HKU CSIS Tech Report TR-2002-17

Upper Bound Results for Multi-Label Interval Routing on Planar Graphs [*]

Savio S.H. Tse and Francis C.M. Lau[†]
Department of Computer Science and Information Systems
The University of Hong Kong
{sshtse,fcmlau}@csis.hku.hk

December 10, 2002

Abstract

Interval routing is a space-efficient routing method for computer networks. In this paper, all graphs are assumed to be planar graphs, unless specified otherwise. We have four upper bound results in this paper. First, for $D \geq 3$, we prove the existence of an $O(D^4)$-IRS on arbitrary graphs whose longest path is bounded by D, where D is the diameter not less than three. With a little modification, we can reduce the number of labels used to $O(D^3)$ with the length of longest path being increased to $(1+\alpha)D$, where α is any constant in $(0, 1)$. Together with the result in Theorem 4 of [14], this result implies an $O(n^{\frac{3}{4}})$-IRS on arbitrary graphs whose longest path is bounded by $(1+\alpha)D$, where n is the number of nodes in the graph. It was proved in [3] that for some non-planar graphs, there is a lower bound of $\frac{3}{2}D - 1$ on the longest path for any M-IRS, $M = O(\frac{n}{D \log \frac{n}{D}})$. Comparing these two results, we conclude that interval routing can perform strictly better in planar graphs with $D = O(\frac{\sqrt[4]{n}}{\log n})$. For larger diameters, the difference between planar and non-planar graphs has yet to be explored. For completeness, we also construct a 6-IRS for arbitrary graphs with $D = 2$.

Keywords: Distributed systems, graph theory, interval routing, network protocols, planar graphs, complexity tradeoffs.

[*] A preliminary version of this paper appeared in *Proc. SIROCCO 2002*, June 2002, pp. 293-308. This work is supported by an RGC CERG grant.

[†] Correspondence: F.C.M. Lau, Department of Computer Science and Information Systems, The University of Hong Kong, Hong Kong / Email: fcmlau@csis.hku.hk / Fax: (+852) 2858 4141.

1 Introduction

Interval routing is a space-efficient routing method for computer networks [10, 8]. The idea is to assign a unique node number (from a cyclicly-ordered set) to every node and then attach a range of node numbers—called an *interval label*—to every outgoing link. To route a message, the destination node number is compared against the interval labels at a node to decide on the link to traverse next. Figure 1 gives a simple example. The figure shows the routing path of a message that travels from Node 2 to Node 0. An interval label of the form $\langle p, q\rangle$ corresponds to the range of node numbers from p to q; intervals of the form $\langle r\rangle$ contain the single node number r. The message first takes the edge to Node 3 because 0 is contained in the interval $\langle 3, 0\rangle$, and then takes the edge to Node 4 because 0 is contained in $\langle 4, 0\rangle$, and so on. Note that the label $\langle 3, 0\rangle$ represents the interval spanning $\{3, 4, 0\}$. It is obvious that $O(d \log n)$ space is sufficient at a node, where d is the node's degree, and n the number of nodes. Whereas using the traditional approach, the routing table at a node could be as large as $O(n \log n)$. Interval routing clearly has an advantage in terms of space, especially for networks with a relatively small degree.

The above labeling of nodes and links and the subsequent routing of messages based on these node numbers and labels is referred to as an *interval routing scheme* (IRS). A valid IRS is one that can route a message from any node to any other node.

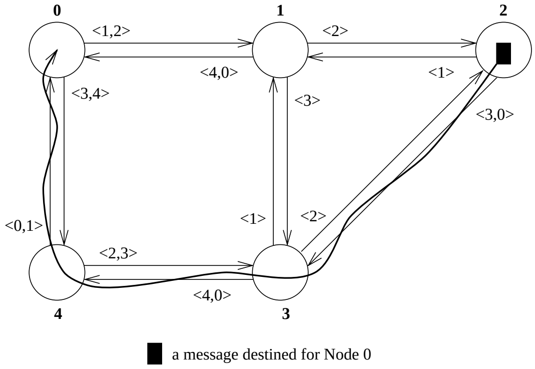

Figure 1: Example of interval routing

Interval routing is indeed a practical method. It has been adopted as the routing method in a commercial routing chip [11], and has thus attracted a fair amount of attention in recent years.

This paper addresses an important question concerning interval routing: given a graph and the best IRS for the graph, what is the quality of the routing paths thus generated? Optimal 1-label IRS's, or simply 1-IRS's, are known to exist for a number of well-known graphs including the tree, the ring, and the two-dimensional mesh [10, 8, 2]. But there are also graphs that are known to admit no optimal 1-IRS, which include the globe graph [9]. For the class of all the graphs that do not admit an optimal 1-IRS, it is meaningful to use more interval labels on each edge. Figure 2 shows a graph which has no optimal 1-label interval routing as proved by Fraigniaud and Gavoille in [2], but has optimal 2-label interval routing. It is believed that the routing path will be shorter if we allow more routing information in the nodes. The study of multi-label interval routing is to find a good trade-off between routing information storage and the path lengths. However, some lower bound results show that we need $\Omega(\sqrt{n})$ labels for optimal IRS [12]. This $\Omega(\sqrt{n})$ lower bound applies even if only a necessary condition of optimality—optimal longest paths—is considered.

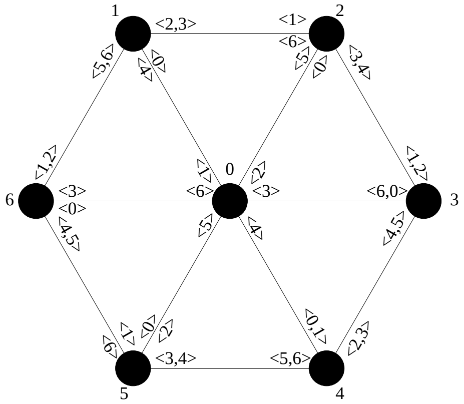

Figure 2: A circular-arc graph with 2 labels in each edges.

Longest path analysis is another way of measuring performance. The first result on longest path analysis is by Ružička [9, 13]. His result—$\frac{3}{2}D - O(1)$—still stands as the best lower bound for 1-IRS after more than ten years. The length of an optimal longest path is obviously the diameter D and so we will use it as an important parameter. Tse and Lau extended the result to multi-label IRS, or M-IRS, and proved a lower bound of $\frac{2M+1}{2M}D - 1$ for $M = O(\sqrt[3]{n})$, and a lower bound of $\frac{2(1+\delta)M+1}{2(1+\delta)M}D$ for $M = O(\sqrt{n})$, for any constant $\delta \in (0, 1]$. The latter result directly implies the lower bound of $\Omega(\sqrt{n})$.

Having studied many lower bound results, one may want to turn to upper bound results. In this paper, we first present an $O(D^4)$-IRS for graphs of $D \geq 3$, whose longest path is bounded by D. This result implies that for all constant-diameter graphs, there exists an $O(1)$-IRS with longest path bounded by D. However, we cannot generalize it to non-planar graphs. A counter example is in [3]. With a little modification, we can reduce the number of labels used to $O(D^3)$-IRS with the length of the longest path increased to $(1+\alpha)D$, where α is any constant in $(0, 1)$ (Section 5). Together with the result in [14] which says that there exists an $(\lfloor \frac{2n}{\alpha D} \rfloor + 1)$-IRS such that the longest path is bounded by $(1 + \alpha)D$, we can directly deduce an $O(n^{\frac{3}{4}})$-IRS whose longest path is bounded by $(1+\alpha)D$. Putting the small α aside, there is an observable gap between the lower bound $\Omega(\sqrt{n})$ and the upper bound $O(n^{\frac{3}{4}})$. Further research is needed to narrow this gap. This paper does not touch on the efficiency of any labeling algorithm. Further research on this issue is needed. For completeness, we also construct a 6-IRS for arbitrary graphs with $D=2$ in Section 6.

2 Definitions and Properties

A graph we consider for interval routing is a connected simple graph, $G=(V,E)$, where V is the set of nodes and E is the set of directed edges such that $(u,v)\in E \Leftrightarrow (v,u)\in E$. In other words, G is an undirected graph. There are n nodes in V and each node has a unique label from the set $\{0,1,\ldots,n-1\}$ denoted as Γ_V. The node labels are cyclicly ordered, denoted as $0 \prec 1 \prec \cdots \prec n-1 \prec 0$. We now define *interval routing scheme* (IRS) formally.

Definition 1 *Let $k\in[2,n]$. For any distinct $v_1,v_2,\ldots,v_k\in\{0,1,\ldots,n-1\}$, $v_1\prec v_2\prec\cdots\prec v_k$ if $\exists p_1,p_2,\ldots,p_{k-1}$ such that $\sum_{i=1}^{i=k-1}p_i<n$ and $v_{i+1}\equiv v_i+p_i \pmod n$ for $i=1$ to $k-1$.*

We further define the expression $u\prec\{v,w\}\prec x$ to be two simultaneous relations based on the cyclic order: $u\prec v\prec x$ and $u\prec w\prec x$.

Definition 2 *An interval $\langle a,b\rangle$ is the set $\{a,a+1,\ldots,b \pmod n\}$. The elements a,b are called the marginal elements of the interval. In particular, $\langle a,a\rangle=\langle a\rangle=\{a\}$, and $\emptyset$ is an empty interval.*

We refer to such a set as an *interval set*. A set $A\subset\Gamma_V$ is not an interval if and only if A is a proper subset of every interval set containing it.

Definition 3 *Given B is an interval. A set is a sub-interval of an interval B if it is an interval and is a subset of B. A is a proper sub-interval of B if A is a sub-interval of B and the marginal elements of A are non-marginal elements of B.*

Definition 4 *Two intervals A and B are non-overlapping if $A\cap B=\emptyset$.*

Definition 5 *Two intervals A and B are disjoint if $A\cup B$ are not an interval.*

Any two disjoint intervals are non-overlapping.

Definition 6 *Let $\mathcal{I}$ be the set containing every possible interval subset of Γ_V.*

Example:

$$\begin{aligned}\Gamma_V = {} & \{0,1,2,3,4\},\\ \mathcal{I} = {} & \{\emptyset,\langle 0\rangle,\langle 1\rangle,\langle 2\rangle,\langle 3\rangle,\langle 4\rangle,\langle 0,1\rangle,\langle 1,2\rangle,\langle 2,3\rangle,\langle 3,4\rangle,\langle 4,0\rangle,\\ & \langle 0,2\rangle,\langle 1,3\rangle,\langle 2,4\rangle,\langle 3,0\rangle,\langle 4,1\rangle,\langle 0,3\rangle,\langle 1,4\rangle,\langle 2,0\rangle,\langle 3,1\rangle,\langle 4,2\rangle,\langle 0,4\rangle\}.\end{aligned}$$

Definition 7 *For $M \geq 2$, let $\mathcal{I}^M$ be the set containing every union of M elements of $\mathcal{I}$.*

Referring to the previous example,

$$\begin{aligned}\mathcal{I}^2 = \quad & \mathcal{I} \cup \{\langle 0\rangle \cup \langle 2\rangle, \langle 0\rangle \cup \langle 3\rangle, \langle 1\rangle \cup \langle 3\rangle, \langle 1\rangle \cup \langle 4\rangle, \langle 2\rangle \cup \langle 4\rangle, \\ & \langle 0,1\rangle \cup \langle 3\rangle, \langle 1,2\rangle \cup \langle 4\rangle, \langle 2,3\rangle \cup \langle 0\rangle, \langle 3,4\rangle \cup \langle 1\rangle, \langle 4,0\rangle \cup \langle 2\rangle\}.\end{aligned}$$

Some unions of intervals are intervals. E.g. $\langle 0\rangle \cup \langle 1\rangle = \langle 0,1\rangle$, $\langle 0,1\rangle \cup \langle 4\rangle = \langle 4,1\rangle$.

Definition 8 *For any V, an node labeling function L is an one-one mapping $L : V \rightarrow \Gamma_V$.*

Definition 9 *Let $M \geq 2$. An M-label edge labeling function L_* is a mapping $L_* : E \rightarrow \mathcal{I}^M$.*

For each $(u,v) \in E$, $L_*(u,v)$ is a union of M intervals. We refer to each of these M intervals as M interval labels of u on (u,v). Since the union of two non-disjoint intervals is an interval, in other words, $L_*(u,v)$ is a union of at most M disjoint intervals.

Definition 10 *An M-interval routing scheme, or M-IRS, on a graph $G = (V,E)$ is an order pair (L, L_*) where L is a node labeling function and L_* is an M-label edge labeling function such that the following are satisfied.*

- *$\forall u, v \in V$, $u \neq v$, $\exists$ a simple path $u, x_1, x_2, \ldots, x_k, v$ in G such that $L(v) \in L_*(u,x_1) \cap L_*(x_1,x_2) \cap \ldots \cap L_*(x_k,v)$, and*
- *$\forall u \in V$, if $(u,v_1),(u,v_2) \in E$, and $v_1 \neq v_2$, then $L_*(u,v_1) \cap L_*(u,v_2) = \emptyset$.*

Directly from Definition 10, we have the following properties.

Property 1 *(Complete) The set of interval labels for edges directed from a node u is complete. That is, $\forall u \in V$, $\Gamma_V - \{L(u)\} \subset \cup_{(u,v)\in E} L_*(u,v)$.*

Property 2 *(Deterministic) The interval labels for edges directed from a node u are disjoint. That is, for $u \neq v$, $L(v)$ is contained in exactly one of these interval labels.*

Property 3 *(No bouncing) $\forall (u,v) \in E$, $L_*(u,v) \cap L_*(v,u) = \emptyset$.*

It should be noted that these two properties are necessary but not sufficient for a valid IRS for general graphs.

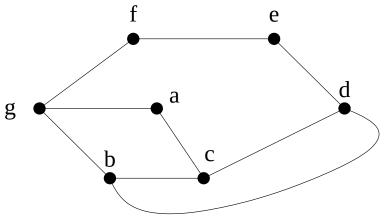

Figure 3: Only nodes a and c are surrounded.

3 An $O(D^4)$-IRS, for graphs of $D \geq 3$

By definition, every planar graph has a geometric representation which can be drawn on a plane such that no two of its edges intersect [1].

Given a graph $G = (V, E)$, $|V| = n$, we choose an arbitrary geometric planar representation and use it for the discussion in this paper. We choose an arbitrary node R such that R is not "surrounded" (Figure 3). Using BFS, we construct a spanning tree T rooted at R. Assume that T has $d + 1$ levels, and its depths is d. Obviously, $d \leq D$.

Based on T, we are now going to label the nodes. We label the graph level by level, anticlockwisely, using integers from 0 to $n - 1$. R is labeled 0. A node in level $i + 1$ will be labeled with a number greater than any node in level i, where $i \in [0, d]$. For each level, a node visited earlier will be labeled with a smaller integer. So, the values of labels are increasing anticlockwisely in a level. Figure 4 is an example. For simplicity, we use a node label as its identity. We modify the definitions of *ancestor* and *descendant*.

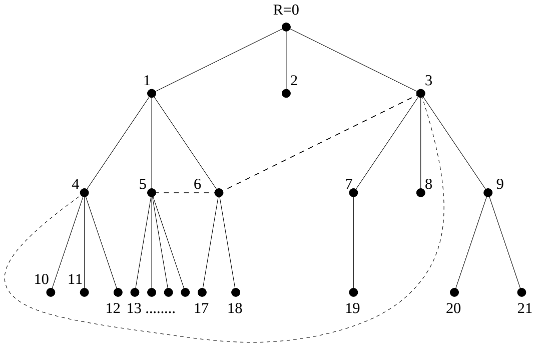

Figure 4: Labels of the nodes. Dotted lines are fronts.

Definition 11 *Given an ith-level node x, a jth-level node y, and $j > i$. If their distance is $j - i$ and all the edges accountable for the distance are geometrically within the i-th and j-th levels, y is called an ancestor of x, and x a descendant of y.*

This definition says that the ancestor-descendant relationship depends on the chosen geometrical representation. Some examples are in Figure 4. Nodes 0, 1 and 3 are ancestors of nodes 6, 17 and 18, and node 5 is an ancestor of nodes 13–16 only; however, node 3 is not an ancestor of nodes 4, 10–12 since the path is not within their levels. The edges not in the spanning tree are called *fronds*.

We are now going to label the edges. For any node u, let A_u be the set of u's descendants, and B_u be the set containing node R and the nodes in the tree path (without fronds) from R to u.

Let C_u be the set $V - A_u - B_u$. Consider the routing from u to the nodes in A_u. Assume that u is at the ith level, then we need at most $d - i$ interval labels in each down edge of u for A_u. The reason is that for each down edge (u, v) of u, $\{v\} \cup A_v$ falls into no more than $d - i - 1$ intervals. An example is shown in Figure 5. For B_u, we need $i - 1$ interval labels in the up edge of u. Therefore, for routing to $A_u \cup B_u$, we need at most $O(D)$ labels in every edge. Obviously, these routing paths are shortest paths.

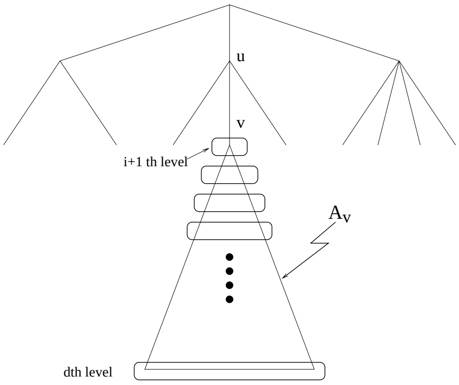

Figure 5: $\{v\} \cup A_v$ falls into $\leq d - i - 1$ intervals.

Consider the last kind of routing which is from u to C_u. First, we put aside the limitation on the number of labels. We label the edges to use the shortest paths from u to x, $\forall x \in C_u, \forall u \in V$.

Definition 12 *Let $\mathcal{S}(a, b)$ be the original shortest routing path from a to b.*

If there are more than one shortest paths for $\mathcal{S}(u, x)$, we will apply the following rules:

R1 $\forall v \in V$, $v \neq u$, $\mathcal{S}(u, v)$ can form a tree rooted at u.

R2 The path must pass through the highest possible ancestor of x, if any.

R3 $\mathcal{S}(x, u)$ should reversely follow as many edges as possible in $\mathcal{S}(u, x)$, as long as **R**2 applies to both paths.

Note that conflict may only arise between **R**2 and **R**3, and **R**2 will have higher priority. Figure 6 shows an example. In the example, $|\mathcal{S}(u,x)| = |\mathcal{S}(x,u)|$ and they use totally different edges. $\mathcal{S}(u,x)$ passes through an ancestor of x, but none of u. Similarly for $\mathcal{S}(x,u)$.

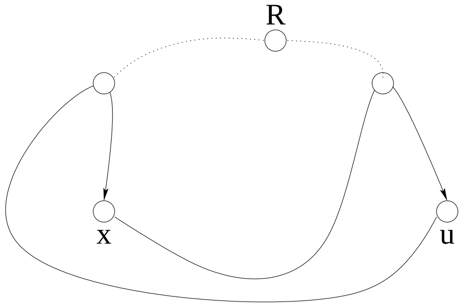

Figure 6: **R**2 overrides **R**3.

According to these shortest paths, we now label the edges of all nodes. We set these interval labels in a way that no interval label contains nodes from different levels, and no interval label contains nodes from C_u as well as $A_u \cup B_u$. It can be done by splitting one interval label into two. Obviously, the routing paths are no longer than D. However, the number of labels in each edge of T may exceed $O(D^4)$. We then show how to reduce the number of labels needed in each edge by relaxing some routing paths.

We divide the nodes in C_u into the left and the right part of $\{u\} \cup A_u \cup B_u$. We consider

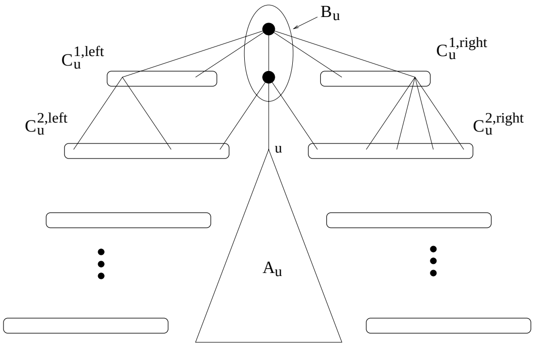

Figure 7: C_u's nodes are divided into left and right parts.

the lth level of the right part. Let $C_u^{l,right}$ be the set of nodes in the lth level of the right part of C_u, $l \in [0,d]$. Similarly for $C_u^{l,left}$. Each $C_u^{l,right}$ forms an interval according to the node labeling method. Figure 7 is an example. Before relaxing any routing paths, they are all shortest paths such that all routings starting from each node form a shortest path tree, *i.e.*, no two routing paths starting from a node will cross each other or touch each other at any middle point. During (and after) path relaxation, part of this property will be conserved as stated in the conditions in Section 4.

Theorem 1 *For each $u \in V$, and for each $l \in [1, d]$, the nodes in $C_u^{l,right}$ will fall into at most $O(D^3)$ interval labels in each edge of u such that all routing paths are no longer than D.*

We prove Theorem 1 in Section 4. We can then apply similar arguments to $C_u^{i,left}$ and deduce Theorem 2 easily.

Theorem 2 *For all graphs of $D \geq 3$, there exists an $O(D^4)$-IRS with all routing paths no longer than D.* □

4 Proof of Theorem 1

Let $X = C_u^{l,right}$. Recall that we set the initial routing paths from u to be all shortest paths. Hence, their lengths are all bounded by D. If a routing path has never been relaxed, it is still the original shortest path. In order to bound the interval labels, we need to relax some of the routing paths having a length less than D. We call such a process *path relaxation*.

We embed the planar representation of G into a Cartesian plane. A region in the Cartesian plane is a set of points surrounded by a set of edges in E. Two points are in the same region surrounded by $E' \subset E$ if and only if they can be connected by a line or curve which does not touch or cross any edges in E'. If $\mathcal{Y}$ is a region, then $\overline{\mathcal{Y}}$ is the complement of $\mathcal{Y}$, *i.e.*, any point $x \in \overline{\mathcal{Y}}$ if and only if $x \notin \mathcal{Y}$.

Definition 13 *Let $\mathcal{R}(a, b)$ be the routing path from a to b.*

For all $s \in V$, we impose a set of invariant conditions **C**1 to **C**4 as follows:

C1 (Path format) $\forall t \in X$, $t \neq s$, if $\mathcal{R}(s, t)$ has never been relaxed, $\mathcal{R}(s, t) = \mathcal{S}(s, t)$. If $\mathcal{R}(s, t)$ has been relaxed, it will have a recursive format $\mathcal{S}(s, x)\mathcal{R}(x, t)$, where $\mathcal{S}(s, x)$ is a subpath of $\mathcal{S}(s, y)$ and y is an ancestor of t.

Hereafter, the expression $\mathcal{R}(s, t) = \mathcal{S}(s, t)$ will refer to the first case that the routing path has never been relaxed. Shortest routing path also refers to the same. The expression $\mathcal{R}(s, t) = \mathcal{S}(s, x)\mathcal{R}(x, t)$ refers to the second case of **C**1. In this case, $\mathcal{S}(s, y)\mathcal{S}(y, t)$ is called the *pseudo-path* of $\mathcal{R}(s, t)$.

C2 (Representative) Each interval label from s to X contains a representative $y \in X$ such that $\mathcal{R}(s, y) = \mathcal{S}(s, y)$. After relaxing $\mathcal{R}(s, x)$, for some $x \in X$, the interval label of s that contains x has a representative $z \in X$ such that $\mathcal{R}(s, z) = \mathcal{S}(s, z) = \mathcal{S}(s, w)\mathcal{S}(w, z)$, where w is an ancestor of x. We say $\mathcal{R}(s, x)$ relies on z. $\mathcal{R}(s, z)$ also passes through one ancestor of each node between x and z. For each node $x' \in X$ if x' is between x and z, $\mathcal{R}(s, x')$ is a relaxed path relying on z.

In particular, if $\mathcal{R}(s,y) = \mathcal{S}(s,y)$, $\mathcal{R}(s,y)$ relies on y.

Definition 14 *Let $\mathcal{R}(s,x)$ be a path where $x \in X$. Define $\mathcal{G}(s,x)$ to be the maximal interval containing x such that $\forall x' \in \mathcal{G}(s,x)$, $x' \in X$, either x' is one of the nodes in $\mathcal{R}(s,x)$, or one ancestor of x' is passed through by $\mathcal{R}(s,x)$.*

C3 If $\mathcal{R}(s,t)$ relies on z, $\mathcal{G}(s,t)$ of the un-relaxed path $\mathcal{S}(s,t)$ is a subset of $\mathcal{G}(s,z)$.

Definition 15 *$\forall x \in X$, let $\|\mathcal{R}(s,x)\|$ be the pseudo-distance of $\mathcal{R}(s,x)$ such that*

$$\|\mathcal{R}(s,x)\| = \begin{cases} |\mathcal{S}(s,x)|, & \text{if } \mathcal{R}(s,x) = \mathcal{S}(s,x); \\ |\mathcal{S}(s,z)|, & \text{if } \mathcal{R}(s,x) \text{ is a relaxed path relying on } z. \end{cases}$$

C4 The number of interval labels of all edges of s does not increase.

C1, **C**2 and **C**3 govern the format of relaxed paths. Figure 8 shows two cases. For the case (a), $\mathcal{R}(s',t)$ relies on z', implying that $\mathcal{R}(s',z') = \mathcal{S}(s',z')$ and $\mathcal{R}(s',z)$ relies on z'. Then, $\mathcal{R}(s',z) \neq \mathcal{S}(s',z)$, implying $\mathcal{R}(s,z) \neq \mathcal{S}(s,z)$. A contradiction to **C**1. For the case (b), **C**2 implies that $\mathcal{R}(s',z)$ is a relaxed path relying on z'. However, $\mathcal{R}(s',z) \neq \mathcal{S}(s',z)$ implies $\mathcal{R}(s,z) \neq \mathcal{S}(s,z)$, which is a contradition to **C**1. The importance of **C**4 is to guarantee that we will not increase the number of labels in any node.

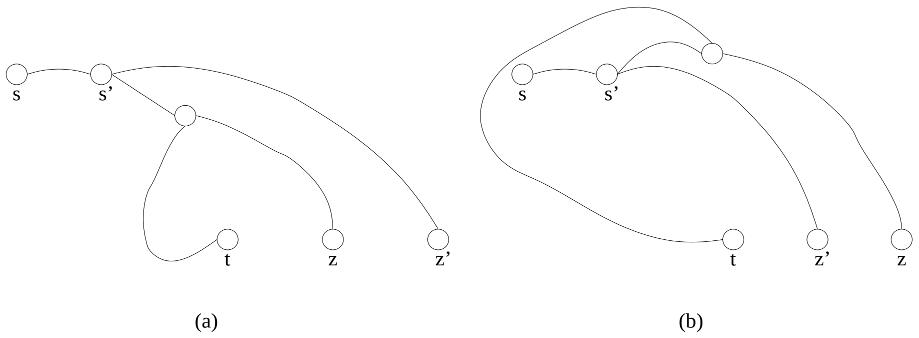

Figure 8: Impossible cases: $\mathcal{R}(s,t) = \mathcal{S}(s,s')\mathcal{R}(s',t)$ relies on z and $\mathcal{R}(s',t)$ relies on z'.

For all $s \in V$, these four conditions must be true during path relaxation. Initially, the above conditions hold. It should be noted that if $\mathcal{R}(u,x) = u, u', \ldots, x$, relaxing $\mathcal{R}(u,x)$ does not always imply the relaxation of $\mathcal{R}(u',x)$. Each routing path can be relaxed many times, as long as the above four conditions conserve.

We show how the number of interval labels for X in any edges of u is bounded by $12D^2(D+2)$. For any $v \neq u$, the number of interval labels for X in any edges of v is not increased but the content of interval labels may change. However, it is guaranteed that any changes will not produce a routing path longer than D and will not add extra interval labels to any edges. Intuitively, our aim is to settle the interval labels in u's edges for routing to X without adversely affecting the others.

Assume that (u, v) has γ interval labels which are subsets of X, where $\gamma > 12D^2(D+2)$. Let the nodes $x_1, x_2, \ldots, x_\gamma$ be the representatives of these γ interval labels from left to right, respectively. Their existence is guaranteed by **C**2.

In each round of relaxation of routing paths, we repeat the arguments in Section 4.1 until the number of interval labels in the edge (u, v) is bounded by $12D^2(D+2)$.

4.1 Path Relaxation

The pre-requisite for each *round* of path relaxation is that $\exists j, k \in \{1, \ldots, \alpha\}$, $j \neq k$, such that $\mathcal{R}(u, x_j) = \mathcal{S}(u, x_j)$, $\mathcal{R}(u, x_k) = \mathcal{S}(u, x_k)$, $|\mathcal{S}(u, x_j)| = |\mathcal{S}(u, x_k)|$ and $\mathcal{S}(u, x_j)$ contains a common ancestor of x_k and x_j, $\overline{x}$. The way to achieve this pre-requisite will be discussed in Section 4.2.

The subpath $\mathcal{S}(u, \overline{x})$ is a common prefix of $\mathcal{S}(u, x_j)$ and $\mathcal{S}(u, x_k)$; otherwise, rule **R**1 will be violated. Let the elements between x_k and x_j be $a_1, a_2, \ldots, a_p$ from left to right, respectively. We also let $a_0 = x_k$ and $a_{p+1} = x_j$. $\overline{x}$ is also their common ancestor. Ultimately, we will relax $\mathcal{R}(u, a_q)$, and make it follow $\mathcal{S}(u, \overline{x})\mathcal{S}(\overline{x}, a_q)$, $\forall q \in [1, p]$. We will do the path relaxation of $\mathcal{R}(u, a_q)$, $\forall q \in [1, p]$, round by round. At the beginning of a round, we walk through the nodes from $\overline{x}$ to u, exclusively and reversely along $\mathcal{S}(u, \overline{x})$. Suppose b_0 is the first node such that the path $\mathcal{R}(b_0, a_q)$ does not pass through $\overline{x}$, $q \in [1, p]$. It implies that if a node v' lies between $\overline{x}$ and b_0, excluding b_0, $\mathcal{R}(v', a_q)$ will pass through $\overline{x}$, $\forall q \in [1, p]$. If no such b_0 exists, path relaxation has been done already.

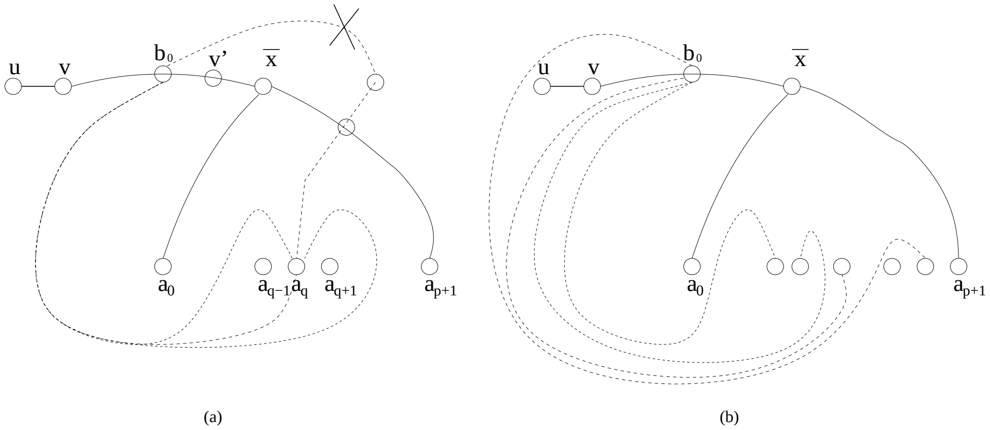

Figure 9: (a) Possible paths of $\mathcal{R}(b_0, a_q)$. (b) Groups of b_0 on $\{a_1, a_2, \ldots, a_p\}$.

Let us investigate the path structure of $\mathcal{R}(b_0, a_q)$, $\forall q \in [1, p]$. If a relaxed path $\mathcal{R}(b_0, a_q)$ relies on a_β, β must fall in $[1, p]$. The reason is that according to the choice of b_0, β cannot be

0 or p, and if $\beta \notin [1, p]$, according to **C**2, $\mathcal{S}(b_0, a_0)$ or $\mathcal{S}(b_0, a_{p+1})$ will be a relaxed path relying on a_β. It is then followed by a contradiction to the pre-requisite $\mathcal{R}(b_0, a_0) = \mathcal{S}(b_0, a_0)$ and $\mathcal{R}(b_0, a_{p+1}) = \mathcal{S}(b_0, a_{p+1})$. Consider an arbitrary $q \in [1, p]$ such that $\mathcal{R}(b_0, a_q) = \mathcal{S}(b_0, a_q)$. $\mathcal{S}(b_0, a_q)$ will not touch $\mathcal{S}(\overline{x}, a_0)$ and $\mathcal{S}(\overline{x}, a_{p+1})$ (Figure 9a), otherwise, rule **R**1 will be violated. Among all shortest routing paths from b_0 to a_q, $q \in [1, p]$, suppose there are exactly p' shortest routing paths. According to **C**2, if $\mathcal{R}(b_0, a_{q'})$, $q' \in [1, p]$, is a relaxed path, it will rely on a node $a_{q''}$, $q'' \in [0, p+1]$, which is the nearest on the left or right side of $a_{q'}$, where $\mathcal{R}(b_0, a_{q''}) = \mathcal{S}(b_0, a_{q''})$. In other words, the set $\{a_1, a_2, \ldots, a_p\}$ is partitioned into p' *groups* which are also disjoint intervals (Figure 9b). We say they are groups of b_0 on $\{a_1, \ldots, a_p\}$. For each group member in a group, the routing path from b_0 to it relies on its group leader.

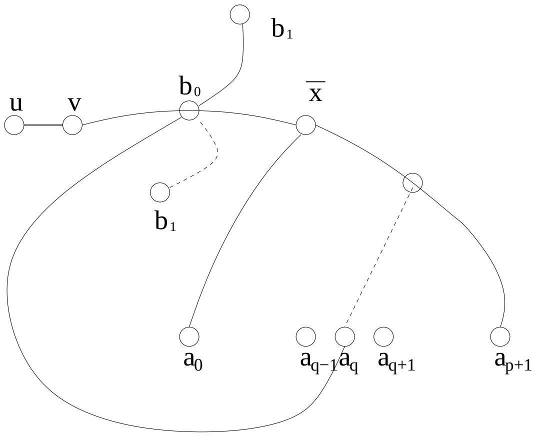

Figure 10: (a)Two possible locations of b_1. (b) A pseudo-path for $\mathcal{R}(b_1, a_q)$.

If we relax $\mathcal{S}(b_0, a_q)$ to follow the path $\mathcal{S}(b_0, \overline{x})$, the new $|\mathcal{R}(b_0, a_q)|$ is the same as the new $\|\mathcal{R}(b_0, a_q)\|$, which is $|\mathcal{S}(b_0, x_j)| \leq D$. However, we could not only relax $\mathcal{S}(b_0, a_q)$ for two reasons. First, there may exist a node b_1 (Figure 10) such that the new $\mathcal{R}(b_1, \overline{a_q})$ is not in a valid format, violating **C**1 or **C**2. It should be noted that the four conditions are conserved everywhere except at b_1. We say b_1 *objects to* the relaxation of $\mathcal{S}(b_0, a_q)$. For such cases, before we change the route for $\mathcal{R}(b_0, a_q)$, we need to find another route for $\mathcal{R}(b_1, a_q)$ without passing through b_0. Then, we relax $\mathcal{R}(b_1, a_q)$. During this relaxation, we may need to relax some other node $a_{q'}$ (a_{q+1} in Figure 10) in order to keep **C**4. However, there may exist a node b_2 objecting to this relaxation.

We extend this idea to b_λ, $\lambda \in Z^+$. The path relaxation will be presented as a recursive function $Relax(b_\lambda, a_{\rho_\lambda}, a_{l_\lambda}, a_{r_\lambda})$. The function is to relax the paths $\mathcal{R}(b_\lambda, a_{\rho_{\lambda+1}})$ to follow the first edge of $\mathcal{R}(b_\lambda, a_{\rho_\lambda})$, $\forall \rho_{\lambda+1} \in [l_\lambda, r_\lambda]$, where $\rho_\lambda \notin [l_\lambda, r_\lambda]$. Before calling the function, we should guarantee the existence of $\mathcal{R}(b_\lambda, a_{\rho_\lambda})$ which passes through the ancestors of $a_{\rho_{\lambda+1}}$, $\forall \rho_{\lambda+1} \in [l_\lambda, r_\lambda]$ (Theorem 3). In implementing $Relax(b_\lambda, a_{\rho_\lambda}, a_{l_\lambda}, a_{r_\lambda})$, the first thing is to extract groups of b_λ on $\{a_{l_\lambda}, \ldots, a_{r_\lambda}\}$. For each group, we will find out those $b_{\lambda+1}$'s objecting to the relaxation of any paths. We will store $b_{\lambda+1}$ in a set B in the form of $(b_{\lambda+1}, f, a_\eta)$, where $f, a_\eta \in X$. We let F be a subset of X that contains f. Then, B implicitly stores the information that $\mathcal{R}(b_{\lambda+1}, f)$ relies on a_η, and $b_{\lambda+1}$ objects to the relaxation of $\mathcal{R}(b_\lambda, f)$. Hence, in particular, $f = a_\eta$ implies $\mathcal{R}(b_{\lambda+1}, f) = \mathcal{S}(b_{\lambda+1}, f)$. We partition B according to the values of a_η—the last attribute of each element. For a particular a_η, consider a $b_{\lambda+1}$ such that the path $b_{\lambda+1}, \ldots, b_\lambda$ is the longest. Among all values of f in $(b_{\lambda+1}, f, a_\eta)$'s, choose the ones that are farthest from a_η in both the left and the right direction. For simplicity, we let the nodes between all f's and a_η be $f^{\lambda+1}_{\min}, \ldots, f^{\lambda+1}_{\max}$, respectively. By **C**2, all nodes inside $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$ are relaxed paths from $b_{\lambda+1}$. Indeed, $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$ is a group of $b_{\lambda+1}$ on $\{a_{l_\lambda}, \ldots, a_{r_\lambda}\}$ with group leader a_η. We will find a path $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$ such that $\mathcal{G}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$ will contain $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$. Intuitively, the path will pass through the ancestors of $f^{\lambda}_{\min}, \ldots, f^{\lambda}_{\max}$, and the nodes in the gap, if any, between them and $a_{\rho_{\lambda+1}}$. The first call in the root of the recursion tree is $Relax(b_0, a_{p+1}, a_1, a_p)$.

```
Function:  $Relax(b_\lambda, a_{\rho_\lambda}, a_{l_\lambda}, a_{r_\lambda})$, where $\rho_\lambda, l_\lambda, r_\lambda \in [1,p]$, $l_\lambda < r_\lambda$.
1.         $B := \emptyset$;
2.         For all $\Delta \in [l_\lambda, r_\lambda]$
2.1            Find all pairs $(b_{\lambda+1}|a_\Delta)$ such that each $\mathcal{R}(b_{\lambda+1}, a_\Delta)$ passes
               through $b_\lambda$ and the new $\mathcal{R}(b_{\lambda+1}, a_\Delta)$ violates C1 or C2;
2.2            Find all pairs $(b_{\lambda+1}|a_\Delta)$ such that each $\mathcal{R}(b_{\lambda+1}, a_\Delta)$ contains
               $(b_{\lambda+1}, b_\lambda) \in E$ and $\mathcal{R}(b_\lambda, a_{\rho_\lambda})$ contains $(b_\lambda, b_{\lambda+1}) \in E$;
2.3            Put all pairs $(b_{\lambda+1}|a_\Delta)$ into $STACK$.
3          While $STACK \neq \emptyset$
3.1            Take out one $(b_{\lambda+1}|a_\Delta)$;
3.2            If $\mathcal{R}(b_{\lambda+1}, a_\Delta) = \mathcal{S}(b_{\lambda+1}, a_\Delta)$
                   Store $(b_{\lambda+1}, a_\Delta, a_\Delta)$ into $B$;
3.3            If $\mathcal{R}(b_{\lambda+1}, a_\Delta) \neq \mathcal{S}(b_{\lambda+1}, a_\Delta)$
3.3.1              Let $\mathcal{S}(y, b_\lambda)$ be the longest subpath of $\mathcal{R}(b_{\lambda+1}, a_\Delta)$
                   such that $\mathcal{R}(y, a_\Delta) = \mathcal{S}(y, b_\lambda)\mathcal{R}(b_\lambda, a_\Delta)$;
3.3.2              $b_{\lambda+1} := y$;
3.3.3              Let $\mathcal{R}(b_{\lambda+1}, a_\Delta)$ relies on $a_\eta$;
3.3.4              Store $(b_{\lambda+1}, a_\Delta, a_\eta)$ into $B$;
4.         Partition $B$ according to the last attributes;
```

Lemma 1 *In Step 3.3.3, $a_\eta \in \{a_{l_\lambda}, \ldots, a_{r_\lambda}\}$ for $\lambda = 0$.*

Proof: By **C**1, $\mathcal{S}(b_{\lambda+1}, a_\eta)$ pass through b_λ. By Step 2, $a_\Delta \in \{a_1, \ldots, a_p\}$. For $\lambda = 0$, if a_η is outside $\{a_1, \ldots, a_p\}$, by **C**2, one of $\mathcal{R}(b_\lambda, a_1)$ and $\mathcal{R}(b_\lambda, a_p)$ must rely on a_η and it violates the pre-requisite that both of them are shortest path. □

Lemma 2 *In Step 3.3.3, assume that* $a_\eta \in \{a_{l_\lambda}, \dots, a_{r_\lambda}\}$ *and we can find a path* $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$ *for step 5.1.6, for* $\lambda = \zeta - 1, \forall \zeta \geq 1$. *Then,* $a_\eta \in \{a_{l_\lambda}, \dots, a_{r_\lambda}\}$ *for the case* $\lambda = \zeta$.

Proof: We denote a_η as $a_{\eta_{\zeta+1}}$ during the execution of $Relax(b_\zeta, a_{\rho_\zeta}, a_{l_\zeta}, a_{r_\zeta})$ and a_{η_ζ} as the ones in its caller $Relax(b_{\zeta-1}, a_{\rho_{\zeta-1}}, a_{l_{\zeta-1}}, a_{r_{\zeta-1}})$. By the algorithm, $\{a_{l_\zeta}, \dots, a_{r_\zeta}\}$ is between a_{ρ_ζ} and the range $[f^\zeta_{\min}, f^\zeta_{\max}]$. In other words, a_Δ is between a_{ρ_ζ} and $a_{\eta_{\zeta-1}}$ since $a_{\eta_{\zeta-1}} \in [f^\zeta_{\min}, f^\zeta_{\max}]$. If $a_{\eta_{\zeta+1}}$ is outside the range, $\mathcal{S}(b_\zeta, a_{\eta_\zeta})$ or $\mathcal{S}(b_\zeta, x)$, for some node $x \in [f^\zeta_{\min}, f^\zeta_{\max}]$, will rely on $a_{\eta_{\zeta+1}}$. By assumption in the lemma statement, both shortest paths exist. Then, it violates **C**2. □

```
5.          For each partition $B'$
5.1            While $B' \neq \emptyset$
5.1.1             Take out a $(b_{\lambda+1}, *, a_\eta) \in B'$ such that the subpath
                  $b_{\lambda+1}, \dots, b_\lambda$ is the longest, where $*$ means any value;
5.1.2             $B'' := \{(b_{\lambda+1}, f, a_\eta) \in B' | f \in F\}$;
5.1.3             $B' := B' - B''$;
5.1.4             $F' := \{f | (b_{\lambda+1}, f, a_\eta) \in B''\} \cup \{a_\eta\}$;
5.1.5             $f^{\lambda+1}_{\max} := \max(F')$;  $f^{\lambda+1}_{\min} := \min(F')$;
5.1.6             Find $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$ such that $a_{\rho_{\lambda+1}}$ is the nearest to
                  the range $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}] \subset \mathcal{G}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$;
5.1.7             If a gap exists between $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$ and $a_{\rho_{\lambda+1}}$
5.1.7.1              Let $a_{l_{\lambda+1}}, \dots, a_{r_{\lambda+1}}$, where $l_{\lambda+1} < r_{\lambda+1}$, be the nodes in
                     the gap which have not relied on $a_{\rho_{\lambda+1}}$;
5.1.7.2              $Relax(b_{\lambda+1}, a_{\rho_{\lambda+1}}, a_{l_{\lambda+1}}, a_{r_{\lambda+1}})$;
5.1.8             Force $\mathcal{R}(b_\lambda, a_\Delta), \forall a_\Delta \in [f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$, to the first edge
                  of $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$;
6.          Force $\mathcal{R}(b_\lambda, a_\Delta), \forall \Delta \in [l_\lambda, r_\lambda]$, to the first edge of $\mathcal{S}(b_\lambda, a_{\rho_\lambda})$;
```

Steps 2, 3 and 4 are to find a set B containing $b_{\lambda+1}$'s objecting to the relaxation of $\mathcal{R}(b_\lambda, a_\Delta)$, $\Delta \in [l_\lambda, r_\lambda]$. Directly from the algorithm, we have the following two observations.

Fact 1 *For all* $\zeta \in [1, \lambda+1]$, $a_{\rho_\zeta} \notin \{a_{l_\zeta}, \dots, a_{r_\zeta}\} \cup [f^\zeta_{\min}, f^\zeta_{\max}]$. □

Fact 2 *For all* $\zeta \in [1, \lambda+1]$, *there is no increase of number of interval labels in the first edge of* $\mathcal{S}(b_\zeta, a_{\rho_\zeta})$.

Proof: In order to keep the number of interval labels unchanged, we group two kinds of nodes in X into the interval containing a_{ρ_ζ}. One is inside $[f^\zeta_{\min}, f^\zeta_{\max}]$. Relaxation of this kind of nodes will not be objected to by others since we choose the b_ζ such that $\mathcal{S}(b_\zeta, b_{\zeta-1})$ is longest. The other kind is the nodes in the gap between $[f^\zeta_{\min}, f^\zeta_{\max}]$ and a_{ρ_ζ}, which have not yet relied on a_{ρ_ζ}. The relaxation of nodes of this kind, ranging from $\{a_{l_\zeta}$ to $a_{r_\zeta})\}$, will be done after the return of the recursive call $Relax(b_\zeta, a_{\rho_\zeta}, a_{l_\zeta}, a_{r_\zeta})$. For those inside the gap and have already relied on a_{ρ_ζ}, we need not handle them. Hence, the result follows. □

Lemma 3 *Consider a destination $a_\triangle$ and the set B created just after step 3. For all $\zeta \in [1, \lambda+1]$, if $\mathcal{R}(b_\zeta, a_\triangle)$ is changed to another path without passing through $b_{\zeta-1}$ for all b_ζ in B such that* **C**1 *to* **C**4 *are conserved, then no nodes in V will object the relaxation of $\mathcal{R}(b_{\zeta-1}, a_\triangle)$.*

Proof: For all b_ζ in B, if $\mathcal{R}(b_\zeta, a_\triangle)$ is changed to another path without passing through $b_{\zeta-1}$ such that **C**1 to **C**4 are conserved, then b_ζ will not object to the relaxation of $\mathcal{R}(b_{\zeta-1}, a_\triangle)$. Now, we consider b_ζ which is not in B, but objects to the relaxation of $\mathcal{R}(b_{\zeta-1}, a_\triangle)$.

If $\mathcal{R}(b_\zeta, a_\triangle) = \mathcal{S}(b_\zeta, a_\triangle)$, according to Step 3.1, b_ζ must be in B. So, we assume $\mathcal{R}(b_\zeta, a_\triangle) \neq \mathcal{S}(b_\zeta, a_\triangle)$.

Let $\mathcal{S}(y, b_{\zeta-1})$ be the longest subpath of $\mathcal{R}(b_\zeta, a_\triangle)$ ending at $b_{\zeta-1}$. Then, $\mathcal{R}(y, a_\triangle)$ is a subpath of $\mathcal{R}(b_\zeta, a_\triangle)$. If there exists another path for $\mathcal{R}(y, a_\triangle)$ without passing through $b_{\zeta-1}$ such that **C**1 to **C**4 are conserved, then by the property of interval routing, there exists another path for $\mathcal{R}(b_\zeta, a_\triangle)$ without passing through $b_{\zeta-1}$, either. □

Step 4 divides the set B into partitions in the way that the last attributes of all elements in each partition are the same. In Step 5, the algorithm will work on all elements in all partitions. Step 5.1.1 takes out the farthest $b_{\lambda+1}$ from B'. That means, in each iteration of the while-loop, $\mathcal{R}(b_{\lambda+1}, a_\triangle)$ will not be a proper subpath of $\mathcal{R}(b'_{\lambda+1}, a_\triangle)$, for all $b'_{\lambda+1}$ in B', and $a_\triangle$ in $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$. Steps 5.1.2 to 5.1.5 provide a range $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$ such that a particular $b_{\lambda+1}$ objects to the relaxation of $\mathcal{R}(b_\lambda, a_\triangle)$, $\forall a_\triangle \in [f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$. The purpose of Step 5.1.6 is to find the other paths for $\mathcal{R}(b_{\lambda+1}, a_\triangle)$, $\forall a_\triangle \in [f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$. We need the condition $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}] \subset \mathcal{G}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$ in order to prove the invariance of **C**1 and **C**2 (Theorem 5). The requirement that the position of $a_{\rho_{\lambda+1}}$ is the nearest or inside $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$ will be useful in proving the existence of $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$ (Theorem 3). Then, we will force $\mathcal{R}(b_{\lambda+1}, a_\triangle)$'s to follow the first edge of $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$. We are going to give a formal proof on the correctness of algorithm.

Consider a particular $(b_1, f, a_{\eta_1}) \in B$ in the implementation of $Relax(b_0, a_{p+1}, a_1, a_p)$. There is a path $\mathcal{R}(b_1, a_{\eta_1}) = \mathcal{S}(b_1, a_{\eta_1})$ passing through b_0. Similarly, for a particular $\lambda > 0$, there is a path $\mathcal{R}(b_{\lambda+1}, a_{\eta_{\lambda+1}}) = \mathcal{S}(b_{\lambda+1}, a_{\eta_{\lambda+1}})$ passing through b_λ in the implementation of $Relax(b_\lambda, a_{\rho_\lambda}, a_{l_\lambda}, a_{r_\lambda})$, $\rho_\lambda, l_\lambda, r_\lambda \in [1, p]$, $l_\lambda < r_\lambda$, Let $\mathcal{S}(b_\lambda, a_{\eta_\lambda})$, $\lambda \geq 1$, be these paths. By the rules **R**1 and **R**3, the paths $\mathcal{S}(b_{\lambda+1}, a_{\eta_{\lambda+1}})$ and $\mathcal{S}(b_\lambda, a_{\eta_\lambda})$ have either exactly one common subpath containing b_λ, or exactly one common node at b_λ. According to the algorithm, $b_{\lambda+1}$ and b_λ are connected by a shortest path which is a subpath of

$\mathcal{S}(b_{\lambda+1}, a_{\eta_{\lambda+1}})$. Then, there is a path $\mathcal{S}(b_{\lambda+1}, b_\lambda)\mathcal{S}(b_\lambda, b_{\lambda-1}) \cdots \mathcal{S}(b_1, b_0)$ which connects b_ζ together, $\forall \zeta \in [0, \lambda+1]$. The path guarantees the existence of a touching point between $\mathcal{S}(b_\zeta, a_{\eta_\zeta})$ and $\mathcal{S}(b_{\zeta-1}, a_{\eta_{\zeta-1}})$, $\forall \zeta \in [2, \lambda+1]$. These touching points guarantee the validity of the following definition.

Definition 16 *Define $\mathcal{P}_0$ to be an empty path ($\emptyset$). Define $\mathcal{P}_1$ to be the path $\mathcal{S}(b_1, b_0)$, and $c_1 = b_0$. For all $\zeta \in [2, \lambda+1]$, define $\mathcal{P}_\zeta$ to be the union of the subpath $\mathcal{S}(b_\zeta, c_\zeta)$ of $\mathcal{S}(b_\zeta, a_{\eta_\zeta})$ and the suffix of $\mathcal{P}_{\zeta-1}$ starting at c_ζ, where c_ζ is defined as the chosen touching point of $\mathcal{S}(b_\zeta, a_{\eta_\zeta})$ and $\mathcal{P}_{\zeta-1}$ such that the suffix is shortest.*

Figure 11 shows two examples.

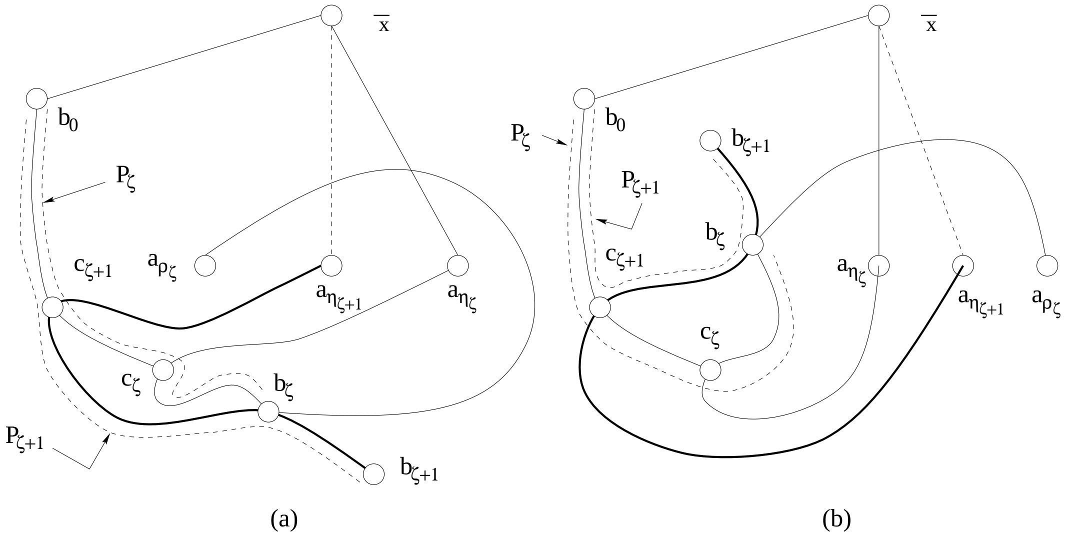

Figure 11: Relationship between $\mathcal{P}_\zeta$ and $\mathcal{P}_{\zeta+1}$, $\forall \zeta \in [1, \lambda+1]$.

Lemma 4 *(Step 5.1.6) There exists $\mathcal{S}(b_1, a_{\rho_1})$ such that (1) a_{ρ_1} and b_1 are in different regions separated by $\mathcal{S}(b_0, \overline{x})$, $\mathcal{S}(\overline{x}, a_{\eta_1})$, and $\mathcal{S}(b_0, a_{\eta_1})$; (2) $[f^1_{\min}, f^1_{\max}] \subset \mathcal{G}(b_1, a_{\rho_1})$; and (3) $\{a_{l_1}, \ldots, a_{r_1}\} \subset \{a_1, \ldots, a_p\}$.*

Proof: Consider the function call $Relax(b_0, a_{p+1}, a_1, a_p)$. The nodes in $[f^1_{\min}, f^1_{\max}]$ form a group of b_1, whose leader is a_{η_1}. Let $\mathcal{X}$ and $\mathcal{Y}$ be the two disjoint regions, and b_1 be in $\mathcal{X}$. Since the boundary paths are all shortest paths and they cannot touch any of $\mathcal{S}(b_0, a_0)$ and $\mathcal{S}(b_0, a_{p+1})$. Hence, a_0 and a_{p+1} are in different regions. We assume that a_0 is in $\mathcal{X}$, and then a_{p+1} is in $\mathcal{Y}$. We leave the other similar case that a_0 is in $\mathcal{Y}$ to the reader.

Consider $\mathcal{R}(b_1, a_{p+1})$. It is done if it touches $\mathcal{S}(b_0, \overline{x})$, *i.e.*, we can use a_{p+1} as a_{ρ_1}. We now consider the un-relaxed $\mathcal{S}(b_1, a_{p+1})$. By **R**1, it cannot touch $\mathcal{S}(b_0, a_{\eta_1})$. Obviously, it cuts $\mathcal{S}(\overline{x}, a_{\eta_1})$ and passes through the ancestors of $\mathcal{G}(b_0, a_{\eta_1})$. It must pass through the ancestors of the nodes in $[f^1_{\min}, f^1_{\max}]$.

- Consider the case that $[f^1_{\min}, f^1_{\max}] \subset \mathcal{G}(b_1, a_{p+1})$. If $\mathcal{R}(b_1, a_{p+1}) = \mathcal{S}(b_1, a_{p+1})$, it is done by letting $a_{\rho_1} = a_{p+1}$; otherwise, we let $\mathcal{R}(b_1, a_{p+1})$ rely on a. Then, it is done by letting $a_{\rho_1} = a$. In both cases, $a_{l_1}, \ldots, a_{r_1}$, the nodes that need to rely on a_{ρ_1}, will reside between a_{η_1} and a_{p+1}, exclusively. It is because we need not relax the nodes between a_{p+1} and a, inclusively.

- Consider the case that $[f^1_{\min}, f^1_{\max}] \not\subset \mathcal{G}(b_1, a_{p+1})$. There exists a node a_ρ between a_{η_1} and a_{p+1} such that $a_\rho \notin \mathcal{G}(b_1, a_{\eta_1})$ and $\mathcal{R}(b_1, a_\rho) = \mathcal{S}(b_1, a_\rho) \Rightarrow [f^1_{\min}, f^1_{\max}] \subset \mathcal{G}(b_1, a_\rho)$

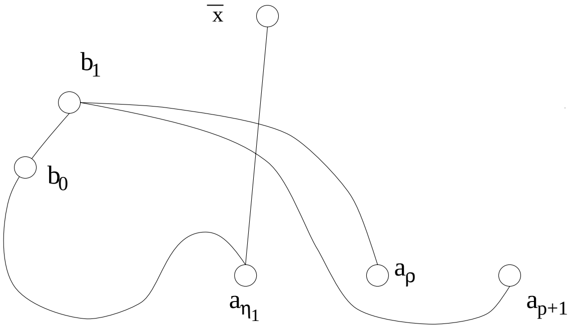

Figure 12: The node a_ρ is a candidate for a_{ρ_1} if $\mathcal{R}(b_1, a_\rho) = \mathcal{S}(b_1, a_\rho)$.

(Figure 12). If $\mathcal{R}(b_1, a_\rho) = \mathcal{S}(b_1, a_\rho)$, it is done by letting $a_{\rho_1} = a_\rho$; otherwise, we let $a_{\rho_1} = a$ where $\mathcal{R}(b_1, a_\rho)$ relies on a. Obviously, $a_{l_1}, \ldots, a_{r_1}$ are between a_{η_1} and a_{p+1}, exclusively.

Note that all candidates for a_{ρ_1} is outside $\mathcal{X}$, *i.e.*, b_1 and a_{ρ_1} are in different regions. □

Lemma 5 *If c_2 is a node on $\mathcal{S}(b_1, a_{\eta_1})$, and $c_2 \neq b_0$, then $\mathcal{G}(b_2, a_{\eta_2}) \subset \mathcal{G}(c_2, a_{\eta_1})$ and $a_{\eta_1} \notin \mathcal{G}(b_2, a_{\eta_2})$.*

Proof: The node c_2 is on the path $\mathcal{S}(b_1, b_0)$. By **R**1, $\mathcal{S}(c_2, a_{\eta_1})$ cannot touch $\mathcal{S}(c_2, a_{\eta_2})$, except at c_2. Consider two regions $\mathcal{X}$ and $\mathcal{Y}$ separated by paths $\mathcal{S}(b_0, \overline{x})$, $\mathcal{S}(\overline{x}, a_{\eta_1})$, and $\mathcal{S}(b_0, a_{\eta_1})$. The node a_{η_2} must reside in the same region as b_1, b_2 and c_2. Assuming the contrary, and $\mathcal{S}(b_2, a_{\eta_2})$ will touch $\mathcal{S}(b_0, \overline{x})$ or $\mathcal{S}(\overline{x}, a_{\eta_1})$. If $\mathcal{S}(b_2, a_{\eta_2})$ touches $\mathcal{S}(b_0, \overline{x})$, its subpath $\mathcal{S}(b_1, a_{\eta_2})$ can be a better choice for Step 5.1.6, which is a contradiction because by Fact 1, $a_{\rho_1} \notin [f^1_{\min}, f^1_{\max}]$. If $\mathcal{S}(b_2, a_{\eta_2})$ touches $\mathcal{S}(\overline{x}, a_{\eta_1})$, then $\mathcal{S}(b_2, a_{\eta_2})$ passes through the ancestors of $\mathcal{G}(b_1, a_{\eta_1})$. If $\mathcal{G}(b_1, a_{\eta_1}) \subset \mathcal{G}(b_1, a_{\eta_2})$, we can use $\mathcal{S}(b_1, a_{\eta_2})$ as a better choice for Step 5.1.6. If $\mathcal{G}(b_1, a_{\eta_1}) \not\subset \mathcal{G}(b_1, a_{\eta_2})$, then there exists a node $a_{\eta'}$ between a_{η_2} and $\mathcal{G}(b_1, a_{\eta_1})$ such that $a_{\eta'} \notin \mathcal{G}(b_2, a_{\eta_2})$. Choose an $a_{\eta'}$ such that it is closest (next) to

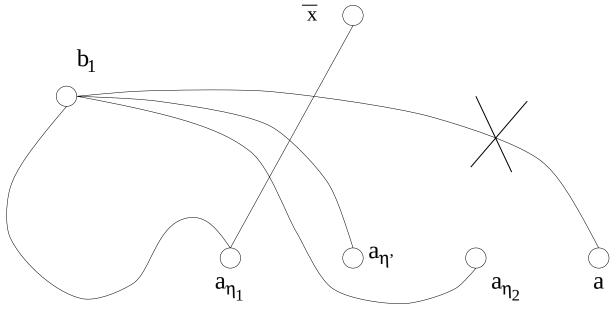

Figure 13: $\mathcal{G}(b_1, a_{\eta_1}) \subset \mathcal{G}(b_1, a_{\eta'})$ and a_{η_2} cannot lie between $a_{\eta'}$ and a.

$\mathcal{G}(b_1, a_{\eta_1})$. Then, $\mathcal{R}(b_1, a_{\eta'}) = \mathcal{S}(b_1, a_{\eta'}) \Rightarrow \mathcal{G}(b_1, a_{\eta_1}) \subset \mathcal{G}(b_1, a_{\eta'})$ (Figure 13). Therefore, if $\mathcal{R}(b_1, a_{\eta'}) = \mathcal{S}(b_1, a_{\eta'})$, $\mathcal{S}(b_1, a_{\eta'})$ is a better choice for Step 5.1.6, contradicting the choice of a_{ρ_1}. Consider the case $\mathcal{R}(b_1, a_{\eta'}) \neq \mathcal{S}(b_1, a_{\eta'})$. Let $\mathcal{R}(b_1, a_{\eta'})$ rely on a. a must reside between a_{η_1} and a_{η_2}; otherwise by **C**2, either $\mathcal{R}(b_1, a_{\eta_1}) \neq \mathcal{S}(b_1, a_{\eta_1})$ or $\mathcal{R}(b_1, a_{\eta_2}) \neq \mathcal{S}(b_1, a_{\eta_2})$. Now, $\mathcal{R}(b_1, a) = \mathcal{S}(b_1, a)$ implies $\mathcal{S}(b_1, a)$ is a better choice for Step 5.1.6. A contradiction. Hence, b_1, b_2, c_2 and a_{η_2} are in the same region.

By Lemma 4, a_{ρ_1} is in the opposite region as b_1. Hence, a_{η_2} and a_{ρ_1} are also in opposite regions. Since a_{η_2} is between a_{η_1} and a_{ρ_1}, there are two cases, $a_{\eta_1} < a_{\eta_2} < a_{\rho_1}$ and $a_{\rho_1} < a_{\eta_2} < a_{\eta_1}$. Consider $a_{\eta_1} < a_{\eta_2} < a_{\rho_1}$. By Lemma 4, $a_{\eta_1} \in [f^1_{\min}, f^1_{\max}] \subset \mathcal{G}(b_1, a_{\rho_1})$. From the Definition 14, $a_{\eta_2} \in \mathcal{G}(b_1, a_{\rho_1})$. By **R**1, $\mathcal{S}(b_1, a_{\rho_1})$ and $\mathcal{S}(b_1, a_{\eta_1})$ has only one segment of contact at b_1. Hence, $\mathcal{S}(b_1, a_{\eta_1})$ cannot pass through a_{ρ_1}, nor any of its ancestors. It must pass through a_{η_2} or one of its ancestors in order to separate a_{η_2} and a_{ρ_1} in different regions (Figure 14). If $a_{\eta_2} \notin \mathcal{G}(b_1, a_{\eta_1})$, there exists some node a' between a_{η_1} and a_{η_2}, which can take up the role of a_{ρ_1} but nearer to a_{η_1} (Figure 14). A contradiction. Hence, $a_{\eta_2} \in \mathcal{G}(b_1, a_{\eta_1})$, which implies $\mathcal{G}(b_2, a_{\eta_2}) \subset \mathcal{G}(c_2, a_{\eta_1})$.

Since, by **R**1, $\mathcal{S}(b_2, a_{\eta_2})$ does not touch $\mathcal{S}(b_1, a_{\eta_1})$, except at b_1. Hence, $a_{\eta_2} \in \mathcal{G}(b_1, a_{\eta_1})$ implies $a_{\eta_1} \notin \mathcal{G}(b_2, a_{\eta_2})$. □

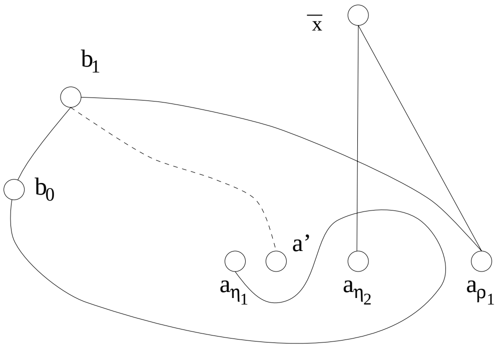

Figure 14: Separation of a_{η_2} and a_{ρ_1} makes $\mathcal{S}(b_1, a_{\eta_1})$ pass a_{η_2} or one of its ancestor.

Theorem 3 *(Step 5.1.6) For all $\zeta \in [1, \lambda+1]$, there exists $\mathcal{S}(b_\zeta, a_{\rho_\zeta})$ such that (1) a_{ρ_ζ} and b_ζ are in different regions separated by $\mathcal{S}(b_0, \overline{x})$, $\mathcal{S}(\overline{x}, a_{\eta_\zeta})$, $\mathcal{S}(c_\zeta, a_{\eta_\zeta})$ and the suffix of $\mathcal{P}_\zeta$ starting at c_ζ; (2) $[f^\zeta_{\min}, f^\zeta_{\max}] \subset \mathcal{G}(b_\zeta, a_{\rho_\zeta})$; (3) $\{a_{l_\zeta}, \ldots, a_{r_\zeta}\} \subset \{a_1, \ldots, a_p\}$; and (4) if $c_{\zeta+1} \neq b_0$, then $\exists \zeta' \leq \zeta$ such that $c_{\zeta+1}$ is a node on $\mathcal{S}(b_{\zeta'}, a_{\eta_{\zeta'}})$, $\mathcal{G}(b_{\zeta+1}, a_{\eta_{\zeta+1}}) \subset \mathcal{G}(c_{\zeta+1}, a_{\eta_{\zeta'}})$, and $a_{\eta_{\zeta'}} \neq a_{\eta_{\zeta+1}}$.*

Proof: We will prove cases (1) to (4) by induction on ζ. The base cases of (1), (2) and (3) are done in Lemma 4. The base case of (4) is done in Lemma 5. Assume the theorem is true for cases $\zeta < \lambda+1$. By induction assumption (2), $[f^\zeta_{\min}, f^\zeta_{\max}] \subset \mathcal{G}(b_\zeta, a_{\rho_\zeta})$, for all $\zeta < \lambda+1$. Hence, $\mathcal{G}(b_1, a_{\eta_1})$, $\mathcal{G}(b_2, a_{\eta_2})$,. . ., $\mathcal{G}(b_{\lambda+1}, a_{\eta_{\lambda+1}})$ are defined. Consider a particular $\pi \in [3, \lambda]$. Now we prove that the theorem is true for the case $\zeta = \pi + 1$.

According to Step 5.1.6 of the algorithm, $a_{\eta_\pi} \in \mathcal{G}(b_\pi, a_{\rho_\pi})$, and $a_{\eta_{\pi+1}}$ is between a_{η_π} and a_{ρ_π}. Consider the path $\mathcal{S}(b_{\pi+1}, a_{\eta_{\pi+1}})$. It cannot touch $\mathcal{S}(b_\pi, a_{\eta_\pi})$ due to **R**1. It cannot touch $\mathcal{S}(\overline{x}, a_{\eta_\pi})$ because it will imply another better choice of $\mathcal{S}(b_\pi, a_{\rho_\pi})$ (as the dotted $\mathcal{S}(b_1, a')$ in Figure 14). It cannot touch $\mathcal{S}(b_0, \overline{x})$ because it will violate **R**1 by cutting $\mathcal{S}(b_\pi, a_{\rho_\pi})$ (as $\mathcal{S}(b_\zeta, a_{\rho_\zeta})$ cuts the dotted path $\mathcal{S}(\overline{x}, a_{\eta_{\zeta+1}})$ in Figure 11). Hence, if $a_{\eta_{\pi+1}}$ is in different regions with $b_{\pi+1}$, $\mathcal{S}(b_{\pi+1}, a_{\eta_{\pi+1}})$ must pass through a node $c_{\pi+1}$ in $\mathcal{P}_\pi$, where $c_{\pi+1} \neq b_\pi$ (Figure 11). In the case when $c_{\pi+1} = b_\pi$, $a_{\eta_{\pi+1}}$ and $b_{\pi+1}$ are in the same region. The path $\mathcal{S}(b_{\pi+1}, c_{\pi+1})$ is a subpath of $\mathcal{P}_{\pi+1}$.

We define σ_μ to be some integer within the range $[1, \lambda+1]$, $\forall \mu \in [1, \lambda+1]$, and $\sigma_{\mu_1} > \sigma_{\mu_2} \Leftrightarrow \mu_1 > \mu_2$. If $c_{\pi+1} \neq b_0$, directly from Definition 16, we have a sequential view of $\mathcal{P}_{\pi+1}$,

$$\mathcal{S}(b_{\pi+1}, c_{\sigma_\nu})\mathcal{S}(c_{\sigma_\nu}, c_{\sigma_{\nu-1}})\mathcal{S}(c_{\sigma_{\nu-1}}, c_{\sigma_{\nu-2}}) \cdots \mathcal{S}(c_{\sigma_1}, c_{\sigma_0}),$$

where $0 \leq \nu \leq \pi+1$, $c_{\sigma_0} = b_0$, and $\sigma_\nu = \pi+1$. $\mathcal{P}_{\pi+1}$ is composed of $\nu+1$ shortest paths. The value of ν depends on the initial choice of all (shortest) routing paths. Note that $\forall \mu \in [1, \nu]$, $\mathcal{S}(c_{\sigma_\mu}, c_{\sigma_{\mu-1}})$ is a subpath of $\mathcal{S}(b_{\sigma_{\mu-1}}, a_{\eta_{\sigma_{\mu-1}}})$, and $\mathcal{S}(b_{\pi+1}, c_{\sigma_\nu})$ is a subpath of $\mathcal{S}(b_{\pi+1}, a_{\eta_{\sigma_\nu}})$.

By induction assumption (4) and **R**1, we can easily show that the paths $\mathcal{S}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$, $\mathcal{S}(c_{\sigma_{\nu-1}}, a_{\eta_{\sigma_{\nu-1}}}) \cdots \mathcal{S}(c_{\sigma_1}, a_{\eta_{\sigma_1}})$ are structured in layers and that the path $\mathcal{S}(c_{\sigma_\mu}, a_{\eta_{\sigma_\mu}})$ does not touch the path $\mathcal{S}(c_{\sigma_{\mu'}}, a_{\eta_{\sigma_{\mu'}}})$ (Figure 15), if $\mu, \mu' \in [1, \nu]$ and $\mu \neq \mu'$.

We define $\mathcal{X}$ to be a minimal region that contains the paths $\mathcal{S}(b_0, a_{\eta_1})$, $\mathcal{S}(c_{\sigma_1}, a_{\eta_{\sigma_1}})$,. . ., $\mathcal{S}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$, the suffix of $\mathcal{P}_\pi$ starting from c_{σ_ν}, and the shortest line joining $a_{\eta_1}, a_{\eta_{\sigma_1}}, \ldots, a_{\eta_{\sigma_\nu}}$, but does not contain the paths $\mathcal{S}(\overline{x}, a_0)$, $\mathcal{S}(\overline{x}, a_{p+1})$, and $\mathcal{S}(b_0, \overline{x})$ excluding b_0. Considering all possible positions of $b_{\pi+1}$, we have the following two cases.

1. $b_{\pi+1}$ is not in $\mathcal{X}$. Figure 16 and 17 show some examples.

   Consider a node a' which is outside $\mathcal{X}$, and in a different region with $b_{\pi+1}$, where the region is bounded by the paths $\mathcal{S}(b_0, \overline{x})$, $\mathcal{S}(\overline{x}, a_{\eta_{\sigma_\mu}})$, $\mathcal{S}(c_{\sigma_\mu}, a_{\eta_{\sigma_\mu}})$, and the suffix of $\mathcal{P}_\pi$ starting from c_{σ_μ}. Obviously, $a' \in \{a_0, \ldots, a_{p+1}\}$. Consider the un-relaxed

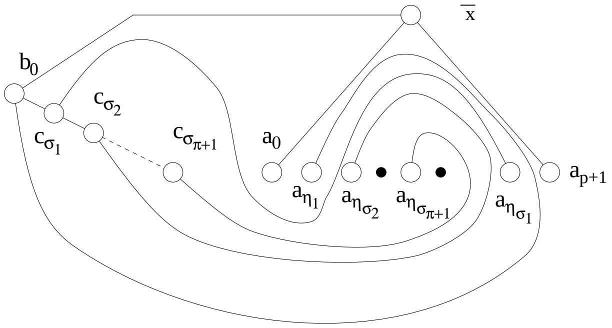

Figure 15: The paths are organized in distinct layers.

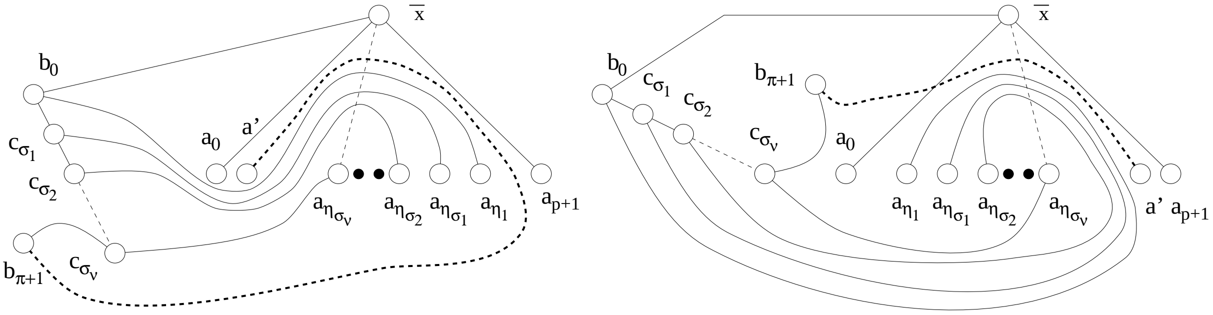

Figure 16: $b_{\pi+1}$ is not in $\mathcal{X}$.

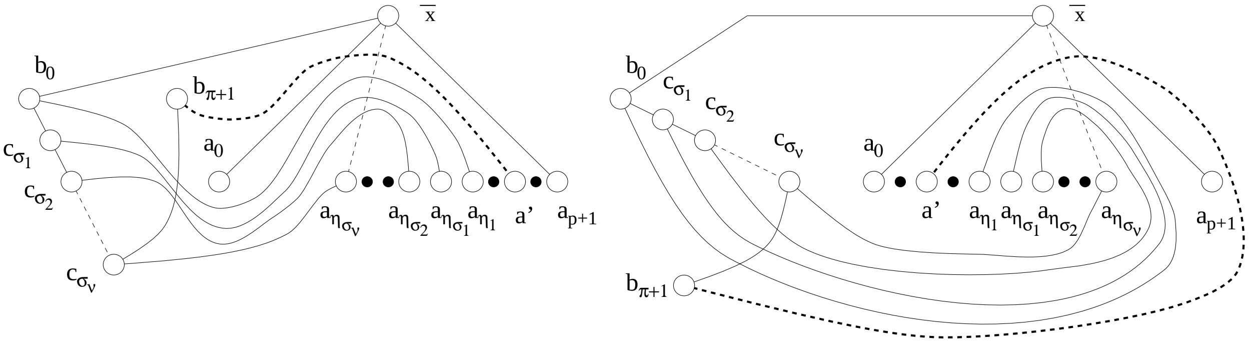

Figure 17: $b_{\pi+1}$ is not in $\mathcal{X}$.

(shortest) path from $b_{\pi+1}$ to a'. If it passes through b_0, take $a'' \in \{a_0, a_{p+1}\}$ such that a'' and $b_{\pi+1}$ are in different regions. We will consider the node between a' and a'', including a'' but excluding a', and set a' to be that node. Iteratively, we must be able to find an a' such that the path does not pass through b_0. If it touches $\mathcal{S}(b_0, \overline{x})$, it is done because it passes through $\overline{x}$ which is the ancestor of every node in $\{a_1, \ldots, a_p\}$. Because of **R**1, it cannot touch $\mathcal{S}(c_{\pi+1}, a_{\eta_{\pi+1}})$ if the last overlapping node is before $c_{\pi+1}$. If it touches $\mathcal{S}(c_{\sigma_\mu}, c_{\sigma_{\mu-1}})$, $\mu \in [1, \nu]$, then it cannot touch $\mathcal{S}(c_{\sigma_\mu}, a_{\eta_{\sigma_\mu}})$ because of **R**1. Then, it is "captured" by $\mathcal{S}(c_{\sigma_\mu}, a_{\eta_{\sigma_\mu}})$ and cannot reach a' without passing through $\mathcal{S}(\overline{x}, a_{\eta_{\sigma_{\pi+1}}})$. Therefore, if it does not touch $\mathcal{S}(b_0, \overline{x})$, it must pass through $\mathcal{S}(\overline{x}, a_{\eta_{\pi+1}})$. If $\mathcal{R}(b_{\pi+1}, a') = \mathcal{S}(b_{\pi+1}, a')$, it satisfies (1) and (2) in the theorem statement. Otherwise, we can choose $\mathcal{S}(b_{\pi+1}, a'')$, where $\mathcal{R}(b_{\pi+1}, a')$ relies on a''. In both cases, a' and $a_{\eta_{\sigma_{\pi+1}}}$ are in $\{a_0, \ldots, a_{p+1}\}$, which implies that $a_{l_{\pi+1}}, \ldots, a_{r_{\pi+1}}$ are within $\{a_0, \ldots, a_{p+1}\}$, if they exist. Hence, (3) is proved.

2. $b_{\pi+1}$ is in $\mathcal{X}$. Figure 18 shows an example.

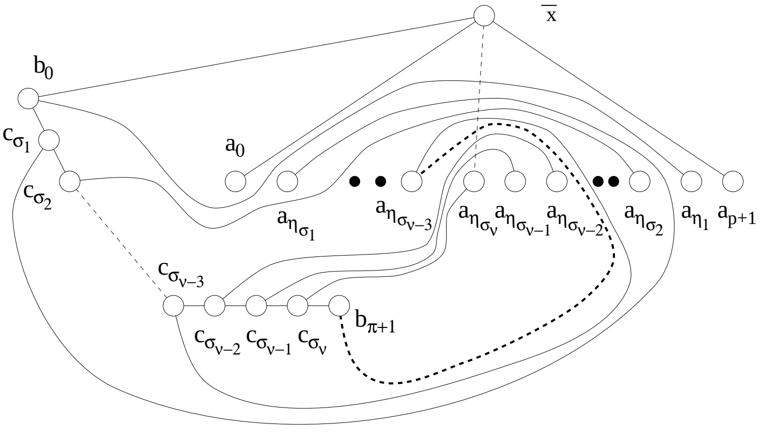

Figure 18: The case for $\mu = \nu - 3$, *i.e.*, $a_{\eta_{\sigma_{\nu-3}}}$ and $b_{\pi+1}$ are in different regions.

We define $\mathcal{X}_\mu$, $\mu \in [1, \nu]$, to be a minimal region that contains the paths $\mathcal{S}(c_{\sigma_\mu}, a_{\eta_{\sigma_\mu}})$, $\mathcal{S}(c_{\sigma_{\mu+1}}, a_{\eta_{\sigma_{\mu+1}}})$,..., $\mathcal{S}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$, the subpath of $\mathcal{P}_\pi$ starting from c_{σ_ν}, and ending at c_{σ_μ}, and the shortest line joining $a_{\eta_{\sigma_\mu}}, a_{\eta_{\sigma_{\mu+1}}}, \ldots, a_{\eta_{\sigma_\nu}}$, but does not contain the paths $\mathcal{S}(\overline{x}, a_0)$, $\mathcal{S}(\overline{x}, a_{p+1})$, $\mathcal{S}(b_0, \overline{x})$ (excluding b_0), $\mathcal{S}(c_{\sigma_1}, a_{\eta_{\sigma_1}})$,..., $\mathcal{S}(c_{\sigma_{\mu-1}}, a_{\eta_{\sigma_{\mu-1}}})$, and the suffix of $\mathcal{P}_\pi$ starting from $c_{\sigma_{\mu-1}}$. Obviously, $\forall \mu \in [1, \nu]$, $\mathcal{X}_\mu \subset \mathcal{X}$.

We choose the greatest μ such that $b_{\pi+1}$ is inside $\mathcal{X}_\mu$. Note that $a_{\eta_{\sigma_\mu}}$ and $b_{\pi+1}$ are in different regions which are surrounded by $\mathcal{S}(b_0, \overline{x})$, $\mathcal{S}(\overline{x}, a_{\eta_{\sigma_\nu}})$, $\mathcal{S}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$ and the suffix of $\mathcal{P}_\pi$ starting at c_{σ_ν}.

Consider $\mathcal{S}(b_{\pi+1}, a_{\eta_{\sigma_\mu}})$. If it touches $\mathcal{S}(c_{\sigma_\mu}, a_{\eta_{\sigma_\mu}})$, it is done. If it touches neither

> $\mathcal{S}(c_{\sigma_\mu}, a_{\eta_{\sigma_\mu}})$ nor $\mathcal{S}(\overline{x}, a_{\eta_{\pi+1}})$, it will then either cut $\mathcal{S}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$ or $\mathcal{S}(c_{\sigma_{\mu'}}, c_{\sigma_{\mu'-1}})$, $\nu \leq \mu' \leq \mu$. The former violates **R**1, and the latter implies that it cannot cut $\mathcal{S}(c_{\sigma_{\mu'-1}}, a_{\eta_{\sigma_{\nu-1}}})$, due to **R**1 again. Then, $\mathcal{S}(b_{\pi+1}, a_{\eta_{\sigma_\mu}})$ will cut $\mathcal{S}(\overline{x}, a_{\eta_{\pi+1}})$ twice and violate **R**3. A contradiction.
>
> Therefore, $\mathcal{S}(b_{\pi+1}, a_{\eta_{\sigma_\mu}})$ must pass through one ancestor of every node in $[f^{\pi+1}_{\min}, f^{\pi+1}_{\max}]$. Cases (1) and (2) are proved if we set $a_{\rho_{\pi+1}}$ to be $a_{\eta_{\sigma_\mu}}$. The positions of $a_{\eta_{\sigma_{\pi+1}}}$ and $a_{\eta_{\pi+1}}$ imply that $a_{l_{\pi+1}}, \dots, a_{r_{\pi+1}}$ are within $\{a_0, \dots, a_{p+1}\}$, if they exist. Then, case (3) is true.

Note that we have just shown the existence of a path for case (2). Actually, we will choose the path with minimum gap size in the algorithm.

Now, we prove Case (4) of the theorem statement. We have just proved Case (3), which is $\{a_{l_{\pi+1}}, \dots, a_{r_{\pi+1}}\} \subset \{a_1, \dots, a_p\}$. Hence, since the paths, $\mathcal{S}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$, $\mathcal{S}(c_{\sigma_{\nu-1}}, a_{\eta_{\sigma_{\nu-1}}}) \cdots \mathcal{S}(c_{\sigma_1}, a_{\eta_{\sigma_1}})$, will not touch each other, they divide the set $\{a_{l_{\pi+1}}, \dots, a_{r_{\pi+1}}\}$ into at most $\nu + 1$ partitions which are not in the same region as $b_{\pi+1}$, where the region concerned here is bounded by $\mathcal{S}(b_0, \overline{x})$, $\mathcal{S}(\overline{x}, a_{\eta_{\sigma_\nu}})$, $\mathcal{S}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$, and the suffix of $\mathcal{P}_{\pi+1}$ starting at c_{σ_ν}. There is at most one partition which is in the same region as $b_{\pi+1}$. For this kind of nodes, we can find a leader $a_{\eta_{\pi+2}}$ and the path $\mathcal{S}(b_{\pi+1}, a_{\eta_{\pi+2}})$ is in the same region as $b_{\pi+1}$. Hence, $\mathcal{G}(b_{\pi+1}, a_{\eta_{\pi+2}}) \subset \mathcal{G}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$. By **R**1, $a_{\eta_{\sigma_\nu}} \neq a_{\eta_{\pi+2}}$. Case (4) is done.

Consider a partition which is not in the same region as $b_{\pi+1}$ and is bounded by $\mathcal{S}(c_{\sigma_\nu}, a_{\eta_{\sigma_\nu}})$, $\mathcal{S}(\overline{x}, a_0)$ and $\mathcal{S}(\overline{x}, a_{p+1})$. The shortest path from $b_{\pi+1}$ to any leader element $a_{\eta_{\pi+2}}$ in the partition will pass through b_0 because of **R**1. Then, it is out of the scope of Case (4).

Consider a partition which is not in the same region as $b_{\pi+1}$ and is bounded by $\mathcal{S}(c_{\sigma_\mu}, a_{\eta_{\sigma_\mu}})$ and $\mathcal{S}(c_{\sigma_{\mu-1}}, a_{\eta_{\sigma_{\mu-1}}})$, where $1 < \mu \leq \nu$. The shortest path from $b_{\pi+1}$ to any leader element $a_{\eta_{\pi+2}}$ in the partition will cut $\mathcal{S}(c_{\sigma_\mu}, c_{\sigma_{\mu-1}})$ at $c_{\pi+2}$, because of **R**1. Again, due to **R**1, $\mathcal{G}(c_{\pi+2}, a_{\eta_{\pi+2}}) \subset \mathcal{G}(c_{\sigma_{\mu-1}}, a_{\eta_{\sigma_{\mu-1}}})$. We need to prove that $a_{\eta_{\sigma_{\mu-1}}} \neq a_{\eta_{\pi+2}})$. Assume the contrary. By **R**1, $\mathcal{S}(c_{\pi+2}, a_{\eta_{\pi+2}})$ is a subpath of $\mathcal{S}(c_{\sigma_{\mu-1}}, a_{\eta_{\sigma_{\mu-1}}})$. If $a_{\eta_{\sigma_{\mu-1}}}$ is in the same region as $b_{\pi+1}$, then it is a contradiction because the partition (of $\{a_{l_{\pi+1}}, \dots, a_{r_{\pi+1}}\}$) is not in the same region of $b_{\pi+1}$. If $a_{\eta_{\sigma_{\mu-1}}}$ is not in the same region as $b_{\pi+1}$, by the way that we choose $a_{\rho_{\pi+1}}$, the partition cannot include $a_{\eta_{\sigma_{\mu-1}}}$ if $a_{\rho_{\pi+1}} = a_{\eta_{\sigma_{\mu-1}}}$. If $a_{\rho_{\pi+1}} \neq a_{\eta_{\sigma_{\mu-1}}}$, $a_{\rho_{\pi+1}}$ must reside between $a_{\eta_{\sigma_{\mu-1}}}$ and $a_{\eta_{\sigma_\nu}}$, exclusively, and therefore, the partition cannot include $a_{\eta_{\sigma_{\mu-1}}}$, either. Hence, $a_{\eta_{\sigma_{\mu-1}}} \neq a_{\eta_{\pi+2}})$. Case (4) is done.

If $c_{\pi+1} = b_0$, by induction assumption (3), $\{a_{l_\pi}, \dots, a_{r_\pi}\} \subset \{a_1, \dots, a_p\}$. According to the algorithm, we need to divide $\{a_{l_\pi}, \dots, a_{r_\pi}\}$ into groups, and each group has a group leader. If a group leader $a_{\eta_{\pi+1}}$ is outside the set $\{a_1, \dots, a_p\}$, by **C**2, $\mathcal{R}(b_0, a_0)$ or $\mathcal{R}(b_0, a_{p+1})$ will rely on $a_{\eta_{\pi+1}}$. This is a contradiction because $\mathcal{R}(b_0, a_0) = \mathcal{S}(b_0, a_0)$ and $\mathcal{R}(b_0, a_{p+1}) = \mathcal{R}(b_0, a_{p+1})$. Hence, $a_{\eta_{\pi+1}}$ is inside the set $\{a_1, \dots, a_p\}$, implying that $\mathcal{G}(c_{\pi+1}, a_{\eta_{\pi+1}}) \subset \{a_1, \dots, a_p\}$. We then borrow the proof of Lemmas 4 and 5, and replace

b_1 by $b_{\pi+1}$, η_1 by $\eta_{\pi+1}$, ρ_1 by $\rho_{\pi+1}$, l_1 by $l_{\pi+1}$, r_1 by $r_{\pi+1}$, b_2 by $b_{\pi+2}$, c_2 by $c_{\pi+2}$, $[f^1_{\min}, f^1_{\max}]$ by $[f^{\pi+1}_{\min}, f^{\pi+1}_{\max}]$. Hence, the theorem is proved. □

Theorem 4 *The function call $Relax(b_0, a_{p+1}, a_1, a_p)$ will terminate.*

Proof: Recall that Definition 16 gives a definition for $\mathcal{P}_\zeta$ and c_ζ, $\forall\zeta \in [1, \lambda+1]$. We construct a tree $\mathcal{T}$ rooted at 0. The internal nodes and leaves of $\mathcal{T}$ are from $[1, \lambda + 1]$. There are three types of edges. The first type has only one instance $(0, 1)$. The second type has (ζ', ζ), $\zeta \in [2, \lambda + 1]$, where ζ' is the greatest number such that $1 \le \zeta' < \zeta$, $c_{\zeta'}$ is a node on $\mathcal{P}_{\zeta-1}$, and $\mathcal{G}(c_\zeta, a_{\eta_\zeta}) \subset \mathcal{G}(c_{\zeta'}, a_{\eta_{\zeta'}})$. The third type has $(0, \zeta)$, $\zeta \in [1, \lambda + 1]$, where $c_\zeta = b_0$, and $\forall\zeta' < \zeta$, $[c_{\zeta'}$ is a node on $\mathcal{P}_{\zeta-1}] \Rightarrow [\mathcal{G}(c_\zeta, a_{\eta_\zeta}) \not\subset \mathcal{G}(c_{\zeta'}, a_{\eta_{\zeta'}})]$.

The tree $\mathcal{T}$, indeed, records the recursion call chain from the call $Relax(b_0, a_{p+1}, a_1, a_p)$ to $Relax(b_\lambda, a_{\rho_\lambda}, a_{l_\lambda}, a_{r_\lambda})$. If ζ is a leave, $\zeta \in [1, \lambda]$, then the simple path from 0 to ζ records $\mathcal{P}_\zeta$. That means, any $\mu \ne 0$ on the path will imply c_μ on $\mathcal{P}_\zeta$. We called this simple path $\mathcal{SP}$. By the definition of c_ζ, $\zeta \in [1, \lambda + 1]$, in Definition 16 and Case (4) of Theorem 3, for any μ, μ' on $\mathcal{SP}$, excluding 0, such that $\mu < \mu'$, $\mathcal{G}(c_{\mu'}, a_{\eta_{\mu'}})$ is a proper subset of $\mathcal{G}(c_\mu, a_{\eta_\mu})$ because $a_{\eta_\mu} \notin \mathcal{G}(c_{\mu'}, a_{\eta_{\mu'}})$. Hence, the length of $\mathcal{SP}$ is no more than the number of nodes in $\{a_1, \ldots, a_p\}$. In other words, the depth of $\mathcal{T}$ is bounded by $O(n)$.

We now want to find the bound for the degrees of each internal node. We want to prove that for every two distinct children of a parent, ζ' and ζ, if $c_{\zeta'} = c_\zeta$, then $a_{\eta_{\zeta'}} \ne a_{\eta_\zeta}$.

Assumption 1 *(For contradiction) There exists a parent in $\mathcal{T}$ with two distinct children ζ and ζ' where ζ' is the smallest child of its parent and ζ is the smallest sibling of ζ' such that $\zeta' < \zeta$ and $c_\zeta = c_{\zeta'}$, but $a_{\eta_{\zeta'}} = a_{\eta_\zeta}$.*

Without loss of generality, we choose a topmost parent satisfying Assumption 1. In other words, $c_{\zeta'}$ is firstly added to $\mathcal{P}_{\zeta'}$ by $\mathcal{S}(b_{\zeta'}, a_{\eta_{\zeta'}})$, and perhaps it is re-visited many times, but the first re-visit is by $\mathcal{S}(b_\zeta, a_{\eta_\zeta})$ where $a_{\eta_\zeta} = a_{\eta_{\zeta'}}$.

Consider the nodes ζ', $\zeta' + 1$, ..., ζ in $\mathcal{T}$. $\forall\mu \in [\zeta', \zeta]$, if $c_\mu = c_{\zeta'}$, then $a_{\eta_\mu} \ne a_{\eta_{\zeta'}}$. $\forall\mu \in [\zeta', \zeta]$, $\mathcal{P}_\mu$ contains $c_{\zeta'}$. Assume for contradiction that $\exists\mu \in [\zeta', \zeta]$ such that $\mathcal{P}_\mu$ does not contain $c_{\zeta'}$. When $c_{\zeta'} = c_\zeta$ appears in $\mathcal{P}_\zeta$, ζ will have a greater parent than ζ'. A contradiction. Therefore, if $c_\mu \ne c_{\zeta'}$, c_μ cannot be a point on $\mathcal{P}_{\zeta'}$; otherwise, $\mathcal{P}_\mu$ will not contain $c_{\zeta'}$. In other words, μ is a descendant of ζ', or some $\mu' \in [\zeta' + 1, \mu - 1]$, where $c_{\mu'} = c_{\zeta'}$.

Claim: $\zeta' < \zeta - 1$. Reason: Since $\zeta' < \zeta$, we assume that $\zeta' = \zeta - 1$. Contradiction will be easily reached by considering that a_{η_ζ} must be located between $a_{\eta_{\zeta-1}}$ and $a_{\rho_{\zeta-1}}$, exclusively. ♢

Claim: $\zeta - 1$ is not a descendant of ζ'. Reason: Assume the contrary that $\zeta - 1$ is a descendant of ζ'. Recall that the node $c_{\zeta'}$ is on $\mathcal{P}_{\zeta-1}$ and $a_{\eta_{\zeta-1}} \ne a_{\eta_{\zeta'}}$. By Theorem 3, $b_{\zeta-1}$ and

$a_{\rho_{\zeta-1}}$ are in different regions which are separated by $\mathcal{S}(b_0, \overline{x})$, $\mathcal{S}(\overline{x}, a_{\eta_{\zeta-1}})$, $\mathcal{S}(c_{\zeta-1}, a_{\eta_{\zeta-1}})$, and the suffix of $\mathcal{P}_{\zeta-1}$ starting from $c_{\zeta-1}$. The node $a_{\eta_{\zeta'}}$ $(= a_{\eta_\zeta})$ is in different region from $b_{\zeta-1}$. The node $a_{\eta_{\zeta'}}$ is in the gap between $a_{\eta_{\zeta-1}}$ and $a_{\rho_{\zeta-1}}$, exclusively. Consider the path $\mathcal{S}(b_\zeta, a_{\eta_\zeta})$ $(= \mathcal{S}(b_\zeta, a_{\eta'_\zeta}))$. It passes through $b_{\zeta-1}$ and then c_ζ $(= c_{\zeta'})$. Then, $\mathcal{G}(b_{\zeta-1}, a_{\eta_{\zeta'}})$ contains $a_{\eta_{\zeta-1}}$ because $\mathcal{G}(c_{\zeta-1}, a_{\eta_{\zeta-1}}) \subset \mathcal{G}(c_{\zeta'}, a_{\eta_{\zeta'}})$, by the construction of $\mathcal{T}$. Hence, $a_{\eta_{\zeta'}}$ is a better choice than $a_{\rho_{\zeta-1}}$. Figure 19 shows an example. A contradiction. $\diamondsuit$

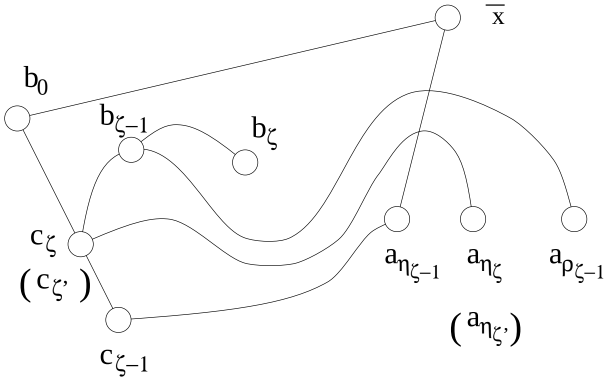

Figure 19: Interaction of two shortest paths violates **R**3.

Therefore, either $c_{\zeta-1} = c_{\zeta'}$ or $\zeta - 1$ is a descendant of $\mu \in [\zeta'+1, \zeta-1]$ such that $c_\mu = c_{\zeta'}$. Hence, we have some groups of $c_{\zeta'}$ on the minimal range on X that contains $\mathcal{G}(c_{\zeta'}, a_{\eta_{\zeta'}})$, $\mathcal{G}(c_{\zeta'+1}, a_{\eta_{\zeta'+1}})$,..., $\mathcal{G}(c_\zeta, a_{\eta_\zeta})$. Without loss of generality, we assume there is a group at the left of $\mathcal{G}(c_{\zeta'}, a_{\eta_{\zeta'}})$. We choose the leftmost one. Suppose it is a group that contains a_{η_μ} and $c_\mu = c_{\zeta'}$, where $\mu \in [\zeta'+1, \zeta-1]$. Let κ and τ be the smallest and largest number in $[\zeta'+1, \zeta-1]$, respectively, such that $\forall \iota \in [\kappa, \tau]$, $\mathcal{G}(c_\iota, a_{\eta_\iota}) \subset \mathcal{G}(c_\mu, a_{\eta_\mu})$. Obviously, $\mu \in [\kappa, \tau]$.

Since $\mathcal{G}(c_{\kappa-1}, a_{\eta_{\kappa-1}}) \not\subset \mathcal{G}(c_\mu, a_{\eta_\mu})$ but $\mathcal{G}(c_\kappa, a_{\eta_\kappa}) \subset \mathcal{G}(c_\mu, a_{\eta_\mu})$ and $\mathcal{G}(c_\mu, a_{\eta_\mu})$ is the leftmost group, $b_{\kappa-1}$ is outside the region which contains $\mathcal{G}(c_\mu, a_{\eta_\mu})$ and a_{η_κ}, where the region boundary is $\mathcal{S}(b_0, \overline{x})$, $\mathcal{S}(\overline{x}, a_{\eta_{\kappa-1}})$, $\mathcal{S}(c_{\kappa-1}, a_{\eta_{\kappa-1}})$, and the suffix of $\mathcal{P}_{\kappa-1}$ starting from $c_{\kappa-1}$. Applying similar argument to b_τ, b_τ and $b_{\kappa-1}$ must be in different regions. Like the argument for $\mathcal{P}_\tau$, there is a (not necessarily simple) path from $b_{\kappa-1}$ to b_τ. Suppose $\kappa < \tau$. There exists a $\tau' \in [\kappa, \tau-1]$ such that $b_{\tau'}$ and $b_{\tau'+1}$ are in different regions. The path $\mathcal{S}(b_{\tau'+1}, a_{\eta_{\tau'+1}})$ cannot cut any point in $\mathcal{P}_\mu$, excluding c_μ; otherwise, it will violate the fact that $\tau'+1$ is a descendant of μ. Therefore, the path $\mathcal{S}(b_{\tau'+1}, a_{\eta_{\tau'+1}})$ has two cutting points with either $\mathcal{S}(c_{\mu'}, a_{\eta_{\mu'}})$ or $\mathcal{S}(c_{\kappa-1}, a_{\eta_{\kappa-1}})$. Then, **R**3 is violated. Therefore, Assumption 1 is false.

Suppose $\kappa = \tau = \mu$. We simply replace the value τ' by $\mu - 1$ and $\tau' + 1$ by μ. The above argument will lead to the same conclusion that Assumption 1 is false.

Hence, a parent has at most $O(np) = O(n^2)$ children. It is because, firstly, for any two of them ζ and ζ' that are distinct and such that $\zeta' < \zeta$, we have $[c_\zeta = c_{\zeta'}] \Rightarrow [a_{\eta_\zeta} \neq a_{\eta_{\zeta'}}]$.

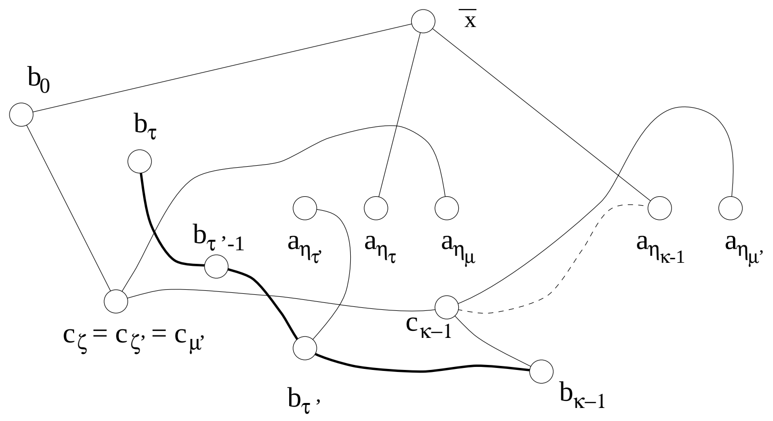

Figure 20: Interaction of two shortest paths violates **R**3.

Second, there are at most $O(n)$ different children for each parent.

Therefore, the size of $\mathcal{T}$ is bounded by $O(n^{2n})$ and the recursion call $Relax(b_0, a_{p+1}, a_1, a_p)$ will terminate. □

Theorem 5 *After each implementation of Step 5.1.8, all conditions* **C**1 *to* **C**4 *are true.*

Proof: We assume that the conditions are true before the execution of Step 5.1.8. Consider the situation after the execution. Consider **C**1. We force some routing from b_λ to follow the first edge of a shortest path. Hence, the first part of a relaxed path is a shortest one. The second part of a relaxed path is also a relaxed path because **C**1 is true before this step.

Consider **C**2. The relaxed path will then rely on $a_{\rho_{\lambda+1}}$ if it is a representative; otherwise, it can rely on the representative which the $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$ is relying on. Hence, a representative exists. Suppose $a_{\rho_{\lambda+1}}$ is the representative. We deal with all nodes in $[f^{\lambda+1}_{\min}, f^{\lambda+1}_{\max}]$ in Step 5.1.8 and all nodes in $\{a_{l_{\lambda+1}}, \ldots, a_{r_{\lambda+1}}\}$ in Step 5.1.7.2, and hence, for all node x in between $a_\triangle$ and $a_{\rho_{\lambda+1}}$, $\mathcal{R}(b_\lambda, x)$ will rely on $a_{\rho_{\lambda+1}}$. Suppose $a_{\rho_{\lambda+1}}$ is not the representative. Let a_ρ be the representative. By similar argument, we have for all node x in between $a_\triangle$ and $a_{\rho_{\lambda+1}}$, $\mathcal{R}(b_\lambda, x)$ will rely on a_ρ. For all node x between $a_{\rho_{\lambda+1}}$ and a_ρ, $\mathcal{R}(b_\lambda, x)$ has already relied on a_ρ, because **C**2 is true before this step 5.1.8.

Consider **C**3. By the way to choose $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}})$, it is true.

Consider **C**4. The interval label does not change until the step. After the step, the labels on the edge for $\mathcal{S}(b_{\lambda+1}, a_{\rho_{\lambda+1}}$ will not increase because the new routing follows the labels of the representative. For the other edges, if a routing is taken out, it will happen to a representative and all its followers. That means, the whole label is taken out completely. In that case, the number of labels decreases. □

Now, $Relax(b_0, a_{p+1}, a_1, a_p)$ has been done already. In the second round, we will re-walk through the nodes from $\overline{x}$ to u, exclusively, along $\mathcal{S}(u, \overline{x})$ reversely. If another b_0

exists, we repeat relaxation of other routing paths, until all are done. Finally the elements in $\{x \in X | x_k \leq x \leq x_j\}$ can be grouped into one interval label in the edge (u, v). It means that we can reduce the number of interval labels of (u, v) by at least one.

4.2 Setup for Path Relaxation

This section provides the pre-requisite for path relaxation discussed in Section 4.1.

Let w be any common ancestor of $x_1, x_2, \ldots, x_\gamma$. Since w can be R, its existence is guaranteed. By the definition of X, u cannot be any descendant of w, and hence, it cannot be inside the "triangle" formed by the shortest paths from w to x_1, to x_γ, and the virtual horizontal bar linking all elements on that level. We call them the left boundary, right boundary, and the bottom, respectively. Figure 21 shows an example.

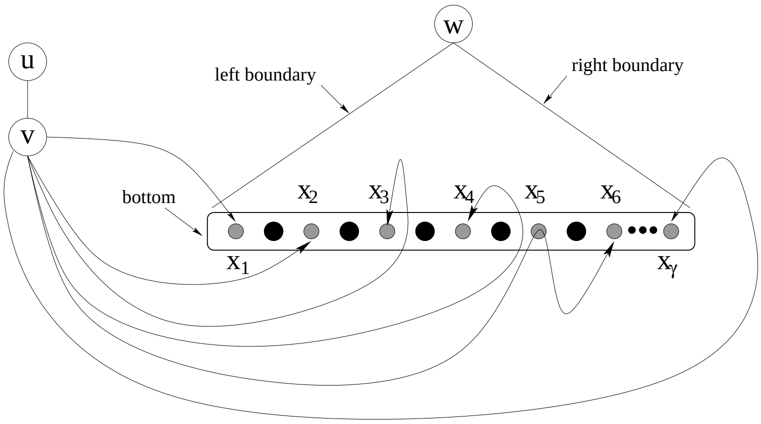

Figure 21: Many ways to reach x_i's.

Definition 17 *Given that $a, b \in [x_1, x_\gamma]$, where $a \prec b$. (That means, a is on the left and b is on the right.) The left boundary $LB(a, b)$ is the shortest path from w to a. The right boundary $RB(a, b)$ is the shortest path from w to b. The bottom $B(a, b)$ is the virtual line linking a, b and all nodes in between.*

$\forall i \in [1, \gamma]$, since $x_i \in L_*(u, v)$, we consider $\mathcal{S}(v, x_i)$ instead of $\mathcal{S}(u, x_i)$. Between adjacent x_i and x_{i+1}, $i \in [1, \gamma - 1]$, there should be an interval which does not belong to $L_*(u, v)$. Let z_i be the representative of this interval (**C**2). The ordering of the nodes in the $B(x_1, x_\gamma)$ is $x_1, z_1, x_2, z_2, \ldots, x_\gamma$, from left to right. Note that $\forall i \in [1, \gamma], j \in [1, \gamma - 1]$, $\mathcal{S}(v, x_i)$ cannot pass through z_j because of rule **R**1.

Definition 18 *Given $a, b \in [x_1, x_\gamma]$, and $a \prec b$. A routing path $\mathcal{S}(v, x_i)$, $i \in [1, \gamma]$, is r-ripple in $B(a, b)$ if it crosses or touches $B(a, b)$ r times and each time it passes through the ancestor of x_j or*

z_k, or it touches x_j, for some $x_j, z_k \in [a,b]$, where $j \neq i$, before arriving at its destination x_i which is also in $B(a,b)$.

Similar definition applies to $\mathcal{S}(u, z_i)$, $i \in [1, \gamma - 1]$. The paths $\mathcal{S}(v, x_4)$ and $\mathcal{S}(v, x_6)$ in Figure 21 are 1-ripple in $B(x_1, x_\gamma)$ because they both cross or touch the bottom and $\mathcal{S}(v, x_4)$ passes through the ancestor of z_4, $\mathcal{S}(v, x_6)$ touches x_5 before arriving at their destinations. $\mathcal{S}(v, x_1)$ and $\mathcal{S}(v, x_\gamma)$ are 0-ripple in $B(x_1, x_\gamma)$ because they never cross or touch the bottom until the end. $\mathcal{S}(v, x_2)$ is also 0-ripple because it never passes through x_j or any ancestors of x_j and z_k, where $j \neq 2$. Similar to $\mathcal{S}(v, x_3)$. Obviously, $\forall$ r-ripple paths, $r \leq D - 1$. Intuitively, every routing path cannot have more than $D - 1$ ripples. Note that for the convenience of discussion, we focus the ripples on x_i's and z_j's, which is different from the common sense that all nodes should be concerned.

Definition 19 *Given the nodes at $B(a,b)$ and their ordering as $x_1, z_1, x_2, z_2, \ldots, x_\gamma$ from left to right. A node x_i, $i \in [2, \gamma - 1]$ is covered above if the only way for $\mathcal{S}(u, x_i)$ to reach x_i is from $B(z_{i-1}, z_i)$. It is covered below if the only way for $\mathcal{S}(u, x_i)$ to reach x_i is from $LB(z_{i-1}, z_i)$ or $RB(z_{i-1}, z_i)$.*

Similar definitions apply to z_i's, $i \in [1, \gamma - 1]$.

Lemma 6 *There are no more than five contiguous nodes at $B(z_1, z_{\gamma-1})$ being covered above.*

Proof: Assume the contrary that there are five contiguous nodes being covered above. Without loss of generality, let the nodes be $x_i, z_i, x_{i+1}, z_{i+1}, x_{i+2}$, and z_{i+2} where $i \in [2, \gamma - 3]$. By rule **R**1, $\mathcal{S}(v, x_i)$, $\mathcal{S}(u, z_i)$, $\mathcal{S}(v, x_{i+1})$, $\mathcal{S}(u, z_{i+1})$, $\mathcal{S}(v, x_{i+2})$, and $\mathcal{S}(u, z_{i+2})$ do not cut/touch one another.

There are three faces. One is formed by $B(x_i, z_{i+2})$, $\mathcal{S}(v, x_i)$ and $\mathcal{S}(v, x_{i+1})$; another one is by $B(x_i, z_{i+2})$, $\mathcal{S}(v, x_{i+1})$ and $\mathcal{S}(v, x_{i+2})$; and the last one is by $B(x_i, z_{i+2})$, $\mathcal{S}(v, x_{i+2})$ and $\mathcal{S}(v, x_i)$.

Consider the case that two of z_i, z_{i+1} and z_{i+2} are in different faces. Without loss of generality, we choose z_i and z_{i+1}. If u resides in the face with z_i, $\mathcal{S}(u, z_{i+1})$ can reach z_{i+1} from the top of $B(x_{i+1}, x_{i+2})$ and passes through $LB(x_{i+1}, x_{i+2})$ or $RB(x_{i+1}, x_{i+2})$. It implies that z_{i+1} cannot be covered above (Figure 22). Similar to the case that u resides in other positions. Hence, all of z_i, z_{i+1} and z_{i+2} must be in the same face.

Let F be the face that contains z_i, z_{i+1} and z_{i+2}. F is formed by the $B(x_i, z_{i+2})$, and some of $\mathcal{S}(v, x_i)$, $\mathcal{S}(v, x_{i+1})$, and $\mathcal{S}(v, x_{i+2})$. None of the paths $\mathcal{S}(v, x_i)$, $\mathcal{S}(v, x_{i+1})$, and $\mathcal{S}(v, x_{i+2})$ can divide F such that z_i, z_{i+1} and z_{i+2} are not in the same partition.

$\mathcal{S}(v, x_i)$ must have ripple(s) in $B(x_i, z_{i+2})$. Consider the contrary that $\mathcal{S}(v, x_i)$ is 0-ripple in $B(x_i, z_{i+2})$ (Figure 23). No matter whether $\mathcal{S}(v, x_{i+1})$ has ripple(s), it will partition F

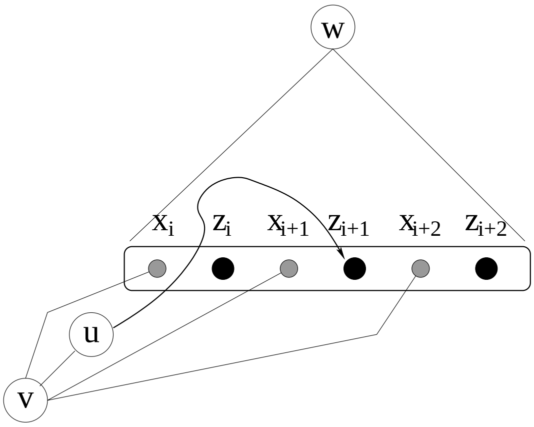

Figure 22: z_i and z_{i+1} are in different faces. A contradiction.

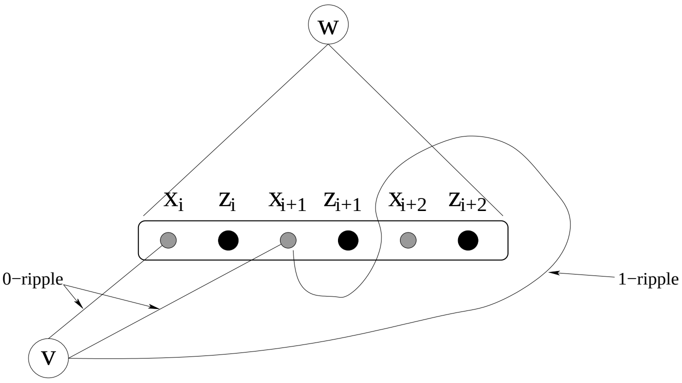

Figure 23: $\mathcal{S}(v, x_i)$ is 0-ripple. A contradiction.

such that z_i will be in different faces with z_{i+1}. A contradiction. Hence, $\mathcal{S}(v, x_i)$ must have ripple(s) in $B(x_i, z_{i+2})$. Recall that $\mathcal{S}(v, x_1)$ cannot divide F which contains z_i, z_{i+1} and z_{i+2}.

Similar argument will apply to $\mathcal{S}(u, z_{i+2})$ that this path cannot divide the face F' which contains x_i, x_{i+1} and x_{i+2}. Figure 24 shows an example. However, the two paths $\mathcal{S}(v, x_1)$ and $\mathcal{S}(u, z_{i+2})$ will cut each other and violating rule **R**1. □

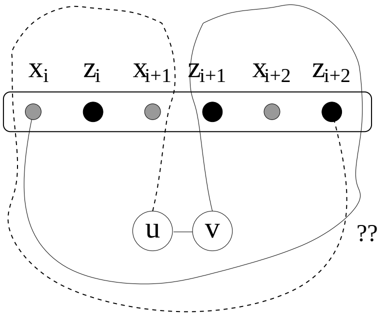

Figure 24: $\mathcal{S}(v, x_1)$ and $\mathcal{S}(u, z_{i+2})$ cut each other. A contradiction.

By similar techniques, we can derive Lemma 7.

Lemma 7 *There are no more than five contiguous nodes at $B(z_1, z_{\gamma-1})$ being covered below.*

Lemma 8 *Every node in $\{z_1, x_2, z_2, \ldots, z_{\gamma-1}\}$ is either bounded above or below.*

Proof: Assume the contrary that $\exists x_i$, $i \in [2, \gamma - 1]$, such that there are two choices for $\mathcal{S}(v, x_i)$ (Figure 25). Consider the routing path $\mathcal{S}(u, z_{i-1})$, u should reside in the face

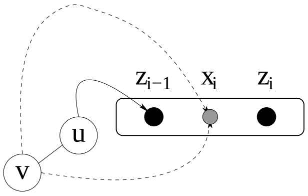

Figure 25: There are two choices for $\mathcal{S}(v, x_i)$. A contradiction.

bounded by $B(z_{i-1}, z_i)$, and the two possible $\mathcal{S}(v, x_i)$'s. Then, z_i is un-reachable by u unless rule **R**1 is violated. A contradiction. Hence, x_i is either covered above or below. □

Since $\gamma > 12D^2(D + 2)$, by Lemmas 6, 7 and 8, we have at least $2D^2(D + 2)$ pairs of x_i, z_i assumed being covered above and below, respectively, where $i \in [2, \gamma - 1]$, without loss of generality. For each z_i belonging to these pairs, $\mathcal{S}(u, z_i)$ can cut the $\mathcal{S}(w, x_{i+1})$ (*i.e.* from right) or $\mathcal{S}(w, x_i)$ (*i.e.* from left). We further narrow our scope to the pairs such that all $\mathcal{S}(u, z_i)$'s are from one side, say the left, and of the same length. Then, there are $D(D + 2)$

pairs such that each $\mathcal{S}(u, z_i)$ cuts $\mathcal{S}(w, x_i)$, and all $\mathcal{S}(u, z_i)$'s are of the same length, where $i \in [2, \gamma - 1]$. Let Z be the set containing the z_i's of these pairs, $i \in [1, \gamma - 1]$. If $\exists z_i, z_j \in Z$, $i \neq j$, such that $\mathcal{S}(u, z_i)$ and $\mathcal{S}(u, z_j)$ pass through a common ancestor of z_i, z_j, then it is done. We are now going to assume the contrary.

Assumption 2 *(for contradiction)* $\forall z_i, z_j \in Z$, *where* $i, j \in [1, \gamma - 1]$, $i \neq j$, $\mathcal{S}(u, z_i)$ *and* $\mathcal{S}(u, z_j)$ *do not pass through any common ancestor of* z_i, z_j.

Consider the case that $\exists z_i \in Z$ such that $\mathcal{S}(u, z_i)$ pass through the ancestors of z_{σ_1}, z_{σ_2}, $\ldots$, z_{σ_D}, where $z_{\sigma_j} \in Z$, $\sigma_j \in [1, \gamma - 1]$, $\sigma_j \neq i$, $\forall j \in [1, D]$. $|\mathcal{S}(u, z_i)|$ is bounded by D obviously. Excluding u and z_i, it can pass through at most $D - 1$ distinct ancestors of z_{σ_1}, z_{σ_2}, $\ldots$, z_{σ_D}. Then, there is at least two nodes $z_{\sigma_j}, z_{\sigma_k}$ having a common ancestor $\overline{z_\sigma}$ which is passed through by $\mathcal{S}(u, z_i)$. According to the initial choice of $\mathcal{S}(u, z_{\sigma_j})$ and $\mathcal{S}(u, z_{\sigma_k})$, they must both have a prefix $\mathcal{S}(u, \overline{z_\sigma})$; otherwise, rule **R**1 will be violated. Then, it is a contradiction to our Assumption 2 by considering $\mathcal{S}(u, z_{\sigma_j})$ and $\mathcal{S}(u, z_{\sigma_k})$. Therefore, $\forall z_i \in Z$, where $i \in [1, \gamma - 1]$, $\mathcal{S}(u, z_i)$ can pass through the ancestors of at most $D - 1$ nodes of Z.

Partition Z into disjoint subsets. If a subset contains exactly a node z_k, first, $\forall z_i \in Z$, $i \neq k$, $\mathcal{S}(u, z_i)$ does not pass through any ancestor of z_k; and second, $\mathcal{S}(u, z_k)$ does not pass through any ancestor of z_i, $\forall z_i \in Z$, $i \neq k$. If a subset Z' contains more than one node, first there exists one $z_i \in Z'$, such that $\mathcal{S}(u, z_i)$ passes through the ancestors of all nodes in $Z' - \{z_i\}$; second, $\forall z_i \in Z'$, $\mathcal{S}(u, z_i)$ does not pass through the ancestors of all nodes in $Z - Z'$; third, $\forall z_i \in Z - Z'$, $\mathcal{S}(u, z_i)$ does not pass through the ancestors of all nodes in Z'. The size of each subset is bounded by D and there are at least $D + 2$ disjoint subsets of Z.

Choose a node from each subset. So, we have at least $D+2$ nodes, which are denoted as $z_{\lambda_1}, z_{\lambda_2}, \ldots, z_{\lambda_{D+2}}$. We consider the routing from v to their adjacent nodes $x_{\lambda_1}, x_{\lambda_2}, \ldots, x_{\lambda_{D+2}}$. Recall that $z_{\lambda_1}, z_{\lambda_2}, \ldots, z_{\lambda_{D+2}}$ are covered below, and their routing paths from u must cut $\mathcal{S}(w, x_{\lambda_1}), \mathcal{S}(w, x_{\lambda_2}), \ldots, \mathcal{S}(w, x_{\lambda_{D+2}})$, respectively. They also belong to different subsets, and therefore, $\mathcal{S}(u, z_{\lambda_i})$ does not pass through the ancestors of z_{λ_j}, $\forall j \in [1, D + 2]$, $j \neq i$. Then, the routing paths $\mathcal{S}(v, x_{\lambda_1}), \mathcal{S}(v, x_{\lambda_2}), \ldots, \mathcal{S}(v, x_{\lambda_{D+1}})$ must come by cutting the right boundary—$\mathcal{S}(u, z_{\lambda_{D+2}})$ (Figure 26). There exist two paths $\mathcal{S}(v, x_{\lambda_i}), \mathcal{S}(v, x_{\lambda_j})$ such that they are of the same length, and $\mathcal{S}(v, x_{\lambda_i})$ passes through the ancestor of x_{λ_j}, $\overline{x_{\lambda_j}}$, where $i, j \in [1, D + 1]$. $\mathcal{S}(\overline{x_{\lambda_j}}, x_{\lambda_j})$ follows the tree edges down to x_{λ_j}. So, $\mathcal{S}(v, x_{\lambda_j})$ cannot pass through it because x_{λ_j} is covered above. $\mathcal{S}(\overline{x_{\lambda_j}}, x_{\lambda_j})$ is also no longer than $\mathcal{S}(\overline{x_{\lambda_j}}, x_{\lambda_i})$ because $|\mathcal{S}(\overline{x_{\lambda_j}}, x_{\lambda_i})|$ is the minimum distance from $\overline{x_{\lambda_j}}$ to any nodes in X. Hence,

$$\begin{aligned} |\mathcal{S}(v, x_{\lambda_j})| &= |\mathcal{S}(v, x_{\lambda_i})| \\ &= |\mathcal{S}(v, \overline{x_{\lambda_j}}| + |\mathcal{S}(\overline{x_{\lambda_j}}, x_{\lambda_i})| \\ &\geq |\mathcal{S}(v, \overline{x_{\lambda_j}}| + |\mathcal{S}(\overline{x_{\lambda_j}}, x_{\lambda_j})|. \end{aligned}$$

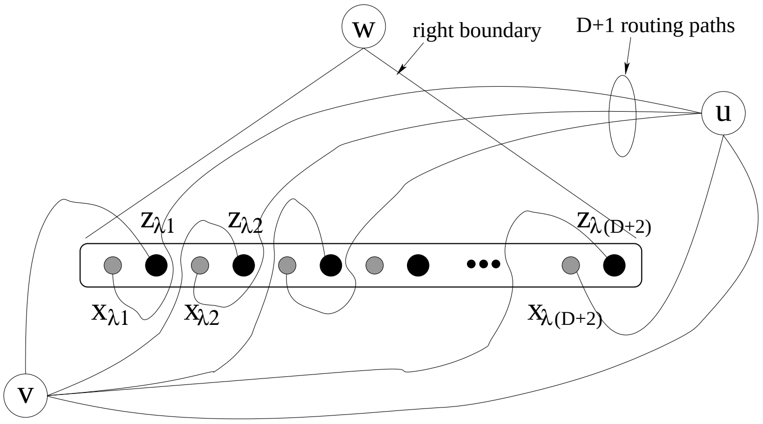

Figure 26: $\forall i \in [1, D+1]$, $\mathcal{S}(v, x_{\lambda_i})$ cuts the right boundary.

This is a contradiction to rule **R**2 on $\mathcal{S}(v, x_{\lambda_j})$ that the highest ancestor $\overline{x_{\lambda_j}}$ should be passed through. Hence, Assumption 2 is false and we have already proved the following theorem.

Theorem 6 *$\exists z_i, z_j \in Z$, $i \neq j$, such that $\mathcal{S}(u, z_i)$ and $\mathcal{S}(u, z_j)$ pass through a common ancestor of z_i, z_j.* □

4.3 Main Result—Combining Path Relaxation and its Setup

Theorem 7 *If there exists an edge (u, v) of u containing more than $12D^2(D+2)$ interval labels for routing to X, we can reduce the number of interval labels by one in each iteration such that the four conditions are kept true.*

Proof: If we assume that at least $2D^2(D+2)$ pairs of x_i and z_i are covered above and below, respectively, where $i \in [2, \gamma - 1]$ (on page 29), Theorem 6 guarantees the existence of $\mathcal{S}(u, z_i)$ and $\mathcal{S}(u, z_j)$ such that they pass through a common ancestor of z_i and z_j, where $i \neq j$. They must go from the same edge of u, say $(u, \hat{v})$, $\hat{v} \neq v$; otherwise rule **R**1 will be violated. Using the procedures of path relaxation, we can combine z_i, z_j and all nodes between them into one interval label of $(u, \hat{v})$ such that the four conditions are kept true. Note that although these two routing paths are not going through (u, v), there is at least one interval between z_i and z_j which belongs to (u, v) and is then taken out from (u, v) after path relaxation. The number of interval labels in (u, v) is then reduced by one.

On the other hand, if we assume that at least $2D^2(D+2)$ pairs of x_i and z_i are covered below and above, respectively, where $i \in [2, \gamma - 1]$, we will have two paths $\mathcal{S}(u, x_i)$ and $\mathcal{S}(u, x_j)$ such that they go through the edge (u, v) and pass through a common ancestor of x_i and x_j, where $i \neq j$. Path relaxation will certainly reduce the number of interval labels in (u, v) by one. □

5 An $O(D^3)$-IRS, for Graphs of $D \geq 3$

Theorem 8 *For all graphs of $D \geq 3$, for any constant $\alpha \in (0,1)$, there exists an $O(D^3)$-IRS with all routing paths no longer than $(1+\alpha)D$.*

Proof: (Outline) Consider the BFS tree. We divide the D levels into $\lceil \frac{1}{\alpha} \rceil$ layers and each layer contains $\lfloor \alpha D \rfloor$ levels. The top level of each layer is considered the anchor level. For each node in the anchor levels, we consider it a root of a subtree of the BFS tree. The subtree is formed by its descendents in its layers. If a node has more than one ancestor, choose the one connected by tree edges (not by fronds).

We label the nodes in the way that the nodes inside a subtree will form a contiguous interval. Like the previous labeling, we label the whole BFS tree layer by layer from top to bottom, and subtree by subtree from left to right in each layer.

Consider the labeling for the edges. We use the idea of shortest path from each node to the nodes in the anchor levels. It then follows by path relaxation and guarantee that the path lengths to all nodes in anchor levels are bounded by D. Inside each subtree, we follow the tree edges and use DFS labeling technique [8]. The last step is to replace an anchor node by the nodes of its subtree in all interval labels, except the ones for routings inside a subtree.

Each routing to a node at an anchor level takes at least D steps. Each routing to a node at a non-anchor level will pass through the root of the corresponding subtree at an anchor level of its layer, and takes $(1+\alpha)D$ steps in total.

Each layer needs $O(D^3)$ interval labels. Totally, we need $O(\frac{D^3}{\alpha}) = O(D^3)$ interval labels. □

6 A 6-IRS for Graphs of $D = 2$

We follow the procedures discussed in Section 3 to find the planar representation, and label the nodes by BFS (Figure 4), and initialize the edge labels by choosing the shortest paths. Recall that if more than one shortest path for a routing, we will choose the one that passes through the highest ancestor of the destination. For routing from R, two labels for each down edge are sufficient because there are only two levels. We now consider the routing paths from a node $u \neq R$ to each level. First, u's up edge should have a label for routing to R.

For each $i \in [1,2]$, we consider the routing to the nodes in i-th level. $\forall u \in V - \{R\}$, from u, there are three types of routing paths of length two.

1. The routing path passes through a node v at the i-th level.
   The node v will take one label. It is because shortest path should be chosen initially.

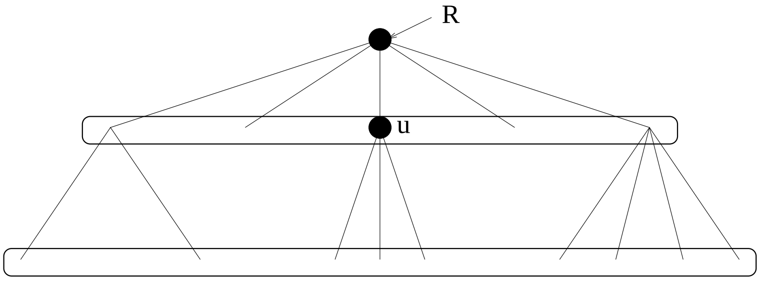

Figure 27: u is in the second level.

The node v divides the level into left and right parts. Consider the right part. We want to show that at most one more label will be needed in the edge (u, v).

Suppose two labels are needed. There are two nodes x_1 and x_2 belonging to the two labels, respectively. Between these two labels, there is at least one node y which disallows the two labels to combine. Figure 28 shows an example. Note that y is covered below. The only way for $\mathcal{S}(u, y)$ is from y's parent $\overline{y}$. The restriction $D = 2$ implies the existence of an edge $(u, \overline{y})$. The edge $(u, \overline{y})$ implies that $\overline{y}$ is also the parent of one of x_1 and x_2. Contradiction to rule **R**2 on $\mathcal{S}(u, x_1)$ or $\mathcal{S}(u, x_2)$ follows.

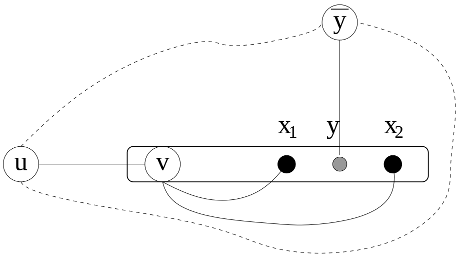

Figure 28: y is between x_1 and x_2.

Hence, one more label is enough for the right part, and one more label will be expected for the left. Totally, three labels for (u, v) are sufficient.

2. The routing path passes through a node v at the $i + 1$-th level, if any.
   The argument for this kind of routing path is similar to the above one, except that the node v will not be individually considered. Hence, two labels are sufficient.

3. The routing path passes through a node v at the $i - 1$-level.
   We are going to show that one label is enough. Assume two labels in the edge (u, v) are necessary. Let x_1 and x_2 be the nodes chosen from the two labels, respectively. Let y be the node between x_1 and x_2 and the routing to y is not through (u, v). According to rule **R**2, the choice of the $\mathcal{S}(u, y)$ implies that it needs only one step from

u to y. Figure 29 shows an example. We apply the technique of path relaxation and

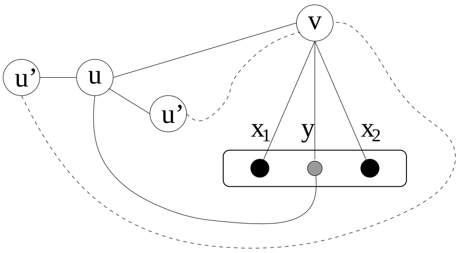

Figure 29: The routing to from u' to x_1 will be a contradiction.

force $\mathcal{R}(u,y)$ to use the edge (u,v). The length of $\mathcal{R}(u,y)$ is still bounded by $|\mathcal{S}(u,x_1)|$ or $|\mathcal{S}(u,x_2)|$. Suppose there is a node u' such that $|\mathcal{R}(u',y)| > D = 2$ after relaxation. When u' and x_1 are in different faces, contradiction follows if we consider the routing from u' to x_1. When u' and x_2 are in the same face, contradiction follows if we consider the routing from u' to x_2. Hence, one label is enough.

Theorem 9 *For all graphs of $D = 2$, there exists a 6-IRS with all routing paths no longer than D.*

Proof: Consider the routing paths of length two. Sum up the number of labels used in the cases for each $i \in [1,2]$. Item 1 implies that each of u's outgoing edges have at least three labels. Item 2 implies that u's fronds and down edges have two more labels. Item 3 implies that u's fronds and up edges have one more label. For routing to the root, all up edges have one more label. Totally, we need six labels. Consider the routing paths of length one. If it is not a proper subpath of any other paths, the edge will contain one label only; otherwise, it is discussed already in the above case. □

References

[1] N. Deo, *Graph Theory*, Prentice Hall, 1974.

[2] P. Fraigniaud and C. Gavoille, Interval Routing Schemes, *Algorithmica*, vol 21, pp.155–182, 1998.

[3] C. Gavoille, “On Dilation of Interval Routing”, *The Computer Journal*, 43(1), 1–7, 2000.

[4] C. Gavoille, “A Survey on Interval Routing”, *Theoretic Computer Science*, 245(2):217–253, 2000.

[5] C. Gavoille and E. Guévremont, “Worst Case Bounds for Shortest Path Interval Routing,” *Journal of Algorithms*, 27:1–25, 1998.

[6] R. Králˇovič, P. Ružička, D. Štefankovič, The Complexity of Shortest Path and Dilation Bounded Interval Routing, in: 3rd International Euro-Par Conference, C. Lengaur, M. Griebl, and S. Gorlatch, eds., vol. 1300 of Lectures Notes in Computer Science, Springer-Verlag, August 1997, 258–265.

[7] Kranakis, E., Krizanc, D. and Ravi, S.S. On Multi-Label Linear Interval Routing Schemes. *The Computer Journal*, **39**, 133–139.

[8] J. van Leeuwen and R.B. Tan, "Interval Routing", *The Computer Journal*, 30:298–307, 1987.

[9] P. Ružička, "A Note on the Efficiency of an Interval Routing Algorithm", *The Computer Journal*, 34:475–476, 1991.

[10] N. Santoro and R. Khatib, "Labelling and Implicit Routing in Networks", *The Computer Journal*, 28:5–8, 1985.

[11] M.D. May, P.W. Thompson, and P.H. Welch (eds.), *Networks, Routers and Transputers*, IOS Press, Amsterdam, 1993.

[12] S.S.H. Tse and F.C.M. Lau, "Lower Bounds for Multi-label Interval Routing", *Proc. 2nd International Colloquium on Structural Information & Communication Complexity (SIROCCO'95)*, 123–134, 1995.

[13] S.S.H. Tse and F.C.M. Lau, "More on the Efficiency of Interval Routing", *The Computer Journal*, Vol. 41, No. 4, 1998, 238–242.

[14] S.S.H. Tse and F.C.M. Lau, "Some Results on the Space Requirement of Interval Routing" *Proc. of 6th International Colloquium on Structural Information and Communication Complexity (SIROCCO 6)*, 264–279, 1999.