

# Minimum Wavelength Assignment for Multicast Traffic in All-Optical WDM Tree Networks

Anuj Rawat, Mark Shayman, Richard La and Steve Marcus

Department of Electrical and Computer Engineering  
University of Maryland, College Park, Maryland 20742  
Email: {anuj,shayman,hyongla,marcus}@umd.edu

**Abstract**—We study the problem of assigning wavelengths to a given set of multicast traffic requests with the objective of minimizing the number of wavelengths used per fiber. We assume that the underlying optical network is a tree and that all-optical networking paradigm is employed. First, we prove that the problem is NP hard even when the underlying network is a simple star or path. Since assigning wavelengths to a given set of unicast traffic requests on star and path networks is easy, this shows that the multicast wavelength assignment problem is fundamentally harder than the unicast scenario. Next we present GREEDY and SUBTREE-BASED: two simple deterministic algorithms for assigning wavelengths to a given set of multicast traffic requests when the underlying network is a tree with maximum node degree 3 and 4 respectively. Of the two wavelength assignment schemes, GREEDY is a  $\frac{5}{2}$ -approximation algorithm, and SUBTREE-BASED is an approximation algorithm with approximation ratio  $\frac{10}{3}$ , 3 and 2 for the cases when the underlying network is a tree with degree 4, 3 and 2, respectively.

## I. INTRODUCTION

Supporting multicast traffic in Wavelength Division Multiplexing (WDM) based optical networks has been an area of active research over the last few years. The reason for this is twofold: firstly, a lot of the current and upcoming applications such as HDTV, multimedia conferencing, distributed computing, etc., are inherently multicast in nature, and secondly, WDM based optical communication networks are capable of providing extremely high data transport capacities at remarkably low error rates. A consequence of this high data transfer rate is that the cost of employing electronic switching at the intermediate network nodes using the current technology is prohibitively high. Hence, it is prudent to switch traffic optically at the intermediate nodes and perform electro-optic (opto-electronic) conversion at only the source (destination) node. This is known as *all-optical* networking.

Multicast traffic requests can be supported in a WDM based all-optical network by constructing individual *light-trees* for each traffic request. A light-tree is a directed tree that is rooted at the source node of the traffic request it supports and spans all the destination nodes of the traffic request. In the absence of wavelength converters at the nodes, which is usually the case due to their high cost, light-trees must respect the *wavelength-continuity constraint*, i.e., a light-tree must use the same wavelength channel on all the optical fibers on which it is routed.

Constructing light-trees requires the network nodes be equipped with light splitters [1] and have tap-and-continue [2] capability. Light splitting is the ability to split the incoming light on a particular wavelength and switch the resulting streams onto multiple outgoing ports. This capability is required at nodes where the light-tree bifurcates. Tap-and-continue is the ability to tap a small amount of the incoming light on a particular wavelength and let the rest of the light go through. This capability is required at nodes that are the destinations for the multicast traffic request being supported by the light-tree, but are not the leaf nodes of the light-tree.

Routing and Wavelength Assignment (RWA) is a problem of extreme importance in WDM based optical networks. In the case of multicast traffic, RWA involves routing the light-trees (corresponding to the given set of multicast traffic requests) over the optical fiber links, and assigning wavelengths to them while maintaining wavelength-continuity. The number of light-trees that can simultaneously share an optical fiber is equal to the number of wavelengths that can be multiplexed onto the fiber. Since this is limited by the WDM technology, it is important to utilize the available wavelengths judiciously. Hence, an important optimization criteria for RWA is minimizing the total number of wavelengths required to support the given set of traffic requests. This is sometimes referred to as the cost of optics in the network.<sup>1</sup>

In this paper, we study the problem of RWA for multicast traffic with the objective of minimizing the number of wavelengths required, when the underlying fiber network is organized as a tree with each tree edge being a pair of antiparallel directed fiber links. Such a network is referred to as *bidirected tree*.

### A. Notation

Some basic notation that is used throughout the paper is introduced in this section. This is not a comprehensive description and we keep on introducing more notations as and when required in the paper.

We denote the cardinality of a finite set  $S$  by  $|S|$ . We denote the image of any mapping  $f : D \rightarrow R$ , restricted to some set  $S \subseteq D$ , by  $f(S)$ . Given a graph  $G$ , we denote the edge set by  $E_G$  and the vertex set by  $V_G$ . An edge between vertices

<sup>1</sup>Cost of electronics, as determined by the number of electronic components such as Line Terminating Equipments (LTEs) required in the network, is also an important optimization criteria and has been well studied.

$u, v \in V_G$  is denoted by  $\{u, v\}$ . Similarly, given a directed graph  $\vec{D}$ , we denote the set of directed edges by  $E_{\vec{D}}$  and the set of vertices by  $V_{\vec{D}}$ . For a pair of vertices  $u, v \in V_{\vec{D}}$ , a directed edge from  $u$  to  $v$  is denoted by  $(u, v)$ . The undirected graph obtained by replacing all the directed arcs of directed graph  $\vec{D}$  by undirected edges is denoted by  $\|\vec{D}\|$ . By  $\bar{G}$  we denote the complement of a given graph  $G$ , i.e.,  $V_{\bar{G}} = V_G$  and  $E_{\bar{G}} = \{\{u, v\} | u, v \in V_G \text{ and } \{u, v\} \notin E_G\}$ . A bidirected tree can be constructed from any given tree  $H$  by replacing all the edges of  $H$  by pairs of antiparallel directed edges. The bidirected tree thus generated is denoted by  $\vec{T}_H$ . The graph  $H$  is then referred to as the *host tree* of the bidirected tree  $\vec{T}_H$ . A rooted subtree  $\vec{R}$  of a given bidirected tree  $\vec{T}_H$ , rooted at vertex  $v \in V_{\vec{T}_H}$ , is a subgraph of  $\vec{T}_H$  such that  $\|\vec{R}\|$  is a subtree of the host tree  $H$ , and every vertex  $u \in V_{\vec{R}} \setminus \{v\}$  has in-degree 1 whereas the root vertex  $v$  has in-degree 0. Let  $\mathcal{R}$  be a set of rooted subtrees of bidirected tree  $\vec{T}_H$ . We denote the set of the all rooted subtrees in  $\mathcal{R}$  that contain directed edge  $(u, v) \in E_{\vec{T}_H}$  by  $\mathcal{R}[(u, v)]$ , i.e.,  $\mathcal{R}[(u, v)] = \{\vec{R} \in \mathcal{R} | (u, v) \in E_{\vec{R}}\}$ . With a slight abuse notation, we denote the set of all the rooted subtrees in  $\mathcal{R}$  that contain directed edges  $(u, v)$  or  $(v, u)$ , by  $\mathcal{R}[\{u, v\}]$ , i.e.,  $\mathcal{R}[\{u, v\}] = \{\vec{R} \in \mathcal{R} | \{u, v\} \in E_{\|\vec{R}\|}\}$ . A set of rooted subtrees  $\mathcal{R}$  of bidirected tree  $\vec{T}_H$  are said to *collide* on directed edge  $(u, v) \in E_{\vec{T}_H}$  if for every  $\vec{R} \in \mathcal{R}$ ,  $(u, v) \in E_{\vec{R}}$ . If the directed edge on which the collision occurs is not important for the subsequent discussion, we simply say that the rooted subtrees collide.

Given a circle  $C$ , an arc  $\hat{A}$  on  $C$  is determined by pair of points  $l_{\hat{A}}, r_{\hat{A}}$  on the circle, referred to as the *left* and *right end points* of  $\hat{A}$ , respectively. Arc  $\hat{A}$  is then denoted by  $(l_{\hat{A}}, r_{\hat{A}})_C$ , and is the set of all points on  $C$  encountered while traversing the circle in clockwise direction starting from point  $l_{\hat{A}}$  and ending at point  $r_{\hat{A}}$ . Arcs  $\hat{A}_i, \hat{A}_j$  on circle  $C$  are said to *overlap*, if they share some common point, i.e., if  $\hat{A}_i \cap \hat{A}_j \neq \emptyset$ . An arc  $\hat{A}_i$  on circle  $C$  is said to *contain* another arc  $\hat{A}_j$  on  $C$ , if all the points of  $\hat{A}_j$  are also in  $\hat{A}_i$ , i.e., if  $\hat{A}_j \subseteq \hat{A}_i$ .

### B. Problem Definition

Observe that if the fiber network is a bidirected tree and all-optical networking paradigm is employed, then specifying a multicast traffic request actually fixes the routing for the corresponding light-tree. The light-tree is the rooted subtree of the bidirected tree, rooted at the source of the multicast traffic request and spanning the set of the destination nodes of the multicast traffic request. Hence, in our problem we model the given set of multicast traffic requests as a given set of rooted subtrees of the bidirected fiber tree.<sup>2</sup>

The problem of assigning wavelengths to a given set of multicast traffic requests in a WDM based all-optical network, where the fiber links are organized as a bidirected trees, with the objective of minimizing the number of wavelengths used, is referred to as Minimum Multicast Wavelength Assignment

(MIN-MC-WA) problem. The exact definition of the MIN-WC-WA problem is presented below.

**Problem 1.1 (MIN-MC-WA):** Given a pair  $\{\vec{T}, \mathcal{R}\}$ , where  $\vec{T}$  is a bidirected tree, and  $\mathcal{R} = \{\vec{R}_1, \dots, \vec{R}_{|\mathcal{R}|}\}$  is a set of rooted subtrees of  $\vec{T}$ ; consider a set of mappings  $\Lambda_{\{\vec{T}, \mathcal{R}\}}$  from  $\mathcal{R}$  to  $\mathbb{N} = \{1, 2, 3, \dots\}$ , such that for any mapping  $\lambda \in \Lambda_{\{\vec{T}, \mathcal{R}\}}$ , if a pair of rooted subtrees  $\vec{R}_i, \vec{R}_j \in \mathcal{R}$  collide, then  $\lambda(\vec{R}_i) \neq \lambda(\vec{R}_j)$ .

Determine the mapping  $\lambda^* = \underset{\lambda \in \Lambda_{\{\vec{T}, \mathcal{R}\}}}{\operatorname{argmin}} |\lambda(\mathcal{R})|$ .

Note that corresponding to any instance  $\{\vec{T}, \mathcal{R}\}$  of the MIN-MC-WA problem, we can define a *conflict graph*  $G_{\{\vec{T}, \mathcal{R}\}}$ , having vertex set  $V_{G_{\{\vec{T}, \mathcal{R}\}}} = \mathcal{R}$ . There is an edge  $\{\vec{R}_i, \vec{R}_j\} \in E_{G_{\{\vec{T}, \mathcal{R}\}}}$  if and only if  $\vec{R}_i, \vec{R}_j \in \mathcal{R}$  collide.

### C. Related Work

Although RWA has long been a problem of interest, like most other problems in optical networks, a majority of the work is concentrated either on developing new heuristics or on writing Integer Linear Programs (ILPs) for various flavors of the problem. Primarily, this is because the problem is extremely hard to solve in the general settings. Hence effort must be made to study the problem in somewhat restrictive, and yet plausible settings. For example, since most of the optical networks are organized as a collection of simple network topologies such as rings, stars, paths and trees, it is prudent to study the problem for these network topologies. Observe that in all these simple topologies, for a particular traffic request, the routing is either fixed (such as in paths, stars, trees, unidirectional rings), or the number of available route options are few (such as in bidirectional rings). Hence, the RWA problem simply reduces to the problem of assigning wavelengths. Another simplification that is sometimes imposed is to assume that the traffic demands are symmetric in nature, i.e., if there is a traffic request with source  $u$  and destination  $v$ , then there is another traffic demand with source  $v$  and destination  $u$ . A further simplification of this problem is to ensure that the pair of corresponding symmetric traffic requests between any pair of nodes are assigned the same wavelength. This problem is sometimes known as the Minimum Undirected Wavelength Assignment (MIN-UWA).

Since the problem of wavelength assignment (or for that matter, almost any resource allocation problem) can be recast as a vertex coloring problem on the corresponding conflict graph, it is not surprising that most of the theoretical results have come from the graph theory community. In particular, [3] established the hardness of MIN-UWA in stars by proving its equivalence to the problem of edge coloring in multigraphs, which is known to be NP hard [4]. In fact, this equivalence also proves that approximating MIN-UWA with an approximation ratio of  $\frac{4}{3} - \epsilon$  in general trees is NP hard for any  $\epsilon > 0$ . But on trees with bounded degree, MIN-UWA can be solved in polynomial time. The best known algorithm for the problem, presented in [5], achieves the approximation ratio of  $\frac{4}{3}$ .

MIN-UWA can be viewed as a restricted case of assigning

<sup>2</sup>Henceforth, in this paper, we use the terms ‘rooted subtree’ and ‘multicast traffic request’ interchangeably. This should not cause any confusion.

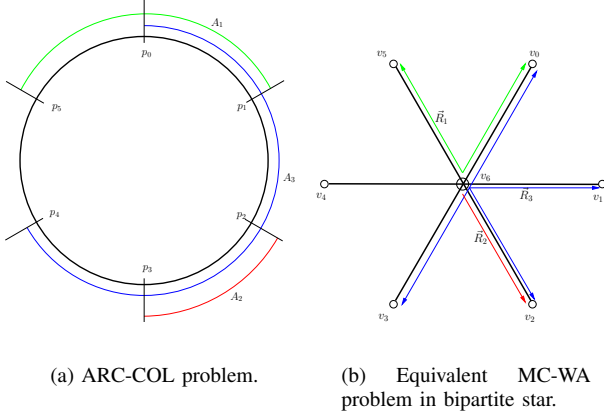


Fig. 1. Construction of an equivalent MC-WA problem in bipartite star from a ARC-COL problem.

wavelengths to a given set of subtrees of a tree.<sup>3</sup> Let us denote this problem as the Minimum Subtree Wavelength Assignment (MIN-SUB-WA) problem. This problem in itself (as well as its variants) is of interest in the graph theoretic community. In [6], it was shown that the problem, when restricted to binary trees, is equivalent to vertex coloring in chordal graphs [7], and therefore is easy to solve [8]. Along similar lines, [9] proved that the problem, when restricted to degree 4 trees, can be treated as vertex coloring in weakly chordal graphs [10], and therefore is also easy to solve [11].

There is another body of results that does not assume that the given set of traffic requests are symmetric. More specifically, consider the problem of MIN-MC-WA, when all the traffic requests are actually unicast. Let us denote this restricted problem as Minimum Unicast Wavelength Assignment (MIN-UC-WA). It was proved in [12] that MIN-UC-WA is NP hard. A stricter result in [13] proved that MIN-UC-WA remains NP hard even under restricted settings, such as when the bidirected tree is restricted to having depth 3, or when the maximum number of traffic requests that can share some common directed edge is restricted to being 3. The best known deterministic algorithm for the problem, presented in [14], has an approximation ratio of  $\frac{5}{3}$ . Moreover, it is known that no greedy, deterministic algorithm can achieve a better approximation ratio than  $\frac{5}{3}$ .

In [15], it was proved that MIN-UC-WA is NP hard even in bidirected binary trees. This is unlike the case in the undirected problem MIN-UWA, which, as stated before, is tractable for bounded degree trees. No deterministic algorithm with an approximation ratio of better than  $\frac{5}{3}$  is known for the problem. This approximation ratio is same as that for MIN-UC-WA in general bidirected trees. A randomized greedy algorithm, with approximation ratio  $\frac{7}{5}$ , for MIN-UC-WA in binary bidirected trees with restricted depth was presented in [16]. Some inapproximability results for the problem were

also presented in the paper. Later, in [17], it was proved that approximating MIN-UC-WA in bidirected binary trees with an approximation ratio  $\frac{4}{3} - \epsilon$  for any  $\epsilon > 0$  is NP hard.

An extensive compilation of complexity results on both MIN-UWA and MIN-UC-WA problems are presented in [13] and [17]. And a comprehensive survey of algorithmic results for the problems is presented in [18], [19] and [20].

#### D. Organization

The rest of the paper is organized as follows. In Section II, we prove a pair of hardness results for restricted cases of the wavelength assignment problem. In Sections III and IV, we present two approximation algorithms for MIN-MC-WA problem  $\{\vec{T}_H, \mathcal{R}\}$ , when the degree of the host tree  $H$  is restricted to being at most 3, and 4, respectively. Finally in Section V, we conclude the paper.

## II. HARDNESS RESULTS

In this section, we prove that MC-WA, the decision version of MIN-MC-WA is NP complete even when the underlying physical topology is restricted to being a bidirected star (Theorem 2.4) or a bidirected path (Theorem 2.5). Here bidirected star and bidirected path have the obvious meanings (as extended from the definition of bidirected trees).

Recall that MIN-UC-WA is the restricted problem of MIN-MC-WA in which all the traffic requests are unicast. The hardness results presented in this section show that the problem of MIN-MC-WA is inherently harder than the problem of MIN-UC-WA. This is because MIN-UC-WA is known to be solvable in polynomial time when the underlying fiber network is either a bidirected star or a bidirected path. Hence simply tweaking the approximation algorithms developed for MIN-UC-WA may not result in good algorithms for MIN-MC-WA, and there is a need to develop and study new techniques for the MIN-MC-WA problem.

First let us define the MC-WA problem.

**Problem 2.1 (MC-WA):** Given a triple  $\{\vec{T}, \mathcal{R}, k\}$  where  $\vec{T}$  is a bidirected tree,  $\mathcal{R}$  is a set of rooted subtrees of  $\vec{T}$  and  $k$  is a positive integer; consider a set of mappings  $\Lambda_{\{\vec{T}, \mathcal{R}\}}$  from  $\mathcal{R}$  to  $\mathbb{N}$ , such that for any mapping  $\lambda \in \Lambda_{\{\vec{T}, \mathcal{R}\}}$ , if a pair of rooted subtrees  $\vec{R}_i, \vec{R}_j \in \mathcal{R}$  collide, then  $\lambda(\vec{R}_i) \neq \lambda(\vec{R}_j)$ .

Is there a mapping  $\lambda^* \in \Lambda_{\{\vec{T}, \mathcal{R}\}}$  such that  $|\lambda^*(\mathcal{R})| \leq k$ ?

We prove both our hardness results by reductions to circular arc coloring (ARC-COL). For completeness, the exact definition of the ARC-COL is given next.

**Problem 2.2 (ARC-COL):** Given a triple  $\{C, \mathcal{A}, l\}$ , where  $C$  is a circle,  $\mathcal{A}$  is a set of arcs on the circle with distinct end points, and  $l$  is a positive integer; consider a set of mappings  $\Theta_{\{C, \mathcal{A}\}}$  from  $\mathcal{A}$  to  $\mathbb{N}$  such that for any mapping  $\theta \in \Theta_{\{C, \mathcal{A}\}}$ , if a pair of arcs  $\hat{A}_i, \hat{A}_j \in \mathcal{A}$  overlap, then  $\theta(\hat{A}_i) \neq \theta(\hat{A}_j)$ .

Is there a mapping  $\theta^* \in \Theta_{\{C, \mathcal{A}\}}$  such that  $|\theta^*(\mathcal{A})| \leq l$ ?

It is known that ARC-COL is NP complete [21].

To show the NP completeness of MC-WA problem in the desired restricted settings, we first prove that the general MC-WA problem is in NP.

**Lemma 2.3:** MC-WA is in NP.

<sup>3</sup>In MIN-UWA, each subtree is in fact a path.

*Proof:* Given any instance of  $\{\vec{T}, \mathcal{R}, k\}$ , of the MC-WA problem, and any mapping  $\lambda : \mathcal{R} \rightarrow \mathbb{N}$ , we can verify in polynomial time, whether  $\lambda$  is a certificate, as defined in the problem definition 2.1, for the given instance of MC-WA problem or not. Hence, MC-WA is in NP. ■

Now we present the two hardness results.

**Theorem 2.4:** MC-WA restricted to bidirected stars is NP complete.

*Proof:* Let  $\{C, \mathcal{A}, k\}$  be any given instance of the ARC-COL problem. Label the end points of the arcs from  $p_0$  to  $p_{2|\mathcal{A}|-1}$  starting with any end point of any arc, and moving clockwise on the circle. Now generate an instance of MC-WA problem as follows:

- (i) Construct a bidirected star  $\vec{S}$  with  $2|\mathcal{A}| + 1$  vertices. Label the leaf vertices from  $v_0$  to  $v_{2|\mathcal{A}|-1}$  starting from any leaf vertex and traversing clockwise through all the leaves. Label the eye of the star as  $v_{2|\mathcal{A}|}$ .
- (ii) Corresponding to each arc  $\hat{A}_i \in \mathcal{A}$ , construct a rooted subtree  $\vec{R}_i$  of  $\vec{S}$  with directed edge set  $E_{\vec{R}_i} = \{(v_{2|\mathcal{A}|}, v_j) | (p_j, p_{j+1})_C \subseteq \hat{A}_i\}$ , and vertex set  $V_{\vec{R}_i} = \{v_{2|\mathcal{A}|}\} \cup \{v_j | (p_j, p_{j+1})_C \subseteq \hat{A}_i\}$ .<sup>4</sup> Let  $\mathcal{R} = \{\vec{R}_1, \dots, \vec{R}_{|\mathcal{A}|}\}$ .

This is a polynomial time construction. An example construction is presented in Figure 1.

By construction, a pair of arcs  $\hat{A}_i, \hat{A}_j \in \mathcal{A}$  overlap if and only if the corresponding pair of rooted subtrees  $\vec{R}_i, \vec{R}_j \in \mathcal{R}$  collide. This is because arc  $(p_a, p_b)_C$  is contained in both  $\hat{A}_i, \hat{A}_j$  if and only if the set of directed edges  $\{(v_{2|\mathcal{A}|}, v_a), (v_{2|\mathcal{A}|}, v_{a+1}), \dots, (v_{2|\mathcal{A}|}, v_{b-1})\}$  is common to both  $\vec{R}_i, \vec{R}_j$ . Hence, there is a bijection between the set of mappings  $\Theta_{\{C, \mathcal{A}\}}$  and  $\Lambda_{\{\vec{S}, \mathcal{R}\}}$ . Moreover for any  $\theta \in \Theta_{\{C, \mathcal{A}\}}$  and the corresponding  $\lambda \in \Lambda_{\{\vec{S}, \mathcal{R}\}}$ ,  $|\theta(\mathcal{A})| = |\lambda(\mathcal{R})|$ . This proves that the instance  $\{C, \mathcal{A}, k\}$  of ARC-COL is equivalent to the instance  $\{\vec{S}, \mathcal{R}, k\}$  of MC-WA where  $\vec{S}$  is a bidirected tree. Hence the problem of ARC-COL is reducible to the problem of MC-WA restricted to bidirected stars. Finally, applying Lemma 2.3 completes the proof. ■

**Theorem 2.5:** MC-WA restricted to bidirected paths is NP complete.

*Proof:* Let  $\{C, \mathcal{A}, k\}$  be any given instance of the ARC-COL problem. From among the  $2|\mathcal{A}|$  end points belonging to all the arcs in  $\mathcal{A}$ , select a point  $p$  that satisfies the following:

- (i) Point  $p$  is a left end point of some arc in  $\mathcal{A}$ .
- (ii) The first end point encountered on traversing the circle  $C$  in anticlockwise direction while starting from point  $p$ , is a right end point of some arc in  $\mathcal{A}$ .

Such an end point must exist because, of the  $2|\mathcal{A}|$  end points belonging to all the arcs in  $\mathcal{A}$ , exactly  $|\mathcal{A}|$  are left end points and  $|\mathcal{A}|$  are right end points for arcs in  $\mathcal{A}$ . Next, label the end points of the arcs from  $p_0$  to  $p_{2|\mathcal{A}|-1}$  starting by labeling the selected end point  $p$  as  $p_0$ , and moving clockwise on the circle.

Partition the set  $\mathcal{A}$  into subsets  $\mathcal{A}^I$  and  $\mathcal{A}^{II}$  where  $\mathcal{A}^I$  is the set of all the arcs in  $\mathcal{A}$  that contain arc  $(p_{2|\mathcal{A}|-1}, p_0)_C$ , and  $\mathcal{A}^{II} = \mathcal{A} \setminus \mathcal{A}^I$ . Without loss of any generality, assume that  $\mathcal{A}^I = \{\hat{A}_1, \dots, \hat{A}_{|\mathcal{A}^I|}\}$ . Therefore,  $\mathcal{A}^{II} = \{\hat{A}_{|\mathcal{A}^I|+1}, \dots, \hat{A}_{|\mathcal{A}|}\}$ . A consequence of the labeling described above is that none of the arcs in the set  $\mathcal{A}^I$  have either  $p_0$  or  $p_{2|\mathcal{A}|-1}$  as an end point.

Next, construct a bidirected path  $\vec{P}$  with  $2|\mathcal{A}| + 4|\mathcal{A}^I|$  vertices that are labeled from  $v_{-2|\mathcal{A}|}$  to  $v_{2(|\mathcal{A}|+|\mathcal{A}^I|)-1}$  starting from one leaf and traversing the path to reach the other leaf. For every arc  $\hat{A}_i \in \mathcal{A}$ , construct a set of rooted subtrees  $\vec{R}_i$  of  $\vec{P}$ . If arc  $\hat{A}_i = (p_{l_i}, p_{r_i})_C \in \mathcal{A}^I = \{\hat{A}_1, \dots, \hat{A}_{|\mathcal{A}^I|}\}$ , then

$$\begin{aligned} \vec{R}_i = & \{\vec{R}_{i,1}, \vec{R}_{i,2}, \vec{R}_{i,3}, \\ & \vec{R}_{i,4,1}, \vec{R}_{i,4,2}, \dots, \vec{R}_{i,4_{k-|\mathcal{A}^I|+i-1}}, \\ & \vec{R}_{i,5,1}, \vec{R}_{i,5,2}, \dots, \vec{R}_{i,5_{k-|\mathcal{A}^I|+i-1}}\} \end{aligned}$$

having

$$\begin{aligned} V_{\vec{R}_{i,1}} &= \{v_{2(|\mathcal{A}|+i)-2}, v_{2(|\mathcal{A}|+i)-3}, \dots, v_{-2i}\}, \\ E_{\vec{R}_{i,1}} &= \{(v_{2(|\mathcal{A}|+i)-2}, v_{2(|\mathcal{A}|+i)-3}), \\ & (v_{2(|\mathcal{A}|+i)-3}, v_{2(|\mathcal{A}|+i)-4}), \\ & \dots, (v_{-2i+1}, v_{-2i})\}, \\ V_{\vec{R}_{i,2}} &= \{v_{l_i}, v_{l_i+1}, \dots, v_{2(|\mathcal{A}|+i)-1}\}, \\ E_{\vec{R}_{i,2}} &= \{(v_{l_i}, v_{l_i+1}), (v_{l_i+1}, v_{l_i+2}), \\ & \dots, (v_{2(|\mathcal{A}|+i)-2}, v_{2(|\mathcal{A}|+i)-1})\}, \\ V_{\vec{R}_{i,3}} &= \{v_{-2i+1}, v_{-2i+2}, \dots, v_{r_i}\}, \\ E_{\vec{R}_{i,3}} &= \{(v_{-2i+1}, v_{-2i+2}), (v_{-2i+2}, v_{-2i+3}), \\ & \dots, (v_{r_i-1}, v_{r_i})\}, \end{aligned}$$

and for every  $j \in \{1, 2, \dots, k - |\mathcal{A}^I| + i - 1\}$ ,

$$\begin{aligned} V_{\vec{R}_{i,4,j}} &= \{v_{2(|\mathcal{A}|+i)-3}, v_{2(|\mathcal{A}|+i)-2}, v_{2(|\mathcal{A}|+i)-1}\}, \\ E_{\vec{R}_{i,4,j}} &= \{(v_{2(|\mathcal{A}|+i)-2}, v_{2(|\mathcal{A}|+i)-3}), \\ & (v_{2(|\mathcal{A}|+i)-2}, v_{2(|\mathcal{A}|+i)-1})\}, \\ V_{\vec{R}_{i,5,j}} &= \{v_{-2i}, v_{-2i+1}, v_{-2i+2}\}, \\ E_{\vec{R}_{i,5,j}} &= \{(v_{-2i+1}, v_{-2i}), (v_{-2i+1}, v_{-2i+2})\}. \end{aligned}$$

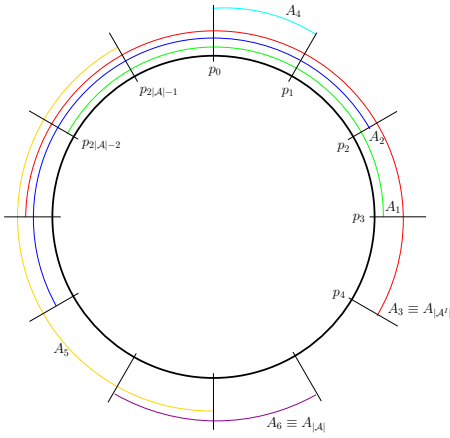
Otherwise, if arc  $\hat{A}_i = (p_{l_i}, p_{r_i})_C \in \mathcal{A}^{II} = \{\hat{A}_{|\mathcal{A}^I|+1}, \dots, \hat{A}_{|\mathcal{A}|}\}$ , then  $\vec{R}_i = \{\vec{R}_i\}$  having

$$\begin{aligned} V_{\vec{R}_i} &= \{v_{l_i}, v_{l_i+1}, \dots, v_{r_i}\}, \\ E_{\vec{R}_i} &= \{(v_{l_i}, v_{l_i+1}), \dots, (v_{r_i-1}, v_{r_i})\}. \end{aligned}$$

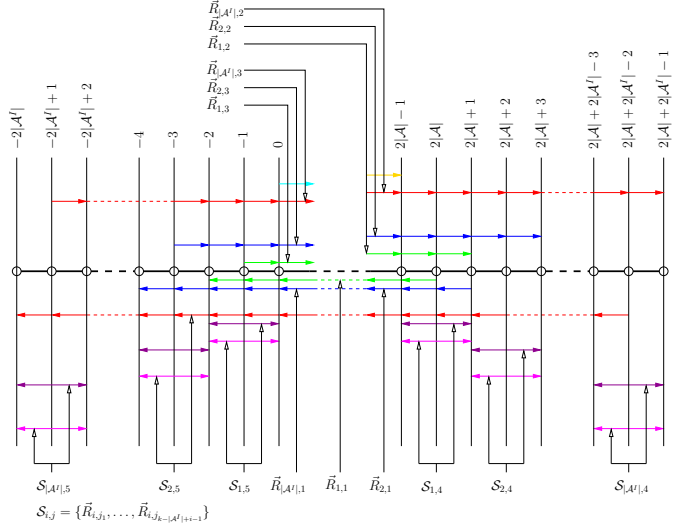
Let  $\mathcal{R} = \bigcup_{i=1}^{|\mathcal{A}|} \vec{R}_i$ . This is a polynomial time construction. An example construction is presented in Figure 2.

We claim that the answer to the ARC-COL problem  $\{C, \mathcal{A}, k\}$  is YES if and only if the answer to the MC-WA problem  $\{\vec{P}, \mathcal{R}, k\}$  is YES. To prove this claim, first assume that the answer to the ARC-COL problem  $\{C, \mathcal{A}, k\}$  is YES. Let  $\theta^*$  be the mapping as described in ARC-COL definition 2.2. Without loss of any generality, assume that  $\theta^*(\mathcal{A}) = \{1, 2, \dots, |\theta^*(\mathcal{A})|\}$ , where  $|\theta^*(\mathcal{A})| \leq k$ . Construct

<sup>4</sup>Here the addition is assumed to be modulo  $2|\mathcal{A}|$ .



(a) ARC-COL problem.



(b) Equivalent MC-WA problem in bipartite path.

Fig. 2. Construction of an equivalent MC-WA problem in bipartite path from a ARC-COL problem.

a mapping  $\lambda : \mathcal{R} \rightarrow \mathbb{N}$  using the mapping  $\theta^*$  as described next. First, for every  $i \in \{1, 2, \dots, |\mathcal{A}^I|\}$ ,

$$\lambda(\vec{R}_{i,1}) = \lambda(\vec{R}_{i,2}) = \lambda(\vec{R}_{i,3}) = \theta^*(\hat{A}_i), \quad (1)$$

and for every  $i \in \{|\mathcal{A}^I| + 1, |\mathcal{A}^I| + 2, \dots, |\mathcal{A}|\}$ ,

$$\lambda(\vec{R}_i) = \theta^*(\hat{A}_i). \quad (2)$$

Next, for every  $i \in \{1, \dots, |\mathcal{A}^I|\}$ ,  $j \in \{1, \dots, k - |\mathcal{A}^I| + i - 1\}$ ,

$$\lambda(\vec{R}_{i,4_j}) = \lambda(\vec{R}_{i,5_j}) = \min \{ \{1, 2, \dots, k\} \setminus \mathbb{F}_{i,j} \}, \quad (3)$$

where

$$\mathbb{F}_{i,j} = \bigcup_{m=1}^{j-1} \{ \lambda(\vec{R}_{i,4_m}) \} \cup \theta^*(\{ \hat{A}_i, \hat{A}_{i+1}, \dots, \hat{A}_{|\mathcal{A}^I|} \}). \quad (4)$$

Later in this proof, we shall show that, for every  $i \in \{1, \dots, |\mathcal{A}^I|\}$ ,  $j \in \{1, \dots, k - |\mathcal{A}^I| + i - 1\}$ , the set  $\{1, 2, \dots, k\} \setminus \mathbb{F}_{i,j} \neq \emptyset$ . Hence, mapping  $\lambda$  is well defined.

Observe that according to our construction, collisions between rooted subtrees in the set  $\mathcal{R}$  can be classified as follows:

- (i) A pair of rooted subtrees  $\vec{R}_i, \vec{R}_j \in \{\vec{R}_{|\mathcal{A}^I|+1}, \dots, \vec{R}_{|\mathcal{A}|}\}$  collide if and only if arcs  $\hat{A}_i, \hat{A}_j \in \mathcal{A}^{II}$  overlap.
- (ii) Let  $\mathcal{S}_i = \{\vec{R}_{1,i}, \dots, \vec{R}_{|\mathcal{A}^I|,i}\}$ , for  $i \in \{1, 2, 3\}$ . For  $i \in \{1, 2, 3\}$ , all the rooted subtrees in the set  $\mathcal{S}_i$  collide. Note that all the arcs in the set  $\mathcal{A}^I = \{\hat{A}_1, \dots, \hat{A}_{|\mathcal{A}^I|}\}$  contain the arc  $(p_{|\mathcal{A}|-1}, p_0)_C$  and therefore, are mutually overlapping.
- (iii) For  $i \in \{|\mathcal{A}^I| + 1, \dots, |\mathcal{A}|\}$  and  $j \in \{1, \dots, |\mathcal{A}^I|\}$ , rooted subtree  $\vec{R}_i$  collides with at least one of the rooted subtrees  $\vec{R}_{j,2}, \vec{R}_{j,3}$  if and only if arcs  $\hat{A}_i \in \mathcal{A}^{II}$  and  $\hat{A}_j \in \mathcal{A}^I$  overlap.

- (iv) Let  $\mathcal{S}_{i,j} = \{\vec{R}_{i,j_1}, \dots, \vec{R}_{i,j_{k-|\mathcal{A}^I|+i-1}}\}$  for  $i \in \{1, \dots, |\mathcal{A}^I|\}$  and  $j \in \{4, 5\}$ . For  $i \in \{1, \dots, |\mathcal{A}^I|\}$  all the rooted subtrees in the set  $\mathcal{S}_{i,4}$  collide with each other, and also with all the rooted subtrees in the sets  $\{\vec{R}_{i,1}, \dots, \vec{R}_{|\mathcal{A}^I|,1}\}$  and  $\{\vec{R}_{i,2}, \dots, \vec{R}_{|\mathcal{A}^I|,2}\}$ ; and all the rooted subtrees in the set  $\mathcal{S}_{i,5}$  collide with each other, and also with all the rooted subtrees in the sets  $\{\vec{R}_{i,1}, \dots, \vec{R}_{|\mathcal{A}^I|,1}\}$  and  $\{\vec{R}_{i,3}, \dots, \vec{R}_{|\mathcal{A}^I|,3}\}$ .

Besides the collisions described above, there can be no other collisions between the rooted subtrees in the set  $\mathcal{R}$ .

Consider a collision of type (i). Since arcs  $\hat{A}_i, \hat{A}_j$  overlap,  $\theta^*(\hat{A}_i) \neq \theta^*(\hat{A}_j)$ . Also, the mapping  $\lambda$  for rooted subtrees  $\vec{R}_i, \vec{R}_j$  is defined according to (2). Hence  $\lambda(\vec{R}_i) \neq \lambda(\vec{R}_j)$ .

Consider a collision of type (ii). Since arcs in the set  $\mathcal{A}^I$  are mutually overlapping,  $\theta^*$  maps distinct arcs in the set to distinct values. Also, the mapping  $\lambda$  for rooted subtrees in the sets  $\mathcal{S}_i$ , for  $i \in \{1, 2, 3\}$  is defined according to (1). Hence, for  $i \in \{1, 2, 3\}$ , distinct rooted subtrees in the set  $\mathcal{S}_i$  are assigned distinct values by the mapping  $\lambda$ .

Consider a collision of type (iii). Since arcs  $\hat{A}_i, \hat{A}_j$  overlap,  $\theta^*(\hat{A}_i) \neq \theta^*(\hat{A}_j)$ . Also, the mapping  $\lambda$  for rooted subtrees  $\vec{R}_i$  is defined according to (2), and for rooted subtrees  $\vec{R}_{j,2}, \vec{R}_{j,3}$ , it is defined according to (1). Hence  $\lambda(\vec{R}_{j,2}), \lambda(\vec{R}_{j,3}) \neq \lambda(\vec{R}_i)$ .

Consider a collision of type (iv). Equation (1) ensures that

$$\theta^*(\{\hat{A}_i, \dots, \hat{A}_{|\mathcal{A}^I|}\}) = \lambda(\{\vec{R}_{i,1}, \vec{R}_{i,2}, \vec{R}_{i,3}, \dots, \vec{R}_{|\mathcal{A}^I|,1}, \vec{R}_{|\mathcal{A}^I|,2}, \vec{R}_{|\mathcal{A}^I|,3}\}). \quad (5)$$

From (4) and (5), we get

$$\mathbb{F}_{i,j} = \lambda \left( \bigcup_{m=1}^{j-1} \{ \vec{R}_{i,4m} \} \cup \{ \vec{R}_{i,1}, \vec{R}_{i,2}, \vec{R}_{i,3}, \dots, \vec{R}_{|\mathcal{A}^I|,1}, \vec{R}_{|\mathcal{A}^I|,2}, \vec{R}_{|\mathcal{A}^I|,3} \} \right). \quad (6)$$

Mapping  $\lambda$  for rooted subtrees in the sets  $\mathcal{S}_{i,j}$ , for  $i \in \{1, \dots, |\mathcal{A}^I|\}$  and  $j \in \{4, 5\}$  is defined according to (3). Hence, for  $i \in \{1, \dots, |\mathcal{A}^I|\}$  and  $j \in \{4, 5\}$ , distinct rooted subtrees in the set  $\mathcal{S}_{i,j}$  are assigned values by the mapping  $\lambda$  that are distinct not only with each other, but also from the values assigned by the mapping  $\lambda$  to the rooted subtrees in the set  $\{ \vec{R}_{i,1}, \vec{R}_{i,2}, \vec{R}_{i,3}, \dots, \vec{R}_{|\mathcal{A}^I|,1}, \vec{R}_{|\mathcal{A}^I|,2}, \vec{R}_{|\mathcal{A}^I|,3} \}$ .

Hence, the mapping  $\lambda$  respects all the collisions among rooted subtrees in the set  $\mathcal{R}$  and is therefore a legal mapping as described in MC-WA definition 2.1, i.e.,  $\lambda \in \Lambda_{\vec{P}, \mathcal{R}}$ .

Now, we shall show that, for every  $i \in \{1, \dots, |\mathcal{A}^I|\}$ ,  $j \in \{1, \dots, k - |\mathcal{A}^I| + i - 1\}$ , the set  $\{1, 2, \dots, k\} \setminus \mathbb{F}_{i,j} \neq \emptyset$ . Hence, mapping  $\lambda$  is well defined. For this, observe that for every  $i \in \{1, \dots, |\mathcal{A}^I|\}$ ,

$$|\mathbb{F}_{i,1}| = |\mathcal{A}^I| - i + 1,$$

and for every  $j \in \{2, \dots, k - |\mathcal{A}^I| + i - 1\}$ ,

$$|\mathbb{F}_{i,j}| = |\mathbb{F}_{i,j-1}| + 1 = |\mathcal{A}^I| - i + j.$$

Hence, for every  $i \in \{1, \dots, |\mathcal{A}^I|\}$

$$\max_{j \in \{1, \dots, k - |\mathcal{A}^I| + i - 1\}} |\mathbb{F}_{i,j}| = |\mathbb{F}_{i,k - |\mathcal{A}^I| + i - 1}| + 1 = k - 1.$$

The above analysis also shows that,  $|\lambda(\mathcal{R})| = k$ . Hence, the answer to the MC-WA problem  $\{\vec{P}, \mathcal{R}, k\}$  is also YES.

Now assume that the answer to the MC-WA problem  $\{\vec{P}, \mathcal{R}, k\}$  is YES. Let  $\lambda^*$  be a mapping as described in MC-WA definition 2.1. First observe that for any legal  $\lambda \in \Lambda_{\vec{P}, \mathcal{R}}$ ,  $|\lambda(\mathcal{R})| \geq k$ . This is because

$$|\lambda(\mathcal{R})| \geq \left| \lambda \left( \bigcup_{m=1}^{k-1} \{ \vec{R}_{|\mathcal{A}^I|,4m} \} \cup \{ \vec{R}_{|\mathcal{A}^I|,2} \} \right) \right| = k.$$

The equality is because there are exactly  $k$  rooted subtrees in the set  $\bigcup_{m=1}^{k-1} \{ \vec{R}_{|\mathcal{A}^I|,4m} \} \cup \{ \vec{R}_{|\mathcal{A}^I|,2} \}$ , and all of them collide on the directed edge  $(v_{2(|\mathcal{A}|+|\mathcal{A}^I|)-2}, v_{2(|\mathcal{A}|+|\mathcal{A}^I|)-1}) \in E_{\vec{P}}$ , therefore the legality of mapping  $\lambda$  forces it to assign distinct values to all the rooted subtrees in the set. Now, since  $\lambda^* \in \Lambda_{\vec{P}, \mathcal{R}}$  is a legal mapping and is also a certificate for MC-WA problem  $\{\vec{P}, \mathcal{R}, k\}$ ,  $|\lambda^*(\mathcal{R})| = k$ .

Now observing all the collisions among the rooted subtrees on directed edges  $(v_{2(|\mathcal{A}|+|\mathcal{A}^I|)-2}, v_{2(|\mathcal{A}|+|\mathcal{A}^I|)-1})$  and  $(v_{2(|\mathcal{A}|+|\mathcal{A}^I|)-2}, v_{2(|\mathcal{A}|+|\mathcal{A}^I|)-3})$ , we note that the distinct rooted subtrees in the set  $\bigcup_{m=1}^{k-1} \{ \vec{R}_{|\mathcal{A}^I|,4m} \}$  are assigned different values according to the mapping  $\lambda^*$ , and also  $\lambda^*(\vec{R}_{|\mathcal{A}^I|,1}) = \lambda^*(\vec{R}_{|\mathcal{A}^I|,2}) \notin \lambda^*(\bigcup_{m=1}^{k-1} \{ \vec{R}_{|\mathcal{A}^I|,4m} \})$ . Continuing similar line of reasoning and observing, for every  $i \in \{|\mathcal{A}^I|, \dots, 1\}$ , pairs of directed edges  $(v_{2(|\mathcal{A}|+i)-2}, v_{2(|\mathcal{A}|+i)-1})$ ,  $(v_{2(|\mathcal{A}|+i)-2}, v_{2(|\mathcal{A}|+i)-3})$ , and

---

### Algorithm 1 GREEDY

---

**Require:** MIN-MC-WA problem instance  $\{\vec{T}_H, \mathcal{R}\}$ , where host tree  $H$  has degree at most 3.  
**Ensure:**  $\lambda^{\text{GDY}} \in \Lambda_{\{\vec{T}_H, \mathcal{R}\}}$   
1: Perform a BFS on host tree  $H$  starting with an arbitrary vertex as the root and enumerate the tree edges in the order of their discovery. Let  $\{e_1, \dots, e_{|E_H|}\}$  be the ordered set of edges  $E_H$ .  
2:  $\mathcal{P}_0 \leftarrow \emptyset$   
3: **for**  $i = 1$  to  $|E_H|$  **do**  
4:  $\mathcal{Q}_i \leftarrow \mathcal{R}[e_i] \setminus \mathcal{P}_{i-1}$   
5: **if** edge  $e_i = \{u, v\} \in E_H^\Pi$  **then**  
6:   Let  $\lambda_1, \lambda_2 \in \Lambda_{\{\vec{T}_H, \mathcal{Q}_i \cup \mathcal{P}_{i-1}\}}$   
7:    $\lambda_1(\vec{R}_j), \lambda_2(\vec{R}_j) \leftarrow \lambda^{\text{GDY}}(\vec{R}_j)$  for  $\vec{R}_j \in \mathcal{P}_{i-1}$  (unassigned otherwise).  
8:   PROCESS-EDGE-1( $\vec{T}_H, \{u, v\}, \mathcal{P}_{i-1}, \mathcal{Q}_i, \lambda_1$ )  
9:   PROCESS-EDGE-2( $\vec{T}_H, \{u, v\}, \{u, w\}, \{u, x\}, \mathcal{P}_{i-1}, \mathcal{Q}_i, \lambda_2$ )  
10:   **if**  $|\lambda_1(\mathcal{P}_{i-1} \cup \mathcal{Q}_i)| \leq |\lambda_2(\mathcal{P}_{i-1} \cup \mathcal{Q}_i)|$  **then**  
11:      $\lambda^{\text{GDY}}(\vec{R}_j) \leftarrow \lambda_1(\vec{R}_j)$  for every  $\vec{R}_j \in \mathcal{Q}_i$   
12:   **else**  
13:      $\lambda^{\text{GDY}}(\vec{R}_j) \leftarrow \lambda_2(\vec{R}_j)$  for every  $\vec{R}_j \in \mathcal{Q}_i$   
14:   **end if**  
15: **else**  
16:   **while**  $\exists$  some unassigned  $\vec{R}_j \in \mathcal{Q}_i$  **do**  
17:      $\lambda^{\text{GDY}}(\vec{R}_j) \leftarrow \min\{l \in \mathbb{N} \mid \nexists \vec{R}_k \in \mathcal{P}_{i-1} \cup \mathcal{Q}_i \text{ such that } \vec{R}_j, \vec{R}_k \text{ conflict and } \lambda^{\text{GDY}}(\vec{R}_k) = l\}$   
18:   **end while**  
19: **end if**  
20:  $\mathcal{P}_i \leftarrow \mathcal{P}_{i-1} \cup \mathcal{Q}_i$   
21: **end for**

---

$(v_{-2i+1}, v_{-2i})$ ,  $(v_{-2i+1}, v_{-2i+2})$ , we have, for distinct  $i, j \in \{|\mathcal{A}^I|, \dots, 1\}$ ,

$$\lambda^*(\vec{R}_{i,1}) = \lambda^*(\vec{R}_{i,2}) = \lambda^*(\vec{R}_{i,3}). \quad (7)$$

and

$$\lambda^*(\vec{R}_{i,1}) \neq \lambda^*(\vec{R}_{j,1}). \quad (8)$$

Consider a mapping  $\theta : \mathcal{A} \rightarrow \mathbb{N}$  defined as

$$\theta(\hat{A}_i) = \lambda^*(\vec{R}_{i,1}) \quad (9)$$

for  $i \in \{1, \dots, |\mathcal{A}^I|\}$ , and

$$\theta(\hat{A}_i) = \lambda^*(\vec{R}_i) \quad (10)$$

for  $i \in \{|\mathcal{A}^I| + 1, \dots, |\mathcal{A}|\}$ . First note that

$$|\theta(\mathcal{A})| \leq |\lambda^*(\mathcal{R})| = k.$$

Next we prove that  $\theta$  is a legal mapping as defined in ARC-COL definition 2.2. Suppose arcs  $\hat{A}_i, \hat{A}_j \in \mathcal{A}$  overlap. If  $\hat{A}_i, \hat{A}_j \in \mathcal{A}^I$  (in which case, they necessarily overlap on arc  $(p_{2|\mathcal{A}|-1}, p_0)_C$ ), then by (8) and (9),  $\theta(\hat{A}_i) \neq \theta(\hat{A}_j)$ . If  $\hat{A}_i, \hat{A}_j \in \mathcal{A}^{II}$ , then by (10) and the fact that  $\lambda^*$  is legal,  $\theta(\hat{A}_i) \neq \theta(\hat{A}_j)$ . If  $\hat{A}_i \in \mathcal{A}^I$  and  $\hat{A}_j \in \mathcal{A}^{II}$ , then  $\hat{A}_i, \hat{A}_j$ 's overlap ensures that rooted subtree  $\vec{R}_j$  collides with at least one of the rooted subtree  $\vec{R}_{i,2}, \vec{R}_{i,3}$ . Hence, by (9), (10) and the fact that  $\lambda^*$  is legal,  $\theta(\hat{A}_i) \neq \theta(\hat{A}_j)$ . This shows that mapping  $\theta$  is indeed legal. Hence, the answer to the ARC-COL problem  $C, \mathcal{A}, k$  is also YES.

This proves that the problem of ARC-COL is reducible to the problem of MC-WA restricted to bipartite paths. Finally, applying Lemma 2.3 completes the proof.  $\blacksquare$

---

**Subroutine 2 PROCESS-EDGE-1**


---

**Require:**  $\{\vec{T}_H, \{u, v\} \in E_H, \mathcal{P} \cup \mathcal{Q}, \lambda\}$  such that the degree of the host tree  $H$  is at most 3,  $\mathcal{P}$  is the set of rooted subtrees of  $\vec{T}_H$  that have already been assigned wavelengths according to the mapping  $\lambda : \mathcal{P} \rightarrow \mathbb{N}$  and  $\mathcal{Q}$  is the set of all the unassigned rooted subtrees of  $\vec{T}_H$  that contain either directed link  $(u, v)$  or  $(v, u)$ .

**Ensure:** Complete the mapping  $\lambda$  to  $\lambda : \mathcal{P} \cup \mathcal{Q} \rightarrow \mathbb{N}$  such that  $\lambda \in \Lambda_{\{\vec{T}_H, \mathcal{P} \cup \mathcal{Q}\}}$ .

- 1:  $B_1 \leftarrow G_{\vec{T}_H, \mathcal{P}[\{u, v\}]} \cup \mathcal{Q}$
- 2: **for all** pairs  $\vec{R}_j, \vec{R}_k \in \mathcal{P}[\{u, v\}] \cup \mathcal{Q}$  such that  $\vec{R}_j, \vec{R}_k$  do not collide **do**
- 3:   **if** any one of the following is true:
  - (i)  $\vec{R}_j, \vec{R}_k \in \mathcal{P}$  and  $\lambda(\vec{R}_j) \neq \lambda(\vec{R}_k)$
  - (ii)  $\vec{R}_j \in \mathcal{Q}, \vec{R}_k \in \mathcal{P}$  and  $\exists \vec{R}_l \in \mathcal{P}$  such that  $\lambda(\vec{R}_l) = \lambda(\vec{R}_k)$  and  $\vec{R}_j, \vec{R}_l$  collide
- 4:     **then**  
        $E_{B_1} \leftarrow E_{B_1} \cup \{\{\vec{R}_j, \vec{R}_k\}\}$
- 5:     **end if**
- 6: **end for**
- 7: Determine a maximum matching  $M_{B_1} \subseteq E_{B_1}$ .  $\{B_1\}$  is bipartite.
- 8: **for all** matched edges  $\{\vec{R}_j, \vec{R}_k\} \in M_{B_1}$  such that  $\vec{R}_j \in \mathcal{Q}$  and  $\vec{R}_k \in \mathcal{P}$  **do**
- 9:    $\lambda(\vec{R}_j) \leftarrow \lambda(\vec{R}_k)$
- 10: **end for**
- 11: **while**  $\exists$  some unassigned  $\vec{R}_j \in \mathcal{Q}$  **do**
- 12:   **if**  $\exists$  matched edge  $\{\vec{R}_j, \vec{R}_k\} \in M_{B_1}$  **then**
- 13:      $\lambda(\vec{R}_j), \lambda(\vec{R}_k) \leftarrow \min\{m \in \mathbb{N} \mid \nexists \vec{R}_l \in \mathcal{P} \cup \mathcal{Q} \text{ such that } \vec{R}_j, \vec{R}_l \text{ or } \vec{R}_k, \vec{R}_l \text{ collide and } \lambda(\vec{R}_l) = m\}$
- 14:   **else**
- 15:      $\lambda(\vec{R}_j) \leftarrow \min\{m \in \mathbb{N} \mid \nexists \vec{R}_l \in \mathcal{P} \cup \mathcal{Q} \text{ such that } \vec{R}_j, \vec{R}_l \text{ collide and } \lambda(\vec{R}_l) = m\}$
- 16:   **end if**
- 17: **end while**

---

### III. GREEDY ASSIGNMENT

In this section, we present a greedy strategy for MC-WA problem  $\{\vec{T}_H, \mathcal{R}\}$ , where the host tree  $H$  has degree at most 3. The scheme is a  $\frac{5}{2}$ -approximation algorithm.

First the host tree edges are ordered according to a Breadth First Searching (BFS) starting from any arbitrary node. Let the edge set ordered according to the BFS be  $E_H = \{e_1, \dots, e_{|E_H|}\}$ . The algorithm then proceeds in *rounds*. In the  $i$ -th round, host tree edge  $e_i$  is *processed*. We denote the set of traffic requests that have been assigned wavelengths at the end of round  $i$  by  $\mathcal{P}_i$ . We let  $\mathcal{P}_0 = \emptyset$ . Processing edge  $e_i$  involves assigning wavelengths to all the traffic requests in the set  $\mathcal{Q}_i = \mathcal{R}[e_i] \setminus \mathcal{P}_{i-1}$ . The wavelength assignment is greedy in the sense that while processing a host tree edge, we try to use as few new wavelengths as possible. The complete scheme is presented as GREEDY (Algorithm 1).

Let the edge  $e_i$  being processed during the  $i$ -th round of GREEDY be  $\{u, v\} \in E_H$ , where vertex  $u$  was discovered before vertex  $v$  according to the BFS. We classify the edge to be in set  $E_H^{\text{II}}$  if vertex  $u$  has degree 3, and the set of adjacent host tree edges  $\{\{u, v\}, \{u, w\}, \{u, x\}\} \in E_H$ , of which edge  $\{u, w\} \in \{e_1, \dots, e_{i-1}\}$  and edge  $\{u, x\} \in \{e_{i+1}, \dots, e_{|E_H|}\}$ . Otherwise, we classify the edge to be in set  $E_H^{\text{I}}$ . While processing any edge, the actual wavelength assignment scheme depends on its classification. If the edge being processed belongs to the set  $E_H^{\text{I}}$ , we assign available

---

**Subroutine 3 PROCESS-EDGE-2**


---

**Require:**  $\{\vec{T}, \{\{u, v\}, \{u, w\}, \{u, x\}\} \subseteq E_H, \mathcal{P} \cup \mathcal{Q}, \lambda\}$  such that the degree of the host tree  $H$  is 3,  $\mathcal{P}$  is the set of rooted subtrees of  $\vec{T}_H$  that have already been assigned wavelengths according to the mapping  $\lambda : \mathcal{P} \rightarrow \mathbb{N}$  and  $\mathcal{Q}$  is the set of all the unassigned rooted subtrees of  $\vec{T}_H$  that contain either directed link  $(u, v)$  or  $(v, u)$ .

**Ensure:** Complete the mapping  $\lambda$  to  $\lambda : \mathcal{P} \cup \mathcal{Q} \rightarrow \mathbb{N}$ .

- 1:  $B_2 \leftarrow G_{\{\vec{T}, (\mathcal{P}[\{u, x\}] \setminus \mathcal{P}[\{u, v\}]) \cup \mathcal{Q}[\{u, x\}]\}}$
- 2: **for all** pairs  $\vec{R}_j, \vec{R}_k \in (\mathcal{P}[\{u, x\}] \setminus \mathcal{P}[\{u, v\}]) \cup \mathcal{Q}[\{u, x\}]$  such that  $\vec{R}_j, \vec{R}_k$  do not collide **do**
- 3:   **if** any one of the following is true:
  - (i)  $\vec{R}_j, \vec{R}_k \in \mathcal{P}$  and  $\lambda(\vec{R}_j) \neq \lambda(\vec{R}_k)$
  - (ii)  $\vec{R}_j \in \mathcal{Q}, \vec{R}_k \in \mathcal{P}$  and  $\exists \vec{R}_l \in \mathcal{P}$  such that  $\lambda(\vec{R}_l) = \lambda(\vec{R}_k)$  and  $\vec{R}_j, \vec{R}_l$  collide
- 4:     **then**  
        $E_{B_2} \leftarrow E_{B_2} \cup \{\{\vec{R}_j, \vec{R}_k\}\}$
- 5:     **end if**
- 6: **end for**
- 7: Determine a maximum matching  $M_{B_2} \subseteq E_{B_2}$ .
- 8: **for all** matched edges  $\{\vec{R}_j, \vec{R}_k\} \in M_{B_2}$  such that  $\vec{R}_j \in \mathcal{Q}$  and  $\vec{R}_k \in \mathcal{P}$  **do**
- 9:    $\lambda(\vec{R}_j) \leftarrow \lambda(\vec{R}_k)$
- 10: **end for**
- 11: **while**  $\exists$  some unassigned  $\vec{R}_j \in \mathcal{Q}[\{u, x\}]$  **do**
- 12:   **if**  $\exists$  matched edge  $\{\vec{R}_j, \vec{R}_k\} \in M_{B_2}$  **then**
- 13:      $\lambda(\vec{R}_j), \lambda(\vec{R}_k) \leftarrow \min\{m \in \mathbb{N} \mid \nexists \vec{R}_l \in \mathcal{P} \cup \mathcal{Q} \text{ such that } \vec{R}_j, \vec{R}_l \text{ or } \vec{R}_k, \vec{R}_l \text{ collide and } \lambda(\vec{R}_l) = m\}$
- 14:   **else**
- 15:      $\lambda(\vec{R}_j) \leftarrow \min\{m \in \mathbb{N} \mid \nexists \vec{R}_l \in \mathcal{P} \cup \mathcal{Q} \text{ such that } \vec{R}_j, \vec{R}_l \text{ collide and } \lambda(\vec{R}_l) = m\}$
- 16:   **end if**
- 17: **end while**
- 18: **while**  $\exists$  some unassigned  $\vec{R}_j \in \mathcal{Q}$  **do**
- 19:    $\lambda(\vec{R}_j) \leftarrow \min\{m \in \mathbb{N} \mid \nexists \vec{R}_l \in \mathcal{P} \cup \mathcal{Q} \text{ such that } \vec{R}_j, \vec{R}_l \text{ collide and } \lambda(\vec{R}_l) = m\}$
- 20: **end while**

---

wavelengths greedily as described in the algorithm. More specifically, in the  $i$ -th round, we randomly select traffic requests one at a time from the set  $\mathcal{Q}_i$  and assign them wavelengths. Preference is given to the wavelengths that have already been assigned to traffic requests during the first  $i - 1$  rounds, or during the current round of wavelength assignment. If no such wavelength is available, a new wavelength is used.

On the other hand, if the edge belongs to the set  $E_H^{\text{II}}$ , then two separate schemes of wavelength assignment, PROCESS-EDGE-1 (Subroutine 2) and PROCESS-EDGE-2 (Subroutine 3) are tested, and the one requiring fewer number of additional wavelengths is finally employed. In PROCESS-EDGE-1, we prefer to reuse wavelengths from the set  $\lambda^{\text{GDY}}(\mathcal{P}_{i-1}[\{u, v\}])$  over reusing wavelengths from the set  $\lambda^{\text{GDY}}(\mathcal{P}_{i-1}[\{u, w\}] \setminus \mathcal{P}_{i-1}[\{u, v\}])$ . Whereas in PROCESS-EDGE-2, it is the other way round.

The detailed description of the algorithm, the intuition behind the scheme, as well as the analysis of the complexity and the approximation ratio is given in [22].

### IV. SUBTREE BASED ASSIGNMENT

Another approach for solving MIN-MC-WA problem  $\{\vec{T}_H, \mathcal{R}\}$  is to treat each traffic request as a subtree of the host tree and assign wavelengths such that if two subtrees share an



---

**Algorithm 4** SUBTREE-BASED

---

**Require:** MIN-MC-WA problem instance  $\{\vec{T}_H, \mathcal{R}\}$ , where host tree  $H$  has degree at most 4.

**Ensure:**  $\lambda^{\text{SUB}} \in \Lambda_{\{\vec{T}_H, \mathcal{R}\}}$

- 1: Determine  $\mathcal{U} = \{\|\vec{R}_i\| \mid \forall \vec{R}_i \in \mathcal{R}\}$ .
  - 2: Solve MIN-SUB-WA problem  $\{H, \mathcal{U}\}$ . Let the wavelength assignment for this problem be  $\lambda$ .
  - 3:  $\lambda^{\text{SUB}}(\vec{R}_i) \leftarrow \lambda(\|\vec{R}_i\|)$  for every  $\vec{R}_i \in \mathcal{R}$
- 

edge, then they are assigned different wavelengths. It is easy to see that this generates a legal wavelength assignment for the MIN-MC-WA problem of interest. More specifically, we generate the set  $\mathcal{U} = \{\|\vec{R}_1\|, \dots, \|\vec{R}_{|\mathcal{R}|}\|\}$  from the given set of traffic requests and then try to determine an assignment  $\lambda : \mathcal{U} \rightarrow \mathbb{N}$  such that if a pair of subtrees  $\|\vec{R}_i\|, \|\vec{R}_j\| \in \mathcal{U}$  share a common edge, then  $\lambda(\|\vec{R}_i\|) \neq \lambda(\|\vec{R}_j\|)$ . This augmented problem, specified by the pair  $\{H, \mathcal{U}\}$ , is the MIN-SUB-WA problem described in Section I. Now we generate a wavelength assignment  $\lambda^{\text{SUB}}$  for the original problem by simply assigning  $\lambda^{\text{SUB}}(\vec{R}_i) = \lambda(\|\vec{R}_i\|)$  for every  $\vec{R}_i \in \mathcal{R}$ .

In case the degree of the host tree  $H$  is not more than 4, we can quickly determine the assignment  $\lambda$  that uses the fewest number of wavelengths for the given instance of the MIN-SUB-WA problem. This is because the problem is equivalent to vertex coloring in weakly chordal graphs (in case the degree of host tree is 4), chordal graphs (in case the degree of host tree is 3) and interval graphs (in case the degree of host tree is 2), all of which can be solved optimally in polynomial time. Moreover, the wavelength assignment thus generated for the MIN-MC-WA problem is an approximation algorithm with approximation ratio  $\frac{10}{3}$ , 3 and 2 in case the degree of host tree is 4, 3 and 2, respectively. The complete algorithm is presented as SUBTREE-BASED (Algorithm 4).

The detailed description of the algorithm, the intuition behind the scheme, as well as the analysis of the complexity and the approximation ratio is given in [22].

## V. CONCLUSION

In this paper, we discussed the problem of assigning wavelengths to a given set of multicast traffic requests in all-optical WDM networks where the fiber links are organized as bidirected trees. The cost criteria that was of interest was the number of wavelengths required per fiber. We first proved that even in very simple bidirected trees, namely the bidirected stars and the bidirected paths, the problem is hard. This is unlike the problem of wavelength assignment for unicast traffic under similar settings. This shows that the problem of multicast wavelength assignment is inherently harder than the unicast problem, and approximation algorithms for the unicast problem may not work well for the multicast case. Next we presented two approximation algorithms for the problem in the case when the degree of the host tree is at most 3 (GREEDY) and at most 4 (SUBTREE-BASED).

## REFERENCES

- [1] S. Laxman and B. Mukherjee, "Light trees: Optical multicasting for improved performance in wavelength routed networks," *IEEE Communications Magazine*, vol. 37, no. 2, pp. 67–73, February 1999.
- [2] M. Ali and J. S. Deogun, "Cost-effective implementation of multicasting in wavelength-routed networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 18, no. 12, pp. 1628–1638, December 2000.
- [3] M. C. Golumbic, M. Lipshteyn, and M. Stern, "The edge intersection graphs of paths in a tree," *Journal of Combinatorial Theory, Series B*, vol. 38, no. 1, pp. 8–22, February 1985.
- [4] I. Holyer, "The NP-completeness of edge-coloring," *SIAM Journal on Computing*, vol. 10, no. 4, pp. 718–720, 1981.
- [5] T. Nishizeki and K. Kashiwaga, "On the 1.1 edge-coloring of multigraphs," *SIAM Journal on Discrete Mathematics*, vol. 3, no. 3, pp. 391–410, August 1990.
- [6] R. E. Jamison and H. M. Mulderb, "Constant tolerance intersection graphs of subtrees of a tree," *Discrete Mathematics*, vol. 290, no. 1, pp. 27–46, January 2005.
- [7] D. R. Fulkerson and O. A. Gross, "Incidence matrices and interval graphs," *Pacific Journal of Mathematics*, vol. 15, no. 3, pp. 835–855, 1965.
- [8] F. Gavril, "Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph," *SIAM Journal on Computing*, vol. 1, no. 2, pp. 180–187, 1972.
- [9] M. C. Golumbic, M. Lipshteyn, and M. Stern, "Finding intersection models of weakly chordal graphs," in *Proceedings of the 32nd International Workshop on Graph-Theoretic Concepts in Computer Science*, ser. LNCS, vol. 4271. Springer, 2006, pp. 241–255.
- [10] R. B. Hayward, "Weakly triangulated graphs," *Journal of Combinatorial Theory, Series B*, vol. 39, no. 3, pp. 200–208, December 1985.
- [11] R. B. Hayward, J. P. Spinrad, and R. Sritharan, "Improved algorithms for weakly chordal graphs," *ACM Transactions on Algorithms*, vol. 3, no. 2, May 2007.
- [12] T. Erlebach and K. Jansen, "Scheduling of virtual connections in fast networks," in *Proceedings of the 4th Parallel Systems and Algorithms Workshop*, 1996.
- [13] S. R. Kumar, R. Panigrahy, A. Russell, and R. Sundaram, "A note on optical routing in trees," *Information Processing Letters*, vol. 62, no. 6, pp. 295–300, June 1997.
- [14] C. Kaklamanis, P. Persiano, T. Erlebach, and K. Jansen, "Constrained bipartite edge coloring with applications to wavelength routing," in *Proceedings of the 24th International Colloquium on Automata, Languages and Programming*, ser. LNCS, vol. 1256. Springer, 1997, pp. 493–504.
- [15] T. Erlebach and K. Jansen, "Call scheduling in trees, rings and meshes," in *Proceedings of the 30th Hawaii International Conference on System Sciences*, 1997, p. 221.
- [16] V. Auletta, I. Caragiannis, C. Kaklamanis, and P. Persiano, "Randomized path coloring on binary trees," in *Proceedings of the 3rd International Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, ser. LNCS, vol. 1913. Springer, September 2000, pp. 407–421.
- [17] T. Erlebach and K. Jansen, "The complexity of path coloring and call scheduling," *Theoretical Computer Science*, vol. 255, no. 1-2, pp. 33–50, March 2001.
- [18] I. Caragiannis, C. Kaklamanis, and P. Persiano, "Wavelength routing in all-optical tree networks: A survey," *Computing and Informatics (formerly Computers and Artificial Intelligence)*, vol. 20, no. 2, pp. 95–120, 2001.
- [19] —, "Approximate path coloring with applications to wavelength assignment in wdm optical networks," in *Proceedings of the 21st International Symposium on Theoretical Aspects of Computer Science*, ser. LNCS, vol. 2996. Springer, 2004, pp. 258–269.
- [20] —, *Approximation Algorithms for Path Coloring in Trees*, ser. LNCS. Springer, 2006, vol. 3484, pp. 74–96.
- [21] M. R. Garey, D. S. Johnson, G. L. Miller, and C. H. Papadimitriou, "The complexity of coloring circular arcs and chords," *SIAM Journal on Matrix Analysis and Applications*, vol. 1, no. 2, pp. 216–227, 1980.
- [22] A. Rawat, M. Shayman, R. La, and S. Marcus, "Coloring rooted subtrees on a bounded degree host tree," available as <http://www.ece.umd.edu/~anuj/COLORING/color.pdf>.