Optimal on-line colorings for minimizing the number of ADMs in optical networks*

Mordechai Shalom[1,3], Prudence W.H. Wong[2], and Shmuel Zaks[1] **

[1] Department of Computer Science, Technion, Haifa, Israel
[cmshalom,zaks]@cs.technion.ac.il
[2] Department of Computer Science, The University of Liverpool, Liverpool, UK
pwong@csc.liv.ac.uk
[3] TelHai Academic College, Upper Galilee, 12210, Israel
cmshalom@telhai.ac.il

Abstract. We consider the problem of minimizing the number of ADMs in optical networks. All previous theoretical studies of this problem dealt with the off-line case, where all the lightpaths are given in advance. In a real-life situation, the requests (lightpaths) arrive at the network on-line, and we have to assign them wavelengths so as to minimize the switching cost. This study is thus of great importance in the theory of optical networks. We present a deterministic on-line algorithm for the problem, and show its competitive ratio to be $\frac{7}{4}$. We show that this result is best possible in general. Moreover, we show that even for the ring topology network there is no on-line algorithm with competitive ratio better than $\frac{7}{4}$. We show that on path topology the competitive ratio of the algorithm is $\frac{3}{2}$. This is optimal for in this topology. The lower bound on ring topology does not hold when the ring is of bounded size. We analyze the triangle topology and show a tight bound of $\frac{5}{3}$ for it. The analyses of the upper bounds, as well as those for the lower bounds, are all using a variety of proof techniques, which are of interest by their own, and which might prove helpful in future research on the topic.

Keywords: *Wavelength Assignment, Wavelength Division Multiplexing (WDM), Optical Networks, Add-Drop Multiplexer (ADM), On-line Algorithms*

1 Introduction

1.1 Background

Optical wavelength-division multiplexing (WDM) is today the most promising technology that enables us to deal with the enormous growth of traffic in communication networks, like the Internet. A communication between a pair of nodes is done via a *lightpath*, which is assigned a certain wavelength. In graph-theoretic terms, a lightpath is a simple path in the network, with a color assigned to it.

Given a WDM network $G = (V, E)$ comprising optical nodes and a set of full-duplex lightpaths $P = \{p_1, p_2, ..., p_N\}$ of G, the wavelength assignment (WLA) task is to assign a wavelength to each lightpath p_i. Most of the studies in optical networks dealt with the issue of assigning colors to lightpaths, so that every two lightpaths that share an edge get different colors.

When the various parameters comprising the switching mechanism in these networks became clearer, the focus of studies shifted, and today a large portion of the studies concentrates on the total hardware cost. The key point here is that each lightpath uses two Add-Drop Multiplexers (ADMs), one at each endpoint. If two adjacent lightpaths, i.e. lightpaths sharing a common endpoint, are assigned the same wavelength, then they can use the same ADM. Because ADMs are designed to be used mainly in ring and path networks in which the degree of a node is at most two, an ADM may be shared by at most two lightpaths. The total cost considered is the total number of ADMs. A more detailed technical explanation can be found in [GLS98].

Lightpaths sharing ADMs in a common endpoint can be thought as concatenated, so that they form longer paths or cycles. These paths/cycles do not use any edge $e \in E$ twice, for otherwise they cannot use the same wavelength which is a necessary condition to share ADMs.

---

* A preliminary version of this paper appeared in [SWZ07].

** This research was partly supported by the EU Project "Graphs and Algorithms in Communication Networks (GRAAL)" - COST Action TIST 293.

The motivation for the on-line problem stems from the need to utilize the cost of use of the optical network. We assume that the switching equipment is installed in the network. Once a lightpath arrives, we need to assign it two ADMs, and our target is to determine which wavelength to assign to it so that we minimize the cost, measured by the total number of ADMs used.

1.2 Previous Work

Minimizing the number of ADMs in optical networks is a main research topic in recent studies. The problem was introduced in [GLS98] for the ring topology. An approximation algorithm for the ring topology with approximation ratio of $\frac{3}{2}$ was presented in [CW02], and was improved in [SZ04,EL04] to $\frac{10}{7}+\epsilon$ and $\frac{10}{7}$, respectively.

For general topology [EMZ02] described an algorithm with approximation ratio of $\frac{8}{5}$. The same problem was studied in [CFW02] and an algorithm with an approximation ratio of $\frac{3}{2}+\epsilon$ was presented. This algorithm is further analyzed in [FSZ06].

The problem of on-line path coloring is studied in earlier works, such as [LV98]. The problem studied in these works has a different objective function, namely the number of colors.

All previous theoretical studies on the problem of minimizing the number of switches dealt with the off-line case, where all the lightpaths are given in advance. An on-line algorithm is said to be c-competitive if for any sequence of lightpaths, the number of ADMs used is at most c times that used by the optimal off-line algorithm (see [BEY98]).

Recently in [BBMELH08] a similar on-line scenario is considered, although in a quite different setting.

1.3 Our Contribution

We present an on-line algorithm with competitive ratio of $\frac{7}{4}$ for any network topology. We prove that no deterministic on-line algorithm has a competitive ratio better than $\frac{7}{4}$ even if the topology is a ring.

We show that the same algorithm has a competitive ratio of $\frac{3}{2}$ in path topologies, and that this is also a lower bound for on-line algorithms in this topology.

The lower bound on ring topology does not hold when the ring is of a bounded size. We study the triangle topology, and show a tight bound of $\frac{5}{3}$ for the competitive ratio on this topology, using another algorithm.

The analyses of the upper bounds, as well as those for the lower bounds, use a variety of proof techniques, which are of interest on their own, and which might prove helpful in future research on the topic.

In Section 2 we describe the problem and some preliminary results. The algorithm and its competitive analysis are presented in Section 3. In Section 4 we present lower bounds for the competitive ratio of the problem on general topology, ring and path topologies. In Section 5 we present tight bounds for triangle networks. We conclude with discussion and open problems in Section 6.

2 Preliminaries

An instance α of the problem is a pair $\alpha=(G,P)$ where $G=(V,E)$ is an undirected graph and P is a set of simple paths in G. In an on-line instance, the graph G is known in advance and the set P of paths is given on-line. In this case we denote $P=\{p_1,p_2,...,p_N\}$ where p_i is the i-th path of the input and $P_i=\{p_j\in P|j\leq i\}$ consists of the first i paths of the input.

Given such an instance we define the following:

Definition 2.1 *The paths $p,p'\in P$ are conflicting or overlapping if they have an edge in common. This is denoted as $p\asymp p'$. The graph of the relation $\asymp$ is called the conflict graph of (G,P).*

Definition 2.2 *A proper coloring (or wavelength assignment) of P is a function $w:P\mapsto\mathbb{N}$, such that $w(p)\neq w(p')$ whenever $p\asymp p'$.*

Note that w is a proper coloring if and only if for any color $c\in\mathbb{N}$, $w^{-1}(c)$ is an independent set in the conflict graph.

Definition 2.3 *A valid chain (resp. cycle) of $\alpha=(G,P)$ is a path (resp.cycle) formed by the concatenation of distinct paths $p_{i_0},p_{i_1},...,p_{i_{k-1}}\in P$ that do not go over the same edge twice. Note that the paths of a valid chain (resp. cycle) constitute an independent set of the conflict graph.*

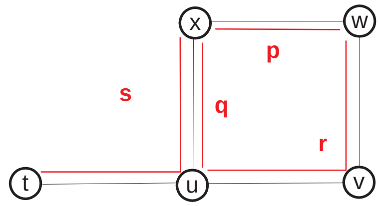


Fig. 1. A sample input.

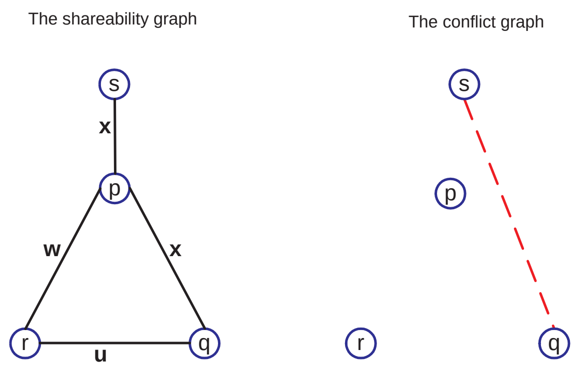


Fig. 2. The shareability and conflict graphs for the input in Figure 1.

Definition 2.4 *A solution S of an instance $\alpha = (G, P)$ is a set of valid chains and valid cycles of P such that each $p \in P$ appears in exactly one of these sets.*

In the sequel we introduce the shareability graph, which together with the conflict graph constitutes another (dual) representation of the instance α. In the sequel, except one exception, we will use the dual representation of the problem.

Definition 2.5 *The shareability graph of an instance $\alpha = (G, P)$, is the edge-labelled multi-graph $\mathcal{G}_\alpha = (P, E_\alpha)$ such that there is an edge $e = (p, q)$ labelled u in E_α if and only if $p \not\asymp q$, and u is a common endpoint of p and q in G.*

Example: Let $\alpha = (G, P)$ be the instance in Figure 1. Its shareability graph $\mathcal{G}_\alpha = (V_\alpha, E_\alpha)$ is the graph at the left side of Figure 2. In this instance $P = \{p, q, r, s\}$, and it constitutes the set of nodes of $\mathcal{G}_\alpha$. The edges together with their labels are $E_\alpha = \{(q, r, u), (p, r, w), (p, q, x), (p, s, x)\}$, because p and q can be joined in their common endpoint x, etc.. Note that, for instance $(q, s, x) \notin E_\alpha$, because although q and s share a common endpoint x, they cannot be concatenated, because they have the edge (x, u) in common. The corresponding conflict graph is the graph at the right side of Figure 2. It has the same node set and one edge, namely (q, s). The paths $q, s \in P$ are conflicting because they have a common edge, i.e. (u, x).

Note that the edges of the conflict graph are not in E_α. This immediately follows from the definitions. Note also that, for any node v of $\mathcal{G}_\alpha$, the set of labels of the edges adjacent to v is of size at most two.

Definition 2.6 *A valid chain (resp. cycle) of $\mathcal{G}_\alpha$ is a simple path $p_{i_0}, p_{i_1}, ..., p_{i_{k-1}}$ of $\mathcal{G}_\alpha$, such that any two consecutive edges in the path (resp. cycle) have distinct labels and its node set is properly colorable with one color (in G), or in other words constitutes an independent set of the conflict graph.*

Note that the valid chains and cycles of $\mathcal{G}_\alpha$ correspond to valid chains and cycles of the instance α. In the above example the chain p, s which is the concatenation of the paths p and s in the graph G, corresponds to the simple path p, s in $\mathcal{G}_\alpha$ and the cycle p, q, r which is a cycle formed by the concatenation of three paths in G corresponds to the cycle p, q, r in $\mathcal{G}_\alpha$. Note that no two consecutive labels are equal in this cycle. On the other hand the paths q, p, s cannot be concatenated to form a chain, because this would require the

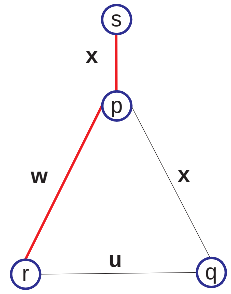


Fig. 3. The sharing graph of the solution $S = \{(s, p, r), (q)\}$ for the input in Figure 1. The thick lines are the edges in the sharing graph.

connection of p to both q and s at node x. The corresponding path q, p, s in $\mathcal{G}_\alpha$ is not a valid chain because the edges (q, p) and (p, s) have the same label, namely x.

Definition 2.7 *The sharing graph of a solution S of an instance $\alpha = (G, P)$, is the following subgraph $\mathcal{G}_{\alpha,S} = (P, E_S)$ of $\mathcal{G}_\alpha$. Two lightpaths $p, q \in P$ are connected with an edge labelled u in E_S if and only if they are consecutive in a chain or cycle in the solution S, and their common endpoint is $u \in V$. We will usually omit the index α and simply write $\mathcal{G}_S$. We denote by $d(p)$ the degree of node p in $\mathcal{G}_S$.*

In our example, $S = \{(s, p, r), (q)\}$ is a solution with two chains. The sharing graph of this solution is depicted in Figure 3. Note that for a chain of size at most two, the distinct labelling condition is satisfied vacuously, and the independent set condition is satisfied because no edge of $\mathcal{G}_\alpha$ can be an edge of the conflict graph.

For any solution S, we partition the set of lightpaths P into disjoint subsets depending on the degree of the corresponding node in $\mathcal{G}_S$. We define:

$$\forall i \in \{0, 1, 2\}, D_i(S) \stackrel{def}{=} \{p \in P \mid d(p) = i\}$$

and

$$d_i(S) \stackrel{def}{=} |D_i(S)| .$$

Note that $d_0(S) + d_1(S) + d_2(S) = |P| = N$.

An edge $(p, q) \in E_S$ with label u corresponds to a concatenation of two paths with the same color at their common endpoint u. Therefore these two endpoints can share an ADM operating at node u, thus saving one ADM. We conclude that every edge of E_S corresponds to a saving of one ADM. When no ADMs are shared, each path needs two ADMs, a total of $2N$ ADMs. Therefore the cost of a solution S is

$$cost(S) = 2|P| - |E_S| = 2N - |E_S| .$$

The objective is to find a solution S such that $cost(S)$ is minimum, in other words $|E_S|$ is maximum.

The following definitions and Lemma appeared in [FSZ06], we repeat them here for completeness.

Given a solution S, $d(p) \leq 2$ for every node $p \in P$. Therefore, the connected components of $\mathcal{G}_S$ are either paths or cycles. Note that an isolated node is a special case of a path. Let $\mathcal{P}_S$ be the set of the connected components of $\mathcal{G}_S$ that are paths. Clearly, $|E_S| = N - |\mathcal{P}_S|$. Therefore

$$cost(S) = 2N - |E_S| = N + |\mathcal{P}_S|$$

Let S^* be a solution with minimum cost. For any solution S we define

$$\epsilon(S) \stackrel{def}{=} \frac{d_0(S) - d_2(S) - 2|\mathcal{P}_{S^*}|}{N}.$$

Lemma 2.1 *For any solution S*

$$cost(S) = cost(S^*) + \frac{1}{2}N(1 + \epsilon(S)).$$

Proof. Clearly $|E_{S^*}| = N - |\mathcal{P}_{S^*}|$. On the other hand $2\,|E_S|$ is the sum of the degrees of the nodes in $\mathcal{G}_S$, namely

$$2\,|E_S| = d_1(S) + 2d_2(S) = N - d_0(S) + d_2(S)$$

We conclude:

$$\begin{aligned} cost(S) - cost(S^*) = |E_{S^*}| - |E_S| &= N - |\mathcal{P}_{S^*}| - \frac{N - d_0(S) + d_2(S)}{2} \\ &= \frac{N}{2} + \frac{d_0(S) - d_2(S) - 2\,|\mathcal{P}_{S^*}|}{2} \\ &= \frac{1}{2}N\left(1 + \frac{d_0(S) - d_2(S) - 2\,|\mathcal{P}_{S^*}|}{N}\right) \end{aligned}$$

□

3 Upper Bounds

In this section we first describe an on-line algorithm, and then show that it is $\frac{7}{4}$-competitive on any network topology and $\frac{3}{2}$-competitive on path topology.

3.1 Algorithm Online-MinADM

In a general network, when the lightpaths are given one-by-one, we adopt a simple coloring procedure. Basically, a new lightpath with endpoints u and v looks for free ADM at its endpoints. A free ADM in u is an ADM serving one lightpath ending in u, but not sharing an edge with the lightpath to be colored. If there are two of the same color, then it first tries to make a cycle with the existing lightpaths, and if this is impossible then it makes a path. If there are free ADMs (at one endpoint, or at both endpoints but of different colors), then it tries to connect to any of them. Otherwise - when there is no free ADM - it is assigned a new color.

When we attempt to color some lightpath p_i, a color λ is said to be *feasible* for p_i, if there is no other lightpath with the same color overlapping with p_i. In other words λ is feasible for p_i, if we can assign $w(p_i) = \lambda$ and w is a proper coloring for P_i.

When a lightpath p_i with endpoints u_i and v_i arrives,

- If there exists a chain of lightpaths of the same color λ whose endpoints are u_i, v_i and λ is feasible for p_i then, assign $w(p_i) = \lambda$.
- Otherwise, if there exists a chain of lightpaths of the same color λ having one endpoint from $\{u_i, v_i\}$ and λ is feasible for p_i then, assign $w(p_i) = \lambda$.
- Otherwise, assign $w(p_i) = \lambda'$, where λ' is an unused color.

Note that, as in the last clause the algorithm resorts to an unused color, it will never construct two chains with the same color. Therefore in the first clause, the algorithm necessarily closes a cycle.

Suppose in the example in Figure 1 the lightpaths arrive in the order of p, q, r, s. Algorithm Online-MinADM would color p, q, r with one color and s with another.

Algorithm Online-MinADM is obviously correct: w is a proper coloring for P_i, because if p_i is colored by one of the first two cases, then it is checked by the algorithm for feasibility, otherwise $w(p_i)$ is assigned an unused color, therefore no other path, in particular no path p_j conflicting with p_i may have $w(p_j) = w(p_i)$.

In this and the following section we prove the following theorem.

Theorem 3.1 *Algorithm* Online-MinADM *is optimal for*

- *general topology, with competitive ratio of* $\frac{7}{4}$,
- *ring topology, with competitive ratio of* $\frac{7}{4}$,
- *path topology, with competitive ratio of* $\frac{3}{2}$.

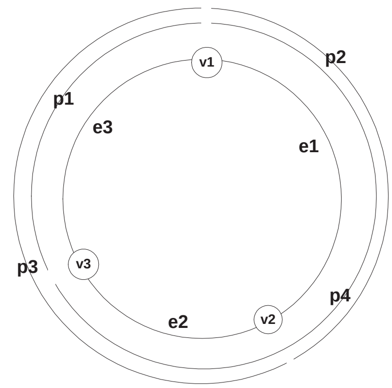


Fig. 4. Lower bound for Online-MinADM.

3.2 Analysis for General Topology

Lemma 3.1 *The competitive ratio of* Online-MinADM *is at least* $\frac{7}{4}$.

Proof. Let G be a cycle of three nodes $V = \{v_1, v_2, v_3\}$, $E = \{e_1, e_2, e_3\}$ where $e_1 = (v_1, v_2), e_2 = (v_2, v_3), e_3 = (v_3, v_1)$ and let $P = \{p_1, p_2, p_3, p_4\}$ where $p_1 = (e_3), p_2 = (e_1), p_3 = (e_2, e_3), p_4 = (e_1, e_2)$. Figure 4 shows the network and the paths. The optimal solution assigns $w(p_1) = w(p_4) = \lambda_1$ and $w(p_2) = w(p_3) = \lambda_2$, and uses 4 ADMs. Recall that Online-MinADM receives the paths of the input one at a time. It assigns $w(p_1) = \lambda_1$, then $w(p_2) = \lambda_1$ because λ_1 is feasible for p_2, then $w(p_3) = \lambda_2$ because λ_1 is not feasible for p_3 and finally $w(p_4) = \lambda_3$, because neither λ_1 nor λ_2 are feasible for p_4. It uses 7 ADMs in total. □

Although the above Lemma is a corollary of Lemma 4.1, the above proof for this special case is provided for ease of exposure.

In the sequel S is a solution returned by the Online-MinADM and S^* is an optimal solution.

Lemma 3.2 *The competitive ratio of* Online-MinADM *is at most* $\frac{7}{4}$.

Proof. We direct each edge of $\mathcal{G}_{S^*}$, such that each path becomes a directed path and each cycle becomes a directed cycle. The direction chosen for every path (resp. cycle) is arbitrary. Let $\overrightarrow{\mathcal{G}}_{S*}$ be the digraph obtained by this process. Unless otherwise stated, $d_{in}(p)$ and $d_{out}(p)$ denote the in and out degrees of p in $\overrightarrow{\mathcal{G}}_{S^*}$, respectively. Clearly, $\forall p \in P$, $d_{in}(p) \leq 1$ and $d_{out}(p) \leq 1$. The following definitions refer to $\overrightarrow{\mathcal{G}}_{S*}$:

$LAST^*$ is the set of nodes that do not have successors in $\overrightarrow{\mathcal{G}}_{S^*}$, namely

$$LAST^* \stackrel{def}{=} \{p \in P | d_{out}(p) = 0\}.$$

Note that $|LAST^*| = |P_{S^*}|$.

The functions $Next^*$ and $Prev^*$ are defined as expected: $Next^*$ (resp. $Prev^*$) maps a node p to the next (resp. previous) node in $\overrightarrow{\mathcal{G}}_{S^*}$ whenever such a node exists, namely:

$$Next^* : P \setminus LAST^* \mapsto P$$

and $Next^*(p)$ is the unique node u such that there is an edge from p to u in $\overrightarrow{\mathcal{G}}_{S*}$. $Prev^* = {Next^*}^{-1}$.

With these definitions in hand, we partition $D_0(S)$, i.e. the set of isolated nodes of $\mathcal{G}_{S^*}$, into sets A, B, C and D using the following classification procedure **CLASSIFY**. This procedure makes use of both graphs $\mathcal{G}_{S^*}$ and $\overrightarrow{\mathcal{G}}_{S*}$:

CLASSIFY($p \in D_0(S)$) **{**
  If $p \in LAST^*$ then {
    $p \in A$; $f_A(p) = p$;

```
      } else {
        q = Next*(p);
        If q ∈ D_2(S) then {
          p ∈ B; f_B(p) = q;
        } else if q ∈ D_1(S) then {
          p ∈ C; f_C(p) = {p, q};
        } else { // q ∈ D_0(S)
          p ∈ D;
        }
      }
    }
```

It is immediate from the code that **CLASSIFY** is a classification procedure: it partitions $D_0(S)$ into $A \uplus B \uplus C \uplus D$. Moreover it is also immediate from the code that $f_A : A \mapsto LAST^*$, $f_B : B \mapsto D_2(S)$ and $f_C : C \mapsto 2^P$.

We first show that $D = \emptyset$. Assume, by contradiction that $p \in D$ for some $p \in D_0(S)$. Then there is $q \in D_0(S)$ such that $q = Next^*(p)$, therefore $(p, q) \in E_{S^*} \subseteq E_\alpha$. Online-MinADM assigned unique colors to each of p and q. Assume without loss of generality that q comes later than p in the input sequence. p is assigned a unique color, therefore it is the only element in its chain. Then $w(p)$ is feasible for q. Then the algorithm should assign $w(q) = w(p)$, a contradiction.

$f_A(p) = p$, therefore it is a one-to-one function, i.e. $|A| \leq |LAST^*| = |\mathcal{P}_{S^*}|$.

$f_B(p) = Next^*(p)$. $Next^*$ is one-to-one, therefore f_B is one-to-one, i.e. $|B| \leq |D_2(S)| = d_2(S)$.

We will now show that the sets $f_C(p)$ are pairwise disjoint. Note that $f_C(p) = \{p, q\}$ where $p \in D_0(S)$ and $q \notin D_0(S)$. Assume that $f_C(p) \cap f_C(p') \neq \emptyset$. Let $f_C(p) = \{p, q\}$ and $f_C(p') = \{p', q'\}$. Then either $p = p'$ or $q = q'$. In the latter case $q = Next^*(p) = Next^*(p') = q'$, then $p = p'$. In both cases, we have $p = p'$. We conclude that if $p \neq p'$, $f_C(p) \cap f_C(p) = \emptyset$. As the sets $f_C(p)$ contain exactly 2 elements, we have $|C| \leq \frac{N}{2}$.

We have $d_0(S) = |D_0(S)| = |A| + |B| + |C| + |D| \leq |\mathcal{P}_{S^*}| + d_2(S) + \frac{N}{2}$. Then

$$\epsilon(S) = \frac{d_0(S) - d_2(S) - 2\,|\mathcal{P}_{S^*}|}{N} \leq \frac{1}{2}.$$

Substituting this in Lemma 2.1 and recalling that $cost(S^*) \geq N$ we get

$$Cost(S) \leq Cost(S^*) + \frac{1}{2}N\left(1 + \frac{1}{2}\right) = Cost(S^*) + \frac{3}{4}N \leq \frac{7}{4}Cost(S^*).$$

□

Note: The reader is referred to [WCLF00], Lemma 4, that gives a bound on related off-line algorithms for ring networks, its proof can be applied to derive an alternative proof of Lemma 3.2. We believe that our proof supplies more insight into the structure of the solution obtained by the on-line algorithm and its relation to an optimal solution.

3.3 Analysis for Path Topology

Lemma 3.3 Online-MinADM *is $\frac{3}{2}$-competitive in path topology.*

Proof. Let $V = \{v_1, v_2, ...\}$ be the nodes of the path from left to right, and σ_i (resp. τ_i) be the set of paths having v_i as their right (resp. left) endpoint. It is well known that the number of ADMs used by an optimal solution is $\Sigma_i \max\{|\sigma_i|, |\tau_i|\}$. In an optimal solution, at each node v_i, exactly $\min\{|\sigma_i|, |\tau_i|\}$ pairs of paths are assigned one color per pair. In fact these pairs constitute a maximum matching MM_i of the complete bipartite graph $(\sigma_i, \tau_i, \sigma_i \times \tau_i)$. The solution saves $|MM_i| = \min\{|\sigma_i|, |\tau_i|\}$ ADMs at node v_i, in other words $E_{S^*} = \uplus_i MM_i$. Note that every matching of a complete bipartite graph can be augmented to a maximum matching. Recall that S is the output of the algorithm. Let S^* be an optimal solution, such that the matching in each node is obtained by augmenting the matching done by S to a maximum matching, i.e. $E_S \subseteq E_{S^*}$.

We will now define a function $f : (E_{S^*} \setminus E_S) \mapsto E_S$. In the following discussion, consult Figure 5.

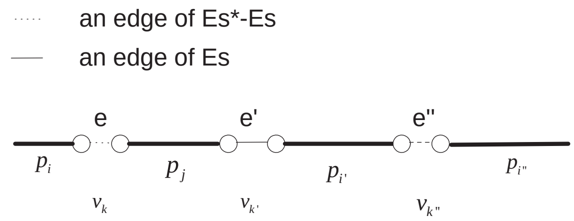


Fig. 5. Competitive ratio of ONLINE-MINADM in the path topology.

Let $e = (p_i, p_j) \in E_{S^*} \setminus E_S$. $e \in E_{S^*} = \uplus_i MM_i$. Let $e \in MM_k$ for some node $v_k \in V$. Assume without loss of generality that $i < j$, i.e. path p_i appears before p_j in the input. As $e \notin E_S$, none of p_i, p_j are paired with any path at node v_k. Therefore when p_j appears in the input $w(p_i)$ is feasible for p_j, if it is not assigned color $w(p_i)$, this can be only because it is assigned color $w(p_j) = w(p_{i'})$, for some $i' < j$. Let the common node of p_j and $p_{i'}$ be $v_{k'}$. Then $e' = (p_j, p_{i'}) \in E_S$. We define $f(e) = e'$. Note that e' is defined uniquely because there cannot be a third path except p_j and $p_{i'}$ getting the same color and ending at node $v_{k'}$. Necessarily $k' \neq k$, because we know that p_j is not paired at node v_k.

We claim that f is one-to-one. Assume, by contradiction that there is some $e'' \neq e$, such that $f(e'') = e'$. Then $e'' \in E_{S^*}$, therefore $e'' \in MM_{k''}$ for some node $v_{k''}$. By the construction of f, k'' is the other endpoint of $p_{i'}$. Let $e'' = (p_{i'}, p_{i''})$. By the discussion in the previous paragraph, symmetrically it follows that $j < i'$, a contradiction. Therefore f is one-to-one, i.e. $|E_{S^*}| - |E_S| = |E_{S^*} \setminus E_S| \leq |E_S|$, thus $|E_S| \geq \frac{1}{2} |E_{S^*}|$.

We conclude as follows. $Cost(S) - Cost(S^*) = |E_{S^*}| - |E_S| \leq \frac{|E_{S^*}|}{2} \leq \frac{N}{2} \leq \frac{Cost(S^*)}{2}$, therefore:

$$Cost(S) \leq \frac{3}{2} Cost(S^*).$$

□

4 Lower Bounds

4.1 General Topology

Lemma 4.1 *There is no deterministic on-line algorithm with competitive ratio* $< \frac{7}{4}$.

Proof. Assume ALG is a deterministic on-line algorithm, with competitive ratio ρ. We show that $\rho \geq \frac{7}{4}$. For colors we use numbers $1, 2, \ldots$. The color assigned to a lightpath a by ALG is denoted by $w(a)$. We use the network depicted in Figure 6. The first lightpath in the input is EFG. Without loss of generality, assume $w(EFG) = 1$.

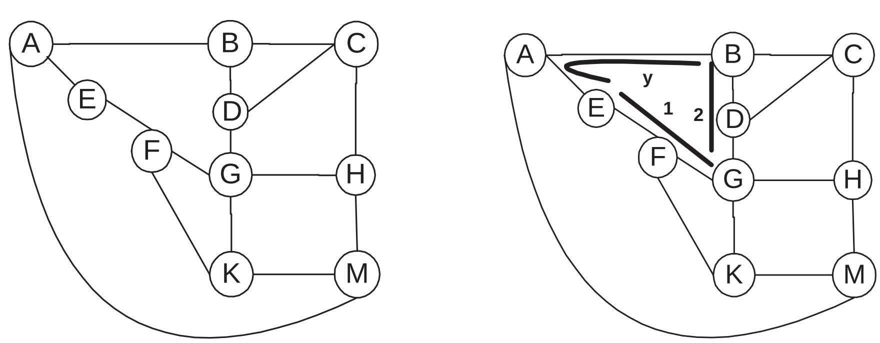


Fig. 6. A network used for the proof of the $\frac{7}{4}$ lower bound on general topology.

The second lightpath in the input is BDG. First assume $w(BDG) = 1$. In this case if lightpath EABDG arrives, we have $w(EABDG) = 2$, then when lightpath GFEAB arrives we have $w(GFEAB) = 3$. ALG thus uses 7 ADMs, while it is easy to see the an optimal solution can use only 4 ADMs, thus $\rho \geq \frac{7}{4}$, a contradiction. Hence, $w(BDG) = 2$.

When the third lightpath in the input y=BAE arrives the situation is as depicted in the right hand side of Figure 6. It is clear that $w(y) \neq 3$, since otherwise $\rho \geq \frac{6}{3} > \frac{7}{4}$, a contradiction. Thus $w(y) = 1$ or $w(y) = 2$.

- case a: $w(y) = 1$
  Let z=EFKMHG be the next lightpath in the input sequence. Clearly $w(z) \neq 1$. Hence $w(z) = 2$ or $w(z) = 3$.
  - $w(z) = 2$
    In this case, when lightpaths GFEAB, EABDG, BDGFE and EABCDG arrive, we get $w(GFEAB) = 3, w(EABDG) = 4, w(BDGFE) = 5, w(EABCDG) = 6$, and $\rho = \frac{14}{8} = \frac{7}{4}$, a contradiction.
  - $w(z) = 3$
    In this case, for u=EABDCHG we have $w(u) = 4$, and $\rho \geq \frac{9}{5} > \frac{7}{4}$, a contradiction.
- case b: $w(y) = 2$
  Let z=BDCHG. Clearly $w(z) \neq 2$. Hence $w(z) = 1$ or $w(z) = 3$.
  - $w(z) = 1$
    When lightpaths EABDG, GFEAB, GKFEAB, and EFGDB arrive, we have $w(EABDG) = 3, w(GFEAB) = 4, w(GKFEAB) = 5, w(EFGDB) = 6$, and $\rho \geq \frac{14}{8} = \frac{7}{4}$, a contradiction.
  - $w(z) = 3$
    For u=GHMKFEAB we have $w(u) = 4$. Then $\rho \geq \frac{9}{5} > \frac{7}{4}$, a contradiction.

□

4.2 Ring Topology

The result in Lemma 4.1 can be proven, though asymptotically even for ring topologies.

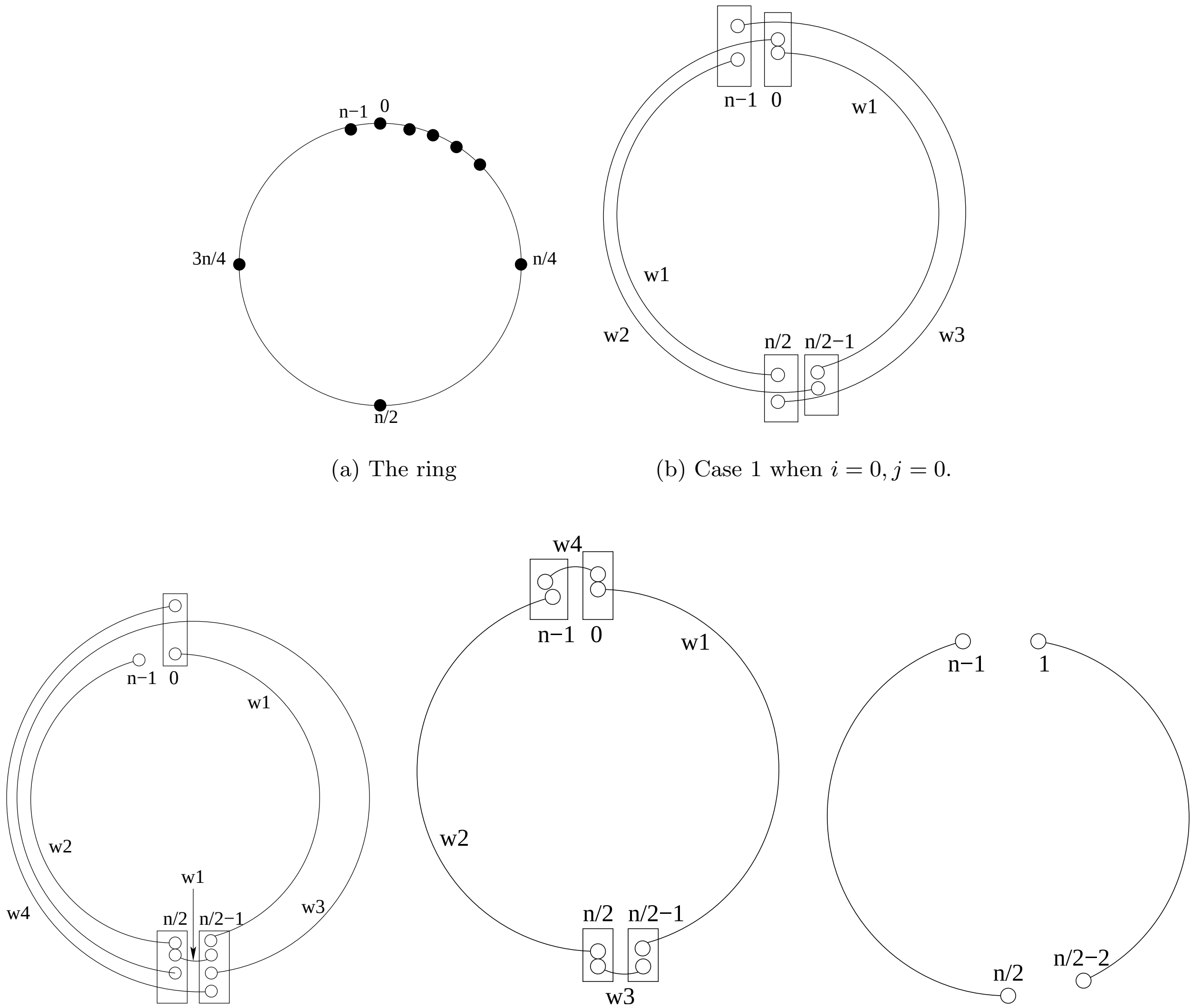
Lemma 4.2 *No deterministic on-line algorithm has a competitive ratio better than* $7/4$*, even for the ring topology.*

Overview. We first give the intuitive ideas behind the adversary. Suppose we divide the ring into four segments A, B, C and D. The adversary first requests lightpaths A and C.

1. If the on-line algorithm assigns the same color to A and C, we then request two lightpaths (B, C, D) and (D, A, B). The on-line algorithm has to use 2 new colors and thus uses 8 ADMs while the off-line algorithm can use 4 ADMs.
2. If the on-line algorithm assigns different colors to A and C, we then request B.
   (a) If the on-line algorithm assigns a third color to B, we further request D forcing the on-line algorithm to use at least 7 ADMs and the off-line algorithm to use 4 ADMs only.
   (b) If the on-line algorithm assigns one of the colors of A or C, w.l.o.g., assume the color of A is assigned, the adversary requests two lightpaths (B, C, D) and (C, D, A). Neither of these can share ADMs with existing lightpaths. The on-line algorithm uses 7 ADMs for lightpaths A, B, (B, C, D), and (C, D, A) plus 2 ADMs for C. The off-line algorithm uses 4 ADMs for A, B, (B, C, D), and (C, D, A) plus 2 ADMs for C.

The only problematic case for the adversary is 2(b). In this case the adversary then repeats in stages, taking the lightpaths A, B, (B, C, D) as the first stage, C as the second stage and proceed to the second stage by requesting A and repeating the above process k times, for some arbitrary large value k. If the on-line algorithm does not reuse any color from previous stages, then the on-line algorithm uses at least $7k+2$ ADMs and the off-line algorithm uses at most $4k+2$ ADMs. With sufficiently large k, this gives a competitive ratio at least $\frac{7}{4} - \epsilon$ for any $\epsilon > 0$. The question is how the adversary can force the on-line algorithm not to reuse colors in previous stages.

The rough idea is to shift the endpoints of the four segments A, B, C and D. By shifting the segment endpoints in a careful manner, we attempt to ensure lightpaths released in a certain stage either cannot share a common endpoint or cannot share a common color (due to overlapping edges) of lightpaths in previous stages. In this case, we can ensure no sharing of colors is possible across stages. In the following description,



(a) The ring

(b) Case 1 when $i = 0, j = 0$.

(c) Case 2 when $i = 0, j = 0$.

(d) Case 4 when $i = 0, j = 0$.

(e) The case when $i = 1, j = 0$.

Fig. 7. Different cases for the 7/4 lower bound for the ring topology.

the adversary runs in stages. In Stage k, the ring is partitioned into four (not necessarily equal) segments, A_k, B_k, C_k, and D_k. In subsequent stages, A_k and C_k keep shrinking while B_k and D_k keep extending. The extent of the shrink and extension in each stage is governed by how the on-line algorithm responds to the adversary. Details are as follows.

The adversary. Consider a ring with n nodes, named $0, 1, \cdots, n-1$ (see Figure 7 (a)). For simplicity, we assume that n is a multiple of 4. The adversary runs in stages, requesting lightpaths which are segments of the ring. For any $0 \le i, j < n/4$, we define the following segments. Note that addition and substraction are all modulo n.

- $A_{i,j} = \langle i, i+1, \cdots, n/2 - 1 - i \rangle$
- $B_{i,j} = \langle n/2 - 1 - i, n/2 - i, \cdots, n/2 + j \rangle$
- $C_{i,j} = \langle n/2 + j, n/2 + j + 1, \cdots, n - 1 - j \rangle$
- $D_{i,j} = \langle n - 1 - j, n - j, \cdots, i \rangle$

For example, when $i = j = 0$, the four segments are $A_{0,0} = \langle 0, 1, \cdots, n/2 - 1 \rangle$, $B_{0,0} = \langle n/2 - 1, n/2 \rangle$, $C_{0,0} = \langle n/2, n/2 + 1, \cdots, n - 1 \rangle$, and $D_{0,0} = \langle n - 1, 0 \rangle$. As i and j increase, $A_{i,j}$ and $C_{i,j}$ shrink while $B_{i,j}$

and $D_{i,j}$ extend. For any lightpath X, we denote the complement of X as $\bar{X}$ (X and $\bar{X}$ do not overlap and they together cover the ring).

Let k denote the current stage number, starting from 0. Starting from $i = j = 0$, the adversary runs as long as $i < n/4$ and $j < n/4$. In each stage, some of the lightpaths requested in this stage will be *marked*. Later on, we will show that marked lightpaths cannot share ADM with lightpaths in other stages (they either overlap or do not share common node). At the end of each stage, at least one of the values i and j is increased, so both B_k and D_k extend in each stage while at least one of A_k and C_k shrinks.

1. Set $k = i = j = 0$.
2. Set $A_k = A_{i,j}$, $B_k = B_{i,j}$, $C_k = C_{i,j}$, $D_k = D_{i,j}$.
3. Release two requests one for lightpath A_k and the other C_k if there is no such lightpaths released so far.
4. If $w(A_k) = w(C_k)$, // Case 1 (see Figure 7 (b))
   - Request lightpaths $\bar{A}_k$ and $\bar{C}_k$.
     The on-line algorithm has to use two different colors other than $w(A_k)$.
   - Mark A_k, C_k, $\bar{A}_k$ and $\bar{C}_k$.
   - Increment both i and j by 1.
     I.e., both A_k and C_k shrink in the next stage.
5. Otherwise, i.e., $w(A_k) \neq w(C_k)$, then request B_k.
   (a) If $w(A_k) = w(B_k)$, // Case 2 (see Figure 7 (c); and Figure 7 (e) for the next round)
      - Request $\bar{A}_k$ and $\bar{B}_k$.
        The on-line algorithm has to use two different colors other than $w(A_k)$ and $w(C_k)$.
      - Mark A_k, B_k, $\bar{A}_k$ and $\bar{B}_k$.
        I.e., only C_k is not marked.
      - Increment i by 1 and keep j unchanged.
        I.e., only A_k shrinks in the next stage.

   (b) If $w(B_k) = w(C_k)$, // Case 3, symmetric to the previous case
      - Request $\bar{B}_k$ and $\bar{C}_k$.
      - Mark B_k, C_k, $\bar{B}_k$ and $\bar{C}_k$.
      - Increment j by 1 and keep i unchanged.

   (c) If $w(A_k)$, $w(B_k)$, and $w(C_k)$ are all different, // Case 4 (see Figure 7 (d))
      - Request D_k.
      - Mark A_k, B_k, C_k, D_k.
      - Increment both i and j by 1.
        I.e., both A_k and C_k shrink in the next stage.
6. Increment k by 1 and repeat from Step 2 if $i < n/4$ and $j < n/4$.

Note that the total number of stages is between $n/4$ and $n/2$.

Analysis. The following observation and lemmas together prove Lemma 4.2. Observation 4.1 is a direct consequence of how the requests are released.

Observation 4.1 *At the end of each stage, at most one lightpath is unmarked.*

Lemma 4.3 *Consider a particular stage k, we have $s_k \geq (7/4)s_k^*$, where s_k and s_k^* are the number of ADMs used by the online algorithm and the optimal off-line algorithm, respectively, for the lightpaths marked in this stage.*

Proof. We consider the ratio by cases. Case 1: $s_k = 8$ and $s_k^* = 4$. Cases 2 and 3: $s_k = 7$ and $s_k^* = 4$. Case 4: $s_k \geq 7$ and $s_k^* = 4$. Therefore, the lemma follows. □

Lemma 4.4 *Suppose s and s^* are the number of ADM switches used by the online algorithm and the optimal off-line algorithm. Then $s/s^* \geq 7/4 - O(1/n)$.*

Proof. Note that $s^* \leq \sum_k s_k^* + 2$ since at most one lightpath is left unmarked at the end of the last stage. On the other hand, we claim that $s \geq \sum_k s_k$. With this claim, the fact that there are between $n/4$ and $n/2$ stages and Lemma 4.3, the lemma follows. To prove the claim, we will show that for any two lightpaths marked in two stages, they cannot share the same color.

Consider Stage k. If Stage k is of Case 1 or 4, then lightpaths requested in later stages do not share any common node with the lightpaths in Stage k. If Stage k is of Case 2 (the argument for Case 3 is similar),

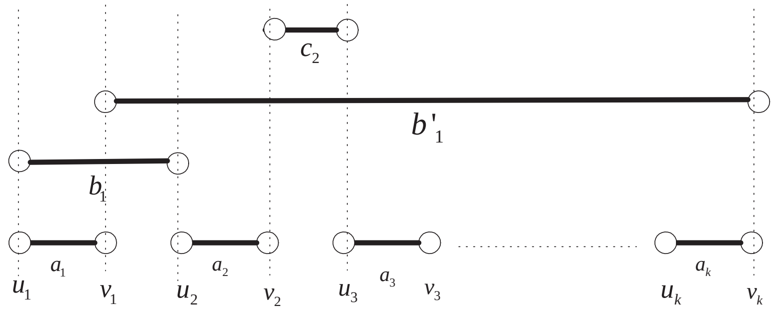


Fig. 8. Adversary for deterministic on-line algorithms in the path topology.

we can observe that among the marked lightpaths in Stage k, A_k and $\bar{A}_k$ do not share any common node with lightpaths in later stages. While $\bar{B}_k$ may share a common node with lightpaths in later stages, $\bar{B}_k$ also share a common edge with all these lightpaths and cannot share ADMs.

The remaining case is B_k. Suppose $h > k$. First, $B_k \asymp B_h$, $B_k \asymp \bar{A}_h$ and $B_k \asymp \bar{C}_h$. Since $w(B_k) = w(A_k)$, $A_k \asymp A_h$ and $A_k \asymp \bar{B}_h$, B_k cannot share the same color as either A_h or $\bar{B}_h$. Further, if Stage h is of Case 3, we need to consider B_k and C_h. Note that $w(C_h) = w(B_h)$ and $B_h \asymp B_k$, so again B_k cannot share the same color with C_h. Therefore, we can conclude that any lightpath marked in Stage k cannot share the same color with any lightpath marked in Stage h for $h > k$. □

4.3 Path Topology

Lemma 4.5 *For any $\epsilon > 0$, there is no $(\frac{3}{2} - \epsilon)$-competitive deterministic algorithm for path topology.*

Proof. Consult Figure 8 for the following description of the adversary. Let G be a path with $2k$ nodes $u_1, v_1, u_2, v_2, ..., u_k, v_k$. Let ALG be any deterministic algorithm. The value of k will be determined later.

The adversary works in two phases. In the first phase the input is $a_1, a_2, ..., a_k$ where $\forall i, a_i = (u_i, v_i)$. In the second phase the input depends on the decisions made by ALG during the first phase. For every $1 \leq i < k$, if $w(a_i) = w(a_{i+1})$ then the input contains two paths $b_i = (u_1, u_{i+1})$ and $b'_i = (v_i, v_k)$, otherwise the input contains one path $c_i = (v_i, u_{i+1})$. In Figure 8 the case $w(a_1) = w(a_2) \neq w(a_3)$ is depicted.

Let $0 \leq x \leq k-1$ be the number of times $w(a_i) = w(a_{i+1})$ is satisfied. Then $w(a_i) \neq w(a_{i+1})$ is satisfied $k - 1 - x$ times.

During the first phase the algorithm uses $2k$ ADMs, one for each node.

For the paths b_i and b'_i, let $\lambda = w(a_i)(= w(a_{i+1}))$. λ is not feasible neither for b_i nor for b'_i. Then the algorithm assigns other colors to b_i and b'_i, and it uses 4 ADMs, for a total of $4x$ ADMs.

For the path c_i, let $\lambda = w(a_i)$ and $\lambda' = w(a_{i+1})(\neq \lambda)$, coloring c_i with one of these colors ALG uses one ADM, otherwise it uses 2 ADMs. Therefore for the paths c_i, ALG uses at least $k - 1 - x$ ADMs.

Summing up, we get that ALG uses at least $2k + 4x + (k - 1 - x) = 3(k + x) - 1$ ADMs.

On the other hand the following solution is possible. For any consecutive paths $c_i, c_{i+1}, ..., c_{i+j}$ color such that $w(b_{i-1}) = w(a_i) = w(c_i) = w(a_{i+1}) = w(c_{i+1}) = ... = w(c_{i+j}) = w(a_{i+j+1}) = w(b'_{i+j+1})$. This solutions use $2k + 2x$ ADMs, one ADM at each u_i, v_i, x additional ADMs at u_1, and x additional ADMs at v_k.

Therefore the competitive ratio of ALG is at least $\frac{3(k+x)-1}{2(k+x)} = \frac{3}{2} - \frac{1}{2(k+x)} \geq \frac{3}{2} - \frac{1}{2k}$. For any $\epsilon > 0$ we can choose $k > \frac{1}{2\epsilon}$, so that the competitive ratio of ALG is bigger then $\frac{3}{2} - \epsilon$. □

5 Triangle Topology

In the previous sections we have shown that algorithm Online-MinADM has an optimal competitive ratio, in general topologies, ring and path topologies. In this section we show an example of topology for which Online-MinADM is not optimal. Note that the proof of Lemma 3.1 implies that Online-MinADM is $\frac{7}{4}$-competitive in the triangle topology. We will show in this section a tight bound of $\frac{5}{3}$ for this topology. Note that the lower bound proof for ring networks requires the ring to be of unbounded size. The proof does not hold for rings of a bounded size. In this section we show that this lower bound does not hold for triangles, and give an optimal algorithm for this topology.

Lemma 5.1 *There is no on-line algorithm with competitive ratio $< \frac{5}{3}$ for triangle topology.*

Proof. Consider a triangle with edge set $\{e_1, e_2, e_3\}$. We will use the following adversary.

Release two lightpaths each of length 1, on edges e_1 and e_2. If $w(e_1) = w(e_2)$, then we continue as in Lemma 3.1, namely release two lightpaths of length 2 each $\{(e_2, e_3), (e_1, e_3)\}$, and we get a competitive ratio of $7/4 > 5/3$.

Otherwise $w(e_1) \neq w(e_2)$, w.l.o.g. assume $w(e_1) = 1, w(e_2) = 2$. Release a lightpath on edge e_3. If $w(e_3) \notin \{1, 2\}$ then the competitive ratio is $6/3 = 2 > 5/3$, otherwise w.l.o.g $w(e_3) = 1$. In this case we have $w(e_1) = w(e_3) = 1$ using 3 ADMs, $w(e_2) = 2$ using 2 ADMs, for a total of 5 ADMs. The competitive ratio is $5/3$. □

5.1 Algorithm Online-Triangle

For the triangle topology, let us name the three edges in the triangle network e_1, e_2, and e_3. There are only six types of lightpaths, namely, (e_1), (e_2), (e_3), (e_1, e_2), (e_2, e_3) and (e_1, e_3). For any lightpath p, we say that p is *length*-1 if it contains one edge, and *length*-2 if it contains two edges.

We now present another algorithm Online-Triangle and show that it is 5/3-competitive for triangle topology. Roughly speaking, the algorithm gives highest priority to a pair of length-2 and length-1 lightpaths to share the same color whenever possible. For length-1 lightpaths, we have seen in the lower bound of Online-MinADM in Lemma 3.1 that, if an on-line algorithm always colors two adjacent length-1 lightpaths with the same color, the competitive ratio of the algorithm is at least $\frac{7}{4}$. To overcome this barrier, when a length-1 lightpath, say $p_i = (e_1)$, arrives, Online-Triangle does not always color p_i with an adjacent length-1 lightpath using the same color. However, if we color three length-1 lightpaths on a cycle each with a different color, this will result in a competitive ratio of 2. Therefore, if there are two lightpaths $p_j = (e_2)$ and $p_k = (e_3)$ with different colors, then Online-Triangle should color p_i with either of these colors if it is feasible. We formalize this concept by "marking" the three lightpaths to represent they are grouped together and should not be further considered when other length-1 lightpaths arrive. As we will show below, only length-1 lightpaths are marked.

Formally, the algorithm runs as follows. When a request of lightpath p_i with endpoints u_i and v_i arrives,

1. In case p_i is length-2,
   - If there exists a length-1 (marked or unmarked) lightpath with color λ with endpoints u_i, v_i, and λ is feasible for p_i, then assign $w(p_i) = \lambda$.
   - Otherwise, assign $w(p_i) = \lambda'$, where λ' is an unused color.
2. In case p_i is length-1,
   - If there exists a length-2 lightpath with color λ with endpoints u_i, v_i, and λ is feasible for p_i, then assign $w(p_i) = \lambda$.
   - Otherwise, if there exists a valid chain of two unmarked length-1 lightpaths with different colors λ_1 and λ_2 with endpoints u_i, v_i, and λ_1 or λ_2 is feasible for p_i (w.l.o.g. assume λ_1 is feasible), then assign $w(p_i) = \lambda_1$ and mark all three lightpaths involved.
   - Otherwise, assign $w(p_i) = \lambda'$, where λ' is an unused color.

Example: Suppose $P = \{p_1, p_2, \cdots, p_7\}$ where p_i is, in order, (e_1), (e_2), (e_3), (e_2), (e_1), (e_3), (e_1, e_3). Then Online-Triangle will first assign $w(p_1) = \lambda_1$, $w(p_2) = \lambda_2$, $w(p_3) = \lambda_1$ and mark all three p_1, p_2 and p_3. Next, we assign $w(p_4) = \lambda_3$ because there is no unmarked lightpath available. We further assign $w(p_5) = \lambda_4$ and $w(p_6) = \lambda_3$. Finally, we assign $w(p_7) = \lambda_2$ because p_7 and p_2 form a cycle.

5.2 Analysis of Online-Triangle

To analyze the performance of Online-Triangle, we first observe how lightpaths are colored in an optimal solution. The proof of the following lemma follows immediately from the definitions.

Lemma 5.2 *The optimal solution S^* always colors (e_1, e_2) and (e_3) with the same color if possible and similarly for the two other symmetric cases. Any remaining length-2 lightpath is colored a distinct color. If there are some length-1 lightpaths remained after this, cycles of three length-1 lightpaths are colored the same color; followed by chains of two length-1 lightpaths with same color and finally remaining length-1 lightpaths with distinct colors. It can be verified such coloring uses the minimum number of ADMs.*

Overview. We then compare S and S^* as follows. We first give a rough idea before formally prove it in Lemma 5.3. Roughly speaking, we consider different cases of how S and S^* color certain set of lightpaths and then compare the ratio case by case. In S, a length-2 lightpath X can always share an ADM with a length-1 lightpath unless the length-1 lightpath has been marked and assigned a color the same as another length-1 lightpath, say Y, which overlaps with X. In this case, S^* also has to use extra ADMs for this length-1 lightpath Y, and S^* and S will be shown to use a comparable number of ADMs. As mentioned before, Online-Triangle does not always color adjacent length-1 lightpaths using same color to avoid the $\frac{7}{4}$ lower bound. Nevertheless, in S, there is no marked cycle of length-1 lightpaths with three different colors. So for any marked cycle, S uses at most 5 ADMs while S^* uses at least 3, which will be shown later to be the worst case leading to the $\frac{5}{3}$-competitive ratio. When S^* has a chain of two length-1 lightpaths with the same color (using 3 ADMs), S needs at most 4 ADMS, giving a ratio of at most $\frac{4}{3}$.

Consider the solution S, the lightpaths can be partitioned into five disjoint sets according to how they are colored. We define these five sets, namely A, B, C, D and E, in this order and any lightpaths that have been classified in a set defined earlier would not be further considered in sets defined later. Let A be the set of cycles containing a length-1 lightpath and a length-2 lightpath with the same color; B be the set of remaining length-2 lightpaths (all with distinct colors). Then the lightpaths left are all length-1. Among these, let C be the set of marked cycles (i.e., two length-1 lightpaths of same color and a third lightpath of a different color); D be the set of remaining marked chains with two length-1 lightpaths of same color. Finally E contains the rest of the length-1 lightpaths. See Figure 9(a) for an illustration. In the example given above, A contains p_7 and p_2; C contains p_4, p_5 and p_6; D contains p_1 and p_3; B and E are empty. We denote the cardinality of A, B, C, D, and E by a, b, c, d and e, respectively. Note that $cost(S) = 2a + 2b + 5c + 3d + 2e$. We then make the following observations.

Property 1 W.l.o.g. the solution S^* satisfies (i) while S satisfies (ii) and (iii).

i. In S^*, every length-2 lightpath forms a cycle with a corresponding length-1 lightpath with the same color.
ii. In S, $d \le a$
iii. In S, $b \le 2d$; hence, $b \le 2a$.

Proof. (i) For any input sequence of lightpaths, if the number of lightpaths (e_1, e_2) is more than e_3, by Lemma 5.2, some lightpaths (e_1, e_2) in S^* will be colored a distinct color without sharing ADMs with other lightpaths; at least the same number of these lightpaths are also colored a distinct color in the solution S returned by Online-Triangle. Removing these lightpaths from the input sequence only increases the ratio $cost(S)/cost(S^*)$. Therefore, we can assume without loss of generality that the number of lightpaths (e_i, e_j) in P is at most that of (e_k) where i, j, k are all different. In other words, in S^*, every length-2 lightpath forms a cycle with a corresponding length-1 lightpath with the same color.

(ii) According to the way Online-Triangle colors lightpaths, we observe that $d \le a$ because each chain in D corresponds to one cycle in A.

(iii) Furthermore, by Lemma 5.2, S^* colors every lightpath X in B using the same color as some length-1 lightpath Y. Note that Y is in D, otherwise, Online-Triangle would have colored X and Y with the same color and then X is not in B. Since every chain in D contains two length-1 lightpaths, it may be paired with two length-2 lightpaths in B in S^*, therefore, we have $b \le 2d$, implying that $b \le 2a$, by (ii). □

Analysis. We are now ready to prove the competitive ratio of Online-Triangle in the triangle topology. The proof is a case analysis of the four cases depending on the set B.

Lemma 5.3 Online-Triangle *is $\frac{5}{3}$-competitive in the triangle topology.*

Proof. We consider four cases depending on the set B. It is useful to recall that Online-Triangle gives highest priority to color a length-2 lightpath and a length-1 lightpath that can form a cycle. The only case such length-2 lightpath cannot form a cycle of same color is only when all the length-1 lightpaths that are its complement have all been assigned the same color as another length-1 lightpath.

Case 1: B is empty, in other words, every length-2 lightpath is colored the same color as a length-1 lightpath (see Figure 9(b)); by Property 1 (i), this is also the case in S^*. Suppose there are s_1, s_2 and s_3 length-1 lightpaths (e_1), (e_2), (e_3), respectively, apart from those that have the same color as some length-2 lightpaths and belong to set A. W.l.o.g., we assume that $s_1 \geq s_2 \geq s_3$. For length-1 lightpaths, by Lemma 5.2, S^* colors all possible cycles of 3 lightpaths in the same color using 3 ADMs, then chains of 2 lightpaths with same color using 3 ADMs, and finally 1 lightpath with its own color using 2 ADMs. S^* uses $2a + 3s_3 + 3(s_2 - s_3) + 2(s_1 - s_2)$ ADMs. On the other hand, for S, $c = s_3$ and S uses 5 ADMs for C while D and E together contain $(s_2 - s_3) + (s_1 - s_3)$ length-1 lightpaths. So S uses at most $2(s_2 - s_3) + 2(s_1 - s_3)$ ADMs for D and E. Therefore S uses at most $2a + 5s_3 + 4(s_2 - s_3) + 2(s_1 - s_2)$ ADMs. We conclude that $\frac{cost(S)}{cost(S^*)} \leq \frac{2a+5s_3+4(s_2-s_3)+2(s_1-s_2)}{2a+3s_3+3(s_2-s_3)+2(s_1-s_2)} \leq \frac{5}{3}$.

Case 2: B contains all three types of length-2 lightpaths (see Figure 9(c)). In this case, both C and E must be empty, otherwise, there exists a length-2 lightpath $p \in B$ that Online-Triangle can color p with its complement in C or E to form a cycle and p should be in A instead. So, $cost(S) = 2a + 2b + 3d$. In this case S^* outperforms S by grouping lightpaths in B with lightpaths in D. Even if so, there are still $2d - b$ length-1 lightpaths left unpaired in D because there are $a + 2d$ length-1 lightpaths in P, $a + b$ of them share ADMs with A and B. Therefore, $cost(S^*) \geq 2a + 2b + (2d - b) = 2a + 2d + b$. Then $\frac{cost(S)}{cost(S^*)} \leq \frac{2a+2b+3d}{2a+2d+b} = 2 - \frac{2a+d}{2a+2d+b} \leq 2 - \frac{2a+d}{4a+2d} = \frac{3}{2}$; the last inequality is due to $b \leq 2a$, Property 1 (iii).

Case 3: B contains two types of length-2 lightpaths only, w.l.o.g., assume they are (e_1, e_2) and (e_2, e_3) (see Figure 9(d)). To simplify the discussion, we assume B contains the same number of each of them; the other case can be handled similarly. According to how Online-Triangle assigns color, C only contains the cycle consisting of (e_2) with one color and $(e_1), (e_3)$ with another color; otherwise, there exists a lightpath $p \in B$ that Online-Triangle would assign the same color as its complement in C to form a cycle and then p should be in A instead. Similarly, E only contains (e_2). In S^*, lightpaths in B must be colored the same color as some lightpaths in C or D (by Lemma 5.2). The number of (e_2) left after all length-2 lightpaths are colored in S^* equals to $c+d+e$, while the total number of (e_1) and (e_3) left equals to $2c+d-b$. The number of cycles formed from the remaining length-1 lightpaths is at most $\frac{2c+d-b}{2}$. Therefore, $cost(S^*) \geq (2a+2b) + \frac{3(2c+d-b)}{2} + 2(c + d + e - \frac{2c+d-b}{2}) = 2a + 3b/2 + 3c + 5d/2 + 2e$. Hence, $\frac{cost(S)}{cost(S^*)} \leq \frac{2a+2b+5c+3d+2e}{2a+3b/2+3c+5d/2+2e} \leq \frac{5}{3}$.

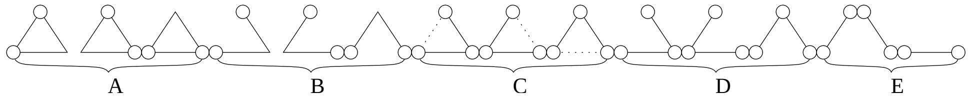
Case 4: B contains one type of length-2 lightpaths only; w.l.o.g., assume it is (e_1, e_2) (see Figure 9(e)). Then C does not contain the cycle consisting of (e_3) with one color and $(e_1), (e_2)$ with another color while E does not contain the lightpath (e_3). In S^*, lightpaths in B must be colored the same color as some lightpaths in C or D (by Lemma 5.2). The number of (e_3) left after all length-2 lightpaths are colored in S^* equals to $c + d - b$, while the total number of (e_1) and (e_2) left equals to $2c + d + e$. Therefore, $cost(S^*) \geq (2a+2b)+3(c+d-b)+\frac{3(2c+d+e-2(c+d-b))}{2} = 2a+2b+3c+3d/2+3e/2 \geq (6a/5+4d/5)+2b+3c+3d/2+3e/2$ because $d \leq a$, by Property 1 (ii). Then $\frac{cost(S)}{cost(S^*)} \leq \frac{2a+2b+5c+3d+2e}{6a/5+2b+3c+23d/10+3e/2} \leq \frac{5}{3}$. □

Theorem 5.1 *Algorithm* Online-Triangle *is optimal for triangle topology, with competitive ratio of* $\frac{5}{3}$.

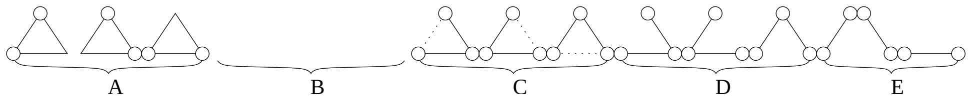
6 Conclusion and Possible Improvements

In this paper we presented an on-line algorithm with competitive ratio of $\frac{7}{4}$ for any network topology, and proved that no algorithm has a competitive ratio better than $\frac{7}{4}$, even if the topology is a ring. We showed that the same algorithm has a competitive ratio of $\frac{3}{2}$ in path topologies, and that this is also a lower bound for any on-line algorithm on this topology. The lower bound on ring topology does not hold when the ring is of a bounded size; we showed an optimal bound of $\frac{5}{3}$ for the competitive ratio for the triangle topology, using a different algorithm as a first attempt to provide tight bounds for rings of bounded size. The analyses of the upper bounds, as well as those for the lower bounds, are all using a variety of proof techniques, which are of interest by their own, and which might prove helpful in future research on the topic.

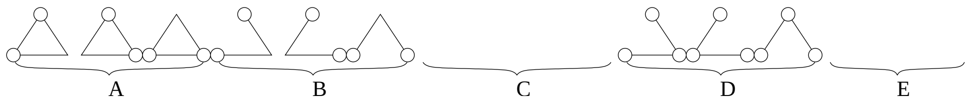
Our bounds pertain to deterministic on-line algorithms. It may be interesting to explore probabilistic algorithms and obtain better bounds. Following our study, it might be interesting to determine the exact complexity of the on-line problem for tree topologies, as a function of some parameter of the tree, and of networks (e.g., rings or paths) of bounded size. An important extension is to consider the on-line version of



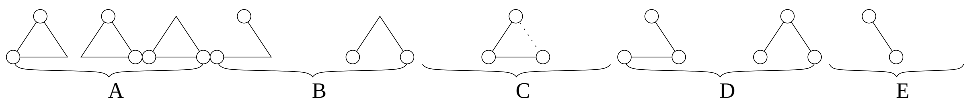
(a) Partition of lightpaths in S



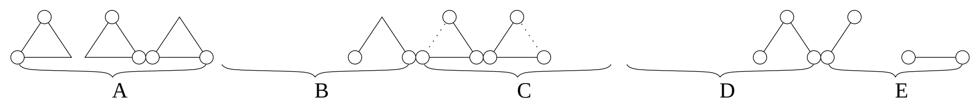
(b) Case 1: B is empty.



(c) Case 2: B contains all three types of lightpaths.



(d) Case 3: B contains two types of lightpaths.



(e) Case 4: B contains one type of lightpaths.

Fig. 9. Different cases for the analysis of Online-Triangle.

the problem when grooming is allowed; in graph-theoretic terms, this amounts to coloring the paths so that at most g of them are crossing any edge, and where each ADM can serve up to g paths that come from at most two of its adjacent edges (see [GRS98,ZM03]). Another direction of extension is to the case where more involved switching functions are under consideration.

References

[BBMELH08] K. Benson, B. Birnbaum, E. Molina-Estolano, and R. Libekind-Hadas. Competitive analysis of online traffic grooming in WDM rings. *IEEE/ACM Transactions on Networking*, 16(4):984–997, August 2008.

[BEY98] Allan Borodin and Ran El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, The Pitt Building, Trumpington Street, Cambridge, United Kingdom, 1998.

[CFW02] G. Călinescu, O. Frieder, and P.-J. Wan. Minimizing electronic line terminals for automatic ring protection in general WDM optical networks. *IEEE Journal of Selected Area on Communications*, 20(1):183–189, Jan 2002.

[CW02] G. Călinescu and P-J. Wan. Traffic partition in WDM/SONET rings to minimize SONET ADMs. *Journal of Combinatorial Optimization*, 6(4):425–453, 2002.

[EL04] L. Epstein and A. Levin. Better bounds for minimizing SONET ADMs. In *2nd Workshop on Approximation and Online Algorithms, Bergen, Norway*, September 2004.

[EMZ02] T. Eilam, S. Moran, and S. Zaks. Lightpath arrangement in survivable rings to minimize the switching cost. *IEEE Journal of Selected Area on Communications*, 20(1):172–182, Jan 2002.

[FSZ06] M. Flammini, M. Shalom, and S. Zaks. On minimizing the number of ADMs in a general topology optical network. In *20th International Symposium on Distributed Computing (DISC), Stockholm, Sweden*, pages 459–473, September 2006. To appear in Discrete Applied Mathematics.

[GLS98] O. Gerstel, P. Lin, and G. Sasaki. Wavelength assignment in a WDM ring to minimize cost of embedded SONET rings. In *INFOCOM'98, Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 69–77, 1998.

[GRS98] O. Gerstel, R. Ramaswami, and G. Sasaki. Cost effective traffic grooming in WDM rings. In *INFOCOM'98, Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies*, 1998.

[LV98] Stefano Leonardi and Andrea Vitaletti. Randomized lower bounds for online path coloring. In *RANDOM '98: Proceedings of the Second International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 232–247, London, UK, 1998. Springer-Verlag.

[SWZ07] M. Shalom, P.W.H. Wong, and S. Zaks. Optimal on-line colorings for minimizing the number of ADMs in optical networks. In *21st International Symposium on Distributed Computing (DISC), Lemesos, Cyprus*, pages 435–449, September 2007.

[SZ04] M. Shalom and S. Zaks. A $10/7 + \epsilon$ approximation scheme for minimizing the number of ADMs in SONET rings. In *First Annual International Conference on Broadband Networks, San-José, California, USA*, pages 254–262, October 2004. To appear in IEEE Trans. on Networking.

[WCLF00] P.J. Wan, G. Călinescu, L.-W. Liu, and O. Frieder. Grooming of arbitrary traffic in SONET/WDM BLSRs. *IEEE Journal of Selected Area on Communications*, 18(10):1995–2003, 2000.

[ZM03] K. Zhu and B. Mukherjee. A review of traffic grooming in WDM optical networks: Architecture and challenges. *Optical Networks Magazine*, 4(2):55–64, March-April 2003.