J Sched (2018) 21:327–336
https://doi.org/10.1007/s10951-017-0514-4

Flexible bandwidth assignment with application to optical networks

Hadas Shachnai[1] · Ariella Voloshin[1] · Shmuel Zaks[1]

Published online: 4 March 2017

Abstract We introduce two scheduling problems, the *flexible bandwidth allocation problem* (FBAP) and the *flexible storage allocation problem* (FSAP). In both problems, we have an available resource, and a set of requests, each consists of a minimum and a maximum resource requirement, for the duration of its execution, as well as a profit accrued per allocated unit of the resource. In FBAP, the goal is to assign the available resource to a feasible subset of requests, such that the total profit is maximized, while in FSAP we also require that each satisfied request is given a contiguous portion of the resource. Our problems generalize the classic *bandwidth allocation problem* (BAP) and *storage allocation problem* (SAP) and are therefore NP-hard. Our main results are a 3-approximation algorithm for FBAP and a $(3+\epsilon)$-approximation algorithm for FSAP, for any fixed $\epsilon > 0$. These algorithms make nonstandard use of the *local ratio* technique. Furthermore, we present a $(2+\epsilon)$-approximation algorithm for SAP, for any fixed $\epsilon > 0$, thus improving the best known ratio of $\frac{2e-1}{e-1} + \epsilon$. Our study is motivated also by critical resource allocation problems arising in all-optical networks.

Keywords Approximation algorithms · Local ratio · Resource allocation · All-optical networks

A preliminary version of this paper appeared in the proceedings of the 39th *International Symposium on Mathematical Foundations of Computer Science(MFCS), 2014*.

✉ Ariella Voloshin
variella@cs.technion.ac.il

Hadas Shachnai
hadas@cs.technion.ac.il

Shmuel Zaks
zaks@cs.technion.ac.il

1 Department of Computer Science, Technion IIT, 3200003 Haifa, Israel

1 Introduction

1.1 Background

Scheduling activities with resource demands arise in a wide range of applications. In these problems, we have a set of activities competing for a reusable resource. Each activity utilizes a certain amount of the resource for the duration of its execution and frees it upon completion. The problem is to find a feasible schedule for a subset of the activities which satisfies certain constraints, including the requirement that the total amount of resource allocated simultaneously for executing activities never exceeds the amount of available resource. Two classic problems that fit in this scenario are the *bandwidth allocation problem* (BAP) and the *storage allocation problem* (SAP). In BAP, the goal is to assign the available resource to a feasible subset of activities, such that the total profit is maximized, while in SAP it is also required that any satisfied activity is given the same contiguous portion of the resource for its entire duration (for references and further discussion see Sect. 1.4). We introduce two variants of these problems where each activity has a minimum and a maximum possible request size, as well as a profit per unit of the resource allocated to it. We refer to these variants as the *flexible bandwidth allocation problem* (FBAP) and the *flexible storage allocation problem* (FSAP).

1.2 Problem statement

In this work, we study FBAP and FSAP on a path network. In graph-theoretical terms, the input for FBAP and FSAP consists of a path $P = (V, E)$ and a set $\mathcal{I}$ of n intervals on P. Each interval $I \in \mathcal{I}$ requires the utilization of a given, limited, resource. The amount of resource available, denoted by $W > 0$, is fixed over P. Each interval $I \in \mathcal{I}$ is defined by the following parameters: (i) a left endpoint, $l(I) \geq 0$, and a right endpoint, $r(I) > l(I)$. Thus, I is associated with the half-open interval $[l(I), r(I))$ on P. (ii) The amount of resource range required by each interval I, where $a(I)$, $b(I)$ are integers satisfying $0 \leq a(I) \leq b(I) \leq W$. Thus, I can take any value in the *possible range* for I, given by $[a(I), b(I)]$, or 0. (iii) The profit $w(I)$ gained for each unit of the resource allocated to I, where $w(I) \geq 0$ is an integer.[1]

A *feasible* allocation has to satisfy the following conditions. (i) Each assigned interval $I \in \mathcal{I}$ is allocated an amount of the resource in its possible range or is not allocated at all. (ii) The specific resources allocated to an interval are fixed along the interval. (iii) The total amount of the resource allocated at any time does not exceed the available amount W. In FBAP, we seek a feasible allocation which maximizes the total profit accrued by the intervals. In FSAP, we add the requirement that the allocation to each interval is a contiguous block of the resource for the entire duration. Note that a solution for FSAP is a solution for FBAP, while the converse is not necessarily true.

Example Consider a path consisting of the nodes $\{0, 1, \ldots, 5\}$, and the set of intervals in the form $I = ([l(I), r(I)), a(I), b(I), w(I))$, $I_1 = ([0, 1), 1, 2, 2)$, $I_2 = ([0, 2), 1, 2, 2)$, $I_3 = ([1, 3), 1, 1, 2)$, $I_4 = ([1, 4), 1, 1, 2)$, $I_5 = ([0, 4), 1, 1, 1)$, $I_6 = ([2, 4), 1, 1, 2)$, $I_7 = ([3, 5), 1, 2, 2)$, and $I_8 = ([4, 5), 1, 2, 2)$. The amount of resource available is $W = 4$. A possible FBAP and FSAP solution is illustrated in Fig. 1a, it has a total profit of 19. Figure 1b illustrates a better FBAP solution, which assigns one more resource block to I_2 and I_7, while the interval I_5 is not assigned at all. This assignment has a total profit of 22. We note that this cannot be a feasible solution for FSAP, since interval I_8 is assigned a non-contiguous block of the resource. To have a feasible FSAP solution, the assignment for interval I_8 is reduced to one resource unit, as illustrated in Fig. 1c. This assignment has a total profit of 20.

Approximation algorithms We develop approximation algorithms and analyze their worst case performance. For $\rho \geq 1$, a ρ-*approximation* algorithm for a maximization problem Π yields in polynomial time a solution whose value is at least within a factor $1/\rho$ of the optimum, for any input for Π.

[1] We note that our results can be adapted also to instances when $a(I)$, $b(I)$ and $w(I)$ are non-integers.

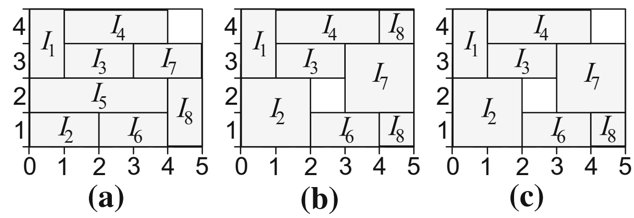

Fig. 1 Example for FBAP and FSAP, where **a** is a possible FBAP and FSAP solution, **b** is an optimal FBAP solution, and **c** is an optimal FSAP solution

1.3 Applications of FSAP and FBAP

The problems FSAP and FBAP have important applications in real-time scheduling. Consider, for example, a reusable resource of fixed size and activities that have a minimum and a maximum range for contiguous or non-contiguous resource requirement. The resource may be memory, computing units, servers in a Cloud, or network bandwidth. The allocated amount of resource for the activities actually determines it performance, quality of service, or processing time. In the following, we present the application of FBAP and FSAP in optimizing spectrum assignment in all-optical networks.

Spectrum assignment in all-optical networks In modern optical networks, several high-speed signals are sent through a single optical fiber. A signal transmitted optically from some source node to some destination node over a wavelength is termed *lightpath* (for comprehensive surveys on optical networks, see, e.g., Ramaswami et al. 2009; Ali Norouzi and Ustundag 2011). Traditionally, the spectrum of light that can be transmitted through the fiber has been divided into frequency intervals of *fixed* width, with a gap of unused frequencies between them. In this context, the term wavelength refers to each of these predefined frequency intervals. An emerging architecture, which moves away from the rigid model toward a flexible model, was suggested in Jinno et al. (2009), Gerstel (2010). In this model, the usable frequency intervals are of *variable* width (even within the same link). Every lightpath has to be assigned a frequency interval (sub-spectrum), which remains fixed through all of the links it traverses. As in the traditional model, two different lightpaths using the same link must be assigned disjoint sub-spectra. This technology is termed *flex-grid* (or *flex-spectrum*), as opposed to the *fixed-grid* (or *fixed-spectrum*) traditional technology. The network implications of this new architecture are described in detail in Gerstel (2011). The following spectrum assignment problems arising in the fixed-grid and flex-grid technology correspond to FBAP and FSAP, respectively. We are given a set of flexible connection requests, each with a lower and upper bound on the width of its spectrum request, as well as an associated positive profit per allocated spectrum unit. In the fixed-grid (or flex-grid) technology, the goal is to find a non-contiguous (or contiguous) spectrum assignment for a subset of requests that maximizes the total profit. A detailed description of this module is given in Shachnai et al. (2014).

1.4 Related work

Bandwidth allocation problem (BAP) We are given a network having some available bandwidth and a set of connection requests. Each request consists of a path in the network, a bandwidth requirement, and a weight. The goal is to feasibly assign bandwidth to a maximum weight subset of requests. BAP is strongly NP-hard even for uniform profit on a path network Chrobak et al. (2012). Bar-Noy et al. (2001) presented a 3-approximation algorithm for the problem. Călinescu et al. (2011) gave a randomized approximation algorithm with expected performance ratio of $2+\epsilon$, for any $\epsilon > 0$. The best known result is a deterministic $(2+\epsilon)$-approximation algorithm due to Chekuri et al. (2007). The generalized version of BAP, in which edge capacities are non-uniform, is known as the unsplittable flow problem (UFP). Bansal et al. (2006) developed a deterministic quasi-polynomial time approximation scheme for UFP on the line, assuming a quasi-polynomial bound on all edge capacities and demands in the input instance.

Storage allocation problem (SAP) In this special case of BAP, we require that each activity is allocated a single contiguous block of resource for all of its edges. SAP is NP-hard. It was first studied by Bar-Noy et al. (2001) and by Leonardi et al. (2000). An approximation algorithm of Bar-Noy et al. (2001) yields a ratio of 7. Chen et al. (2002) studied the special case in which all resource requirements are multiples of $1/K$, for some integer $K \geq 1$. They presented an $O(n(nK)^K)$ time dynamic programming algorithm to solve this special case and also gave an approximation algorithm with ratio $\frac{e}{e-1}+\epsilon$, for any $\epsilon > 0$, assuming the maximum resource requirement of any activity is $O(1/K)$. Bar-Yehuda et al. (2009) presented a randomized $(2+\epsilon)$-approximation algorithm for SAP, along with a deterministic $(\frac{2e-1}{e-1}+\epsilon)$-approximation algorithm, for any fixed $\epsilon > 0$. Mömke and Wiese (2015) studied the generalized version of SAP, in which edge capacities are non-uniform. They presented a randomized LP-based approximation algorithm with expected performance ratio of $2+\epsilon$, for any $\epsilon > 0$.

The flex non-contiguous (FNC) and flex contiguous (FC) problems FNC and FC are restricted variants of FBAP and FSAP, respectively, in which all intervals have to be assigned an amount of resource in their required range, i.e., for each interval $I \in \mathcal{I}$, the amount of the assigned resource is at least $a(I)$. We note that the special case of FNC and FC in which $a(I)=0$, for all $I \in \mathcal{I}$, is also a special case of FBAP and FSAP, respectively. The papers Shalom et al. (2013), Shachnai et al. (2014), Katz et al. (2016) consider FNC and FC. Shalom et al. (2013) showed that FNC is polynomially solvable. In contrast, Shachnai et al. (2014) observed that FC cannot be approximated within any bounded ratio, unless $P = NP$. They showed that FC is NP-hard in the strong sense for the subclass of instances where $a(I)=0$ for all $I \in \mathcal{I}$, and presented a $(2+\epsilon)$-approximation algorithm for such instances, for any fixed $\epsilon > 0$. For this subclass, Katz et al. (2016) presented a $(5/4+\epsilon)$-approximation algorithm in the special case where the input is a proper interval graph, for any fixed $\epsilon > 0$. The paper Katz et al. (2016) also extends our hardness result for FSAP, as given in the preliminary version of this paper, by showing that FC is NP-hard even if for all $I \in \mathcal{I}$, $a(I)=0$, $b(I)=\text{Max}$ for some $1 \leq \text{Max} \leq W$, and $w(I)=1$. For this special case, the authors obtain a $(\frac{2k}{2k-1})$-approximation algorithm, where $k=\lceil \frac{W}{Max} \rceil$, and show that this subclass admits a *polynomial time approximation scheme (PTAS)*.

1.5 Our results

We study the scheduling problems FBAP and FSAP. We note that both problems are NP-hard and show that, in fact, FSAP is NP-hard in the strong sense for any instance $\mathcal{I}$ where $a(I) \neq b(I)$ for all $I \in \mathcal{I}$. We first give a 3-approximation algorithm for FBAP. We then show that this algorithm can be extended to yield a $(3+\epsilon)$-approximation for FSAP, for any fixed $\epsilon > 0$. Finally, we consider SAP, the special case of FSAP where $a(I) = b(I)$ for all $I \in \mathcal{I}$. We present a $(2+\epsilon)$-approximation algorithm, for any fixed $\epsilon > 0$, thus improving the best known ratio of $\frac{2e-1}{e-1}+\epsilon$, due to Bar-Yehuda et al. (2009).

Techniques In developing our approximation algorithm for FBAP, we make nonstandard use of the *local ratio* technique. In particular, we apply the technique for a maximization problem, where instances are associated with profit per unit vectors, and the output solution has a continuous as well as discrete component. Indeed, this is due to the fact that the amount of resource allocated to each request can either be in its possible range, or else equal to zero. To the best of our knowledge, this interpretation of the local ratio technique is given here for the first time. We note that a straightforward application of an algorithm of Bar-Noy et al. (2001), combined with allocation of integral powers of $(1+\epsilon)$ in the range $[a(I), b(I)]$, for any $I \in \mathcal{I}$, yields a polynomial time $(4+\epsilon)$-approximation for FBAP. Our algorithm, which allows *any* allocation in the continuous range $[a(I), b(I)]$, has better running time and yields an improved ratio of 3.

Organization of the paper In Sect. 2, we give some definitions and notation, our proof of hardness for FSAP and a short overview of the local ratio technique. We study FBAP in Sect. 3, FSAP in Sect. 4, and SAP in Sect. 5. We conclude in Sect. 6 with a summary and directions for future work.

2 Preliminaries

Throughout the paper, we use graph-theoretical coloring terminology. Specifically, the set of requests $\mathcal{I}$ is represented as a set of n intervals on a path $P = (V, E)$. For an interval I, we denote by $l(I)$ and $r(I)$ its left endpoint and right endpoint, respectively. The amount of available resource, W, can be viewed as the amount of available distinct colors. Each interval $I \in \mathcal{I}$ has a minimum $a(I)$ and a maximum $b(I)$ color requirements, $0 \le a(I) \le b(I) \le W$, and a positive profit per allocated color (or profit per unit) $w(I)$, where $a(I), b(I)$ and $w(I)$ are nonnegative integers.

The set of available colors is $\Lambda = \{1, \ldots, W\}$. A contiguous range of colors is any set $\Lambda_i^j = \{t : 1 \le i \le t \le j \le W\}$, and is termed an *interval* of colors. A *(multi)coloring* is a function $\sigma : \mathcal{I} \mapsto 2^{\Lambda}$ that assigns to each interval $I \in \mathcal{I}$ a subset of the set Λ of colors. A coloring σ is *feasible* if for every $I \in \mathcal{I}$ $a(I) \le |\sigma(I)| \le b(I)$, or $|\sigma(I)| = 0$, and for any two intervals $I, I' \in \mathcal{I}$ such that $I \cap I' \ne \emptyset$ we have $\sigma(I) \cap \sigma(I') = \emptyset$. A *contiguous color assignment* is a coloring σ that assigns a contiguous range of colors. For any disjoint subsets $\mathcal{I}', \mathcal{I}'' \subseteq \mathcal{I}$, a coloring function σ for $\mathcal{I}'$ can be expanded to a coloring function $\overline{\sigma}$ for $\mathcal{I}' \cup \mathcal{I}''$ such that $\overline{\sigma}(I) = \sigma(I)$ for each $I \in \mathcal{I}'$ and $\overline{\sigma}(I) = \emptyset$ for each $I \notin \mathcal{I}'$. The total profit of a feasible coloring σ of $\mathcal{I}' \subseteq \mathcal{I}$ is given by $profit^{\sigma}(\mathcal{I}') \stackrel{def}{=} \sum_{I \in \mathcal{I}'} |\sigma(I)| \cdot w(I)$. When there is no ambiguity regarding the set of intervals, we simply write $profit^{\sigma}$. We denote by $E(I)$ the set of edges that are contained in I. For an edge e, we denote by $\mathcal{I}_e$ the subset of $\mathcal{I}$ consisting of the intervals containing e.

The problems We first introduce the following coloring problem FBAP:

> FBAP($\mathcal{I}$, W)
> **Input:** A tuple $(\mathcal{I}, W)$, where $\mathcal{I}$ is a set of intervals, and W is a positive integer.
> **Output:** A feasible coloring function σ for $\mathcal{I}$.
> **Objective:** Maximize $profit^{\sigma}(\mathcal{I})$.

A solution S for FBAP consists of the intervals that were assigned at least one color, i.e., a set of pairs, where the first entry of each pair is an interval $I \in \mathcal{I}$, and the second entry is the interval coloring size $|\sigma(I)|$.

The second problem is the contiguous color assignment variant, FSAP, in which the goal is to find a feasible contiguous coloring σ for $\mathcal{I}$ that maximizes $profit^{\sigma}(\mathcal{I})$.

> FSAP($\mathcal{I}$, W)
> **Input:** A tuple $(\mathcal{I}, W)$, where $\mathcal{I}$ is a set of intervals, and W is a positive integer.
> **Output:** A feasible contiguous coloring function σ for $\mathcal{I}$.
> **Objective:** Maximize $profit^{\sigma}(\mathcal{I})$.

A solution S for FSAP consists of the intervals that were assigned at least one color, i.e., a set of triples, where the first entry of each triple is an interval $I \in \mathcal{I}$, the second entry is the interval coloring size $|\sigma(I)|$, and the third entry is its lowest color index.

Given a solution S to FSAP (or FBAP), we denote by $\mathcal{I}_S$ the intervals of S and by σ_S their coloring function. Note that for FBAP it is impossible to return a solution of the exact color assignment σ in polynomial time. For example, suppose we are given a path and a set of intervals $\mathcal{I} = \{I\}$ where $a(I) = 0$ and $b(I) = W$. Presenting a coloring solution σ such that $|\sigma(I)| = polylog(W)$ is not polynomial in the input size. Therefore, following the description of the algorithm for FBAP, we explain how to achieve the exact color assignment.

We note that BAP and SAP are special cases of FBAP and FSAP, respectively (where $a(I) = b(I)$ for every $I \in \mathcal{I}$). BAP and SAP are NP-hard since they include the *Knapsack* problem as a special case, where all intervals share the same edge; thus, we have that FSAP and FBAP are NP-hard. In addition, following the hardness result of Katz et al. (2016), FSAP is NP-hard for the subclass in which for all $I \in \mathcal{I}$, $a(I) = 0, b(I) = \text{Max}$ for some $1 \le \text{Max} \le W$, and $w(I) = 1$. For another subclass, where $a(I) \ne b(I)$ for all $I \in \mathcal{I}$, we show that the problem remains NP-hard in the strong sense, even where all intervals have the same unit profit.

Lemma 1 *FSAP is NP-hard in the strong sense, even for uniform profit instances, where $a(I) \ne b(I)$ for all $I \in \mathcal{I}$.*

Proof We show that the decision version of FSAP is NP-complete.

FSAP($\mathcal{I}$, W, B) Decision Problem
Input: A tuple $(\mathcal{I}, W, B)$, where $\mathcal{I}$ is a set of intervals, and W and B are positive integer.
Question: Is there a feasible contiguous coloring function σ for $\mathcal{I}$ such that $profit^{\sigma}(\mathcal{I}) \ge B$?

The reduction is from the *dynamic storage allocation (DSA)* problem. The DSA problem deals with packing a given set of rectangles that can only move vertically, into a horizontal strip of minimum height, such that no two rectangles overlap. The problem is known to be NP-complete in the strong sense (problem SR2 in Garey and Johnson 1979). Formally, the input for the DSA problem is a tuple $(\mathcal{R}, D)$, where $\mathcal{R}$ is the set of items to be stored; each item $r \in \mathcal{R}$ has a size $s(r) \in \mathbb{Z}^+$, an arrival time $ar(r) \in \mathbb{Z}_0^+$, and a departure time $de(r) \in \mathbb{Z}^+$. $D \in \mathbb{Z}^+$ is the storage size. The goal is to determine whether there is a feasible allocation of storage for $\mathcal{R}$, i.e., a function $h : \mathcal{R} \mapsto \{1, 2, \ldots, D\}$, such that (i) for each $r \in \mathcal{R}$, the allocated storage interval $I(r) = \{h(r), \ldots, h(r) + s(r) - 1\}$ is contained in $\{1, \ldots, D\}$, and (ii) for each $r, r' \in \mathcal{R}$, if $I(r) \cap I(r') \ne \emptyset$ then either $de(r) \le ar(r')$ or $de(r') \le ar(r')$.

Given an instance $I = (\mathcal{R}, D)$ of the DSA problem, we form a corresponding instance $I' = (\mathcal{I}, W, B)$ of FSAP as follows. For each $r \in \mathcal{R}$, we define an interval I_r where $l(I_r) = ar(r)$, $r(I_r) = de(r)$, $a(I_r) = s(r) - 1$, $b(I_r) = s(r)$, and $w(I_r) = 1$. We set the amount of available colors $W = D$, and finally set the profit $B = \sum_{r \in \mathcal{R}} s(r)$. We have to show that there exists a solution for I iff there exists a solution for I'.

Assume that there is a feasible allocation storage function h for I as described above. Then, the contiguous color assignment σ for the corresponding interval $I_r \in \mathcal{I}$ is $\sigma(I_r) = \{h(r), \ldots, h(r) + s(r) - 1\}$. The profit of this allocation is $\sum_{I \in \mathcal{I}} |\sigma(P)| \cdot 1 = \sum_{r \in \mathcal{R}} s(r) = B$. Therefore, there exists a solution for I'.

Conversely, suppose there is a feasible contiguous coloring function σ for I' with profit of at least $B = \sum_{r \in \mathcal{R}} s(r)$. Thus, each interval $I_r \in \mathcal{I}$ got a color interval of size $b(I_r)$ as follows: $\sigma(I_r) = \{i, \ldots, i + b(I_r) - 1\}$ for some i, $1 \leq i \leq W + 1 - b(I_r)$. Then, the solution for the DSA problem, i.e., the function h for the corresponding item $r \in \mathcal{R}$ is $h(r) = i$. This allocation is contained in $\{1, \ldots, D\}$. In addition, since σ is a feasible contiguous color assignment, for each $r, r' \in \mathcal{R}$, if $I(r) \cap I(r') \neq \emptyset$ then either $de(r) \leq ar(r')$ or $de(r') \leq ar(r)$. Therefore, there is a feasible allocation of storage for I. We note that this is a polynomial time transformation. In addition, given a solution for FSAP, its correctness can be verified in polynomial time; therefore, the FSAP decision problem is NP-complete. □

Narrow and wide intervals In deriving our approximation results, for a given set of interval requests, we form two new interval sets. We solve separately the problem for each set, and then the solution of higher profit is expanded into a solution for the original instance. Formally, given a set of intervals $\mathcal{I}$ and a parameter $\delta \in (0, 1]$, we form two sets of intervals $\mathcal{I}_{\text{narrow}}$ and $\mathcal{I}_{\text{wide}}$ as follows. For any $I \in \mathcal{I}$ for which $a(I) \leq \delta W$, we define an interval I_{narrow} with the same left and right endpoint as I, $a(I_{\text{narrow}}) = a(I)$, $b(I_{\text{narrow}}) = \min\{b(I), \lfloor \delta W \rfloor\}$, and $w(I_{\text{narrow}}) = w(I)$. We call this set of intervals $\mathcal{I}_{\text{narrow}}$. For any $I \in \mathcal{I}$ for which $b(I) > \delta W$, we define an interval I_{wide} with the same left and right endpoint as I, $a(I_{\text{wide}}) = \max\{a(I), \lfloor \delta W \rfloor + 1\}$, $b(I_{\text{wide}}) = b(I)$, and $w(I_{\text{wide}}) = w(I)$. This set of intervals is termed $\mathcal{I}_{\text{wide}}$.

Given feasible colorings of $\mathcal{I}_{\text{narrow}}$ and $\mathcal{I}_{\text{wide}}$, we choose the set that yields higher profit and assign the same colors to the corresponding intervals in $\mathcal{I}$. Formally, the feasible coloring σ_{narrow} (or σ_{wide}) of the instance $(\mathcal{I}_{\text{narrow}}, W)$ (or $(\mathcal{I}_{\text{wide}}, W)$) is expanded it to a feasible coloring $\sigma^{\text{expand}}_{\text{narrow}}$ (or $\sigma^{\text{expand}}_{\text{wide}}$) for $(\mathcal{I}, W)$ such that, for any $I \in \mathcal{I}$, if $a(I) \leq \delta W$ (or $b(I) > \delta W$) then $\sigma^{\text{expand}}_{\text{narrow}}(I) = \sigma_{\text{narrow}}(I)$ (or $\sigma^{\text{expand}}_{\text{wide}}(I) = \sigma_{\text{wide}}(I)$); otherwise, $\sigma^{\text{expand}}_{\text{narrow}}(I) = \emptyset$ (or $\sigma^{\text{expand}}_{\text{wide}}(I) = \emptyset$).

We note that our technique generalizes a well-known approximation technique, in which the input is partitioned into two subsets; the problem is then solved separately for each set, and the output is the solution of larger profit (see, e.g., Bar-Noy et al. 2001; Chen et al. 2002; Bar-Yehuda et al. 2009; Călinescu et al. 2011; Mömke and Wiese 2015). In contrast, our technique forms two new interval sets $\mathcal{I}_{\text{narrow}}$ and $\mathcal{I}_{\text{wide}}$, i.e., we do not necessarily have that $\mathcal{I}_{\text{narrow}} \subseteq \mathcal{I}$, or $\mathcal{I}_{\text{wide}} \subseteq \mathcal{I}$. Indeed, it may be the case that our transformation changes the range of possible colors for some intervals in the original instance. The next lemma shows that the approximation ratio guaranteed by the common *partitioning technique* holds also for our technique.

Lemma 2 *Let $(\mathcal{I}, W)$ be an instance of* FBAP *(or* FSAP*). For any $\delta \in (0, 1]$, let σ_{narrow} and σ_{wide} be a ρ_{narrow}-approximate solution for the instance $(\mathcal{I}_{\text{narrow}}, W)$ and a ρ_{wide}-approximate solution for the instance $(\mathcal{I}_{\text{wide}}, W)$, respectively, for $\rho_{\text{narrow}}, \rho_{\text{wide}} \geq 1$. Then, the solution of larger profit can be expanded to a $(\rho_{\text{narrow}} + \rho_{\text{wide}})$-approximate solution for the instance $(\mathcal{I}, W)$.*

Proof Let σ^*, σ^*_{narrow}, and σ^*_{wide} be optimal solutions for the instances $(\mathcal{I}, W)$, $(\mathcal{I}_{\text{narrow}}, W)$, and $(\mathcal{I}_{\text{wide}}, W)$, respectively.

Given a feasible coloring function σ of $(\mathcal{I}, W)$ we derive feasible coloring function σ^{narrow} and σ^{wide} for $(\mathcal{I}_{\text{narrow}}, W)$ and $(\mathcal{I}_{\text{wide}}, W)$, respectively, as follows. For the instance $(\mathcal{I}_{\text{narrow}}, W)$, for any interval $I_{\text{narrow}} \in \mathcal{I}_{\text{narrow}}$ if the corresponding interval $I \in \mathcal{I}$ was assigned with $|\sigma(I)| \leq \delta W$ colors, then $\sigma^{\text{narrow}}(I_{\text{narrow}}) = \sigma(I)$. Otherwise, $\sigma^{\text{narrow}}(I_{\text{narrow}}) = \emptyset$. For the instance $(\mathcal{I}_{\text{wide}}, W)$, for any interval $I_{\text{wide}} \in \mathcal{I}_{\text{wide}}$ if the corresponding interval $I \in \mathcal{I}$ was assigned with $|\sigma(I)| > \delta W$ colors, then $\sigma^{\text{wide}}(I_{\text{wide}}) = \sigma(I)$. Otherwise, $\sigma^{\text{wide}}(I_{\text{wide}}) = \emptyset$. Thus we have that for any feasible coloring σ of $(\mathcal{I}, W)$, $\mathit{profit}^{\sigma}(\mathcal{I}) = \mathit{profit}^{\sigma^{\text{narrow}}}(\mathcal{I}_{\text{narrow}}) + \mathit{profit}^{\sigma^{\text{wide}}}(\mathcal{I}_{\text{wide}})$. Therefore, we conclude that $\mathit{profit}^{\sigma^*}(\mathcal{I}) \leq \mathit{profit}^{\sigma^*_{\text{narrow}}}(\mathcal{I}_{\text{narrow}}) + \mathit{profit}^{\sigma^*_{\text{wide}}}(\mathcal{I}_{\text{wide}})$. Assume w.l.o.g that $\mathit{profit}^{\sigma_{\text{narrow}}}(\mathcal{I}_{\text{narrow}}) \geq \mathit{profit}^{\sigma_{\text{wide}}}(\mathcal{I}_{\text{wide}})$. Then,

$$
\begin{aligned}
&\mathit{profit}^{\sigma^*}(\mathcal{I}) \\
&\quad\leq \mathit{profit}^{\sigma^*_{\text{narrow}}}(\mathcal{I}_{\text{narrow}}) + \mathit{profit}^{\sigma^*_{\text{wide}}}(\mathcal{I}_{\text{wide}}) \\
&\quad\leq \rho_{\text{narrow}} \cdot \mathit{profit}^{\sigma_{\text{narrow}}}(\mathcal{I}_{\text{narrow}}) \\
&\qquad+ \rho_{\text{wide}} \cdot \mathit{profit}^{\sigma_{\text{wide}}}(\mathcal{I}_{\text{wide}}) \\
&\quad\leq (\rho_{\text{narrow}} + \rho_{\text{wide}}) \cdot \mathit{profit}^{\sigma_{\text{narrow}}}(\mathcal{I}_{\text{narrow}}) \\
&\quad= (\rho_{\text{narrow}} + \rho_{\text{wide}}) \cdot \mathit{profit}^{\sigma^{\text{expand}}_{\text{narrow}}}(\mathcal{I}).
\end{aligned}
$$

□

The local ratio technique The technique, initially developed by Bar-Yehuda and Even (1985), with later extensions, e.g., in Bafna et al. (1999), Bar-Yehuda (2000), Bar-Noy et al.

(2001), is based on the Local Ratio Theorem. Let $\mathbf{w} \in \mathbb{R}^n$ be a profit vector, and let F be a set of feasibility constraints on vectors $\mathbf{x} \in \mathbb{R}^n$. A vector $\mathbf{x} \in \mathbb{R}^n$ is a *feasible solution* to a given problem instance $(F, \mathbf{w})$ if it satisfies all of the constraints in F. Its value is the inner product $\mathbf{w} \cdot \mathbf{x}$.

Theorem 1 (Bar-Noy et al. (2001)) *Let F be a set of constraints and let $\mathbf{w}$, $\mathbf{w_1}$, and $\mathbf{w_2}$ be profit vectors such that $\mathbf{w} = \mathbf{w_1} + \mathbf{w_2}$. Then, if $\mathbf{x}$ is an r-approximate solution with respect to $(F, \mathbf{w_1})$ and with respect to $(F, \mathbf{w_2})$, then it is an r-approximate solution with respect to $(F, \mathbf{w})$.*

In this paper, we apply the technique, taking $\mathbf{w}$ to be the vector of profit per unit for the elements, and the solution vector $\mathbf{x}$ specifies the amount of resource units allocated to the input elements. The amount of resource allocated to each element can either be in its possible range, or else equal to zero.

3 A 3-approximation algorithm for FBAP

In this section, we present a polynomial time 3-approximation algorithm for FBAP. Given an instance $(\mathcal{I}, W)$ of FBAP, the algorithm starts by forming two sets of intervals: $\mathcal{I}_{\text{wide}}$ and $\mathcal{I}_{\text{narrow}}$, using $\delta = 1/2$ as defined in Sect. 2. For the wide intervals, $\mathcal{I}_{\text{wide}}$, the algorithm reduces the problem to *maximum weight independent set* (MWIS) on interval graphs, which has an optimal polynomial time algorithm (Golumbic 1980). Since each interval requires at least more than half of the resource, no pair of intersecting intervals can be colored together; therefore, by reducing it to an interval with a width of its maximal resource requirement, we get an optimal solution. For the narrow intervals, $\mathcal{I}_{\text{narrow}}$, we present a 2-approximation algorithm. The algorithm returns expansion to $\mathcal{I}$ of the color assignment of larger profit. By Lemma 2, we obtain a 3-approximation for FBAP$(\mathcal{I}, W)$.

FBAP on narrow intervals In the following, we describe algorithm NarrowFBAP and then prove that it yields a 2-approximation for the narrow intervals. The input for the algorithm is a tuple $(\mathcal{I}, w)$, where $\mathcal{I}$ is a set of intervals and w is the profit per unit function of $\mathcal{I}$. Algorithm NarrowFBAP uses the local ratio technique; it is recursive and works as follows. The algorithm starts by removing all intervals of non-positive profit per unit value as they do not change the optimum value. If no intervals remain, then it returns $\emptyset$. Otherwise, it chooses an interval $\tilde{I}$ with the minimum right endpoint. It constructs a new profit per unit function w_1, which assign profit only to intervals which intersect $\tilde{I}$ and solves the problem recursively on $w_2 = w - w_1$. Then, if the solution that was computed recursively has at least $a(\tilde{I})$ colors available for $\tilde{I}$, it adds $\tilde{I}$ to the solution with the maximal amount of colors such that the feasibility is maintained.

For a profit per unit function w, the total profit of a feasible coloring function σ of a subset $\mathcal{I}' \subseteq \mathcal{I}$ is denoted by $profit^{\sigma}(\mathcal{I}', w)$. Given a solution S, the load of an edge e in S is defined as $load(S, e) \stackrel{def}{=} \sum_{I \in \mathcal{I}_e \cap \mathcal{I}_S} |\sigma_S(I)|$.

Algorithm 1 NarrowFBAP$(\mathcal{I}, w)$

1: $\mathcal{I} \leftarrow \mathcal{I} \setminus \{I \in \mathcal{I} : w(I) \leq 0\}$
2: **If** $\mathcal{I} = \emptyset$ **then** return $\emptyset$
3: Select an interval $\tilde{I} \in \mathcal{I}$ with a minimum right endpoint
4: Define for each $I \in \mathcal{I}$

$$w_1(I) = w(\tilde{I}) \cdot \begin{cases} 1 & I = \tilde{I}, \\ \frac{b(\tilde{I})}{W - a(\tilde{I})} & I \neq \tilde{I}, I \cap \tilde{I} \neq \emptyset, \\ 0 & \text{otherwise.} \end{cases}$$

and $w_2 = w - w_1$
5: $S' \leftarrow$ NarrowFBAP$(\mathcal{I}, w_2)$
6: $\tilde{e} \leftarrow \text{argmax}_{e \in E(\tilde{I})}\, load(S', e)$
7: **If** $a(\tilde{I}) \leq W - load(S', \tilde{e})$ **then** $S \leftarrow S' \cup \{(\tilde{I}, \min\{b(\tilde{I}), W - load(S', \tilde{e})\})\}$
8: **Else** $S \leftarrow S'$
9: **Return** S

We note that algorithm NarrowFBAP returns the number of colors assigned to each interval. The algorithm can be easily modified to return the coloring of the intervals, by changing Line 7 to add to the solution the list of assigned colors, rather than their number.

Theorem 2 *Algorithm* NarrowFBAP$(\mathcal{I}, w)$ *computes in polynomial time a 2-approximate solution for any* FBAP *instance in which $b(I) \leq W/2$ for all $I \in \mathcal{I}$.*

Proof Clearly, the first step, in which intervals of non-positive profit are deleted, does not change the optimum value. Thus, it is sufficient to show that S is a 2-approximation with respect to the remaining intervals. The proof is by induction on the number of recursive calls. At the basis of the recursion, the solution returned is optimal and is a 2-approximation, since $\mathcal{I} = \emptyset$. For the inductive step, we show that the returned solution S is a 2-approximation with respect to w_1 and w_2, and thus, by the Lemma 2, it is 2-approximation with respect to w. Assuming that S' is a 2-approximation with respect to w_2, we have to show that S is a 2-approximation with respect to w_2. Since $w_2(\tilde{I}) = 0$ and $S' \subseteq S$, it follows that S is a 2-approximation with respect to w_2.

We now show that S is a 2-approximation with respect to w_1. In order to prove this, we need the following claims.

Claim 1 *For the solution S, $profit^{\sigma_S}(\mathcal{I}_S, w_1) \geq w_1(\tilde{I}) \cdot b(\tilde{I})$.*

Proof The claim holds since, either $\tilde{I} \in \mathcal{I}_S$ and $a(\tilde{I}) \leq \left|\sigma_S(\tilde{I})\right| \leq b(\tilde{I})$, or $S' \cup \{(\tilde{I}, a(\tilde{I}))\}$ is infeasible. For the case that $\tilde{I} \in \mathcal{I}_S$, if $\left|\sigma_S(\tilde{I})\right| = b(\tilde{I})$, then $profit^{\sigma_S}(\mathcal{I}_S, w_1) \geq w_1(\tilde{I}) \cdot b(\tilde{I})$; else, $\tilde{I} \in \mathcal{I}_S$ and thus the profit accrued by the intervals intersecting $\tilde{I}$ is $w_1(\tilde{I}) \cdot \frac{b(\tilde{I})}{W-a(\tilde{I})} \cdot (W - \left|\sigma_S(\tilde{I})\right|)$. In addition, $\left|\sigma_S(\tilde{I})\right| < b(\tilde{I})$, and since $b(\tilde{I}) \leq W/2$, we have that $a(\tilde{I}) + b(\tilde{I}) \leq W$, and thus

$$
\begin{aligned}
&\mathit{profit}^{\sigma_S}(\mathcal{I}_S, w_1)\\
&= w_1(\tilde{I}) \cdot \left|\sigma_S(\tilde{I})\right| + w_1(\tilde{I}) \cdot \frac{b(\tilde{I})}{W - a(\tilde{I})} \cdot \left(W - \left|\sigma_S(\tilde{I})\right|\right)\\
&= w_1(\tilde{I}) \cdot \left|\sigma_S(\tilde{I})\right| \cdot \left(1 - \frac{b(\tilde{I})}{W - a(\tilde{I})}\right)\\
&\quad + w_1(\tilde{I}) \cdot b(\tilde{I}) \cdot \frac{W}{W - a(\tilde{I})}\\
&\geq w_1(\tilde{I}) \cdot b(\tilde{I}).
\end{aligned}
$$

Consider now the case where $\tilde{I} \notin \mathcal{I}_S$. Since $S' \cup \{(\tilde{I}, a(\tilde{I}))\}$ is infeasible, it follows that $profit^{\sigma_S}(\mathcal{I}_S, w_1) \geq w_1(\tilde{I}) \cdot \frac{b(\tilde{I})}{W-a(\tilde{I})} \cdot (W - a(\tilde{I}) + 1) \geq w_1(\tilde{I}) \cdot b(\tilde{I})$. Therefore we conclude that $profit^{\sigma_S}(\mathcal{I}_S, w_1) \geq w_1(\tilde{I}) \cdot b(\tilde{I})$. □

Claim 2 *For any optimal solution* S^*, $profit^{\sigma_{S^*}}(\mathcal{I}_{S^*}, w_1) \leq 2 \cdot w_1(\tilde{I}) \cdot b(\tilde{I})$.

Proof The claim holds since if $\left|\sigma_{S^*}(\tilde{I})\right| \geq a(\tilde{I})$, then

$$
\begin{aligned}
&\mathit{profit}^{\sigma_{S^*}}(\mathcal{I}, w_1)\\
&\leq w_1(\tilde{I}) \cdot \left|\sigma_{S^*}(\tilde{I})\right| + w_1(\tilde{I}) \cdot \frac{b(\tilde{I})}{W - a(\tilde{I})} \cdot \left(W - \left|\sigma_{S^*}(\tilde{I})\right|\right)\\
&\leq w_1(\tilde{I}) \cdot b(\tilde{I}) + w_1(\tilde{I}) \cdot b(\tilde{I}) \cdot \left(\frac{W - \left|\sigma_{S^*}(\tilde{I})\right|}{W - a(\tilde{I})}\right)\\
&\leq 2 \cdot w_1(\tilde{I}) \cdot b(\tilde{I}).
\end{aligned}
$$

Else, $\left|\sigma_{S^*}(\tilde{I})\right| = 0$, and thus we have that $profit^{\sigma_{S^*}}(\mathcal{I}, w_1) \leq w_1(\tilde{I}) \cdot \frac{b(\tilde{I})}{W-a(\tilde{I})} \cdot W$, and since $a(\tilde{I}) \leq W/2$ we get that $profit^{\sigma_{S^*}}(\mathcal{I}_{S^*}, w_1) \leq 2 \cdot w_1(\tilde{I}) \cdot b(\tilde{I})$. □

Combining Claims 1 and 2, we have that S is a 2-approximate solution with respect to w_1. By Theorem 1, S is also a 2-approximate solution with respect to w. The running time is polynomial, since the number of recursive call is at most the number of input intervals, and each call requires linear time. □

Combining the exact algorithm for MWIS in interval graphs of Golumbic (1980), Theorem 2, and Lemma 2, we have

Theorem 3 *There exists a polynomial time* 3*-approximation algorithm for* FBAP.

4 A $(3 + \epsilon)$-approximation algorithm for FSAP

We now show that our result for FBAP can be extended to yield almost the same bound for FSAP. Given an instance $(\mathcal{I}, W)$ of FSAP, we form two sets of intervals: $\mathcal{I}_{\text{narrow}}$ and $\mathcal{I}_{\text{wide}}$ (as defined in Sect. 2), using a parameter $\delta > 0$ (to be determined). For the wide intervals, $\mathcal{I}_{\text{wide}}$, we present a $(1+\epsilon)$-approximation algorithm for any fixed $\epsilon > 0$. For the narrow intervals, $\mathcal{I}_{\text{narrow}}$, we give a $(2 + \epsilon)$-approximation algorithm for any fixed $\epsilon > 0$. The algorithm selects the color assignment of larger profit among the assignments found for $\mathcal{I}_{\text{narrow}}$ and $\mathcal{I}_{\text{wide}}$. This assignment is then expanded into a solution for the original instance $\mathcal{I}$. By Lemma 2, we obtain a $(3+\epsilon)$-approximate solution for FSAP, for any fixed $\epsilon > 0$. We note that any future improvements in the approximation ratio for FBAP on narrow intervals would improve also the approximation ratio of our algorithm for FSAP.

FSAP on wide intervals We describe below a $(1 + \epsilon)$-approximation algorithm for a wide instance of FSAP. The algorithm is based on rounding data combined with dynamic programming. We note that dynamic programming algorithms are widely used for this class of allocation problems (see, e.g., Chen et al. 2002; Bar-Yehuda et al. 2009). Given an instance $(\mathcal{I}, W)$ of FSAP, we say that σ is a feasible rounded color assignment if it is a feasible coloring and for every $I \in \mathcal{I}$, $|\sigma(I)| = k_\sigma \cdot \lfloor \delta^2 \cdot W \rfloor$, such that $k_\sigma \in \{0, \ldots, \lfloor \frac{1}{\delta^2} \rfloor\}$. A rounded solution for FSAP is a solution having a rounded color assignment. We present an optimal polynomial time dynamic programming algorithm for this rounded version of FSAP on the $\mathcal{I}_{\text{wide}}$ intervals and prove that it yields a $(1+\epsilon)$-approximation algorithm for the original $\mathcal{I}_{\text{wide}}$ instance of FSAP.

Lemma 3 *Given an instance* $(\mathcal{I}, W)$ *of* FSAP *and* $\delta \in (0, 1]$, *where* $a(I) \geq \lfloor \delta W \rfloor + 1$ *for all* $I \in \mathcal{I}$, *there is a polynomial time algorithm that computes an optimal rounded color assignment for* $(\mathcal{I}, W)$.

Proof Assume σ is a feasible contiguous color assignment for $(\mathcal{I}, W)$. Then, at any given point on the path, there are at most $1/\delta$ intervals that were assigned at least one color. Therefore, the constant $L = \lfloor 1/\delta \rfloor$ bounds the number of such intervals. Let $v \in \{0, \cdots, n\}$. We denote by IN^v the instance obtained from $(\mathcal{I}, W)$ by restricting the interval set to contain intervals that start at the point v or earlier. Formally, $IN^v = (\mathcal{I}^v, W)$, where $\mathcal{I}^v = \{I \in \mathcal{I} : l(I) \leq v\}$.

We say that two color assignments σ, σ' of the subsets Q, Q' *agree* if $\sigma(I) = \sigma'(I)$ whenever $I \in Q \cap Q'$, and we use the notation $\sigma \sim \sigma'$. For a rounded color assignment σ, let σ_{e_v} be the color assignment of the intervals $\mathcal{I}_{e_v}$, where e_v denotes the edge $(v, v + 1)$, and $\mathcal{I}_{e_v}$ denotes the intervals containing e_v. We denote by $OPT(IN^v, \sigma_{e_v})$ the profit of an optimum FSAP rounded solution for $textitIN^v$ that agrees with σ_{e_v}. Consider a contiguous rounded color assignment

σ^v of $\mathcal{I}^v$, and a contiguous rounded color assignment σ^{v-1} of $\mathcal{I}^{v-1}$ that agrees with σ^v. We have

$$profit^{\sigma^v}(\mathcal{I}^v) = profit^{\sigma^{v-1}}\left(\mathcal{I}^{v-1}\right) + \sum_{I \text{ s.t } l(I)=v} \left|\sigma^v(I)\right| \cdot w(I).$$

Therefore, given a rounded color assignment σ_{e_v} of $\mathcal{I}_{e_v}$, the value $OPT(IN^v, \sigma_{e_v})$ is obtained as follows.

$$OPT(IN^v, \sigma_{e_v}) = \max_{\sigma_{e_v} \sim \sigma_{e_{v-1}}} \left\{OPT\left(IN^{v-1}, \sigma_{e_{v-1}}\right)\right\} + \sum_{I \text{ s.t } l(I)=v} \left|\sigma_{e_v}(I)\right| \cdot w(I). \quad (1)$$

Thus, we have that

$$\max\left\{OPT(IN^{n-1}, \sigma) \;:\; \sigma \text{ is a contiguous rounded coloring of } \mathcal{I}_{e_{n-1}}\right\}$$

is an optimal rounded solution for FSAP.

We note that any contiguous rounded color assignment σ_{e_v} of the intervals $\mathcal{I}_{e_v}$ assigns colors to at most L intervals, and for each $I \in \mathcal{I}$, $\sigma_{e_v}(I) = \{i \cdot \lfloor \delta^2 W \rfloor, \ldots, j \cdot \lfloor \delta^2 W \rfloor : L \le i \le j \le L^2\}$. Therefore, the number of possible nonzero color assignments is bounded by $\sum_{l=0}^{L} \binom{L^2}{l} = O(L^{2L+2})$. In addition, there are at most n^L possibilities for choosing the intervals for the assignment, and the number of their permutations is bounded by $L! < L^L$. Thus, we have that for each e_v such that $v \in \{0, \cdots, n\}$ there are $O(n^L L^{3L+2})$ rounded color assignments of the intervals $\mathcal{I}_{e_v}$.

The dynamic programming table is of size $O(n^{L+1} L^{3L+2})$, and it is defined as follows. For each $v \in \{0, \cdots, n-1\}$, and for each feasible rounded color assignment σ of $\mathcal{I}_{e_v}$, we have an entry containing the value $OPT(IN^v, \sigma)$. We initialize the table by setting for each feasible rounded color assignment σ of $\mathcal{I}_{e_0}$ $OPT(IN^0, \sigma) = \sum_{I \text{ s.t } l(I)=0} |\sigma(I)| \cdot w(I)$. We use the recursive formulation in (1) to compute all other entries. To compute each entry $OPT(IN^i, \sigma_{e_i})$, we need to go through all the optimal rounded solutions of all possible rounded color assignment of $\mathcal{I}_{e_{i-1}}$. There are at most $O(n^L L^{3L+2})$ possibilities. Therefore, the total running time is $O(n^{2L+1} L^{6L+4})$. □

Applying the algorithm of Lemma 3 we have

Lemma 4 *Given an instance* $(\mathcal{I}, W)$ *of* FSAP, *for any fixed* $\epsilon > 0$, *there exists* $\delta \in (0, 1]$, *such that there is a polynomial time* $(1+\epsilon)$*-approximation algorithm for* $(\mathcal{I}_{\text{wide}}, W)$.

Proof Let S be the returned solution after applying the algorithm of Lemma 3. For each $I \in \mathcal{I}_S$, there exists an integer $k_S(I) \in \{\lfloor \frac{1}{\delta} \rfloor, \ldots, \lfloor \frac{1}{\delta^2} \rfloor\}$, such that $|\sigma_S(I)| = k_S(I) \cdot \lfloor \delta^2 W \rfloor$. Let S^* be an optimal solution for $\mathcal{I}_{\text{wide}}$ and $OPT = profit^{\sigma_{S^*}}(\mathcal{I}_{\text{wide}})$. For each $I \in \mathcal{I}_{S^*}$, there exists an integer $k_{S^*}(I) \in \{\lfloor \frac{1}{\delta} \rfloor, \ldots, \lfloor \frac{1}{\delta^2} \rfloor\}$, such that $k_{S^*}(I) \cdot \lfloor \delta^2 W \rfloor < |\sigma_{S^*}(I)| \le (k_{S^*}(I)+1) \cdot \lfloor \delta^2 W \rfloor$, and thus

$$|\sigma_{S^*}(I)| - \left\lfloor \delta^2 W \right\rfloor \le k_{S^*}(I) \cdot \left\lfloor \delta^2 W \right\rfloor. \quad (2)$$

Furthermore, we can derive a feasible rounded solution from S^*, $S^{*\text{round}}$, such that $\mathcal{I}_{S^{*\text{round}}} = \mathcal{I}_{S^*}$ and for each $I \in \mathcal{I}_{S^*}$ where $\sigma_{S^*}(I) = \{i, \ldots, j\}$, $\sigma_{S^{*\text{round}}}(I) = \{\lceil \frac{i}{\delta^2 W} \rceil \lfloor \delta^2 W \rfloor, \ldots, \lfloor \frac{j}{\delta^2 W} \rfloor \lfloor \delta^2 W \rfloor\}$. Thus,

$$\left|\sigma_{S^{*\text{round}}}(I)\right| \ge (k_{S^*}(I) - 2) \cdot \left\lfloor \delta^2 W \right\rfloor. \quad (3)$$

We note that, for each $I \in \mathcal{I}_{\text{wide}}$, $\delta W < a(I)$, and thus

$$\sum_{I \in \mathcal{I}_{S^*}} \delta W w(I) < profit^{\sigma_{S^*}}(\mathcal{I}_{S^*}). \quad (4)$$

Hence,

$$\begin{aligned}
profit^{\sigma_S}(\mathcal{I}_{\text{wide}}) &= \sum_{I \in \mathcal{I}_S} k_S(I) \cdot \left\lfloor \delta^2 W \right\rfloor \cdot w(I) \\
&\ge \sum_{I \in \mathcal{I}_{S^{*\text{round}}}} \left|\sigma_{S^{*\text{round}}}(I)\right| \cdot w(I) \\
\text{(by Ineq. (3))} &\ge \sum_{I \in \mathcal{I}_{S^{*\text{round}}}} (k_{S^*}(I) - 2) \cdot \left\lfloor \delta^2 W \right\rfloor \cdot w(I) \\
&= \sum_{I \in \mathcal{I}_{S^*}} (k_{S^*}(I) - 2) \cdot \left\lfloor \delta^2 W \right\rfloor \cdot w(I) \\
\text{(by Ineq. (2))} &\ge \sum_{I \in \mathcal{I}_{S^*}} \left(|\sigma_{S^*}(I)| - 3\left\lfloor \delta^2 W \right\rfloor\right) \cdot w(I) \\
\text{(by Ineq. (4))} &> profit^{\sigma_{S^*}}(\mathcal{I}_{S^*}) - 3\delta profit^{\sigma_{S^*}}(\mathcal{I}_{S^*}) \\
&= OPT - 3\delta OPT.
\end{aligned}$$

By choosing $\delta \le \frac{\epsilon}{3(1+\epsilon)}$, we have a polynomial time $(1+\epsilon)$-approximation algorithm for $\mathcal{I}_{\text{wide}}$. □

FSAP on narrow intervals We now show how to obtain a $(2+\epsilon)$-approximate solution for the $\mathcal{I}_{\text{narrow}}$ intervals. Recall that, in BAP, we are given a path having one unit of available resource, and a set $\mathcal{I}$ of intervals on the path. Each interval $I \in \mathcal{I}$ consists of an arrival time and departure time, a resource requirement $s(I) \in [0, 1]$, and a profit $p(I) \in \mathbb{Z}$. The goal is to assign the resource to a maximum weight subset of requests. A solution S for BAP consists of the assigned intervals. The profit S is given by $profit(S) \overset{def}{=} \sum_{I \in S} p(I)$. SAP is a special case of BAP in which we require that each interval is allocated a single contiguous block of resource for its entire duration.

We use as subroutines algorithm NarrowFBAP (of Sect. 3) and a subroutine of an algorithm for SAP due to Bar-Yehuda et al. (2009), which transforms in polynomial time a BAP solution into a SAP solution, as formulated in the following lemma.

Lemma 5 (Bar-Yehuda et al. 2009) *There exists constants* $\delta_0 \in (0, 1]$ *and* $C_0 > 0$*, such that* S *is a solution for* BAP *on intervals* $\mathcal{I}$ *for which* $s(I) \leq \delta$ *for all* $I \in \mathcal{I}$*, where* $\delta \in (0, \delta_0)$*, then* S *can be transformed in polynomial time into a solution for* SAP *with profit at least* $(1 - C_0\delta^{\frac{1}{7}})$ *profit* (S).

Combining Theorem 2 and Lemma 5, we have

Lemma 6 *Given an instance* $(\mathcal{I}, W)$ *of* FSAP*, for any fixed* $\epsilon > 0$*, there exists* $\delta > 0$*, such that there is a polynomial time* $(2 + \epsilon)$*-approximation algorithm for* $(\mathcal{I}_{\text{narrow}}, W)$.

Proof Given $\epsilon > 0$, choose δ, such that $\delta < \min\{\delta_0, \frac{\epsilon}{C_0(2+\epsilon)})^7\}$. Then, applying as a subroutine algorithm NarrowFBAP, we have a non-contiguous color assignment for the $\mathcal{I}_{\text{narrow}}$ intervals achieving a ratio of 2 to the optimum for FBAP. Taking this assignment as an input for the subroutine of Lemma 5, we have a contiguous color assignment with ratio $\frac{1-C_0\delta^{\frac{1}{7}}}{2} > \frac{1}{2+\epsilon}$. Overall, we have a polynomial running time, since we use as subroutines two polynomial time algorithms. □

Combining Lemmas 4, 6, and 2, we obtain

Theorem 4 *There exists a polynomial time* $(3 + \epsilon)$ *-approximation algorithm for* FSAP*, for any fixed* $\epsilon > 0$.

5 A $(2 + \epsilon)$-approximation algorithm for SAP

In this section, we consider SAP, the special case of FSAP where $a(I) = b(I)$ for all $I \in \mathcal{I}$. We present a polynomial time $(2 + \epsilon)$-approximation algorithm for any fixed $\epsilon > 0$.

In deriving the algorithm, we use a technique similar to the one used for solving general instances of FSAP. Given an instance $\mathcal{I}$ of SAP, we partition the intervals into two sets: *narrow* and *wide*, using a parameter $\delta > 0$ (to be determined). Specifically, *narrow* intervals are those for which $|s(I)| \leq \delta W$ and *wide* intervals are those for which $|s(I)| > \delta W$. For the *wide* intervals, we use the $(1 + \epsilon)$-approximation algorithm of Lemma 4 (for FSAP on $\mathcal{I}_{\text{wide}}$ intervals). For the *narrow* intervals, we show how to obtain a $(1 + \epsilon)$-approximate solution. The algorithm returns the color assignment of greater profit. By Lemma 2, this yield a $(2 + \epsilon)$-approximation algorithm for SAP.

SAP on narrow intervals In the following, we show how to obtain a $(1 + \epsilon)$-approximate solution for the *narrow* intervals. We use as subroutines two known algorithms: for BAP and SAP. For BAP, the paper (Chekuri et al. 2007) presents a $(2+\epsilon)$-approximation algorithm, for any $\epsilon > 0$. The authors obtain the result by dividing the input intervals into *wide* and *narrow* intervals, for some $\delta \in (0, 1)$. They use dynamic programming to compute an optimal solution for the *wide* intervals, and LP-based algorithm to obtain a $(1+O(1)\sqrt{\delta})$-approximate solution for the *narrow* intervals, as stated in the next result.

Lemma 7 (Chekuri et al. 2007) *There exist constants* $\delta_1 \in (0, 1)$ *and* $C_1 > 0$*, such that for any* $\delta \in (0, \delta_1)$ *there exists a* $(1+C_1\sqrt{\delta})$*-approximation algorithm for the* narrow *intervals of* BAP.

The second subroutine that we use is an algorithm for SAP of Bar-Yehuda et al. (2009), which transforms in polynomial time a BAP solution into a SAP solution (Lemma 5).

Combining Lemmas 7 and 5, we have

Lemma 8 *For any fixed* $\epsilon > 0$*, there exists* $\delta > 0$*, such that there is a polynomial time* $(1 + \epsilon)$*-approximation algorithm for the narrow intervals of* SAP.

Proof Given $\epsilon > 0$, choose δ, such that such that $\delta < \min\{\delta_0, \delta_1, (\frac{\epsilon}{C_1+C_0(1+\epsilon)})^7\}$, where the constants $\delta_0 \in (0, 1]$ and $C_0 > 0$ are of Lemma 5 and $\delta_1 \in (0, 1)$ and $C_1 > 0$ are of Lemma 7. Then, calling as a subroutine the algorithm of Lemma 7 yields a resource assignment for the *narrow* intervals with ratio $(1 + C_1\sqrt{\delta})$. Taking this assignment as an input for the subroutine of Lemma 5, we have a contiguous resource assignment with ratio $\frac{1-C_0\delta^{\frac{1}{7}}}{1+C_1\sqrt{\delta}} > \frac{1}{1+\epsilon}$. Overall, we have a polynomial running time, since we use as subroutines two polynomial time algorithms. □

Summarizing the above discussion, we have

Theorem 5 *For any fixed* $\epsilon > 0$*, there is a polynomial time* $(2 + \epsilon)$*-approximation algorithm for* SAP.

Proof Given an instance for SAP, by Lemma 8, there exists a constant $\delta > 0$ for which there is a polynomial time $(1+\epsilon)$-approximation algorithm for the *narrow* intervals. A $(1 + \epsilon)$-approximate solution for the *wide* intervals can be found using the algorithm of Lemma 4. By Lemma 2, taking the better of the two solutions, we obtain a $(2+\epsilon)$-approximation algorithm for any fixed $\epsilon > 0$. □

6 Summary and future work

In this paper, we studied FBAP and FSAP. We observed that both problems are NP-hard even for highly restricted inputs, and presented a 3-approximation and a $(3+\epsilon)$-approximation algorithm for general inputs of FBAP and FSAP, respectively. We point to a few of the many problems that remain open.

- We showed that FSAP is NP-hard for the subclass of instances where $a(I) \neq b(I)$ for all $I \in \mathcal{I}$. For this case, it would be interesting to obtain a better approximation ratio than the one derived for general instances.
- Our results for intervals on a line call for the study of FBAP and FSAP in other graph, especially those that are relevant in optical networks.
- The flex-grid technology enables to combine non-contiguous and contiguous spectrum assignment; thus, a request can be assigned either a contiguous or non-contiguous set of colors. In this setting, a non-contiguous color assignment for any request requires accumulatively more spectrum than contiguous color assignment of the same request (due to the gap of unused frequencies between wavelengths). This affects the total amount of the spectrum used and may imply different profits per unit for each assignment. This practical setting opens up an unexplored terrain for future study.
- It would be interesting to extend FSAP and FBAP to the case of variable bandwidth available per interval. This case is not only challenging from theoretical point of view, but also of practical interest.
- Finally, as stated above, FSAP and FBAP are the flexible variants of classic SAP and BAP, respectively. There is much importance in exploring the implications of these new problems and our results in the context of resource allocation in emerging computing and network technologies.

Acknowledgements We thank Dror Rawitz for valuable discussions.

References

Ali Norouzi, A. Z., & Ustundag, B. B. (2011). An integrated survey in optical networks: Concepts, components and problems. *International Journal of Computer Science and Network Security*, *11*(1), 10–26.

Bafna, V., Berman, P., & Fujito, T. (1999). A 2-approximation algorithm for the undirected feedback vertex set problem. *SIAM Journal on Discrete Mathematics*, *12*(3), 289–297.

Bansal, N., Chakrabarti, A., Epstein, A., & Schieber, B. (2006). A quasi-PTAS for unsplittable flow on line graphs. In: *Proceedings of the 38th annual ACM symposium on theory of computing (STOC)* (pp. 721–729).

Bar-Noy, A., Bar-Yehuda, R., Freund, A., Naor, J., & Schieber, B. (2001). A unified approach to approximating resource allocation and scheduling. *Journal of the ACM*, *48*(5), 1069–1090.

Bar-Yehuda, R. (2000). One for the price of two: A unified approach for approximating covering problems. *Algorithmica*, *27*(2), 131–144.

Bar-Yehuda, R., Beder, M., Cohen, Y., & Rawitz, D. (2009). Resource allocation in bounded degree trees. *Algorithmica*, *54*(1), 89–106.

Bar-Yehuda, R., & Even, S. (1985). A local-ratio theorem for approximating the weighted vertex cover problem. In *Analysis and design of algorithms for combinatorial problems. North-Holland mathematics studies* (vol. 109, pp. 27–45). North-Holland.

Călinescu, G., Chakrabarti, A., Karloff, H. J., & Rabani, Y. (2011). An improved approximation algorithm for resource allocation. *ACM Transactions on Algorithms*, *7*(4), 48.

Chekuri, C., Mydlarz, M., & Shepherd, F. B. (2007). Multicommodity demand flow in a tree and packing integer programs. *ACM Transaction on Algorithms* **3**(3).

Chen, B., Hassin, R., & Tzur, M. (2002). Allocation of bandwidth and storage. *IIE Transactions*, *34*(5), 501–507.

Chrobak, M., Woeginger, G. J., Makino, K., & Xu, H. (2012). Caching is hard—Even in the fault model. *Algorithmica*, *63*(4), 781–794.

Garey, M., & Johnson, D. S. (1979). *Computers and Intractability. A guide to the theory of NP-completeness*. San Francisco: Freeman.

Gerstel, O. (2010). Flexible use of spectrum and photonic grooming. In *Integrated photonics research, silicon and nanophotonics and photonics in switching* (p. PMD3).

Gerstel, O. (2011). Realistic approaches to scaling the ip network using optics. In *Optical fiber communication conference and exposition and the national fiber optic engineers conference* (p. OWP1).

Golumbic, M. C. (1980). *Algorithmic graph theory and perfect graphs*. New York: Academic Press.

Jinno, M., Takara, H., Kozicki, B., Tsukishima, Y., Sone, Y., & Matsuoka, S. (2009). Spectrum-efficient and scalable elastic optical path network: Architecture, benefits, and enabling technologies. *IEEE Communications Magazine*, *47*(11), 66–73.

Katz, D., Schieber, B., & Shachnai, H. (2016). Brief announcement: Flexible resource allocation for clouds and all-optical networks. In *proceedings of the 28th ACM symposium on parallelism in algorithms and architectures (SPAA)* (pp. 225–226).

Leonardi, S., Marchetti-Spaccamela, A., & Vitaletti, A. (2000). Approximation algorithms for bandwidth and storage allocation problems under real time constraints. In *proceedings of the 20th conference on foundations of software technology and theoretical computer science* (pp. 409–420).

Mömke, T., & Wiese, A. (2015). A $(2+\epsilon)$-approximation algorithm for the storage allocation problem. In *proceedings of the 42nd international colloquium on automata, languages, and programming (ICALP)* (pp. 973–984).

Ramaswami, R., Sivarajan, K., & Sasaki, G. (2009). *Optical networks: A practical perspective* (3rd ed.). Los Altos: Morgan Kaufmann.

Shachnai, H., Voloshin, A., & Zaks, S. (2014) Optimizing bandwidth allocation in flex-grid optical networks with application to scheduling. In *Proceedings of the 28th IEEE international parallel and distributed processing symposium (IPDPS)* (pp. 862–871).

Shalom, M., Wong, P. W. H., & Zaks, S. (2013). Profit maximization in flex-grid all-optical networks. In *Proceedings of the 20th international colloquium on structural information and communication complexity (SIROCCO)* (pp. 249–260). Berlin: Springer.