UNIVERSITY
OF TRENTO

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

38050 Povo – Trento (Italy), Via Sommarive 14
http://www.dit.unitn.it

ALLOCATING SERVERS IN INFOSTATIONS FOR BOUNDED SIMULTANEOUS REQUESTS

Alan Bertossi, Cristina M. Pinotti, Romeo Rizzi
and Phalguni Gupta

September 2002

Technical Report # DIT-02-0074

Allocating Servers in Infostations for Bounded Simultaneous Requests *

A.A. Bertossi[†] M.C. Pinotti[‡] R. Rizzi[‡] P. Gupta[§]

Abstract

The Server Allocation with Bounded Simultaneous Requests problem arises in infostations, where mobile users going through the coverage area require immediate high-bit rate communications such as web surfing, file transferring, voice messaging, email and fax. Given a set of service requests, each characterized by a temporal interval and a category, an integer k, and an integer h_c for each category c, the problem consists in assigning a server to each request in such a way that at most k mutually simultaneous requests are assigned to the same server at the same time, out of which at most h_c are of category c, and the minimum number of servers is used. Since this problem is computationally intractable, a 2-approximation on-line algorithm is exhibited which asymptotically gives a $\left(2 - \frac{h}{k}\right)$-approximation, where $h = min\{h_c\}$. Generalizations of the problem are considered, where each request r is also characterized by a bandwidth rate w_r, and the sum of the bandwidth rates of the simultaneous requests assigned to the same server at the same time is bounded, and where each request is characterized also by a gender bandwidth. Such generalizations contain Bin-Packing and Multiprocessor Task Scheduling as special cases, and they admit on-line algorithms providing constant approximations.

Keywords: Infostations, Server allocation, Bin-Packing, Greedy algorithms, Approximation algorithms, On-line algorithms, Interval coloring.

1 Introduction

Infostation is a new concept for fourth generation of mobile communication which should support high-speed and high-quality services at *many-time many-where*. Such a concept has been introduced to overcome the difficulties due to typical high bit-rate connections such as web surfing, file transferring, voice messaging, email and fax. Indeed, it is not possible to provide such high bit-rate connections at *anytime anywhere* as it is today for cellular systems which support only low bit-rate communications. An infostation is an isolated pocket area with small coverage (about a hundred of meters) of high bandwidth connectivity (at least a megabit per second) that collects information requests of mobile users and delivers data while users are going through the coverage area. The available bandwidth depends on the distance between the mobile user and the center

*This work is partially supported by MIUR-RealWine Research Program.

†Department of Computer Science, University of Bologna, Mura Anteo Zamboni, 7, 40127 Bologna, ITALY, bertossi@cs.unibo.it

‡Department of Computer Science and Telecommunications, University of Trento, Via Sommarive, 14, 38050 Povo, Trento, ITALY, pinotti@science.unitn.it, rrizzi@science.unitn.it

§Department of Computer Science and Engineering, Indian Institute of Technology Kampur, Kampur-208016, INDIA, pg@cse.iitk.ac.in

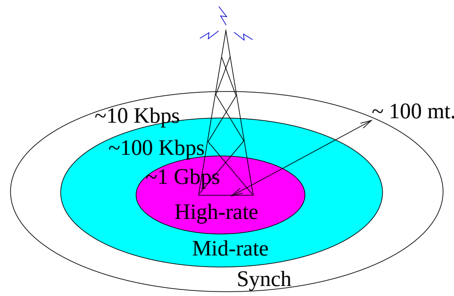

Figure 1: Ranges and bit-rates for Infostation coverage area.

of the coverage area, increasing with decreasing distance, as depicted in Figure 1 [4, 8, 9]. Infostations could be located along roadways, at airports, in campuses, and they can provide access ports to Internet and/or access to services managed locally.

Two infostation models have been proposed so far. In the *isolated* model, the infostations are irregularly scattered on the land and a user approaching one infostation does not know when another infostation will be met. In the *multiple* model, there are several infostations regularly distributed on the land and the mobile user encounters an infostation very frequently thus experiencing a much larger coverage area than in the isolated model. As shown in Figure 2, the infostation system may retrieve the requested data from remote gateways, may provide Internet access, and may home local services (such as building access, credit card transactions, and map downloads). The mobile user connection starts when it first senses the infostation's presence and finishes when it leaves the coverage area. Depending on the mobility options, three kinds of users are characterized: *drive-through*, *walk-through*, and *sit-through*. According to the mobility options, the bit-rate connection is high variable for drive-through, low variable for walk-through, and fixed for sit-through. In addition to radio broadcast communication, infostations create opportunities to deliver new wireless information services dedicated to single-users, which could be supported for example by infrared technologies. In summary, several communication paradigms are possible. Communications can be broadcast or dedicated to a single user, data can be locally provided or retrieved from a remote gateway, and the bit-rate transmission can be fixed or variable, depending on the infostation model and on the mobility kind of the user.

Each mobile user going through the infostation may require a data service out of a finite set of possible service categories available. The admission control, i.e., the task of deciding whether or not a certain request will be admitted, is essential. In fact, a user going through an (isolated) infostation to obtain a (toll) service is not disposed to have its request delayed or refused. Hence, the service dropping probability must be kept as low as possible. For this purpose, many admission control and bandwidth allocation schemes for infostations maintain a pool of servers so that when a request arrives it is immediately and irrevocably assigned to a server thus clearing the service dropping probability. Precisely, once a request is admitted, the infostation assigns a temporal interval and a proper bandwidth for serving the request, depending on the service category, on the size of the data required and on the mobility kind of the user, as shown in Table 1 for a sample of requests with their actual parameters. Moreover, the infostation decides whether the request may be served locally or through a remote gateway. In both cases, a server (either in the

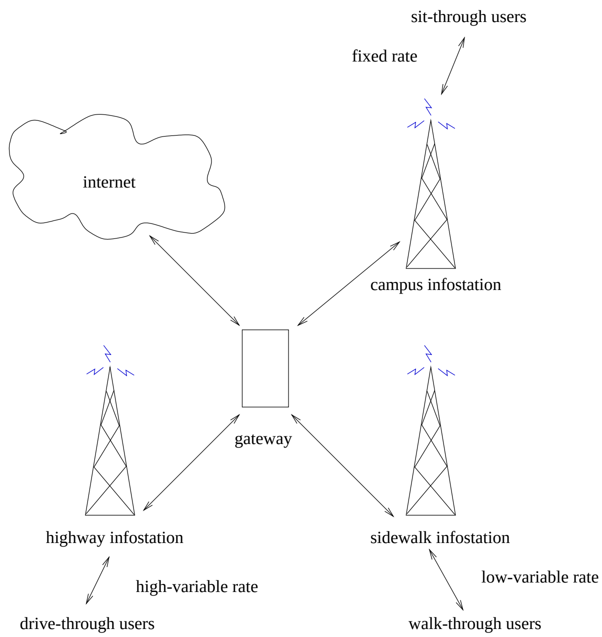

Figure 2: An Infostation system.

Category	**Size (Kbps)**	**Time (seconds)**	
		Low rate	**High rate**
FTP download	10000	100	10
Video streams	5000	50	5
Audio streams, E-mail attachments	512	5	.5
E-mails, Web Browsing	64	.6	.06

Table 1: Example of actual time intervals required to serve different kinds of requests.

infostation or in the gateway) is allocated on demand to the request during the assigned temporal interval. The request is immediately assigned to its server without knowing the future, namely with no knowledge of the next request.

The server, selected out of the predefined server pool, serves the requests on-line, that is in an ongoing manner as they become available. Moreover, each server may serve more than one request simultaneously but it is subject to some architecture constraints. For example, no more than k requests could be served simultaneously by a local server supporting k infrared channels or by a gateway server connected to k infostations. Similarly, no more than h services of the same category can be delivered simultaneously due to access constraints on the original data, such as software licenses, limited on line subscriptions and private access.

In this paper, a particular problem arising in the design of infostation systems is faced which consists in finding scheduling algorithms for allocating the minimum number of servers to the user requests in such a way that the temporal, architectural and data constraints are satisfied. In details, a service *request* r will be modeled by a service *category* c_r and a temporal *interval*

$I_r = [s_r, e_r)$ with *starting time* s_r and *ending time* e_r. Two requests are simultaneous if their temporal intervals overlap. The input of the problem consists of a set R of service requests, a bound k on the number of mutually simultaneous requests to be served by the same server at the same time, and a set C of service categories with each category c characterized by a bound h_c. The output is a mapping from the requests in R to the servers that uses the minimum possible number of servers to assign all the requests in R subject to the constraints that the same server receives at most k mutually simultaneous requests at the same time, out of which at most h_c are of category c. Such a problem is called in this paper *Server Allocation with Bounded Simultaneous Requests*.

It is worthy to note that, equating servers with bins, and requests with items, the above problem is similar to a generalization of Bin-Packing, known as *Dynamic Bin-Packing* [1], where in addition to size constraints on the bins, the items are characterized by an arrival and a departure time, and repacking of already packed items is allowed each time a new item arrives. The problem considered in this paper, in contrast, does not allow repacking and has capacity constraints also on the bin size for each category. Furthermore, equating servers with processors and requests with tasks, the above problem becomes a generalization of deterministic multiprocessor scheduling with task release times and deadlines [6], where in addition each processor can execute more than one task at the same time.

The rest of this paper is structured as follows. In Section 2, it is shown that Server Allocation with Bounded Simultaneous Requests is computationally intractable and therefore a solution using the minimum number of servers cannot be found in polynomial time. Section 3 deals with α-approximation algorithms, that is polynomial time algorithms that provide solutions which are guaranteed to never be greater than α times optimal solutions. It is shown that Server Allocation with Bounded Simultaneous Requests cannot be α-approximated with $\alpha < \frac{4}{3}$. Moreover, a 2-approximation on-line algorithm is exhibited which asymptotically gives a $\left(2 - \frac{h}{k}\right)$-approximation, where $h = \min_{c \in C} h_c$. In Section 4, a generalization of the problem is considered where each request r is also characterized by an integer *bandwidth rate* w_r, and the bounds on the number of simultaneous requests to be served by the same server are replaced by bounds on the sum of the bandwidth rates of the simultaneous requests assigned to the same server. For this problem, on-line and off-line algorithms are proposed which give a constant approximation. Other two generalizations are proposed in Section 5 in which each request is characterized either by a multi-dimensional bandwidth rate or by both a bandwidth rate and a gender bandwidth associated to the category of the request. Again, these problems admit on-line algorithms providing a constant approximation.

2 Computational Intractability

The Server Allocation with Bounded Simultaneous Requests problem on a set $R = \{r_1, \dots, r_n\}$ of requests can be formulated as a coloring problem on the corresponding set $I = \{I_1, \dots, I_n\}$ of temporal intervals. Indeed, equating servers with colors, the original server allocation problem is equivalent to the following coloring problem:

Problem 1 (Interval Coloring with Bounded Overlapping). *Given a set I of intervals each belonging to a category, an integer k, and an integer h_c for each category c, assign a color to each interval in such a way that at most k mutually overlapping intervals receive the same color, at most*

h_c mutually overlapping intervals all having category c receive the same color, and the minimum number of colors is used.

In order to prove that Problem 1 is computationally intractable, the following simplified decisional formulation is considered, where there is a bound $h_c = 1$ for each category c.

Problem 2 (Interval Coloring with Unit Bounded Categories). *Given a set I of intervals each belonging to a category, and two integers k and b, decide whether b colors are enough to assign a color to each interval in such a way that at most k mutually overlapping intervals receive the same color and no two overlapping intervals with the same category receive the same color.*

In the following it is proved that Problem 2 is NP-complete by exhibiting a reduction from a variant of a graph coloring problem. For this purpose some preliminary results are needed. The graph coloring problem below is well-known to be NP-complete [2], even if $b \geq 3$.

Problem 3 (Chromatic Number). *Given an integer b and an undirected graph $G = (V, E)$, decide whether the nodes in V can be colored with b colors in such a way that adjacent nodes receive different colors.*

Recall that a *stable set* of a graph $G = (V, E)$ is a subset S of nodes in V such that no two nodes in S are adjacent. Consider the variant below of Problem 3, where one wants to employ every color exactly the same number of times.

Problem 4 (Balanced Coloring). *Given two integers b and k, and an undirected graph $G = (V, E)$ of kb nodes, decide whether V can be partitioned into b stable sets each of size k.*

Lemma 2.1. *Balanced Coloring is NP-complete, even if $b \geq 3$.*

Proof: Chromatic Number is reduced in polynomial time to Balanced Coloring as follows. Given an instance of Chromatic Number, namely b and $G = (V, E)$, let $k = |V|$ and $G' = (V', E')$ be the graph obtained by considering b disjoint copies of G. Clearly, G can be colored with b colors if and only if V' can be partitioned into b stable sets each of size k. Hence, solving the instance of Balanced Coloring corresponds to solving the instance of Chromatic Number. □

In order to show that Interval Coloring with Unit Bounded Categories is NP-complete, a polynomial time reduction from Balanced Coloring is given.

Theorem 2.2. *Interval Coloring with Unit Bounded Categories is NP-complete, even if $b \geq 3$.*

Proof: Problem 4 is reduced in polynomial time to Problem 2 as follows. Given an instance of Balanced Coloring, that is b, k, and $G = (V, E)$, an instance of Problem 2 is constructed in such a way that there are as many categories as there are nodes in V, a subset of intervals correponds to each node in V so that all such intervals are forced to receive the same color, and some pairs of intersecting intervals belong to the same category if and only if their corresponding nodes in V are adjacent.

Specifically, let $kb = |V|$ and $m = |E|$. Take the set of categories as $C = \{1, 2, \ldots, kb\}$. Construct $2kb + 2m$ intervals as follows. For each generic node $v \in V$, let $v_1, \ldots, v_\ell$ be the neighbours of v (indexed, for the sake of simplicity, so that $v_i < v_{i+1}$). The following $\ell + 2$ intervals correspond to node v:

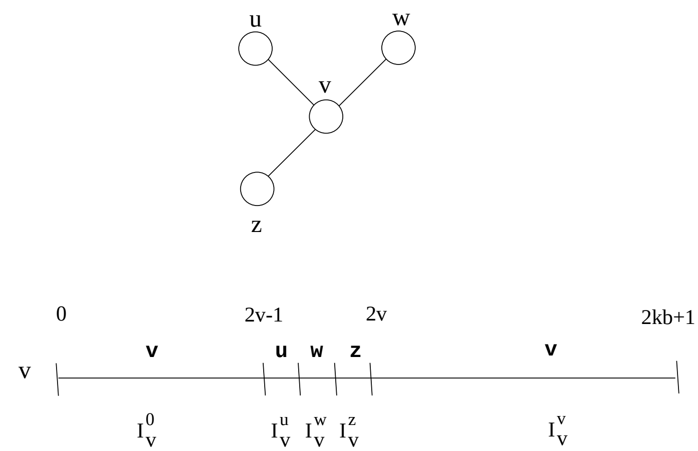

Figure 3: The 5 intervals corresponding to a node v with 3 neighbours u, w, z (categories are represented by literals above the intervals).

- $I_v^0 = [0, 2v - 1)$ with category $c_v^0 = v$,
- $I_v^v = [2v, 2kb + 1)$ with category $c_v^v = v$,
- $I_v^{v_i} = \left[2v - 1 + \frac{i-1}{\ell}, 2v - 1 + \frac{i}{\ell}\right)$ with category $c_v^{v_i} = v_i$, for $i = 1, \dots, \ell$.

If a node v is isolated, that is it has no neighbour, then a single interval $I_v = [0, 2kb + 1)$ with category $c_v = v$ corresponds to it.

Note that $[v, u] \in E$, with $v < u$, if and only if both the following conditions hold:

1. $I_v^u \cap I_u^0 \neq \emptyset$ and $c_v^u = c_u^0 = u$, and
2. $I_u^v \cap I_v^v \neq \emptyset$ and $c_u^v = c_v^v = v$.

Thus, adjacency between two nodes u and v in G is coded by two pairs of overlapping intervals in I with categories u and v, which cannot be colored the same.

As an example, Figure 3 shows the 5 intervals corresponding to a node v with 3 neighbours u, w, z. Instead Figure 4 depicts all the $2kb+2m$ intervals of I corresponding to a graph $G = (V, E)$ with $kb = 6$ nodes and $m = 10$ edges (in such a figure, $k = 2$ and $b = 3$).

The graph G can be colored with b colors if and only if the intervals in I can be colored with b colors, as proved below.

Assume that G can be colored with b colors. Assign to all the intervals $I_v^0, I_v^v, I_v^{v_1}, \dots, I_v^{v_\ell}$ the same color that node v has in G. Clearly, if nodes v and u in G are colored the same, then they are not adjacent. Two cases arise: either v and u have no common neighbour, or they have at least one common neighbour w. In the former case, all the intervals corresponding to v and u belong to different categories, whereas in the latter case the intervals I_v^w and I_u^w, although belonging to the same category w, do not intersect. In both cases all the intervals corresponding to v and u can receive the same color. Moreover, since the same color appears in G exactly k times, exactly k mutually overlapping intervals of I receive the same color.

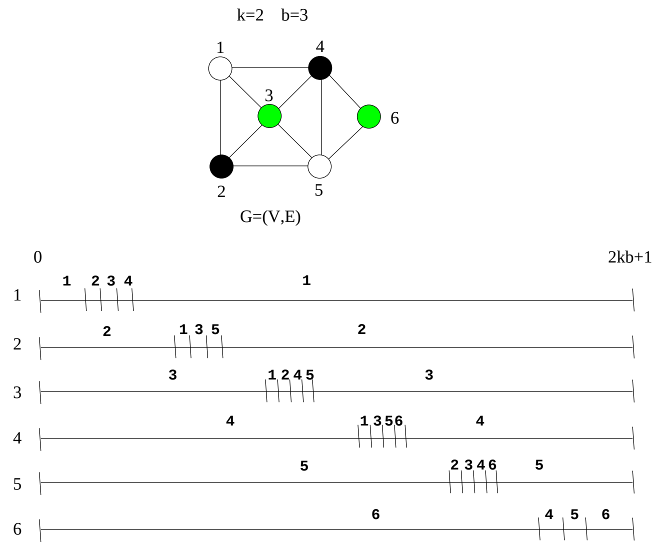

Figure 4: Example of reduction from Balanced Coloring to Interval Coloring with Unit Bounded Categories (categories are represented by numbers above the intervals).

Conversely, assume that all the intervals in I can be colored with b colors. Observe that for each v, all the intervals $I_v^0, I_v^v, I_v^{v_1}, \ldots, I_v^{v_\ell}$ must be colored the same (otherwise, by construction, at least $b+1$ colors would be required). Therefore, assign such a color also to the node v of G. If $I_v^0, I_v^v, I_v^{v_1}, \ldots, I_v^{v_\ell}$ and $I_u^0, I_u^u, I_u^{u_1}, \ldots, I_u^{u_p}$ have the same color, then either all such intervals belong to different categories or there are at least two intervals of the same category but they do not intersect. In both cases, nodes v and u are not adjacent in G, and thus they can receive the same color. Moreover, since k mutually overlapping intervals are colored the same, the same color is used in G exactly k times.

Summarizing, solving the instance of Interval Coloring with Unit Bounded Categories corresponds to solving the instance of Balanced Coloring. □

By the above result, Problem 2, and hence Server Allocation with Bounded Simultaneous Requests, is computationally intractable. Therefore, one is forced to abandon the search for fast algorithms that find optimal solutions. Thus, one can devise fast algorithms that provide solutions which are fairly close to optimal. This strategy is followed in the next section, where a polynomial-time approximation algorithm is exhibited for providing solutions that will never differ from optimal by more than a specified percentage.

3 Approximation Algorithm for Interval Coloring with Bounded Overlapping

An α-approximation algorithm for a minimization problem is a polynomial-time algorithm producing a solution of value $appr(x)$ on input x such that, for all the inputs x,

$$appr(x) \leq \alpha \cdot opt(x),$$

where $opt(x)$ is the value of the optimal solution on x. In other words, the approximate solution is guaranteed to never be greater than α times the optimal solution [2]. For the sake of simplicity, from now on, $appr(x)$ and $opt(x)$ will be simply denoted by $appr$ and opt, respectively.

The following result shows that the optimization version of Problem 2 cannot be α-approximated for any $\alpha < \frac{4}{3}$.

Corollary 3.1. *The optimization version of Interval Coloring with Unit Bounded Categories admits no α-approximation algorithm for $\alpha < \frac{4}{3}$.*

Proof: By the reduction in the proof of Lemma 2.2, if there is an α-approximation algorithm with $\alpha < \frac{4}{3}$, then there is a decision algorithm for Balanced Coloring with $b = 3$. Indeed, if Balanced Coloring admits a 3-coloring, then $opt = 3$, and the α-approximation algorithm gives a 3-coloring since $appr < \frac{4}{3}3 = 4$. Otherwise, if Balanced Coloring does not admit any 3-coloring, then the α-approximation algorithm gives a solution with $appr \geq 4$. □

Assume that the intervals in I arrive one by one, and are indexed by non-decreasing starting times. When an interval I_i arrives, it is immediately and irrevocably colored, and the next interval I_{i+1} becomes known only after I_i has been colored. An algorithm that works in such an ongoing manner is said *on-line* [5]. On-line algorithms are opposed to *off-line* algorithms, where the intervals are not colored as they become available, but they are all colored only after the entire sequence I of intervals is known.

A simple polynomial-time on-line algorithm for the more general Interval Coloring with Bounded Overlapping can be designed based on the following greedy strategy:

Algorithm 1 GREEDY (I_i)

- To color interval I_i use, if possible, a color already used for previous intervals, otherwise use a brand new color.

Theorem 3.2. *Algorithm Greedy provides a 2-approximation and, asymptotically, a $\left(2 - \frac{h}{k}\right)$-approximation, where $h = \min_{c \in C} h_c$, for Interval Coloring with Bounded Overlapping.*

Proof: Let $appr = \phi$ be the solution given by the algorithm and assume the colors $1, \ldots, \phi$ have been introduced in this order. Let $I_r = [s_r, e_r)$ be the first interval colored ϕ. Let Ω_1 be the set of intervals in I containing s_r and let Ω_2 be the set of intervals in I containing s_r and with category c_r. Hence, $\Omega_2 \subseteq \Omega_1$. Let $\omega_1 = |\Omega_1|$ and $\omega_2 = |\Omega_2|$. Clearly, $opt \geq \lceil \frac{\omega_1}{k} \rceil$ and $opt \geq \lceil \frac{\omega_2}{h_{c_r}} \rceil$. Color ϕ was introduced to color I_r because, for every $\gamma \in \{1, \ldots, \phi - 1\}$, at least one of the following two conditions held:

1. at least k intervals in Ω_1 have color γ;

2. at least h_{c_r} intervals in Ω_2 have color γ.

For $i = 1, 2$, let n_i be the number of colors in $\{1, \ldots, \phi - 1\}$ for which condition i holds (if for a color both conditions hold, then choose one of them arbitrarily). Hence, $n_1 + n_2 = \phi - 1$ or, equivalently, $appr = \phi = n_1 + n_2 + 1$. Clearly, $\omega_1 \geq kn_1 + h_{c_r} n_2 + 1$ and $\omega_2 \geq h_{c_r} n_2 + 1$. Therefore:

$$opt \geq \max\left\{\left\lceil\frac{\omega_1}{k}\right\rceil ; \left\lceil\frac{\omega_2}{h_{c_r}}\right\rceil\right\} \geq \max\left\{\left\lceil\frac{kn_1 + h_{c_r} n_2 + 1}{k}\right\rceil ; \left\lceil\frac{h_{c_r} n_2 + 1}{h_{c_r}}\right\rceil\right\}$$

$$\geq \max\left\{n_1 + \left\lceil\frac{h_{c_r} n_2 + 1}{k}\right\rceil ; n_2 + 1\right\} \geq \max\left\{n_1 + \frac{h}{k} n_2 ; n_2 + 1\right\},$$

where $h = \min_{c \in C} h_c$.

If $n_2 + 1 \geq n_1 + \frac{h}{k} n_2$, then:

$$\frac{appr}{opt} \leq \frac{n_1 + n_2 + 1}{n_2 + 1} \leq \frac{n_2(1 - \frac{h}{k}) + 1 + n_2 + 1}{n_2 + 1} = 2 - \frac{h}{k}\frac{n_2}{n_2 + 1} \leq 2$$

If $n_2 + 1 \leq n_1 + \frac{h}{k} n_2$, then:

$$\frac{appr}{opt} \leq \frac{n_1 + n_2 + 1}{n_1 + \frac{h}{k} n_2} \leq \frac{n_1 + n_1 + \frac{h}{k} n_2}{n_1 + \frac{h}{k} n_2} = 1 + \frac{n_1}{n_1 + \frac{h}{k} n_2} \leq 2.$$

Therefore, Algorithm 1 gives a 2-approximation.

To achieve the asymptotic approximation, first observe that $opt \geq \max\left\{n_1 + \frac{h}{k} n_2 ; n_2\right\}$. Moreover, when $opt \to \infty$, also $\phi \to \infty$, and hence $\phi' = \phi - 1 \to \infty$, too. The ratio $\frac{appr}{opt}$ is maximum when opt is minimum, that is, since $opt \geq n_2$ and $opt \geq n_1 + \frac{h}{k} n_2$, for $opt' = n_2 = n_1 + \frac{h}{k} n_2$. Thus,

$$\frac{appr}{opt} \to \frac{\phi'}{opt'} = \frac{n_1 + n_2}{n_2} = \frac{(1 - \frac{h}{k}) n_2 + n_2}{n_2} = 2 - \frac{h}{k}.$$

Hence, asymptotically, Algorithm 1 gives a $\left(2 - \frac{h}{k}\right)$-approximation. □

The following Corollary 3.3 shows that the $\left(2 - \frac{h}{k}\right)$-approximation is the best possible for Algorithm 1, even in the case that $h = 1$, $k = 2$, and no interval properly contains another interval.

Corollary 3.3. *Algorithm Greedy admits no α-approximation with $\alpha < 2 - \frac{1}{k}$ for Interval Coloring with Unit Bounded Categories.*

Proof: It is shown that there is an instance for which the $(2 - \frac{1}{k})$-approximation is achievable. Consider the particular input instance consisting of the k^2 mutually overlapping intervals $I_1, I_2 \ldots, I_{k^2}$ defined as follows:

- $I_i = [i, i+k^2)$ with category $c_i = \begin{cases} i & \text{if } \ 1 \leq i \leq k^2 - k \\ k^2 - k + 1 & \text{if } \ k^2 - k + 1 \leq i \leq k^2 \end{cases}$

The Greedy algorithm colors the interval I_i as soon as it becomes available, that is at time i, thus assigning color 1 to $I_1, \ldots, I_k$, color 2 to $I_{k+1}, \ldots, I_{2k}$, and so on. In particular, color j is assigned to $I_{(j-1)k+1}, \ldots, I_{jk}$, and color $k-1$ is given to $I_{(k-2)k+1}, \ldots, I_{(k-1)k}$. Moreover, for the remaining intervals $I_{k^2-k+1}, \ldots, I_{k^2}$, k additional colors are employed, one for each interval. Overall, $2k-1$ colors are used. However, an optimal off-line algorithm, that knows in advance the entire sequence of intervals, uses k colors assigning color 1 to intervals $I_1, \ldots, I_{k-1}$ and interval I_{k^2-k+1}, color 2 to intervals $I_k, \ldots, I_{2(k-1)}$ and I_{k^2-k+2}, and so on. In particular, color j is assigned to intervals $I_{(j-1)(k-1)+1}, \ldots, I_{j(k-1)}$ and I_{k^2-k+j}, while color k is given to $I_{(k-1)(k-1)+1}, \ldots, I_{k(k-1)}$ and I_{k^2}. Therefore, $\frac{appr}{opt} = \frac{2k-1}{k} = 2 - \frac{1}{k}$.

□

4 Approximation Algorithm for Weighted Interval Coloring with Bounded Overlapping

Consider now a generalization of Server Allocation with Bounded Simultaneous Requests, where each request r is also characterized by an integer *bandwidth rate* w_r, and the bounds on the number of simultaneous requests to be served by the same server are replaced by bounds on the sum of the bandwidth rates of the simultaneous requests assigned to the same server. Such a problem can be formulated as a weighted generalization of Problem 1 as follows.

Problem 5 (Weighted Interval Coloring with Bounded Overlapping). *Given a set I of intervals, with each interval I_r characterized by a category c_r and an integer weight w_r, an integer k, and an integer h_c for each category c, assign a color to each interval in such a way that the sum of the weights for mutually overlapping intervals receiving the same color is at most k, the sum of the weights for mutually overlapping intervals of category c receiving the same color is at most h_c, and the minimum number of colors is used.*

More formally, denote by

- $I[t]$ the intervals *active* at instant t, that is, $I[t] = \{I_r \in I : s_r \leq t \leq e_r\}$;
- $I[c]$ the intervals belonging to the same category c, that is $I[c] = \{I_r \in I : c_r = c\}$; and
- $I(\gamma)$ the set of intervals colored γ.

Moreover, let $I(\gamma)[t] = I(\gamma) \cap I[t]$ be the intervals colored γ and active at instant t. Finally, let $I(\gamma)[t][c] = I(\gamma)[t] \cap I[c]$ be the intervals of category c, colored γ, and active at instant t.

The constraints on the sum of the weights for mutually overlapping intervals receiving the same color can be stated as follows:

$$\sum_{I_r \in I(\gamma)[t]} w_r \leq k \quad \forall \gamma, \forall t \tag{1}$$

$$\sum_{I_r \in I(\gamma)[t][c]} w_r \leq h_c \quad \forall \gamma, \forall t, \forall c \tag{2}$$

Note that Problem 1 is a particular case of Problem 5, where $w_r = 1$ for each interval I_r.

An approximation on-line algorithm for Problem 5 (which contains Bin-Packing [1] as a special case) is presented below.

Algorithm 2 FIRST-COLOR (I_i)

- To color interval I_i use the lowest possible indexed color among those already used for previous intervals. If no such color exists, use a brand new color.

Theorem 4.1. *Algorithm First-Color asymptotically provides a constant approximation for Weighted Interval Coloring with Bounded Overlapping.*

Proof: Assume the on-line First-Color algorithm employs ϕ colors. Consider the first interval $I_r = [s_r, e_r)$ which is colored ϕ. At time s_r, I_r cannot be colored with any color in $\{1, \dots, \phi - 1\}$ since otherwise at least one of the constraints (1) and (2) would be violated. Two cases may occur, depending on whether w_r is smaller or larger than $\frac{h}{2}$, where $h = \min_{c \in C} h_c$.

Case 1: Suppose $w_r \leq \frac{h}{2}$. Let Ω_1 be the set of intervals in I containing s_r and let Ω_2 be the set of intervals in I containing s_r and having category c_r. Let $w(\Omega_1)$ (resp., $w(\Omega_2)$) be the sum of the weights of the intervals in Ω_1 (resp., Ω_2). Clearly, $opt \geq \lceil \frac{w(\Omega_1)}{k} \rceil$ and $opt \geq \lceil \frac{w(\Omega_2)}{h_{c_r}} \rceil$.

Color ϕ was used for I_r because, for every $\gamma \in \{1, \dots, \phi - 1\}$, at least one of the following two conditions held:

1. $w(\Omega_1(\gamma)) \geq \frac{k}{2}$, where $\Omega_1(\gamma)$ are the intervals in Ω_1 already colored γ,
2. $w(\Omega_2(\gamma)) \geq \frac{h_{c_r}}{2}$, where $\Omega_2(\gamma)$ are the intervals in Ω_2 with category c_r already colored γ.

For $i = 1, 2$, let n_i be the number of colors in $\{1, \dots, \phi - 1\}$ for which i holds (if for a color both conditions hold, then choose one of them arbitrarily). Hence, $n_1 + n_2 = \phi - 1$. Clearly, $w(\Omega_1) \geq n_1 \frac{k}{2} + n_2 \frac{h_{c_r}}{2}$ and $w(\Omega_2) \geq n_2 \frac{h_{c_r}}{2}$. Therefore,

$$opt \geq \max\left\{ \left\lceil \frac{w(\Omega_1)}{k} \right\rceil ; \left\lceil \frac{w(\Omega_2)}{h_{c_r}} \right\rceil \right\} \geq \max\left\{ \left\lceil \frac{n_1 k + n_2 h}{2k} \right\rceil ; \left\lceil \frac{n_2}{2} \right\rceil \right\}$$

If $\frac{n_2}{2} \geq \frac{n_1 k + n_2 h}{2k}$, then:

$$\frac{appr}{opt} \leq 2 \frac{n_1 + n_2 + 1}{n_2} \leq 2 \frac{n_2(2 - \frac{h}{k}) + 1}{n_2} \rightarrow 2\left(2 - \frac{h}{k}\right)$$

If $\frac{n_2}{2} \leq \frac{n_1 k + n_2 h}{2k}$, then:

$$\frac{appr}{opt} \leq \frac{n_1 + n_2 + 1}{\frac{n_1 k + n_2 h}{2k}} = \frac{2(n_1 + n_2 + 1)}{n_1 + \frac{h}{k} n_2} \leq 4$$

by the bound proved in Theorem 3.2.

Case 2: Suppose $w_r > \frac{h}{2}$. Two further subcases may come up.

Case 2.1: For each color $\gamma \in \{\lceil \frac{\phi}{2} \rceil, \ldots, \phi - 1\}$ at least one of the two following conditions holds:

$$\sum_{I_r \in I(\gamma)[s_r]} w_r \geq \frac{k}{2} \tag{3}$$

$$\sum_{I_r \in I(\gamma)[s_r][c]} w_r \geq \frac{h_c}{2} \quad \text{for some } c \tag{4}$$

Therefore, the sum of the weights of the intervals active at time s_r is at least $\frac{h}{2} \lfloor \frac{\phi}{2} \rfloor$. Thus,

$$opt \geq \left\lceil \frac{\frac{h}{2} \lfloor \frac{\phi}{2} \rfloor}{k} \right\rceil \qquad \text{and} \qquad \frac{appr}{opt} \leq \frac{\phi}{\frac{\frac{h}{2} \lfloor \frac{\phi}{2} \rfloor}{k}} \rightarrow \frac{5k}{h}$$

Case 2.2: There is a color $\overline{\gamma} \in \{\lceil \frac{\phi}{2} \rceil, \ldots, \phi - 1\}$ for which both conditions (3) and (4) do not hold. Thus, there is an interval $I_{\overline{r}} = [s_{\overline{r}}, e_{\overline{r}})$, colored $\overline{\gamma}$, of weight $w_{\overline{r}} < \frac{h}{2}$. When $I_{\overline{r}}$ was colored, $\overline{\gamma}$ was the lowest possible indexed color. Therefore, this subcase reduces to Case 1 above loosing a factor of 2.

In conclusion, algorithm First-Color asymptotically gives a constant approximation. □

It is worthy to note that in the case there are no constraints on the total weight of mutually overlapping intervals of the same category, the above algorithm yields a 4-approximation. This can be easily checked assuming $h = k$ in the proof of Theorem 4.1 and observing that, in the Case 2.1, only condition (3) holds for all the colors γ.

Moreover, note that the worst approximation constant of Algorithm 2 is given by $\frac{5k}{h}$ when $\frac{k}{h} > \frac{8}{5}$, and by 8 otherwise. In the next subsection, an off-line algorithm for Problem 5 is proposed which guarantees an 8-approximation even in the case that $\frac{k}{h} > \frac{8}{5}$.

4.1 Off-Line Approximation Algorithm

An off-line approximation algorithm for Problem 5 runs three passes over the entire input sequence I. Each pass scans the intervals in I by non decreasing starting times and delivers a new set of colors. Hence, denoted by ϕ_i the number of colors delivered in pass i, the total number of colors employed is $\phi = \phi_1 + \phi_2 + \phi_3$.

Algorithm 3 THREE-PASS-GREEDY (I)

PASS 1: Color all those intervals I_r such that $w_r > \frac{k}{2}$;

PASS 2: Color all those remaining intervals I_r such that $w_r > \frac{1}{2}h_{c_r}$;

PASS 3: Color all the remaining intervals.

Note that, in the first pass, the same color cannot be assigned to two overlapping intervals, and hence the problem reduces to coloring an interval graph, which can be done optimally in polynomial time [3]. Therefore, $\phi_1 \leq \mathit{opt}$. In the second and third passes, instead, Algorithm 1 is employed. For the second pass alone, an approximation of 3 can be shown as follows, obtaining $\phi_2 \leq 3 \cdot \mathit{opt}$.

Lemma 4.2. *The second pass of Algorithm Three-Pass-Greedy uses $\phi_2 \leq 3 \cdot \mathit{opt}$ colors.*

Proof: Assume the colors $1, \ldots, \phi_2$ have been introduced in this order and let $I_r = [s_r, e_r)$ be the first interval colored ϕ_2. Observe that since the intervals with weight greater than $\frac{k}{2}$ have been colored in the first pass, it follows that in the second pass only intervals with weight smaller than or equal to $\frac{k}{2}$ are considered.

Let $w(\Omega_1)$ and ω_2 be defined as in the proofs of Theorem 4.1 and 3.2, respectively, but restricted to the intervals colored during the second pass. Clearly, $\mathit{opt} \geq \lceil \frac{w(\Omega_1)}{k} \rceil$ and $\mathit{opt} \geq \lceil \omega_2 \rceil$ since two intervals with the same category cannot be colored the same during the second pass. Color ϕ_2 was introduced because for every $\gamma \in \{1, \ldots, \phi_2 - 1\}$ at least one of the following two conditions held:

1. $w(\Omega_1(\gamma)) > \frac{k}{2}$;

2. there is an interval with category c_r already colored γ.

For $i = 1, 2$, let n_i be the number of colors in $\{1, \ldots, \phi_2 - 1\}$ for which condition i holds. Clearly, $n_1 + n_2 = \phi_2 - 1$, $w(\Omega_1) \geq n_1 \frac{k}{2} + h n_2$, and $\omega_2 \geq n_2 + 1$. Therefore,

$$\mathit{opt} \geq \max\left\{ \left\lceil \frac{w(\Omega_1)}{k} \right\rceil ; n_2 + 1 \right\} \geq \max\left\{ \frac{n_1}{2} + \frac{h}{k} n_2 ; n_2 + 1 \right\}$$

which, combined with $\mathit{appr} = \phi_2 = n_1 + n_2 + 1$, is enough to prove a 3-approximation by a simple modification of the proof of Theorem 3.2. □

For the third pass alone, the analysis on the approximation guarantee of Theorem 3.2 can again be easily adapted by loosing a factor of 2, thus having $\phi_3 \leq 4 \cdot \mathit{opt}$. As a consequence, $\mathit{appr} = \phi_1 + \phi_2 + \phi_3 \leq 8 \cdot \mathit{opt}$, and the following result holds.

Corollary 4.3. *Algorithm Three-Pass-Greedy provides an 8-approximation for Weighted Interval Coloring with Bounded Overlapping.* □

5 Approximation Algorithms for Further Generalizations

This section considers two generalizations of the Server Allocation with Bounded Simultaneous Requests problem, where each request r is characterized by real bandwidths, normalized in $[0,1]$ for analogy with the Bin-Packing problem [1].

In the first generalization, which contains Multi-Dimensional Bin-Packing as a special case, each request r is characterized by a *k-dimensional* bandwidth rate $\mathbf{w}_r = (w_r^{(1)}, \ldots, w_r^{(k)})$, where the c-th component specifies the bandwidth needed for the c-th category and k is the number of categories, i.e. $k = |C|$. The overall sum of the bandwidth rates of the simultaneous requests of the same category assigned to the same server at the same time is bounded by 1, which implies that the total sum of the bandwidth rates over all the categories is bounded by k.

In the second generalization, each request r is characterized by a *gender bandwidth rate* g_{r,c_r} associated to the category c_r and by a bandwidth rate w_r. The overall sum of the bandwidth rates of the simultaneous requests assigned to the same server at the same time is bounded by 1, as well as the overall sum of the gender bandwidth rates of the simultaneous requests of the same category assigned to the same server at the same time, which is also bounded by 1.

5.1 Multi-Dimensional Weighted Interval Coloring with Unit Overlapping

The first generalization of the server allocation problem can be formulated as the following variant of the interval coloring problem.

Problem 6 (Multi-Dimensional Weighted Interval Coloring with Unit Overlapping). *Given a set I of intervals, with each interval I_r characterized by a k-dimensional weight $\mathbf{w}_r = (w_r^{(1)}, \ldots, w_r^{(k)})$, where $w_r^{(c)} \in [0,1]$, for $1 \leq c \leq k$, assign a color to each interval in such a way that the overall sum of the weights of the same category for mutually overlapping intervals receiving the same color is bounded by 1.*

More formally, according to the notations introduced in Section 4, the constraints on the sum of the weights of the same category for mutually overlapping intervals receiving the same color can be stated as follows:

$$\sum_{I_r \in I(\gamma)[t][c]} w_r^{(c)} \leq 1 \qquad \forall \gamma,\, \forall t,\, \forall c \tag{5}$$

Note that the above constraints, added up over all the categories in C, imply the following redundant constraints:

$$\sum_{c=1}^{k} \sum_{I_r \in I(\gamma)[t][c]} w_r^{(c)} \leq k \qquad \forall \gamma,\, \forall t \tag{6}$$

which are analogous to constraints (1) of Problem 5. Problem 6 can also be solved on-line by Algorithm 2, introduced in the previous section.

Theorem 5.1. *The First-Color algorithm provides a 4k-approximation for Multi-Dimensional Weighted Interval Coloring with Unit Overlapping.*

Proof: Assume the on-line algorithm employs ϕ colors. Consider the first interval $I_{\overline{r}} = [s_{\overline{r}}, e_{\overline{r}})$ which is colored ϕ. At time $s_{\overline{r}}$, $I_{\overline{r}}$ cannot be colored with any color in $\{1, \dots, \phi - 1\}$.

Consider two cases.

Case 1: There is a color $\overline{\gamma}$ among $\{\lceil \frac{\phi}{2} \rceil, \dots, \phi - 1\}$ such that for each component c, with $1 \le c \le k$:

$$\sum_{I_r \in I(\overline{\gamma})[s_{\overline{r}}][c]} w_r^{(c)} \le \frac{1}{2}$$

Let $I_{r'}$ be any interval in $I(\overline{\gamma})[s_{\overline{r}}][c]$. Clearly, $w_{r'}^{(c)} \le \frac{1}{2}$, for all c. Consider now instant $s_{r'}$, when interval $I_{r'}$ was colored $\overline{\gamma} \ge \lceil \frac{\phi}{2} \rceil$. Since $I_{r'}$ cannot be colored with any color in $\{1, \dots, \lceil \frac{\phi}{2} \rceil - 1\}$, then for every $\gamma \in \{1, \dots, \lceil \frac{\phi}{2} \rceil - 1\}$ and for every c, with $1 \le c \le k$:

$$\sum_{I_r \in I(\gamma)[s_{r'}][c_{r'}]} w_r^{(c)} > \frac{1}{2}$$

Hence,

$$opt \ge \sum_{I_r \in I[s_{r'}]} \frac{1}{k} \sum_{c=1}^{k} w_r^{(c)} \ge \frac{1}{k} \sum_{\gamma=1}^{\lceil \frac{\phi}{2} \rceil - 1} \sum_{I_r \in I(\gamma)[s_{r'}]} \sum_{c=1}^{k} w_r^{(c)} \ge \frac{1}{k} \sum_{\gamma=1}^{\lceil \frac{\phi}{2} \rceil - 1} \frac{1}{2} \ge \frac{1}{k} \frac{\phi}{2} \frac{1}{2} \ge \frac{\phi}{4k}$$

Case 2: For every color γ in $\{\lceil \frac{\phi}{2} \rceil, \dots, \phi - 1\}$, there is a category c, with $1 \le c \le k$, such that

$$\sum_{I_r \in I(\gamma)[s_r][c]} w_r^{(c)} > \frac{1}{2}.$$

By a reasoning analogous to Case 1, it follows that:

$$opt \ge \sum_{I_r \in I[s_{r'}]} \frac{1}{k} \sum_{c=1}^{k} w_r^{(c)} \ge \frac{1}{k} \sum_{\gamma=\lceil \frac{\phi}{2} \rceil}^{\phi - 1} \sum_{I_r \in I(\gamma)[s_{r'}]} \sum_{c=1}^{k} w_r^{(c)} \ge \frac{1}{k} \sum_{\gamma=\lceil \frac{\phi}{2} \rceil}^{\phi - 1} \frac{1}{2} \ge \frac{1}{k} \frac{\phi}{2} \frac{1}{2} \ge \frac{\phi}{4k}$$

Since $appr = \phi$, an approximation of $4k$ holds. □

The above problem, when considered as an off-line problem, is APX-hard since it contains Multi-Dimensional Bin-Packing as a special case, which has been shown to be APX-hard [7] already for $k = 2$.

5.2 Double Weighted Interval Coloring with Unit Overlapping

The second generalization of the server allocation problem that is considered can be stated as an interval coloring problem as follows.

Problem 7 (Double Weighted Interval Coloring with Unit Overlapping). *Given a set* I *of intervals, with each interval* I_r *characterized by a gender weight* $g_{r,c_r} \in (0,1]$ *associated to the category* c_r *and by a bandwidth weight* $w_r \in (0,1]$*, assign a color to each interval in such a way that the overall sum of the gender weights for mutually overlapping intervals of the same category receiving the same color is bounded by* 1*, the overall sum of the bandwidth weights for mutually overlapping intervals receiving the same color is bounded by* 1*, and the minimum number of colors is used.*

Formally, the constraints of Problem 7 are given below:

$$\sum_{I_r \in I(\gamma)[t]} w_r \leq 1 \qquad \forall \gamma, \forall t \tag{7}$$

$$\sum_{I_r \in I(\gamma)[t][c]} g_{r,c} \leq 1 \qquad \forall \gamma, \forall t, \forall c \tag{8}$$

Problem 7 can again be solved on-line by Algorithm 2, introduced in the previous section.

Theorem 5.2. *The First-Color algorithm provides a* 10*-approximation for Double Weighted Interval Coloring with Unit Overlapping.*

Proof: Assume the on-line algorithm employs ϕ colors. Consider the first interval $I_{\overline{r}} = [s_{\overline{r}}, e_{\overline{r}})$ which is colored ϕ. At time $s_{\overline{r}}$, $I_{\overline{r}}$ cannot be colored with any color in $\{1, \dots, \phi - 1\}$.

Consider three cases.

Case 1: Let $w_{\overline{r}} \leq \frac{1}{2}$ and $g_{\overline{r},c_{\overline{r}}} \leq \frac{1}{2}$.

For every color γ in $\{1, \dots, \phi - 1\}$, at least one of the following two conditions holds:

$$\sum_{I_r \in I(\gamma)[s_{\overline{r}}]} w_r > \frac{1}{2} \tag{9}$$

$$\sum_{I_r \in I(\gamma)[s_{\overline{r}}][c_{\overline{r}}]} g_{r,c_{\overline{r}}} > \frac{1}{2} \tag{10}$$

If there are at least $\left\lceil \frac{\phi}{2} \right\rceil$ colors for which Condition (9) holds, then

$$opt \geq \sum_{\gamma=1}^{\phi-1} \sum_{I_r \in I(\gamma)[s_{\overline{r}}]} w_r \geq \frac{\phi}{2}\frac{1}{2} = \frac{\phi}{4}.$$

If there are at least $\left\lceil \frac{\phi}{2} \right\rceil$ colors for which Condition (10) holds, then

$$opt \geq \sum_{\gamma=1}^{\phi-1} \sum_{I_r \in I(\gamma)[s_{\overline{r}}][c_{\overline{r}}]} g_{r,c_{\overline{r}}} \geq \frac{\phi}{2}\frac{1}{2} = \frac{\phi}{4}.$$

Case 2: Let $w_{\overline{r}} \leq \frac{1}{2}$ and $g_{\overline{r},c_{\overline{r}}} > \frac{1}{2}$.

For every color γ in $\{\left\lceil \frac{\phi}{2} \right\rceil, \ldots, \phi - 1\}$, at least one of the following two conditions holds:

$$\sum_{I_r \in I(\gamma)[s_{\overline{r}}]} w_r > \frac{1}{2} \tag{11}$$

$$\sum_{I_r \in I(\gamma)[s_{\overline{r}}][c_{\overline{r}}]} (g_{r,c_{\overline{r}}} + g_{\overline{r},c_{\overline{r}}}) > 1. \tag{12}$$

Note that, if for a color both Conditions (11) and (12) hold, then Condition (11) is preferred. Two further subcases may arise:

Case 2.1: For at least half of the colors in $\{\left\lceil \frac{\phi}{2} \right\rceil, \ldots, \phi - 1\}$, Condition (11) holds. In such a case:

$$opt \geq \sum_{\gamma=\lceil \frac{\phi}{2} \rceil}^{\phi-1} \sum_{I_r \in I(\gamma)[s_{\overline{r}}]} w_r \geq \frac{\phi}{4}\frac{1}{2} = \frac{\phi}{8}.$$

Case 2.2: For at least half of the colors in $\{\left\lceil \frac{\phi}{2} \right\rceil, \ldots, \phi - 1\}$, Condition (12) holds. Then, either for all such colors Condition (10) holds, and therefore a bound of $\frac{\phi}{8}$ easily follows, or there is one color $\overline{\gamma}$ such that

$$\sum_{I_r \in I(\overline{\gamma})[s_{\overline{r}}][c_{\overline{r}}]} g_{r,c_{\overline{r}}} < \frac{1}{2}.$$

In this subcase, there is an interval $I_{r'} = [s_{r'}, e_{r'})$, colored $\overline{\gamma}$, with $w_{r'} < \frac{1}{2}$ and $g_{r',c_{r'}} < \frac{1}{2}$. When $I_{r'}$ was colored, $\overline{\gamma}$ was the lowest possible indexed color. Therefore, this subcase reduces to Case 1 above loosing a factor of 2.

Case 3: Let $w_{\overline{r}} > \frac{1}{2}$.

Let j be the largest integer such that, for every color γ in $\{\phi - j, \ldots, \phi - 1\}$, it holds

$$\sum_{I_r \in I(\gamma)[s_{\overline{r}}]} w_r > \frac{1}{2}.$$

Then, $j + 1 \leq 2 \cdot opt$ readily follows. Moreover, color $\phi - j - 1$ verifies

$$\sum_{I_r \in I(\phi-j-1)[s_{\overline{r}}]} w_r < \frac{1}{2}.$$

Thus, by Cases 1 and 2, it follows that $\phi - j - 1 \leq 8 \cdot opt$. Therefore, since $\phi = (\phi - j - 1) + (j + 1) \leq 8 \cdot opt + 2 \cdot opt$, a 10-approximation is obtained. □

6 Conclusions

This paper has considered several on-line approximation algorithms for problems arising in infostations, where a set of requests characterized by categories and temporal intervals have to be assigned to servers in such a way that a bounded number of simultaneously requests are assigned to the same server and the number of servers is minimized. However, several questions still remain open. For instance, one could lower the approximation bounds derived for Problems 5, 6 and 7. Moreover, one could consider the scenario in which the number of servers is given in input, each request has a deadline, and the goal is to minimize the overall completion time for all the requests.

References

[1] E.G. Coffman, G. Galambos, S. Martello & D. Vigo, "Bin Packing Approximation Algorithms: Combinatorial Analysis", in *Handbook of Combinatorial Optimization*, D.Z. Du & P.M. Pardalos (Editors), 1999, Kluwer, Dordrecht, pp. 151-207.

[2] M.R. Garey & D.S. Johnson, *Computers and Intractability*, 1979, Freeman, San Francisco.

[3] M.C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, 1980, Academic Press, New York.

[4] D.J. Goodman, J. Borras, N.B. Mandayam & R.D. Yates, "INFOSTATIONS: A New System Model for Data and Messaging Services", *IEEE Vehicular Technology Conference*, 1997, pp. 969-973.

[5] R.M. Karp, "On-Line Algorithms Versus Off-Line Algorithms: How Much is it Worth to Know the Future?", in *Proceedings of the IFIP 12th World Computer Congress. Volume 1: Algorithms, Software, Architecture*, Jan van Leeuwen (Editor), 1992, Elsevier Science Publishers, Amsterdam, pp. 416-429.

[6] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, & H. Shmoys, *Sequencing and Scheduling: Algorithms and Complexity*, 1993, North-Holland, Amsterdam.

[7] G.J. Woeginger, "There is No Asymptotic PTAS for Two-Dimensional Vector Packing", *Information Processing Letters*, 1997, pp. 293-297.

[8] G. Wu, C.W. Chu, K. Wine, J. Evans & R. Frenkiel, "WINMAC: A Novel Transmission Protocol for Infostations", *IEEE Vehicular Technology Conference*, 1999, pp. 1-15.

[9] J. Zander,"Trends and Challenges in Resource Management Future Wireless Networks", *IEEE Wireless Communications & Networks Conference*, 2000.