Interval Scheduling to Maximize Bandwidth Provision

Mordechai Shalom [*] Prudence W.H. Wong (Speaker) [†] Shmuel Zaks [‡]

1 Introduction

The problem. We study an interval scheduling problem in which each job j is associated with a time interval $I_j = [s_j, t_j]$, a minimum required bandwidth a_j, a maximum required bandwidth b_j, and a weight w_j, where s_j, t_j, a_j, b_j and w_j are all integers. The length of the interval I_j is denoted by $|I_j|$ and is defined as $t_j - s_j$. We are given an integer W that denotes the number of colors (amount of bandwidth) available. We denote the set of available colors as $\Lambda = [0, W-1]$. A *coloring* c is to assign to each job j a subset $c(j)$ of the set Λ of colors during the whole interval I_j. A coloring c is *valid* if (1) for any job j the number of colors assigned is between a_j and b_j, i.e., $a_j \leq |c(j)| \leq b_j$; and (2) each color is assigned at each time to at most one interval, i.e., for any two jobs j_1 and j_2 with I_{j_1} and I_{j_2} overlapping, we have $c(j_1) \cap c(j_2) = \emptyset$.

The *weighted bandwidth* allocated to a job j is defined as $|I_j| \cdot w_j \cdot |c(j)|$. The *weighted bandwidth* of a coloring is the sum of the weighted bandwidth of all jobs. The objective of the problem is to find a valid coloring such that the weighted bandwidth is maximum.

We say that a coloring c is *contiguous* if for each job j, the set of colors assigned to j forms an interval, i.e., $c(j) = \{x, x+1, x+2, \cdots, y\}$, for some integers $0 \leq x \leq y < W$. We say that c is *circularly contiguous* if the set of colors $c(j)$ forms an interval $c(j) = \{x, x+1, \cdots, y\}$ for $0 \leq x \leq y < W$ or forms a circular interval, i.e., $c(j) = \{x, x+1, \cdots, W-1, 0, 1, \cdots y\}$, for some integers $0 \leq y < x < W$. We can then define variants of the problem requiring the coloring to be contiguous or circularly contiguous.

Motivation. Bandwidth allocation is common in many network applications such as content distribution networks or mobile clients, which require bandwidth reservations to support hangovers for streaming video [1, 2]. In particular our problem is motivated by the DWDM (dense wavelength division multiplexing) network [4, 5, 7]. In optical networks, high-speed signals are sent through optical fibers using WDM technology in which a signal transmitted from a source to a destination is given some wavelength. The spectrum of light that can be transmitted through the fiber is divided into frequency intervals. When the underlying network topology is a path, it is in analogy to the time line while the available wavelength is in analogy to the available colors in our scheduling problem.

[*] cmshalom@telhai.ac.il. Tel Hai College, Upper Galilee, 12210, Israel.

[†] pwong@liverpool.ac.uk. Department of Computer Science, University of Liverpool, Liverpool, UK.

[‡] zaks@cs.technion.ac.il. Department of Computer Science, Technion, Haifa, Israel. The research of this author was supported in part by a Catedra de Excelencia that he is holding in Universidad Carlos III de Madrid, Departmento de Ingenieria Telematica.

2 Maximizing weighted bandwidth

We first consider non-contiguous coloring of intervals. If the minimum and maximum required bandwidth a_j and b_j are zero and W, respectively, for every job j, the problem is trivial and can be solved by finding a subset of non-overlapping intervals such that the total weighted bandwidth is maximum. In other words, we have to find a maximum weight independent set in the intersection graph of the input intervals.

When $a_j > 0$ and $b_j = W$, the problem becomes non-trivial. Any valid coloring can be considered as the union of two disjoint colorings c_1 and c_2 where c_1 assigns exactly a_j colors to each job j and c_2 assigns additional colors to the jobs. The weighted bandwidth of c_1 is fixed at $\sum_j |I_j| \cdot w_j \cdot a_j$ and so the question is to find a coloring c_2 that gives the maximum weighted bandwidth. We describe the idea of finding c_2 here. Suppose λ is a certain color in the coloring c_1. Consider the set of jobs colored with λ from left to right. The time interval between two such jobs forms a gap, in which the best we can do is to find a maximum weight independent set in this gap and color them λ.

Roughly speaking, we use this idea of "gap" to construct a *gap graph* that represents all possible gaps as a bipartite graph. The two sets of vertices in the bipartite graph are the sets of ending points and starting points of jobs in the input, respectively. The end point of a job j_1 is connected to the start point of another job j_2 and the edge weight is the maximum weight independent set in the gap $[t_1, s_2]$. We then show that the problem can be reduced to finding a maximum weighted matching in this gap graph satisfying certain properties. As we can find maximum weighted matching in polynomial time, our problem can be solved in polynomial time. Thus we get:

Theorem 1 *There is a polynomial time algorithm that finds a (non-contiguous) coloring with maximum weighted bandwidth when $a_j \geq 0$ and $b_j = W$ for all jobs j.*

We further consider the special case when the weight of jobs is the same as the interval size, i.e., $w_j = |I_j|$. Compared to arbitrary weight, we can release the restriction that $b_j = W$ and we can show:

Lemma 2 *When $w_j = |I_j|$, the problem with $a_j \geq 0$ and $b_j \leq W$ can be reduced to a minimum cost maximum flow problem, and is thus solvable in polynomial time.*

3 Contiguous coloring

The optimal coloring found in Section 2 may not be contiguous. In this section we consider contiguous coloring. We first observe that our problem is NP-hard as it generalizes the NP-hard problem interval coloring of interval graphs, which is also known as the dynamic storage allocation problem [3] or the ship-building problem [6]. Let ℓ_t be the number of jobs whose intervals contain a certain time unit $[t, t+1]$. We call ℓ_t the *load* at t. A contiguous coloring c defines a permutation of the ℓ_t jobs and divides the W available colors into ℓ_t segments. There are $\ell_t!$ permutations and the number of possible color assignment is $\binom{W}{\ell_t}$ for each permutation. This gives rise to a dynamic programming approach to solve the problem. We consider, from left to right, each time unit (some time units can be skipped if there is no start and finish of jobs) and keep track of all the possible colorings so far. In each step $[t, t+1]$, for each coloring c for $[t, t+1]$

as described above, we identify all existing colors that "agree"[1] with c and choose the one that results in the maximum weighted bandwidth. We thus get:

Theorem 3 *There is a polynomial time dynamic programming algorithm that finds an optimal contiguous coloring when the load ℓ_t is bounded by a constant for all t.*

We also consider circularly contiguous coloring. We show that a circularly contiguous coloring can be converted to a contiguous coloring with a constant degradation to the weighted bandwidth obtained, as follows.

Theorem 4 *There is a randomized polynomial time algorithm that converts a valid circularly contiguous coloring cc to a valid contiguous coloring c such that the weighted bandwidth of c is at least $\frac{3}{4}$ of that of cc, when $a_j = 1$ for all j.*

We note that this algorithm can be derandomized.

Finally we consider a special instance named "proper" instance. A set of jobs is *proper* if no job whose interval is properly contained in another job. In this case we show that any coloring for a proper instance can be converted to a circularly contiguous coloring with the same weighted bandwidth. Together with Theorem 1, this implies:

Theorem 5 *For proper instances, there is a polynomial time algorithm that finds an optimal circularly contiguous coloring.*

Combining Theorems 4 and 5, we have:

Corollary 6 *For proper instances, there is a polynomial time randomized algorithm for contiguous coloring with approximation ratio $\frac{4}{3}$, when $a_j = 1$ for all j.*

Similar to Lemma 4, this algorithm can be derandomized.

References

[1] L.-O. Burchard, H.-U. Heiss, and C. A. F. D. Rose. Performance issues of bandwidth reservations for grid computing. In *Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*, pages 82–90. IEEE Computer Society, 2003.

[2] D. Ferrari, A. Gupta, and G. Ventre. Distributed advance reservation of real-time connections. In *Wokrshop on Network on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, pages 16–27, 1995.

[3] M. Garey and D. S. Johnson. *Computers and Intractability, A Guide to the Theory of NP-Completeness*. Freeman, 1979.

[4] O. Gerstel. Flexible use of spectrum and photonic grooming. In *Photonics in Switching, OSA (Optical Society of America)Technical Digest*, page PMD3, 2010.

[5] O. Gerstel. Realistic approaches to scaling the IP network using optics. In *Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC)*, pages 1 –3, march 2011.

[6] M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*. North-Holland, 2004.

[7] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, and S. Matsuoka. Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies. *Comm. Mag.*, 47:66–73, nov. 2009.

[1] Two partial colorings agree with each other if they give the same color to each job that both of them color.