Longest Wait First for Broadcast Scheduling

Chandra Chekuri* Sungjin Im† Benjamin Moseley‡

April 12, 2009

Abstract

We consider *online* algorithms for broadcast scheduling. In the pull-based broadcast model there are n unit-sized pages of information at a server and requests arrive online for pages. When the server transmits a page p, all outstanding requests for that page are satisfied. There is a lower bound of $\Omega(n)$ on the competitiveness of online algorithms to minimize average flow-time; therefore we consider resource augmentation analysis in which the online algorithm is given extra speed over the adversary. The *longest-wait-first* (LWF) algorithm is a natural algorithm that has been shown to have good empirical performance [2]. Edmonds and Pruhs showed that LWF is 6-speed $O(1)$-competitive using a very complex analysis; they also showed that LWF is not $O(1)$-competitive with less than 1.618-speed. In this paper we make two main contributions to the analysis of LWF and broadcast scheduling.

- We give an intuitive and easy to understand analysis of LWF which shows that it is $O(1/\epsilon^2)$-competitive for average flow-time with $(4+\epsilon)$ speed. Using a more involved analysis, we show that LWF is $O(1/\epsilon^3)$-competitive for average flow-time with $(3.4+\epsilon)$ speed.
- We show that a natural extension of LWF is $O(1)$-speed $O(1)$-competitive for more general objective functions such as average delay-factor and L_k norms of delay-factor (for fixed k). These metrics generalize average flow-time and L_k norms of flow-time respectively and ours are the first non-trivial results for these objective functions in broadcast scheduling.

*Department of Computer Science, University of Illinois, 201 N. Goodwin Ave., Urbana, IL 61801. chekuri@cs.uiuc.edu. Partially supported by NSF grants CCF-0728782 and CNS-0721899.

†Department of Computer Science, University of Illinois, 201 N. Goodwin Ave., Urbana, IL 61801. im3@uiuc.edu

‡Department of Computer Science, University of Illinois, 201 N. Goodwin Ave., Urbana, IL 61801. bmosele2@uiuc.edu. Partially supported by NSF grant CNS-0721899.

1 Introduction

We consider online algorithms for broadcast scheduling in the pull-based model. In this model there are n pages (representing some form of useful information) available at a server and clients request a page that they are interested in. The server *broadcasts* pages according to some online policy and *all* outstanding requests for a page are satisfied when that page is transmitted/broadcast. This is what distinguishes this model from the standard scheduling models where the server has to process each request separately. Broadcast scheduling is motivated by several applications. Example situations where the broadcast assumption is natural include wireless and satellite networks, LAN based systems and even some multicast systems. See [35, 1, 2, 26] for pointers to applications and systems that are based on this model. In addition to their practical interest, broadcast scheduling has been of much interest in recent years from a theoretical point of view. There is by now a good amount of literature in online and offline algorithms in this model [7, 2, 1, 8, 26]. There is also substantial work in the stochastic and queuing theory literature [18, 17, 33, 34] on related models which make distributional assumptions on the request arrivals. In a certain sense, **LWF** can be shown to be optimal when page arrivals are independent and assumed to have a Poisson distribution [3].

It is fair to say that algorithmic development and analysis for broadcast scheduling have been challenging even in the simplest setting of unit-sized pages; so much so that a substantial amount of technical work has been devoted to the development of *offline* approximation algorithms [28, 23, 24, 25, 4, 5]; many of these offline algorithms are non-trivial and are based on linear programming based methods. Further, most of these offline algorithms, with the exception of [5], are in the resource augmentation model of Kalyanasundaram and Pruhs [27] in which the analysis is done by giving the algorithm a machine with speed $s > 1$ when compared to a speed 1 machine for the adversary. In this paper we are interested in *online* algorithms in the worst-case competitive analysis framework. We consider the problem of minimizing average flow-time (or waiting time) of requests and other more stringent objective functions. It is easy to show an $\Omega(n)$ lower bound on the competitive ratio [28] of any deterministic algorithm and hence we also resort to resource augmentation analysis. For average flow-time three algorithms are known to be $O(1)$-competitive with $O(1)$-speed. The first is the natural *longest-wait-first* (**LWF**) algorithm/policy: at any time t that the server is free, schedule the page p for which the total waiting time of all outstanding requests for p is the largest. Edmonds and Pruhs [21], in a complex and original analysis, showed that **LWF** is a 6-speed $O(1)$-competitive algorithm and also that it is not $O(1)$-competitive with a speed less than $(1 + \sqrt{5})/2$. The same authors also gave a different algorithm called BEQUI in [20] and show that it is a $(4 + \epsilon)$-speed $O(1)$-competitive algorithm; although the algorithm has intuitive appeal, the proof of its performance relies on an involved reduction to an algorithm for a non-clairvoyant scheduling problem [19] whose analysis itself is substantially complex. The recent improved result in [22] for the non-clairvoyant problem when combined with the reduction mentioned above leads to a $(2 + \epsilon)$-speed $O(1)$-competitive algorithm; however the new algorithm requires the knowledge of ϵ and hence is not as natural as the other algorithms. The preemptive algorithms in [20, 22] are also applicable when the page sizes are arbitrary; see [26] for empirical evaluation in this model. At a technical level, a main difficulty in online analysis for broadcast scheduling is the fact shown in [28] that no online algorithm can be *locally*-competitive with an adversary[1].

We focus on the **LWF** algorithm in the setting of unit-sized pages. In addition to being a natural greedy policy, it has been shown to outperform other natural policies [2]; moreover, related variants are known to be optimal in certain stochastic settings. It is, therefore, of interest to better understand its performance. We are motivated by the following questions. Is there a simpler and more intuitive analysis of **LWF** for broadcast scheduling than the analysis presented in [21]? Can we close the gap between the upper and lower bounds on the speed requirement of **LWF** to guarantee constant competitiveness? Can we obtain

[1] An algorithm is locally-competitive if at each time t, its queue size is comparable to that of the queue size of the adversary. Many results in standard scheduling are based on showing local-competitiveness.

competitive algorithms for more stringent objective functions than average flow-time such as L_k norms of flow-time, average delay-factor[2] and L_k norms of delay-factor? We give positive answers to these questions.

Results: Our results are for unit-size pages. We make two contributions.

- We give a simple and intuitive analysis of **LWF** that already improves the speed bound in [21]; the analysis shows that **LWF** is $(4+\epsilon)$-speed $O(1/\epsilon^2)$-competitive for average flow time. Using a more complex analysis, we show that **LWF** is $(3.4+\epsilon)$-speed $O(1/\epsilon^3)$-competitive.
- We show that a natural generalization of **LWF** that we call **LF** is $O(k)$-speed $O(k)$-competitive for minimizing the L_k norm of flow time — these bounds extend to average delay factor and L_k norms of delay factor. These are the first non-trivial results for L_k norms in broadcast scheduling for $k > 1$.

L_k norms for flow-time for some small $k > 1$ such as $k = 2, 3$ have been suggested as alternate and robust metrics of performance; see [6, 31] for more on this. Our results show that **LWF**-like algorithms have reasonable theoretical performance even for these more difficult metrics. We derive these additional results in a unified fashion via a general framework that is made possible by our simpler analysis for **LWF**. In our recent work [11] we show that **LF** is not $O(1)$-competitive with any constant speed for the L_∞-norm of delay factor. This suggests that **LF** may require a speed that increases with k to obtain $O(1)$-competitiveness for L_k norms. We note that the algorithms in [20, 22] that perform well for average flow time do not easily extend to the more general objective functions that we consider.

Our analysis of **LWF** borrows several key ideas from [21], however, we make some crucial simplifications. We outline the main differences in Section 1.1 where we give a brief overview of our approach.

Notation and Formal Definitions: We assume that the server has n distinct unit-sized pages of information. We use $J_{p,i}$ to denote i'th request for a page $p \in \{1, \ldots, n\}$. We let $a_{p,i}$ denote the arrival time of the request $J_{p,i}$. The finish time $f_{p,i}$ of a request $J_{p,i}$ under a given schedule/algorithm is defined to be the earliest time after $a_{p,i}$ when the page p is sequentially transmitted by the scheduler; to avoid notational overload we assume that the algorithm is clear from the context. Note that multiple requests for the same page can have the same finish time. The total flow time for an algorithm over a sequence of requests is now defined as $\sum_p \sum_i (f_{p,i} - a_{p,i})$. Delay-factor is a recently introduced metric in scheduling [12, 9, 15]. In the context of broadcast scheduling, each request $J_{p,i}$ has a soft deadline $d_{p,i}$ that is known upon its arrival. The slack of $J_{p,i}$ is $d_{p,i} - a_{p,i}$. The delay-factor of $J_{p,i}$ with finish time $f_{p,i}$ is defined to be $\max(1, \frac{f_{p,i}-a_{p,i}}{d_{p,i}-a_{p,i}})$; in other words it is the ratio of the waiting time of the request to its slack. It can be seen that delay-factor generalizes flow-time since we can set $d_{p,i} = a_{p,i} + 1$ for each (unit-sized) request $J_{p,i}$. Given a scheduling metric such as flow-time or delay-factor that, for each schedule assigns a value $m_{p,i}$ to a request $J_{p,i}$, one can define the L_k norm of this metric in the usual way as $\sqrt[k]{\sum_{(p,i)} m_{p,i}^k}$. Note that minimizing the sum of flow-times or delay-factors is simply the L_1 norm problem. In resource augmentation analysis, the online algorithm is given a faster machine than the optimal offline algorithm. For $s \geq 1$, an algorithm A is s-speed r-competitive if A when the given s-speed machine achieves a competitive ratio of r.

In this paper we assume, for simplicity, the discrete time model. In this model, at each integer time t, the following things happen exactly in the following order; the scheduler make a decision of which page p to broadcast; the page p is broadcast and all outstanding requests of page p are immediately satisfied, thus having finish time t; new requests arrive. Note that new pages which arrive at t are not satisfied by the broadcasting at the time t. It is important to keep it in mind that all these things happen only at integer times. See [21] for more discussion on discrete time versus continuous time models. For the most part, we assume for simplicity of exposition, that the algorithm is given an integer speed s which implies that the algorithm schedules (at most) s requests in each time slot. For this reason we present our analysis for 5-speed and

[2]Delay-factor is a recently introduced metric and we describe it more formally later.

4-speed which extend to $(4+\epsilon)$-speed and $(3.4+\epsilon)$-speed respectively. Due to space constraints we defer the details of the extensions.

Related Work: We give a very brief description of related work due to space constraints. We refer the reader to the survey on online scheduling by Pruhs, Sgall and Torng [32] for a comprehensive overview of results and algorithms (see also [31]). Broadcast scheduling has seen a substantial amount of research in recent years; apart from the work that we have already cited we refer the reader to [28, 13, 29], the recent paper of Chang et al. [12], and the surveys [32, 31] for several pointers to known results. As we mentioned already, a large amount of the work on broadcast scheduling has been on offline algorithms including NP-hardness results and approximation algorithms (often with resource augmentation). With few exceptions [20], almost all the work has focused on the unit-size page assumption. Apart from the work on average flow-time that has been mentioned before, the other work on online algorithms for flow-time are the following. Bartal and Muthukrishnan [8, 12] showed that the first-come-first-serve rule (FCFS) is 2-competitive for maximum flow-time. More recently, Chekuri and Moseley [15] developed a $(2+\epsilon)$-speed $O(1/\epsilon^2)$-competitive algorithm for maximum delay-factor; we note that this algorithm requires knowledge of ϵ. Constant competitive online algorithms for maximizing throughput are given in [30, 10, 36, 16]. Algorithms to minimize L_k norms of flow-time in the context of standard scheduling have been studied in [6] and [14].

1.1 Overview of Analysis

We give a high level overview of our analysis of **LWF**. Let OPT denote some fixed optimal 1-speed offline solution; we overload notation and use OPT also to denote the value of the optimal schedule. Recall that for simplicity of analysis, we assume the discrete-time model in which requests arrive at integer times. For the same reason we analyze **LWF** with an integer speed $s > 1$. We can assume that **LWF** is never idle. Thus, in each time step **LWF** broadcasts s pages and the optimal solution broadcasts 1 page. We also assume that requests arrive at integer times. At time t, a request is in the set $U(t)$ if it is unsatisfied by the scheduler at time t. In the broadcast setting **LWF** with speed s is defined as the following.

Algorithm: $\mathbf{LWF}_s$

- At any integer time t, broadcast the s pages with the largest waiting times, where the waiting time of page p is $\sum_{J_{p,i} \in U(t)} (t - a_{p,i})$.

Our analysis of **LWF** is inspired by that in [21]. Here we summarize our approach and indicate the main differences from the analysis in [21]. Given the schedule of $\mathbf{LWF}_s$ on a request sequence σ, the requests are partitioned into two disjoint sets S (self-chargeable requests) and N (non-self-chargeable requests). Let the total flow time accumulated by $\mathbf{LWF}_s$ for requests in S and N be denoted by $\mathbf{LWF}_s^S$ and $\mathbf{LWF}_s^N$ respectively. Likewise, let OPT^S and OPT^N be the flow-time OPT accumulates for requests in S and N, respectively. S is the set of requests whose flow-time is comparable to their flow-time in OPT. Hence one immediately obtains that $\mathbf{LWF}_s^S \leq \rho \text{OPT}^S$ for some constant ρ. For requests in N, instead of charging them only to the optimal solution, these requests are charged to the total flow time accumulated by **LWF** *and* OPT. It will be shown that $\mathbf{LWF}_s^N \leq \delta \mathbf{LWF}_s + \rho \text{OPT}^N$ for some $\delta < 1$; this is crux of the proof. It follows that $\mathbf{LWF}_s = \mathbf{LWF}_s^S + \mathbf{LWF}_s^N \leq \rho \text{OPT}^S + \rho \text{OPT}^N + \delta \mathbf{LWF} \leq \rho \text{OPT} + \delta \mathbf{LWF}$. This shows that $\mathbf{LWF}_s \leq \frac{\rho}{1-\delta} \text{OPT}$, which will complete our analysis. Perhaps the key idea in [21] is the idea of charging $\mathbf{LWF}_s^N$ to $\mathbf{LWF}_s$ with a $\delta < 1$; as shown in [28], no algorithm for any constant speed can be locally competitive with respect to all adversaries and hence previous approaches in the non-broadcast scheduling context that establish local competitiveness with respect to OPT cannot work.

In [21], the authors do not charge $\mathbf{LWF}_s^N$ directly to $\mathbf{LWF}_s$. Instead, they further split N into two types and do a much more involved analysis to bound the flow-time of the type 2 requests via the flow-time of type 1 requests. Moreover, they first transform the given instance to canonical instance in a complex way and prove the correctness of the transformation. Our simple proof shows that these complex arguments can

be done away with. We also improve the speed bounds and generalize the proof to other objective functions.

1.2 Preliminaries

To show that $\mathbf{LWF}_s^N \leq \delta\mathbf{LWF}_s + \rho\text{OPT}^N$, we will map the requests in N to other requests scheduled by $\mathbf{LWF}_s$ which have comparable flow time. An issue that can occur when using a charging scheme is that one has to be careful not to overcharge. In this setting, this means for a single request $J_{p,i}$ we must bound of the number of requests in N which are charged to $J_{p,i}$. To overcome the overcharging issue, we will appeal to a generalization of Hall's theorem. Here we will have a bipartite graph $G = (X \cup Y)$ where the vertices in X will correspond to requests in N. The vertices in Y will correspond to all requests scheduled by $\mathbf{LWF}_s$. A vertex $u \in X$ will be adjacent to a vertex $v \in Y$ if u and v have comparable flow time and v was satisfied while u was in our queue and unsatisfied; that is, u can be charged to v. We then use a simple generalization of Hall's theorem, which we call *Fractional Hall's Theorem*. Here a vertex of $u \in X$ is matched to a vertex of $v \in Y$ with weight $\ell_{u,v}$ where $\ell_{u,v}$ is not necessarily an integer. Note that a vertex can be matched to multiple vertices.

Definition 1.1 (c-covering)**.** *Let $G = (X \cup Y, E)$ be a bipartite graph whose two parts are X and Y, and let $\ell : E \to [0,1]$ be an edge-weight function. We say that ℓ is a c-covering if for each $u \in X$, $\sum_{(u,v)\in E} \ell_{u,v} = 1$ and for each $v \in Y$, $\sum_{(u,v)\in E} \ell_{u,v} \leq c$.*

The following lemma follows easily from either Hall's Theorem or the Max-Flow Min-Cut Theorem.

Lemma 1.2 (Fractional Hall's theorem)**.** *Let $G = (V = X \cup Y, E)$ be a bipartite graph whose two parts are X and Y, respectively. For a subset S of X, let $N_G(S) = \{v \in Y | uv \in E,\ u \in S\}$, be the neighborhood of S. For every $S \subseteq X$, if $|N_G(S)| \geq \frac{1}{c}|S|$, then there exists a c-covering for X.*

Throughout this paper we will discuss time intervals and unless explicitly mentioned we will assume that they are closed intervals with integer end points. When considering some contiguous time interval $I = [s,t]$ we will say that $|I| = t - s + 1$ is the length of interval I; in other words it is the number of integers in I. For simplicity, we abuse this notation; when X is a set of closed intervals, we let $|X|$ denote the number of distinct integers in some interval of X. Note that $|X|$ also can be seen as the sum of the lengths of maximal contiguous sub-intervals if X is composed of non-overlapping intervals.

To show that Lemma 1.2 holds in a given setting, we show another lemma which will be used throughout this paper. Lemma 1.3 says that union of some fraction of time intervals is comparable to that of whole time interval.

Lemma 1.3. *Let $X = \{[s_1,t_1], \ldots, [s_k,t_k]\}$ be a finite set of closed intervals and let $X' = \{[s'_1,t_1], \ldots, [s'_k,t_k]\}$ be an associated set of intervals such that for $1 \leq i \leq k$, $s'_i \in [s_i,t_i]$ and $|[s'_i,t_i]| \geq \lambda|[s_i,t_i]|$. Then $|X'| \geq \lambda|X|$.*

2 Minimizing Average Flow Time

We focus our attention to minimizing average flow time. A fair amount of notation is needed to clearly illustrate our ideas. Following [21], for each page, we will partition time into intervals via *events*. Events for page p are defined by $\mathbf{LWF}_s$'s broadcasts of page p. When $\mathbf{LWF}_s$ broadcasts page p a new event occurs. An event x for page p will be defined as $E_{p,x} = \langle b_{p,x}, e_{p,x}\rangle$ where $b_{p,x}$ is the beginning of the event and $e_{p,x}$ is the end. Here $\mathbf{LWF}_s$ broadcast page p at time $b_{p,x}$ and this is the xth broadcast of page p. Then $\mathbf{LWF}_s$ broadcast page p at time $e_{p,x}$ and this is the $(x+1)$st broadcast of page p. This starts a new event $E_{p,x+1}$. Therefore, the algorithm $\mathbf{LWF}_s$ does not broadcast p on the time interval $[b_{p,x}+1, e_{p,x}-1]$. Thus, it can be seen that for page p, $e_{p,x-1} = b_{p,x}$. It is important to note that the optimal offline solution may broadcast page p multiple (or zero) times during an event for page p. See Figure 1.

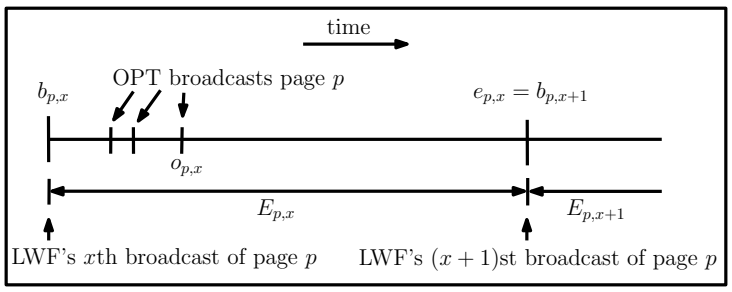

Figure 1: Events for page p.

For each event $E_{p,x}$ we let $\mathcal{J}_{p,x} = \{(p,i) \mid a_{p,i} \in [b_{p,x}, e_{p,x} - 1]\}$ denote the set of requests for p that arrive in the interval $[b_{p,x}, e_{p,x} - 1]$ and are satisfied by $\mathbf{LWF}_s$ at $e_{p,x}$. We let $F_{p,x}$ denote the flow-time in $\mathbf{LWF}_s$ of all requests in $\mathcal{J}_{p,x}$. Similarly we define $F^*_{p,x}$ to be flow time in OPT for all requests in $\mathcal{J}_{p,x}$. Note that OPT may or may not satisfy requests in $\mathcal{J}_{p,x}$ during the interval $[b_{p,x}, e_{p,x}]$.

An event $E_{p,x}$ is said to be self-chargeable and in the set S if $F_{p,x} \leq F^*_{p,x}$ or $e_{p,x} - b_{p,x} < \rho$, where $\rho > 1$ is a constant which will be fixed later. Otherwise the event is non-self-chargeable and is in the set N. Implicitly we are classifying the requests as self-chargeable or non-self-chargeable, however it is easier to work with events rather than individual requests. As the names suggest, self-chargeable events can be easily charged to the flow-time of an optimal schedule. To help analyze the flow-time for non-chargeable events, we set up additional notation and further refine the requests in N.

Consider a non-self-chargeable event $E_{p,x}$. Note that since this event is non-self-chargeable, the optimal solution must broadcast page p during the interval $[b_{p,x} + 1, e_{p,x} - 1]$; otherwise, $F_{p,x} \leq F^*_{p,x}$ and the event is self-chargeable. Let $o_{p,x}$ be the last broadcast of page p by the optimal solution during the interval $[b_{p,x} + 1, e_{p,x} - 1]$. We define $o'_{p,x}$ for a non-self-chargeable event $E_{p,x}$ as $\min\{o_{p,x}, e_{p,x} - \rho\}$. This ensures that the interval $[o'_{p,x}, e_{p,x}]$ is sufficiently long; this is for technical reasons and the reader should think of $o'_{p,x}$ as essentially the same as $o_{p,x}$.

Let $\mathbf{LWF}^S_s = \sum_{p,x:E_{p,x}\in S} F_{p,x}$ and $\mathbf{LWF}^N_s = \sum_{p,x:E_{p,x}\in N} F_{p,x}$ denote the the total flow time for self-chargeable and non self-chargeable events respectively. Similarly, let $\mathrm{OPT}^S = \sum_{p,x:E_{p,x}\in S} F^*_{p,x}$ and $\mathrm{OPT}^N = \sum_{p,x:E_{p,x}\in N} F^*_{p,x}$. For a non-chargeable event $E_{p,x}$ we divide $\mathcal{J}_{p,x}$ into early requests and late requests depending on whether the request arrives before $o'_{p,x}$ or not. Letting $F^e_{p,x}$ and $F^l_{p,x}$ denote the flow-time of early and late requests respectively, we have $F_{p,x} = F^e_{p,x} + F^l_{p,x}$. Let $\mathbf{LWF}^{N^e}_s$ and $\mathbf{LWF}^{N^l}_s$ denote the total flow time of early and late requests of non-self-chargeable events for **LWF**'s schedule, respectively.

The following two lemmas follow easily from the definitions.

Lemma 2.1. $\mathbf{LWF}^S_s \leq \rho\mathrm{OPT}^S$.

Lemma 2.2. $\mathbf{LWF}^{N^l}_s \leq \rho\mathrm{OPT}^N$.

Thus the main task is to bound $\mathbf{LWF}^{N^e}_s$. For a non-chargeable event $E_{p,x}$ we try to charge $F^e_{p,x}$ to events ending in the interval $[o'_{p,x}, e_{p,x} - 1]$. The lemma below quantifies the relationship between $F^e_{p,x}$ and the flow-time of events ending in this interval.

Lemma 2.3. *For any* $0 \leq \lambda \leq 1$, *if* $e_{q,y} \in [\lceil o'_{p,x} + \lambda(e_{p,x} - o'_{p,x})\rceil, e_{p,x} - 1]$ *then* $F_{q,y} \geq \lambda F^e_{p,x}$.

Proof. Let $F_{p,x}(t)$ be the total waiting time accumulated by **LWF** for page p on the time interval $[b_{p,x}, t]$. We divide $F_{p,x}(t)$ into two parts $F^e_{p,x}(t)$ and $F^l_{p,x}(t)$, which are the flow time due to early requests and to late requests, respectively. Note that $F_{p,x}(t) = F^e_{p,x}(t) + F^l_{p,x}(t)$. The early requests arrived before time $o'_{p,x}$, thus, for any $t' \geq \lceil o'_{p,x} + \lambda(e_{p,x} - o'_{p,x})\rceil$, $F^e_{p,x}(t') \geq \lambda F^e_{p,x}(e_{p,x}) = \lambda F^e_{p,x}$.

Since $\mathbf{LWF}_s$ chose to transmit q at $e_{q,y}$ when p was available to be transmitted, it must be the case that $F_{q,y} \geq F_{p,x}(e_{q,y}) \geq F^e_{p,x}(e_{q,y})$. Combining this with the fact that $F^e_{p,x}(e_{q,y}) \geq \lambda F^e_{p,x}$, the lemma follows. □

With the above setup in place, we now prove that $\mathbf{LWF}_s$ is $O(1)$ competitive for $s = 5$ via a clean and simple proof, and for $s = 4$ via a more involved proof. These proofs can be extended to non-integer speeds with better bounds on the speed. In particular, we can show that $\mathbf{LWF}_{3.4+\epsilon}$ is $O(1/\epsilon^3)$-competitive. We omit these extensions in this version.

2.1 Analysis of 5-Speed

This section will be devoted to proving the following main lemma that bounds the flow-time of early requests of non self-chargeable events.

Lemma 2.4. *For $\rho \geq 1$, $\mathbf{LWF}_5^{N^e} \leq \frac{4\rho}{5(\rho-1)}\mathbf{LWF}_5$.*

Assuming the lemma, $\mathbf{LWF}_5$ is $O(1)$-competitive, using the argument outlined earlier in Section 1.1.

Theorem 2.5. $\mathbf{LWF}_5 \leq 90\mathrm{OPT}$.

Proof. By combining Lemma 2.1, 2.2 and 2.4, we have that $\mathbf{LWF}_5 = \mathbf{LWF}_5^S + \mathbf{LWF}_5^{N^l} + \mathbf{LWF}_5^{N^e} \leq \rho\mathrm{OPT}^S + \rho\mathrm{OPT}^N + \frac{4\rho}{5(\rho-1)}\mathbf{LWF}_5$. Setting $\rho = 10$ completes the proof. □

We now prove Lemma 2.4. Let $E_{p,x} \in N$. We define two intervals $I_{p,x} = [o'_{p,x}, e_{p,x} - 1]$ and $I'_{p,x} = [o'_{p,x} + \lceil (e_{p,x} - o'_{p,x})/2 \rceil, e_{p,x} - 1]$. Since $\rho \leq e_{p,x} - o'_{p,x}$, it follows that $|I'_{p,x}| \geq \frac{\rho-1}{2\rho}|I_{p,x}|$. We wish to charge $F^e_{p,x}$ to events (could be in S or N) in the interval $I'_{p,x}$. By Lemma 2.3, each event $E_{q,y}$ that finishes in $I'_{p,x}$ satisfies the property that $F_{q,y} \geq F^e_{p,x}/2$. Moreover, there are $5(\lfloor e_{p,x} - o'_{p,x})/2 \rfloor$ such events to charge to since $\mathbf{LWF}_5$ transmits 5 pages in each time slot. Thus, locally for $E_{p,x}$ there are enough events to charge to if ρ is a sufficiently large constant. However, an event $E_{q,y}$ with $e_{q,y} \in I'_{p,x}$ may also be charged by many other events if we follow this simple local charging scheme. To overcome this overcharging, we resort to a global charging scheme by setting up a bipartite graph G and invoking the fractional Hall's theorem (see Lemma 1.2) on this graph.

The bipartite graph $G = (X \cup Y, E)$ is defined as follows. There is exactly one vertex $u_{p,x} \in X$ for each non-self-chargeable event $E_{p,x} \in N$ and there is exactly one vertex $v_{q,y} \in Y$ for each event $E_{q,y} \in A$, where A is the set of all events. Consider two vertices $u_{p,x} \in X$ and $v_{q,y} \in Y$. There is an edge $u_{p,x}v_{q,y} \in E$ if and only if $e_{q,y} \in I'_{p,x}$. By Lemma 2.3, if there is an edge between $u_{p,x} \in X$ and $v_{q,y} \in Y$ then $F_{q,y} \geq F^e_{p,x}/2$.

The goal is now to show that G has a $\frac{2\rho}{5(\rho-1)}$-covering. Consider any non-empty set $Z \subseteq X$ and a vertex $u_{p,x} \in Z$. Recall that the interval $I_{p,x}$ contains at least one broadcast by OPT of page p. Let $\mathcal{I} = \bigcup_{u_{p,x} \in Z} I_{p,x}$ be the union of the time intervals corresponding to events in Z. Similarly, define $\mathcal{I}' = \bigcup_{u_{p,x} \in Z} I'_{p,x}$.

We claim that $|Z| \leq |\mathcal{I}|$. This is because the optimal solution has 1-speed and it has to do a separate broadcast for each event in Z during $\mathcal{I}$. Now consider the neighborhood of Z, $N_G(Z)$. We note that $|N_G(Z)| = 5|\mathcal{I}'|$ since $\mathbf{LWF}_5$ broadcasts 5 pages for each time slot in $|\mathcal{I}'|$ and each such broadcast is adjacent to an event in Z from the definition of G. From Lemma 1.3, $|\mathcal{I}'| \geq \frac{\rho-1}{2\rho}|\mathcal{I}|$ as we had already observed that $|I'_{p,x}| \geq \frac{\rho-1}{2\rho}|I_{p,x}|$ for each $E_{p,x} \in N$. Thus we conclude that $|N_G(Z)| = 5|\mathcal{I}'| \geq 5\frac{\rho-1}{2\rho}|\mathcal{I}| \geq 5\frac{\rho-1}{2\rho}|Z|$. Since this holds for $\forall Z \subseteq X$, by Lemma 1.2, there must exist a $\frac{2\rho}{5(\rho-1)}$-covering. Let ℓ be such a covering. Finally, we prove that the covering implies the desired bound on $\mathbf{LWF}_5^{N^e}$.

$$\begin{aligned}
\mathbf{LWF}_5^{N^e} &= \sum_{u_{p,x} \in X} F^e_{p,x} \text{ [By Definition]} \\
&= \sum_{u_{p,x}v_{q,y} \in E} \ell_{u_{p,x},v_{q,y}} F^e_{p,x} \text{ [By Def. 1.1, i.e. for } \forall u_{p,x} \in X, \textstyle\sum_{v_{q,y} \in Y} \ell_{u_{p,x},v_{q,y}} = 1]
\end{aligned}$$

$$\leq \sum_{u_{p,x}v_{q,y}\in E} \ell_{u_{p,x},v_{q,y}} 2F_{q,y} \text{ [By Lemma 2.3]}$$

$$\leq \frac{4\rho}{5(\rho-1)} \sum_{v_{q,y}\in Y} F_{q,y} \text{ [Change order of } \textstyle\sum \text{ and } \ell \text{ is a } \frac{2\rho}{5(\rho-1)}\text{-covering]}$$

$$\leq \frac{4\rho}{5(\rho-1)} \mathbf{LWF}_5. \text{ [Since } Y \text{ includes all events]}$$

This finishes the proof of Lemma 2.4.

Remark 2.6. *If non-integer speeds are allowed then the analysis in this subsection can be extended to show that* **LWF** *is* $4+\epsilon$*-speed* $O(1+1/\epsilon^2)$*-competitive.*

2.2 Analysis of 4-Speed

Due to insufficient space, we only sketch the key idea. We remind the reader that early requests of each non-self-chargeable event $E_{p,x}$ were charged to only half the events that ended on $[o'_{p,x}, e_{p,x}-1]$. Thus, fully utilizing all the events, which end during $[o'_{p,x}, e_{p,x}-1]$, can improve the speed. Lemma 2.3, however, does not provide a good comparison between $F^e_{p,x}$ and flow time of event $E_{r,z}$ which is done close to $o'_{p,x}$. We overcome this by further refining the class of non self-chargeable events into `Type1` and `Type2`. For an event $E_{p,x}$ in the interesting class `Type2`, we are able to show that all events in $[o'_{p,x}, e_{p,x}-1]$ have comparable flow-time to that of $E_{p,x}$. This allows us to effectively charge $E_{p,x}$ to events done at $o_{p,x}$; note that for any two events $E_{p,x}$ and $E_{q,y}$ in N, $o_{p,x} \neq o_{q,y}$. The proof is technical and requires several parameters; details can be found in Appendix A.

3 Generalization to Delay-Factor and L_k Norms

In this section, our proof techniques are extended to show that a generalization of **LWF** is $O(1)$-speed $O(1)$-competitive for minimizing the average delay-factor and minimizing the L_k-norm of the delay-factor. Recall that flow-time can be subsumed as a special case of delay-factor. Thus, these results will also apply to L_k norms of flow-time. Instead of focusing on specific objective functions, we develop a general framework and derive results for delay-factor and L_k norms as special cases. First, we set up some notation. We assume that for each request $J_{p,i}$ there is a non-decreasing function $m_{p,i}(t)$ that gives the cost/penalty of that $J_{p,i}$ accumulates if it has *waited* for a time of t units after its arrival. Thus the total cost/penalty incurred for a schedule that finishes $J_{p,i}$ at $f_{p,i}$ is $m_{p,i}(f_{p,i}-a_{p,i})$. For flow-time $m_{p,i}(t)=t$ while for delay-factor it is $\max(1, \frac{t-a_{p,i}}{d_{p,i}-a_{p,i}})$. For L_k norms of delay-factor we set $m_{p,i}(t)=\max(1, \frac{t-a_{p,i}}{d_{p,i}-a_{p,i}})^k$. Note that the L_k norm of delay-factor for a given sequence of requests is $\sqrt[k]{\sum_{p,i} m_{p,i}(f_{p,i}-a_{p,i})}$ but we can ignore the outer k'th root by focusing on the inner sum.

A natural generalization of **LWF** to more general metrics is described below; we refer to this (greedy) algorithm as **LF** for Longest First. We in fact describe $\mathbf{LF}_s$ which is given s speed over the adversary.

Algorithm: $\mathbf{LF}_s$

- At any integer time t, broadcast the s pages with the largest m-waiting times where the m-waiting time of page p at t is $\sum_{J_{p,i}\in U(t)} m_{p,i}(t-a_{p,i})$.

Remark 3.1. *The algorithm and analysis do not assume that the functions* $m_{p,i}$ *are "uniform" over requests. In principle each request* $J_{p,i}$ *could have a different penalty function.*

In order to analyze **LF**, we need a lower bound on the "growth" rate of the functions $m_{p,i}()$. In particular we assume that there is a function $h:[0,1]\rightarrow\mathbb{R}^+$ such that $m_{p,i}(\lambda t)\geq h(\lambda)m_{p,i}(t)$ for all $\lambda\in[0,1]$.

It is not to difficult to see that for flow-time and delay-factor we can choose $h(\lambda) = \lambda$, and for L_k norms of flow-time and delay-factor, we can set $h(\lambda) = \lambda^k$. We also define a function $m : \mathbb{R}^+ \to \mathbb{R}^+$ as $m(x) = \max_{(p,i)} m_{p,i}(x)$. The rest of the analysis depends only on h and m.

In the following subsection we outline a generalization of the analysis from Section 2.1 that applies to $\mathbf{LF}_s$; the analysis bounds various quantities in terms of the functions $h()$ and $m()$. In Section 3.2, we derive the results for minimizing delay-factor and L_k norms of delay-factor.

3.1 Outline of Analysis

To bound the competitiveness of $\mathbf{LF}_s$, we use the same techniques we used for bounding the competitiveness of $\mathbf{LWF}_s$. Events are again defined in the same fashion; $E_{p,x}$ is the event defined by the x'th transmission of p by $\mathbf{LF}_s$. We again partition events into self-chargeable and non self-chargeable events and charge self-chargeable events to the optimal value and charge non-self-chargeable events to $\delta\mathbf{LF}_s + m(\rho)\mathrm{OPT}^N$ for some $\delta < 1$. For an event $E_{p,x}$, let $M_{p,x}(t) = \sum_{J_{p,i} \in \mathcal{J}_{p,x}} m_{p,i}(t - a_{p,i})$ denote the total m-cost of all requests for p that arrive in $[b_{p,x}, e_{p,x} - 1]$ that are satisfied at $e_{p,x}$. We let $M^*_{p,x}(t)$ be the m-cost of the same set of requests for the optimal solution. An event $E_{p,x}$ is self-chargeable if $M_{p,x} \leq m(\rho)M^*_{p,x}$ or $e_{p,x} - b_{p,x} \leq \rho$ for some constant ρ to be optimized later. The remaining events are non self-chargeable. Again, requests for non-self-chargeable events are divided into early requests and late requests based on whether they arrive before $o'_{p,x}$ or not where $o'_{p,x} = \min\{o_{p,x}, e_{p,x} - \rho\}$. Let $M^e_{p,x}$ and $M^l_{p,x}$ be the flow time accumulated for early and late requests of a non-self-chargeable event $E_{p,x}$, respectively. The values of $\mathbf{LF}^N_s$, $\mathbf{LF}^{N^l}_s$, $\mathbf{LF}^{N^e}_s$, and $\mathbf{LF}^S_s$ are defined in the same way as $\mathbf{LWF}^N_s$, $\mathbf{LWF}^{N^l}_s$, $\mathbf{LWF}^{N^e}_s$, and $\mathbf{LWF}^S_s$. Likewise for OPT. The following two lemmas are analogues of Lemmas 2.1 and 2.2 and follow from definitions.

Lemma 3.2. $\mathbf{LF}^S_s \leq m(\rho)\mathrm{OPT}^S$.

Lemma 3.3. $\mathbf{LF}^{N^l}_s \leq m(\rho)\mathrm{OPT}^N$.

We now show a generalization of Lemma 2.3 that states that any event $E_{q,y}$ such that $e_{q,y}$ is close to $e_{p,x}$ has m-waiting time comparable to the m-waiting time of early requests of $E_{p,x}$.

Lemma 3.4. *Suppose $E_{p,x}$ and $E_{q,y}$ are two events such that $e_{q,y} \in [\lceil o'_{p,x} + \lambda(e_{p,x} - o'_{p,x})\rceil, e_{p,x} - 1]$, $M_{q,y} \geq h(\lambda)M^e_{p,x}$.*

Sketch. Consider an early request $J_{p,i}$ in $\mathcal{J}_{p,x}$ and let $t \in [\lceil o'_{p,x} + \lambda(e_{p,x} - o'_{p,x})\rceil, e_{p,x} - 1]$. Since $a_{p,i} \leq o'_{p,x}$, it follows that $t \geq \lambda(e_{p,x} - a_{p,i}) + a_{p,i}$. Hence, $m_{p,i}(t - a_{p,i}) \geq h(\lambda)m_{p,i}(e_{p,x} - a_{p,i})$. Summing over all early requests, it follows that $M^e_{p,x}(t) \geq h(\lambda)M^e_{p,x}$. Since $\mathbf{LF}_s$ chose to transmit q at $t = e_{q,y}$ instead of p, it follows that $M_{q,y} \geq M_{p,x}(e_{q,y}) \geq M^e_{p,x}(e_{q,y}) \geq h(\lambda)M^e_{p,x}$. □

As in Section 2.1, the key ingredient of the analysis is to bound the waiting time of early requests. We state the analogue of Lemma 2.4 below. Observe that we have an additional parameter β. In Lemma 2.4 we hard wire β to be $1/2$ to simplify the exposition. In the more general setting, the parameter β needs to be tuned based on h.

Lemma 3.5. *For any $0 < \beta < 1$, $\mathbf{LF}^{N^e}_s \leq \frac{\rho}{sh(\beta)(\rho(1-\beta)-1)}\mathbf{LF}_s$, where h is some scaling function for m.*

The proof of the above lemma follows essentially the same lines as that of Lemma 2.4. The idea is to charge $M^e_{p,x}$ to events in the interval $[o'_{p,x} + \lceil\beta(e_{p,x} - o'_{p,x})\rceil, e_{p,x} - 1]$. Using Lemma 3.4, each event in this interval is within a factor of $h(\lambda)$ of $M^e_{p,x}$. The length of this interval is at least $\frac{\rho(1-\beta)-1}{\rho}$ times the length of the interval $[o'_{p,x}, e_{p,x} - 1]$. To avoid overcharging we again resort to the global scheme using fractional Hall's theorem after we setup the bipartite graph. We can then prove the existence of a $\frac{\rho}{s(\rho(1-\beta)-1)}$-covering and since each event can pay to within a factor of $h(\beta)$, the lemma follows.

Putting the above lemmas together we derive the following theorem.

Theorem 3.6. *Let $\beta \in (0,1)$ and $\rho > 1$ be given constants. If s is an integer such that $\frac{\rho}{sh(\beta)(\rho(1-\beta)-1)} \leq \delta < 1$, then algorithm $\mathbf{LF}_s$ is s-speed $\frac{m(\rho)}{1-\delta}$-competitive.*

3.2 Results for Delay-Factor and L_k Norms

We can apply Theorem 3.6 with appropriate choice of parameters to show that $\mathbf{LF}_s$ is $O(1)$-competitive with $O(1)$ speed.

For minimizing average delay-factor we have $h(\lambda) = \lambda$ and $m(x) \leq x$. For this reason, average delay-factor behaves essentially the same as average flow-time and we can carry over the results from flow-time.

Theorem 3.7. *The algorithm* **LF** *is 5-speed $O(1)$ competitive for minimizing the average delay-factor. For non-integer speeds it is $4+\epsilon$-speed $O(1/\epsilon^2)$-competitive.*

The analysis in Section A also extends to delay-factor although it does not fall in the general framework that we outlined in Section 3.1. Thus **LF** is $(3.4+\epsilon)$-speed $O(1/\epsilon^3)$-competitive for average delay-factor.

For L_k norms of delay-factor we have $h(\lambda) = \lambda^k$ and $m(x) \leq x^k$. By choosing $\beta = \frac{k}{k+1}$, $\rho = 90(k+1)$ and $s = 3(k+1)$ in Theorem 3.6, we can show that the algorithm **LF** is $3(k+1)$-speed $O(\rho^k)$-competitive for minimizing $\sum_{p,i} m_{p,i}(f_{p,i} - a_{p,i})$. Thus for minimizing the L^k-norm delay factor, we obtain $\sqrt[k]{O(\rho^k)} = O(\rho)$ competitiveness, which shows the following.

Theorem 3.8. *For $k \geq 1$, the algorithm* **LF** *is $O(k)$-speed $O(k)$-competitive for minimizing L_k-norm of delay-factor.*

4 Conclusion

We gave a simpler analysis of **LWF** for minimizing average flow-time in broadcast scheduling. This not only helps improve the speed bound but also results in extending the algorithm and analysis to more general objective functions such a delay-factor and L_k norms of delay-factor. We hope that our analysis is useful in other scheduling contexts.

Our recent work in [11] shows that **LF** is not $O(1)$-competitive with any speed for L_∞-norm of delay factor, which is equivalent to minimizing the maximum delay factor. Thus, we believe the speed requirement for **LF** to obtain $O(1)$-competitiveness needs to grow with k for L_k-norms of delay factor. It would be interesting to formally prove this. This raises the question of whether there is an alternate algorithm that is $O(1)$-speed $O(1)$-competitive for L_k norms of flow time and delay factor. We remark that the lower bound for **LF** [11] applies only to delay factor and it is open whether **LF** is $O(1)$-speed $O(1)$-competitive for L_k norms of flow time. Can the speed bound on **LWF** for $O(1)$-competitiveness be further improved? Edmonds and Pruhs [21] conjecture that their lower bound of $(1+\sqrt{5}/2)$ is tight. Is there an **LWF**-like algorithm that performs well when page sizes are different?

Acknowledgments: We thank Kirk Pruhs for his helpful comments and encouragement.

References

[1] S. Acharya, M. Franklin, and S. Zdonik. Dissemination-based data delivery using broadcast disks. *Personal Communications, IEEE [see also IEEE Wireless Communications]*, 2(6):50–60, Dec 1995.

[2] Demet Aksoy and Michael J. Franklin. "rxw: A scheduling approach for large-scale on-demand data broadcast. *IEEE/ACM Trans. Netw.*, 7(6):846–860, 1999.

[3] M. H. Ammar and J. W. Wong. The design of teletext broadcast cycles. *Performance Evaluation*, 5(4):235–242, 1985.

[4] Nikhil Bansal, Moses Charikar, Sanjeev Khanna, and Joseph (Seffi) Naor. Approximating the average response time in broadcast scheduling. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 215–221, 2005.

[5] Nikhil Bansal, Don Coppersmith, and Maxim Sviridenko. Improved approximation algorithms for broadcast scheduling. In *SODA '06: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 344–353, 2006.

[6] Nikhil Bansal and Kirk Pruhs. Server scheduling in the l_p norm: a rising tide lifts all boat. In *STOC*, pages 242–250, 2003.

[7] Amotz Bar-Noy, Randeep Bhatia, Joseph (Seffi) Naor, and Baruch Schieber. Minimizing service and operation costs of periodic scheduling. *Math. Oper. Res.*, 27(3):518–544, 2002.

[8] Yair Bartal and S. Muthukrishnan. Minimizing maximum response time in scheduling broadcasts. In *SODA '00: Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms*, pages 558–559, 2000.

[9] Michael A. Bender, Raphaël Clifford, and Kostas Tsichlas. Scheduling algorithms for procrastinators. *J. Scheduling*, 11(2):95–104, 2008.

[10] Wun-Tat Chan, Tak Wah Lam, Hing-Fung Ting, and Prudence W. H. Wong. New results on on-demand broadcasting with deadline via job scheduling with cancellation. In Kyung-Yong Chwa and J. Ian Munro, editors, *COCOON*, volume 3106 of *Lecture Notes in Computer Science*, pages 210–218, 2004.

[11] Benjamin Moseley Chandra Chekuri, Sungjin Im. Minimizing maximum response time and delay factor in broadcasting scheduling. Manuscript, 2009.

[12] Jessica Chang, Thomas Erlebach, Renars Gailis, and Samir Khuller. Broadcast scheduling: algorithms and complexity. In *SODA '08: Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 473–482. Society for Industrial and Applied Mathematics, 2008.

[13] Moses Charikar and Samir Khuller. A robust maximum completion time measure for scheduling. In *SODA '06: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 324–333, 2006.

[14] Chandra Chekuri, Ashish Goel, Sanjeev Khanna, and Amit Kumar. Multi-processor scheduling to minimize flow time with epsilon resource augmentation. In László Babai, editor, *STOC*, pages 363–372, 2004.

[15] Chandra Chekuri and Benjamin Moseley. Online scheduling to minimize the maximum delay factor. In *SODA '09: Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithm*, 2009.

[16] Marek Chrobak, Christoph Dürr, Wojciech Jawor, Lukasz Kowalik, and Maciej Kurowski. A note on scheduling equal-length jobs to maximize throughput. *J. of Scheduling*, 9(1):71–73, 2006.

[17] R. K. Deb. Optimal control of bulk queues with compound poisson arrivals and batch service. *Opsearch.*, 21:227–245, 1984.

[18] R. K. Deb and R. F. Serfozo. Optimal control of batch service queues. *Adv. Appl. Prob.*, 5:340–361, 1973.

[19] Jeff Edmonds. Scheduling in the dark. *Theor. Comput. Sci.*, 235(1):109–141, 2000.

[20] Jeff Edmonds and Kirk Pruhs. Multicast pull scheduling: When fairness is fine. *Algorithmica*, 36(3):315–330, 2003.

[21] Jeff Edmonds and Kirk Pruhs. A maiden analysis of longest wait first. *ACM Trans. Algorithms*, 1(1):14–32, 2005.

[22] Jeff Edmonds and Kirk Pruhs. Scalably scheduling processes with arbitrary speedup curves. In *SODA '09: Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithm*, 2009.

[23] Thomas Erlebach and Alexander Hall. Np-hardness of broadcast scheduling and inapproximability of single-source unsplittable min-cost flow. In *SODA '02: Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 194–202, 2002.

[24] Rajiv Gandhi, Samir Khuller, Yoo-Ah Kim, and Yung-Chun (Justin) Wan. Algorithms for minimizing response time in broadcast scheduling. *Algorithmica*, 38(4):597–608, 2004.

[25] Rajiv Gandhi, Samir Khuller, Srinivasan Parthasarathy, and Aravind Srinivasan. Dependent rounding and its applications to approximation algorithms. *J. ACM*, 53(3):324–360, 2006.

[26] Alexander Hall and Hanjo Täubig. Comparing push- and pull-based broadcasting. or: Would "microsoft watches" profit from a transmitter?. In *Proceedings of the 2nd International Workshop on Experimental and Efficient Algorithms (WEA 03)*, pages 148–164, 2003.

[27] Bala Kalyanasundaram and Kirk Pruhs. Speed is as powerful as clairvoyance. *J. ACM*, 47(4):617–643, 2000.

[28] Bala Kalyanasundaram, Kirk Pruhs, and Mahendran Velauthapillai. Scheduling broadcasts in wireless networks. *Journal of Scheduling*, 4(6):339–354, 2000.

[29] Samir Khuller and Yoo Ah Kim. Equivalence of two linear programming relaxations for broadcast scheduling. *Oper. Res. Lett.*, 32(5):473–478, 2004.

[30] Jae-Hoon Kim and Kyung-Yong Chwa. Scheduling broadcasts with deadlines. *Theor. Comput. Sci.*, 325(3):479–488, 2004.

[31] Kirk Pruhs. Competitive online scheduling for server systems. *SIGMETRICS Perform. Eval. Rev.*, 34(4):52–58, 2007.

[32] Kirk Pruhs, Jiri Sgall, and Eric Torng. *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*, chapter Online Scheduling. 2004.

[33] J. Weiss. Optimal control of batch service queues with nonlinear waiting costs. *Modeling and Simulation*, 10:305–309, 1979.

[34] J. Weiss and S. Pliska. Optimal policies for batch service queueing systems. *Opsearch*, 19(1):12–22, 1982.

[35] J. Wong. Broadcast delivery. *Proceedings of the IEEE*, 76(12):1566–1577, 1988.

[36] Feifeng Zheng, Stanley P. Y. Fung, Wun-Tat Chan, Francis Y. L. Chin, Chung Keung Poon, and Prudence W. H. Wong. Improved on-line broadcast scheduling with deadlines. In Danny Z. Chen and D. T. Lee, editors, *COCOON*, volume 4112 of *Lecture Notes in Computer Science*, pages 320–329, 2006.

A Analysis of 4-speed

In this section, we further improve the speed from 5 to 4 in the discrete time model. We assume the speed $s = 4$ throughout this section.

Theorem A.1. **LWF** *is* 4*-speed* $O(1)$*-competitive.*

Proof. By combining Lemma 2.1, 2.2, A.2 and A.8 (Lemma A.2 and A.8 will be proved soon), it follows that

$$\begin{aligned}\mathbf{LWF}_4 &= \mathbf{LWF}_4^S + \mathbf{LWF}_4^{N^l} + \mathbf{LWF}_4^{N_1^e} + \mathbf{LWF}_4^{N_2^e} \\ &\leq \rho\mathrm{OPT}^S + \rho\mathrm{OPT}^N + \frac{4\rho}{\alpha^2}\mathrm{OPT} + \frac{3-8\alpha-8\gamma}{4(1-4\alpha-4\gamma)}\mathbf{LWF}_4 + \frac{\rho}{4\gamma}\mathrm{OPT}^N\end{aligned}$$

Setting $\rho = 128$, $\alpha = 1/32$ and $\gamma = 1/32$ completes the proof. □

In Section 2.1 early requests of each non-self-chargeable event $E_{p,x}$ were charged to events that ended on the last half of $[o'_{p,x}, e_{p,x} - 1]$. This was a compromise between using more events vs. finding quality events. In other words, if we use more events ending in $[o'_{p,x}, e_{p,x} - 1]$, the average quality of those events degrades because events ending close to $o'_{p,x}$ do not have flow time comparable to $F^e_{p,x}$. On the other hand, if we use only quality events, we can only charge to a small faction of events ending on $[o'_{p,x}, e_{p,x} - 1]$. To overcome this issue, we will show that all events ending in $[o'_{p,x}, e_{p,x} - 1]$ have comparable flow time with $F^e_{p,x}$ if only a small number of self-chargeable events end on $[o'_{p,x}, e_{p,x} - 1]$. This will then improve our bound on the speed. For the other case where $E_{p,x}$ has many self-chargeable events, $F^e_{p,x}$ will be directly charged to those self-chargeable events having comparable flow time with $F^e_{p,x}$, thus directly to OPT.

We now describe this idea in more details. Non-self-chargeable events in N are partitioned into two disjoint sets N_1 and N_2 depending on they have many self-chargeable events or not. Formally, Non-self-chargeable event $E_{p,x}$ is said to be `Type1` and in N_1 if it has at least $\alpha s(e_{p,x} - o'_{p,x})$ self-chargeable events where $\alpha < 1$ is some constant to be fixed later. The rest of the events in N are in N_2 and said to be `Type2`. We let $\mathbf{LWF}_4^{N_1 e}$ and $\mathbf{LWF}_4^{N_2 e}$ denote the total flow time of early requests of N_1 and N_2, respectively. As mentioned already, the `Type1` events can be charged to the optimal solution because it has many self-chargeable events. For each `Type2` event, we will bound $F^e_{p,x}$ with events which end at $o_{p,x}$. Recall that Lemma 2.3 cannot compare $F^e_{p,x}$ and $F_{r,z}$, where $E_{r,z}$ is an event ending at $o_{p,x}$, i.e. $e_{r,z} = o_{p,x}$. Thus we find a *bridge* events which start from a way before $o'_{p,x}$ and end close to $e_{p,x}$. Since each bridge event $E_{q,y}$ substantially overlap both with $E_{p,x}$ and with $e_{r,z}$, we can compare $F^e_{p,x}$ with $F_{q,y}$ and $F^e_{q,y}$ with $F_{r,z}$, thereby $F^e_{p,x}$ with $F_{r,z}$. We also observe that each $E_{r,z}$ is charged by one event $E_{p,x}$ such that $o_{p,x} = e_{r,z}$, as each non-self-chargeable event has its unique last broad cast time. Thus we are safe from overcharging.

In the following lemma, we directly charge early requests of `Type1` events to OPT. For the goal, we charge early requests of each `Type1` event $E_{p,x}$ to self-chargeable events having flow time comparable to $F^e_{p,x}$ which end on $[o'_{p,x}, e_{p,x} - 1]$. By the definition of `Type1` events, we know that each `Type1` event $E_{p,x}$ has many events that it can be charged to. However, to the overcharging issue, we resort to a global charging scheme again using the modified Hall's theorem. We separate how to find a covering to Lemma A.3, as we will use it again for charging `Type2` events.

Lemma A.2. *If* $\alpha\rho \geq 4$*, then* $\mathbf{LWF}_4^{N_1 e} \leq \frac{4\rho}{\alpha^2}\mathrm{OPT}$.

Proof. Let $G = (X \cup Y, E)$ be a bipartite graph where $u_{p,x} \in X$ iff $E_{p,x} \in N_1$, $v_{q,y} \in Y$ iff $E_{q,y} \in S$, and $u_{p,x}v_{q,y} \in E$ iff $e_{q,y} \in [o'_{p,x} + \lceil \alpha/2(e_{p,x} - o'_{p,x}) \rceil, e_{p,x} - 1]$. Note that if $u_{p,x}v_{q,y} \in E$, $F^e_{p,x} \leq \frac{2}{\alpha}F_{q,y}$ by Lemma 2.3. It can be observed that each vertex $u_{p,x} \in X$ has at least $2\alpha(e_{p,x} - o'_{p,x}) - 4$ ($\geq \alpha(e_{p,x} - o'_{p,x})$ by the given condition) neighbors. This follows from the observations that at least $\alpha s(e_{p,x} - o'_{p,x})$ self-chargeable events end during $[o'_{p,x}, e_{p,x} - 1]$ by definition of `Type1` and at most $s(\alpha/2(e_{p,x} - o'_{p,x})) + s$

events end during $[o'_{p,x}, o'_{p,x} + \lceil \alpha/2(e_{p,x} - o'_{p,x}) \rceil - 1]$. Thus G has a $\frac{2}{\alpha}$-covering by Lemma A.3. Let ℓ be such a covering. We now prove the final step.

$$\begin{aligned}
\mathbf{LWF}_4^{N_1^e} &= \sum_{u_{p,x} \in X} F^e_{p,x} = \sum_{u_{p,x} v_{q,y} \in E} \ell_{u_{p,x}, v_{q,y}} F^e_{p,x} \text{[By Definition 1.1]} \\
&\leq \sum_{u_{p,x} v_{q,y} \in E} \ell_{u_{p,x}, v_{q,y}} \frac{2}{\alpha} F_{q,y} \text{[By Lemma 2.3]} \\
&\leq \frac{2}{\alpha} \frac{2}{\alpha} \sum_{v_{q,y} \in Y} F_{q,y} \text{[Change order of summation and } \ell \text{ is } \tfrac{2}{\alpha}\text{-covering]} \\
&\leq \frac{4}{\alpha^2} \mathbf{LWF}_4^S \text{ [Since } Y \text{ includes all self-chargeable events]} \\
&\leq \frac{4}{\alpha^2} \rho \mathrm{OPT}^S \text{[By Lemma 2.1]}
\end{aligned}$$

□

The following lemma states, when G is a bipartite graph whose parts are a subset of non-self-chargeable events and a subset of all events respectively, the quality of covering in terms of how many neighbors each non-self-chargeable event has. The main difference from what was done for finding a covering in the proof of Lemma 2.4 is that here each non-self-chargeable event is not required to have all events ending in some sub-interval as its neighbors.

Lemma A.3. *Let A denote all events. Let $G = (X \cup Y, E)$ be a bipartite graph where there exists only one vertex $u_{p,x} \in X$ only if $E_{p,x} \in N$, there exists only one vertex $v_{q,y} \in Y$ only if $E_{q,y} \in A$ and $v_{q,y} \in N_G(u_{p,x})$ only if $e_{q,y} \in [o'_{p,x}, e_{p,x} - 1]$. Suppose that $\exists \lambda > 0$ such that $\forall u_{p,x} \in X, |N_G(u_{p,x})| \geq \lambda(e_{p,x} - o'_{p,x})$. Then there exists $\frac{2}{\lambda}$-covering for X.*

Proof. Consider any non-empty set $Z \subseteq X$ and its neighborhood $N(Z)$. We will show that $|N_G(Z)| \geq \lambda/2|G|$. Let $I_{p,x} = [o'_{p,x}, e_{p,x} - 1]$ and $\mathcal{I} = \bigcup_{u_{p,x} \in Z} I_{p,x}$. For simplicity we assume that $\mathcal{I}$ is a contiguous interval. Otherwise, the proof can be simply reduced to each maximal contiguous interval in $\mathcal{I}$. First we show $N_G(Z) \geq \frac{\lambda}{2}|\mathcal{I}|$. We generously give up intervals in $\mathcal{I}$ which are contained in other intervals in $\mathcal{I}$ and order the remaining intervals in increasing order of their starting points. After picking up the first interval, we greedily pick up the next interval which the least overlaps with the previous chosen interval or starts just after the end of the interval. We index the chosen intervals according to their orders, 1,2,3 and so on. Let $\mathcal{I}_{odd}$ and $\mathcal{I}_{even}$ be the odd-indexed and even-indexed intervals, respectively. Note that no intervals in $\mathcal{I}_{odd}$ overlap with each other. Likewise for $\mathcal{I}_{even}$. We have $|\mathcal{I}_{even}| + |\mathcal{I}_{odd}| \geq |\mathcal{I}|$, since $\mathcal{I} = \mathcal{I}_{even} \cup \mathcal{I}_{odd}$. WLOG, suppose $|\mathcal{I}_{odd}| \geq |\mathcal{I}_{even}|$. Let us consider any interval $I_{p',x'}$ in $\mathcal{I}_{odd}$. We know that $E_{p',x'}$ (or $u_{p',x'}$) has at least $\lambda(e_{p,x} - o'_{p,x})$, so by summing over all intervals in $\mathcal{I}_{odd}$, we can find at least $\lambda|\mathcal{I}_{odd}| \geq \lambda/2|\mathcal{I}|$. Thus we have $|N_G(Z)| \geq \lambda/2|\mathcal{I}|$ Also we have $|Z| \leq |\mathcal{I}|$; this is because the optimal solution has 1-speed and since it has to do a separate broadcast for each event in Z. Combining these two inequalities, if follows that $|N_G(Z)| \geq \frac{\lambda}{2}|\mathcal{Z}|$, and therefore G has $\frac{2}{\lambda}$-covering by Lemma 1.2. □

Our attention is now shifted to `Type2` events. As mentioned previously, the main idea is to find bridge events for each $E_{p,x} \in N_2$. Formally, $E_{q,y}$ is said to be a *bridge* event of $E_{p,x}$ if $o'_{q,y} \leq e_{p,x} - (2 - 4\alpha - 4\gamma)(e_{p,x} - o'_{p,x})$ and $e_{q,y} \in [o'_{p,x} + \lceil 1/2(e_{p,x} - o'_{p,x}) \rceil, e_{p,x} - 1]$, where $0 < \gamma < 1$ is a constant to be decided later. Let $\mathcal{B}(E_{p,x})$ be the set of bridge events of $E_{p,x}$. Recall that we want to compare $E_{p,x}$ with $E_{r,z}$ such that $e_{r,z} = o_{p,x}$. Intuitively, a bridge event $E_{q,y}$ bridges two events $E_{p,x}$ and $E_{r,z}$ by stretching over both events. The following lemma says that every `Type2` event has many bridge events.

Lemma A.4. *If $4\gamma\rho \geq 1$, then for any $E_{p,x} \in N_2$, $|\mathcal{B}(E_{p,x})| \geq 4\gamma(e_{p,x} - o'_{p,x}) \geq 1$.*

Proof. Let $E_{p,x} \in N_2$. Let $I = [o'_{p,x}, e_{p,x}-1]$ and $I' = [o'_{p,x}+\lceil 1/2(e_{p,x}-o'_{p,x})\rceil, e_{p,x}-1]$. Our argument is simple; because there are many non-self-chargeable events ending in I', the last optimal broadcast times of many of those events cannot be contained in I', thus many events start a way earlier than $o'_{p,x}$. For the formal proof, we first show that (1) there are at least $(2-4\alpha)(e_{p,x}-o'_{p,x})-2$ non-self-chargeable events that end during I'. This is because there are at least $s\lfloor 1/2(e_{p,x}-o'_{p,x})\rfloor \geq 2(e_{p,x}-o'_{p,x})-2$ events which end during I' and `Type2` event $E_{p,x}$ has at most $\alpha s(e_{p,x}-o'_{p,x})$ self-chargeable events which end during I by definition. Note that for any non-self-chargeable event $E_{q,y}$ which ends on I', OPT must broadcast page q before $e_{p,x}$, more precisely $o_{q,z} < e_{q,z} < e_{p,x}$, that is $o_{q,z} \leq e_{p,x}-2$. Let $t_b = e_{p,x} - (2-4\alpha-4\gamma)(e_{p,x}-o'_{p,x})$. Finally, (2) there are at most $(2-4\alpha-4\gamma)(e_{p,x}-o'_{p,x})-2$ time slots when OPT can broadcast pages during $[\lceil t_b\rceil, e_{p,x}-2]$. From (1) and (2), we can deduce that $|\mathcal{B}(E_{p,x})| \geq 4\gamma(e_{p,x}-o'_{p,x}) \geq 4\gamma\rho \geq 1$. □

In the next lemma, we show each bridge event $E_{q,y} \in \mathcal{B}(E_{p,x})$ provides a good comparison between $F^e_{p,x}$ and the flow time of any event $F_{r,z}$ which end at $o_{p,x}$.

Lemma A.5. *Suppose that $4\gamma\rho \geq 1$. Let $E_{p,x} \in N_2$, $E_{q,y} \in \mathcal{B}(E_{p,x})$ and $E_{r,z}$ be an event s.t. $e_{r,z} = o_{p,x}$. Then, $F^e_{p,x} \leq \frac{3-8\alpha-8\gamma}{1-4\alpha-4\gamma}F_{r,z} + 2\rho F^*_{q,y}$.*

Proof. We start from an easy case that $e_{r,z} \geq e_{q,y}$. We have $\frac{1}{2}F^e_{p,x} \leq \frac{e_{r,z}-o'_{p,x}}{e_{p,x}-o'_{p,x}}F^e_{p,x} \leq F_{r,z}$. The first inequality comes from that $e_{q,y} \geq o'_{p,x} + \lceil \frac{1}{2}(e_{p,x}-o'_{p,x})\rceil$ and the second by Lemma 2.3. Thus it holds that $F^e_{p,x} \leq 2F_{r,z}$, which clearly satisfies the lemma.

Now let us consider the other case that $e_{r,z} < e_{q,y}$. By comparing $E_{p,x}$ and $E_{q,y}$, using Lemma 2.3, we have (1) $\frac{1}{2}F^e_{p,x} \leq \frac{e_{q,y}-o'_{p,x}}{e_{p,x}-o'_{p,x}}F^e_{p,x} \leq F_{q,y}$. The first inequality holds because $e_{q,y} \geq o'_{p,x} + \lceil \frac{1}{2}(e_{p,x}-o'_{p,x})\rceil$ and the second by Lemma 2.3. Next we compare $E_{q,y}$ and $E_{r,z}$. It follows that (2) $\frac{2(1-4\alpha-4\gamma)}{3-8\alpha-8\gamma}F^e_{q,y} \leq \frac{o'_{p,x}-o'_{q,y}}{e_{q,y}-o'_{q,y}}F^e_{q,y} \leq \frac{e_{r,z}-o'_{q,y}}{e_{q,y}-o'_{q,y}}F^e_{q,y} \leq F_{r,z}$. The first inequality can be shown by easy calculation using the fact that $o'_{q,y} \leq e_{p,x} - (2-4\alpha-4\gamma)(e_{p,x}-o'_{p,x})$ and $e_{q,y} \geq o'_{p,x} + \lceil 1/2(e_{p,x}-o'_{p,x})\rceil$. The second follows from that $o'_{p,x} \leq o_{p,x} = e_{r,z}$. Combining (1) and (2), we get $F^e_{p,x} \leq 2F_{q,y} = 2(F^e_{q,y} + F^l_{q,y}) \leq \frac{3-8\alpha-8\gamma}{1-4\alpha-4\gamma}F_{r,z} + 2\rho F^*_{q,y}$. □

Remark A.6. *Lemma A.5 holds for any event $E_{r,z}$ such that $e_{r,z} \in [o'_{p,x}, e_{p,x}-1]$. But we only need to consider the case where $e_{r,z} = o_{p,x}$ for our charging scheme.*

By taking the average of the inequalities in Lemma A.5 over the $s=4$ events ending at $o_{p,x}$, we have the following corollary.

Corollary A.7. *Suppose that $4\gamma\rho \geq 1$. Let $E_{p,x} \in N_2$ and $E_{q,y} \in \mathcal{B}(E_{p,x})$.*
Then, $F^e_{p,x} \leq \frac{3-8\alpha-8\gamma}{4(1-4\alpha-4\gamma)}(\sum_{E_{r,z}|e_{r,z}=o_{p,x}} F_{r,z}) + \frac{\rho}{2}F^*_{q,y}$

Note that in Lemma A.5, $F^e_{p,x}$ is bounded not only with $F_{r,z}$ but also with $F^*_{q,y}$, which contributes to OPT. If many events use $E_{q,y}$ as their bridges, $E_{q,y}$ can be overcharged. To avoid this, we found many bridge candidates for each `Type2` event in Lemma A.4. Using the modified Hall's theorem, we will bound the number of events which use the same bridge event.

Now we are ready to bound early requests of `Type2` events, i.e. $\mathbf{LWF}^{N^e_2}$. Recall that each `Type2` event $E_{p,x}$ is charged to the $s=4$ events which are finished at $o_{p,x}$. Note that $E_{r,z}$ is used only by $E_{p,x}$ since $E_{p,x}$ is the only event such that $o_{p,x} = e_{r,z}$. Thus $E_{r,z}$ is not overcharged.

Lemma A.8. *If $4\gamma\rho \geq 1$, $\mathbf{LWF}^{N^e_2}_4 \leq \frac{3-8\alpha-8\gamma}{4(1-4\alpha-4\gamma)}\mathbf{LWF}_4 + \frac{\rho}{4\gamma}\mathrm{OPT}^N$.*

Proof. Let $G = (X \cup Y, E)$ be a bipartite graph where $u_{p,x} \in X$ iff $E_{p,x} \in N_2$, $v_{q,y} \in Y$ iff $E_{q,y} \in N$ and $u_{p,x}v_{q,y} \in E$ iff $E_{q,y} \in \mathcal{B}(E_{p,x})$. By Lemma A.4, $u_{p,x} \in X$ has at least $4\gamma(e_{p,x} - o'_{p,x})$ neighbors, hence by Lemma A.3, G has $\frac{1}{2\gamma}$-covering. Let ℓ' be such a covering. Now we are ready to prove the final step. For simplicity, let $k = \frac{3-8\alpha-8\gamma}{4(1-4\alpha-4\gamma)}$.

$$\begin{aligned}
\mathbf{LWF}_4^{N_2^e} &= \sum_{u_{p,x} \in X} F^e_{p,x} = \sum_{u_{p,x}v_{q,y} \in E} \ell'_{u_{p,x},v_{q,y}} F^e_{p,x} \text{[By Definition 1.1]} \\
&\leq \sum_{u_{p,x}v_{q,y} \in E} \ell'_{u_{p,x},v_{q,y}} (k \sum_{E_{r,z} | e_{r,z} = o_{p,x}} F_{r,z} + \frac{\rho}{2} F^*_{q,y}) \text{[By Corollary A.7]} \\
&= k \sum_{u_{p,x} \in X} \sum_{E_{r,z} | e_{r,z} = o_{p,x}} F_{r,z} + \frac{\rho}{2} \sum_{v_{q,y} \in Y} F^*_{q,y} \sum_{u_{p,x} \in X} \ell'_{u_{p,x},v_{q,y}} \\
&\leq k\mathbf{LWF}_4 + \frac{\rho}{2} \sum_{v_{q,y} \in Y} F^*_{q,y} \frac{1}{2\gamma} \text{ [By (*) and } \ell' \text{ being a } \tfrac{1}{2\gamma}\text{-covering]} \\
&\leq k\mathbf{LWF}_4 + \frac{\rho}{4\gamma} \mathrm{OPT}^N \text{ [Since } Y \text{ include all non-self-chargeable events]}
\end{aligned}$$

It holds that (*) $\sum_{u_{p,x} \in X} \sum_{E_{r,z} | e_{r,z} = o_{p,x}} F_{r,z} \leq \mathbf{LWF}_4$, because for each non-self-chargeable $E_{r,z}$ there is only one event $E_{p,x}$ such that $e_{r,z} = o_{p,x}$. □

Remark A.9. *If non-integer speeds are allowed then the analysis in this subsection can be extended to show that* **LWF** *is* $3.4 + \epsilon$*-speed* $O(1 + 1/\epsilon^3)$*-competitive.*

B Omitted Proofs

B.1 Proof of Lemma 1.3

Proof. Let I be the union of all intervals in X. I' is similarly defined for X'. We prove the lemma when I' is a contiguous interval; otherwise we can simply sum over all maximal intervals in I'. WLOG, we can set $I = [s_1, t']$ and $I' = [s', t']$. This is because I must start with one interval in X, say $[s_1, t_1]$ and both I and I' must have the same ending point t' by construction. Since $s \leq s'_1$, it is enough to show that $\frac{t-s'_1+1}{t-s_1+1} \geq \lambda$ and it follows from the given condition that $|[s'_1, t_1]| \geq \lambda |[s_1, t_1]|$, (i.e. $t_1 - s'_1 + 1 \geq \lambda(t_1 - s_1 + 1)$) and $t \geq t_1$. □