An Online Scalable Algorithm for Average Flow Time in Broadcast Scheduling [*]

Sungjin Im[†] Benjamin Moseley[‡]

Abstract

In this paper the online pull-based broadcast model is considered. In this model, there are n pages of data stored at a server and requests arrive for pages online. When the server *broadcasts* page p, all outstanding requests for the same page p are simultaneously satisfied. We consider the problem of minimizing average (total) flow time online where all pages are unit-sized. For this problem, there has been a decade-long search for an online algorithm which is scalable, i.e. $(1+\epsilon)$-speed $O(1)$-competitive for any fixed $\epsilon > 0$. In this paper, we give the *first* analysis of an online scalable algorithm.

Keywords: Broadcast, Average Flow Time, Online, Scalable

1 Introduction

We consider the pull-based broadcast scheduling model. In this model, there are n pages available at a server and requests arrive for pages over time. The server must satisfy all requests. When the server *broadcasts* a page p, all outstanding requests for the same page p are satisfied simultaneously. This is the main difference from standard scheduling settings where the server must process each request separately. The broadcast model is motivated by several applications such as multicast systems and wireless and LAN networks [32, 1, 2, 26]. Besides the practical interest in the model, broadcast scheduling has seen growing interest in algorithmic scheduling literature both in the offline and online settings [6, 2, 1, 7, 26]. Work has also been done in stochastic and queueing theory literature on related models [17, 16, 30, 31].

In this paper we concentrate on the online model with the goal of minimizing the total (or equivalently average) flow time [1]. This is one of the most popular quality of service metrics. The ith request for page p will be denoted $J_{p,i}$. The request $J_{p,i}$ arrives at time $a_{p,i}$ and, in the online model, this is when the server is first aware of the request. Time is slotted and a *single* page can be broadcasted in a time-slot. Notice that this implies that all pages can be broadcast in the same amount of time. Unit (or similar) processing time pages is popular in practice. This model also captures the algorithmic difficulty of the problem and this is almost exclusively the model addressed in previous literature. The total flow time of a given schedule can be written as $\sum_p \sum_i (f_{p,i} - a_{p,i})$, where $f_{p,i}$ is the time when $J_{p,i}$ is satisfied.

It was shown that without resource augmentation any *online* deterministic algorithm is $\Omega(n)$-competitive [28]. Further, any randomized online algorithm has a lower bound of $\Omega(\sqrt{n})$ [3]. Due to these strong lower-bounds we focus on the resource augmentation model [27] where an algorithm A is given $s \geq 1$ speed and is

[*]A preliminary version of this paper is to appear at the 21st ACM-SIAM Symposium on Discrete Algorithms (SODA 2010), Austin, TX.

[†]Department of Computer Science, University of Illinois, 201 N. Goodwin Ave., Urbana, IL 61801. im3@illinois.edu.
Partially supported by NSF grants CCF-0728782, CNS-0721899, and Samsung Fellowship.

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

[1] Flow time is often referred to response time or wait time

compared to an optimal offline solution that has 1 speed. We will let A_s be the flowtime accumulated for an algorithm A when given s speed; sometimes we will allow A_s to denote the algorithm itself with s speed if there is no confusion in the context. In the resource augmentation model, we say that A is s-speed r-competitive if $A_s \leq r\text{OPT}$ for all request sequences, where OPT is an optimal offline solution given 1 speed.

The algorithmic difficulty in broadcast scheduling is that two algorithms may have to do different number of broadcasts to satisfy the same set of requests. For example, consider the algorithm most-requests-first (MRF) which broadcasts the page that has the largest number of unsatisfied requests. This algorithm may seem like the most natural candidate for the problem. However, it was shown that MRF is not $O(1)$-competitive even when given any fixed extra speed [28]. A simple example shows that MRF may repeatedly broadcast the same page, while ignoring requests which eventually accumulate a large amount of flowtime. The optimal solution can take advantage of the broadcast setting and satisfy the requests MRF was busy working on by a single broadcast. This leaves the optimal solution free to work on other requests that are unsatisfied under MRF's schedule.

Further adding to the difficulty of algorithmic development, previous work has shown that the existence of a $O(1)$-speed $O(1)$-competitive online algorithm cannot be proved using standard techniques. An algorithm A is said to be *locally* competitive if the number of requests in A's queue is comparable to the number of requests in the adversary's queue at each time. In [28] it was shown that no online algorithm can be locally competitive with an adversary. Local competitiveness has been one of the most popular methods of analysis in standard scheduling [27, 8, 28].

Even though broadcast scheduling has been studied extensively over the last decade, the complexity of the problem is yet to be well understood. In the *offline* setting, minimizing average flowtime was first studied using non-trivial linear programming techniques coupled with resource augmentation [28, 23, 24, 25]. It was not until later that a complex reduction showed that this problem was in fact NP-Hard [22]. Recently, a simpler proof of this fact was found [10]. Following this line of work, a $(1+\epsilon)$-speed $O(1)$-approximation algorithm was eventually given in [3]. Here, resource augmentation was used even though it is still open if the problem admits an $O(1)$-approximation. The problem is substantially more difficult without resource augmentation. No non-trivial analysis was shown without resource augmentation until Bansal et al. gave a $O(\sqrt{n})$-approximation in [3]. More recently, a $O(\log^2 n / \log\log(n))$-approximation was shown in [4]. We note that this result relies on highly non-trivial algorithmic techniques.

In the online setting, the strong lowerbound without resource augmentation has led previous work to focus on finding $O(1)$-speed $O(1)$-competitive algorithms. The ultimate goal of this line of work is to find a $(1+\epsilon)$-speed $O(1)$-competitive algorithms (for any fixed $\epsilon > 0$). That is, to show an algorithm that achieves $O(1)$-competitiveness with the minimum amount of extra resources. For this reason, an online algorithm, which is $(1+\epsilon)$-speed $O(1)$-competitive, is said to be *scalable*. For problems which have strong lowerbounds without resource augmentation, finding a scalable algorithm is the best positive result that can be achieved using worst-case analysis.

Previously, there have been two main approaches used to avoid a local argument, however both lines of work do not seem to suggest a way to obtain a scalable algorithm. The first was given by Edmonds and Pruhs in [19]. They showed a non-trivial reduction from the problem of minimizing average flowtime in broadcast scheduling to a non-clairvoyant scheduling problem. Their reduction takes an algorithm A that is s-speed c-competitive for the non-clairvoyant scheduling problem and creates an algorithm B that is $2s$-speed c-competitive for the broadcast scheduling problem. Using this reduction, they were able to show an algorithm which is $(4+\epsilon)$-speed $O(1)$-competitive for minimizing the average flowtime in broadcast scheduling [18, 19]. More recently, the same authors used this reduction to show another algorithm is $(2+\epsilon)$-speed $O(1)$-competitive [21]. Both of these algorithms can be extended to the case where pages have varying sizes. Notice that a factor of 2 in the speed is lost in the reduction and, therefore, the reduction cannot be used to show a scalable algorithm.

The algorithm longest-wait-first (LWF) was first introduced in [28]. LWF uses a natural scheduling policy which always schedules the page with the highest flowtime. Edmonds and Pruhs showed that LWF is 6-speed $O(1)$-competitive using a direct analysis that avoided the use of the reduction [20]. In this work, new novel techniques were introduced to avoid a local argument. The techniques presented in the paper were quite complex. In joint work with Chekuri, we were able to simplify these techniques to make the key ideas more transparent. Using this, we were able to show LWF is $(4+\epsilon)$-speed $O(1)$-competitive [12]. In this paper, a generalization of these techniques will be presented which was made possible by our previous simplification. However, LWF was shown to be $n^{\Omega(1)}$-competitive when given speed less than 1.618 [20].

Results: For the problem of minimizing total flowtime in broadcast scheduling, we give the first online scalable algorithm **LA-W**. We prove that **LA-W** is $(1+\epsilon)$-speed $O(1/\epsilon^{11})$-competitive for any $0 < \epsilon \leq 1$, giving a positive answer to a central open problem in the area. Our algorithm **LA-W** is similar to LWF in that it prioritizes pages with large flowtime, however **LA-W** also gives preference to requests which have arrived recently. Favoring requests which have arrived recently has been shown to be useful in [21]. The algorithm **LA-W** focuses on *pages* which have requests that arrived recently. This is fundamentally different from the algorithm given in [21], which focuses on requests that arrived recently without considering the page they are requesting. Unfortunately, in the broadcast setting it is difficult to categorize which pages have requests that arrived recently, since the arrival of requests can be scattered over time. To counter this, we develop a novel and robust way to compare the arrival time of requests between two different pages.

Overview of the Algorithm: Let $F_p(t)$ be the total waiting time of unsatisfied requests for page p at time t and let $F_{max}(t) = \max_p F_p(t)$. LWF schedules a page p such that $F_p(t) = F_{max}(t)$. Notice that LWF schedules the page without considering the number of outstanding requests for the page. Due to this, LWF may broadcast a page with a relatively small number of unsatisfied requests which have been waiting to be scheduled for a long period of time. However, a page with a small number of requests does not accumulate flowtime quickly. In some cases, pages which have a large number of unsatisfied requests should be broadcasted since these requests will rapidly accumulate flowtime. Using this insight, [20] was able to show a lower bound of 1.618 on the speed LWF required to be $O(1)$-competitive.

Our algorithm **LA-W** keeps the main spirit of LWF by always broadcasting pages with flowtime comparable to $F_{max}(t)$ at each time t. However, amongst the pages with flowtime comparable to $F_{max}(t)$, **LA-W** prioritizes pages with requests which have arrived recently. By prioritizing recent requests, we avoid the potentially negative behavior of LWF. This is because a page with requests that arrived recently must have a large number of outstanding requests to have flowtime similar to F_{max}. As mentioned, we develop a new way to compare the arrive time of requests for two different pages. Using this technique, we will be able to break up time into intervals and show when requests arrive on these intervals. Thus allowing us to determine how **LA-W** and the optimal solution must behave on these intervals.

The algorithm **LA-W** broadcasts pages with unsatisfied requests that arrived recently to potentially find pages which have a large number of outstanding requests. The reader may wonder why we chose pages in this manner when we could simply broadcast the page with many outstanding requests. In fact, we have considered an algorithm which schedules the page with the largest number of outstanding requests amongst the pages with flowtime comparable to $F_{max}(t)$. For this algorithm, we have established that it is scalable for the problem of minimizing the *maximum weighted* flowtime in broadcast scheduling [13]. Further, we have preliminary evidence that this algorithm is $O(1)$ competitive for average flowtime when given more than 2 speed. We however were unable to determine its performance for average flowtime when given less than 2 speed.

Other Related Work: Charikar and Khuller considered a generalization of average flowtime where the goal is to minimize the average flowtime for a fraction of the requests [11]. Besides work on minimizing the total flowtime, other objective functions have been considered in the broadcast model. In [7, 10, 13] a 2-competitive algorithm was given for the problem of minimizing the maximum response time. When each request has a deadline, constant competitive algorithms were given by [29, 9, 33, 15] with the the objective of maximizing the number of requests satisfied by their deadlines. For the problem of minimizing maximum weighted flowtime, a $(1+\epsilon)$-speed $O(1)$-competitive algorithm was been given by [13]. The delay factor is a metric closely related to weighted flowtime. For the problem of minimizing the maximum delay factor a $(2+\epsilon)$-speed $O(1)$-

comeptitive algorithm was given by [14] and this was improved to a $(1+\epsilon)$-speed $O(1)$-competitive algorihtm in [13]. For the problem of minimizing the L_k-norms of flowtime and the delay factor [12] gave $O(k)$-speed $O(k)$-competitive algorithms.

2 Time Model and Algorithm

We assume that all requests have unit processing time and without loss of generality this is 1. In our time model we assume that requests arrive only at non-negative integer times. Any scheduling algorithm A with speed $s \geq 1$ schedules a page every $1/s$ time-steps starting from time 0. When A broadcasts a page p at time t, all alive (unsatisfied) requests for page p which arrived *strictly* earlier than t are *immediately* satisfied by the broadcast. If $J_{p,i}$ is a request satisfied by a broadcast, it has flow time $t - a_{p,i}$. Note that under the schedule produced by the optimal solution with 1-speed, every request has flow time at least 1. On the other hand, A with speed $s > 1$ may finish some requests within a delay less than one. Though it would seem fair to force A to schedule requests after at least one time step, we do not assume this because our analysis will hold in either case and this assumption improves the readability of the analysis.

Before introducing our algorithm, we state notation that will be used throughout the paper. For any time interval starting at b and ending at e, we let $|I| = e - b$. For a set of requests R, we will let $F(R)$ be the flowtime accumulated for the requests in R by our algorithm. For a page p we will let $F_p(t)$ be the total flowtime accumulated at time t for unsatisfied requests for page p. We will let $F(R,t)$ be the total flowtime accumulated by our algorithm for requests in the set R at time t. Note that if some requests in R arrive after time t then these requests do not contribute to the value of $F(R,t)$. We let $F^*(R)$ denote the total flow time OPT accumulates for a set of requests R.

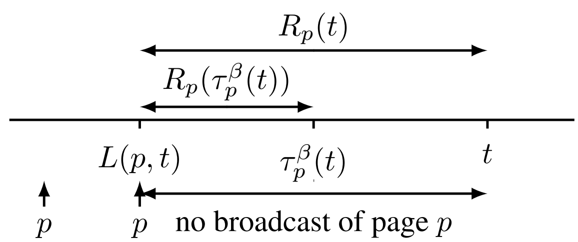

Figure 1: $R_p(t)$ denotes the alive requests of page p at time t, i.e. the requests of page p which arrived during $[L(p,t), t]$. Likewise, $R_p(\tau_p^\beta(t))$ denotes the requests which arrived during $[L(p,t), \tau_p^\beta(t)]$.

We now introduce our algorithm, denoted by **LA-W** for Latest Arrival time with Waiting. We assume that **LA-W** is given $s = 1+\epsilon$ speed where $0 < \epsilon \leq 1$ is a fixed constant. Our algorithm is parameterized by constants $c > 1$ and $\beta < 1$ depending on ϵ, which will be defined later. For each page p and time t, let $R_p(t)$ denote the set of alive requests for page p at time t. Let $L(p,t)$ be the last time before time t that our algorithm broadcasted page p. If there is no such time then $L(p,t) = 0$. Note that $R_p(t)$ is equivalent to the set of the requests for page p which arrived during $[L(p,t), t]$. For a page p and time t let $\tau_p^\beta(t) = \texttt{argmin}_{L(p,t) \leq t' \leq t}(F(R_p(t'), t) \geq (1-\beta)F_p(t))$. In other words, $\tau_p^\beta(t)$ denotes the *earliest* time t' no later than time t and no earlier than time $L(p,t)$ such that the requests in $R_p(t')$ have total flowtime at least $(1-\beta)F_p(t)$ at time t. By this definition, if $R_{[L(p,t), \tau_p^\beta(t)]}$ is the set of requests for page p that arrive on the interval $[L(p,t), \tau_p^\beta(t)]$ and $R_{[L(p,t), \tau_p^\beta(t))}$ is the set of requests for page p that arrive on the interval $[L(p,t), \tau_p^\beta(t))$ then $F(R_{[L(p,t), \tau_p^\beta(t)]}, t) \geq (1-\beta)F_p(t)$ and $F(R_{[L(p,t), \tau_p^\beta(t))}, t) < (1-\beta)F_p(t)$. See Figure 1.

Algorithm: **LA-W**

- Let t be a time where our algorithm is not broadcasting a page.
- Let $F_{max}(t) = \max_p F_p(t)$.
- Broadcast one page according to Rule 2 every $\lfloor \frac{10}{\epsilon} \rfloor$ broadcasts, and broadcast one page according to Rule 1 otherwise.
  - `Rule 1`: broadcast the page $p = \texttt{argmax}_{p' \in Q(t)} \tau_{p'}^{\beta}(t)$, where $Q(t) = \{q \mid F_q(t) \geq \frac{1}{c} F_{max}(t)\}$ breaking ties arbitrarily.
  - `Rule 2`: broadcast a page p where $F_p(t) = F_{max}(t)$ breaking ties arbitrarily.

Our algorithm **LA-W** broadcasts pages mainly according to `Rule 1` while occasionally broadcasting a page according to `Rule 2`. The second rule uses LWF's scheduling policy which broadcasts a page with the highest flowtime. The first rule chooses a page p with the latest time $\tau_p^{\beta}(t)$ among the pages with flowtime close to $F_{max}(t)$. The value of $\tau_p^{\beta}(t)$ can be interpreted as the latest arrival time of any unsatisfied request for page p after discounting requests that arrived recently that have small flowtime. Since the arrival of requests for the same page p can be scattered over time, we will use $\tau_p^{\beta}(t)$ as the representative arrival time of those requests. Notice that if all requests for page p arrive at time t' then $\tau_p^{\beta}(t) = t'$ for any $0 < \beta \leq 1$. We remark that we do not know if `Rule 2` is needed for **LA-W** to be $(1+\epsilon)$-speed $O(1)$-competitive. `Rule 2` will play a crucial role in our analysis, but we do not have a proof that `Rule 1` alone performs poorly.

3 Analysis

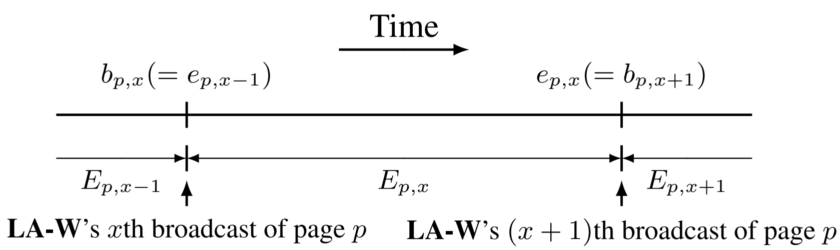

Figure 2: Events for page p.

Let σ be a fixed sequence of requests. OPT denotes a fixed offline optimal solution. We assume **LA-W**$_{1+\epsilon}$ is always busy scheduling pages for the sequence σ. Otherwise, our arguments can be applied to each maximal time interval where **LA-W**$_{1+\epsilon}$ is busy. Following the lead of [20, 12], time is partitioned into *events* for each page p. Events for page p are defined by **LA-W**$_{1+\epsilon}$'s broadcasts of page p. Each time **LA-W**$_{1+\epsilon}$ broadcasts a page, an event begins and an event ends. An event $E_{p,x} = \langle b_{p,x}, e_{p,x} \rangle$ begins at time $b_{p,x}$ and ends at time $e_{p,x}$. Here, **LA-W**$_{1+\epsilon}$ broadcasts page p at time $b_{p,x}$ and at time $e_{p,x}$. These are the xth and $(x+1)$st broadcasts of page p by **LA-W**$_{1+\epsilon}$. The $(x+1)$st broadcast of page p starts a new event $E_{p,x+1}$ and $e_{p,x} = b_{p,x+1}$. On the time interval $(b_{p,x}, e_{p,x})$ **LA-W**$_{1+\epsilon}$ does not broadcast page p. The optimal solution can broadcast page p zero or more times during an event $E_{p,x}$. See Figure 2.

For an event $E_{p,x}$, let $R_{p,x}$ denote the set of requests satisfied by the $(x+1)$st broadcast of page p. Notice that all requests in $R_{p,x}$ arrive during $E_{p,x}$, formally during $[b_{p,x}, e_{p,x})$. Let $F_{p,x} = F(R_{p,x})$ be the total flowtime **LA-W**$_{1+\epsilon}$ accumulates for requests in $R_{p,x}$. Likewise let $F^*_{p,x} = F^*(R_{p,x})$ be the flowtime OPT accumulates for requests in $R_{p,x}$. We refer to $F_{p,x}$ as the flowtime of $E_{p,x}$. Similarly to requests, for a set $\mathcal{E}$ of events we let $F(\mathcal{E}) = \sum_{E_{p,x} \in \mathcal{E}} F_{p,x}$.

For any event $E_{p,x}$, the next lemma will be used to bound the flowtime accumulated for page p at different times during $E_{p,x}$. This will help us to compare the flowtime of $E_{p,x}$ to the flowtime of events ending during $E_{p,x}$. The proof of this lemma follows easily by definition of flowtime.

Lemma 3.1 *For any event $E_{p,x}$, let $R' \subseteq R_{p,x}$. Let t be such that $b_{p,x} \leq t < e_{p,x}$. Suppose that all requests in R' arrive no later than time t. Then for any $0 \leq \eta < 1$, $F(R', t + \eta(e_{p,x} - t)) \geq \eta F(R')$. Further, if $F(R') \geq \upsilon F_{p,x}$, then $F(R', t + \eta(e_{p,x} - t)) \geq \eta\upsilon F_{p,x}$.*

Proof: $F(R', t + \eta(e_{p,x} - t)) = \sum_{J_{p,i} \in R'}(t + \eta(e_{p,x} - t) - a_{p,i}) = \sum_{J_{p,i} \in R'}((1-\eta)t + \eta e_{p,x} - a_{p,i}) \geq \sum_{J_{p,i} \in R'}((1-\eta)a_{p,i} + \eta e_{p,x} - a_{p,i}) = \eta \sum_{J_{p,i} \in R'}(e_{p,x} - a_{p,i}) = \eta F(R')$. The inequality holds, since any request $J_{p,i}$ in R' arrives no later than time t. □

Our goal is to show that $\sum_p \sum_x F_{p,x} \leq O(1)$OPT. We start by partitioning events into two groups. An event $E_{p,x}$ is called *self-chargeable* if $F_{p,x} \leq \gamma F^*_{p,x}$ where $\gamma \geq 1$ is a constant to be fixed later. Let $\mathcal{S}$ be the set of all self-chargeable events. The other events are called *non-self-chargeable* and are in the set $\mathcal{N}$. By definition of self-chargeable events, we can easily bound $F(\mathcal{S})$ by OPT.

Lemma 3.2 $F(\mathcal{S}) \leq \gamma$OPT.

Proof: $F(\mathcal{S}) = \sum_{E_{p,x} \in \mathcal{S}} F_{p,x} \leq \sum_{E_{p,x} \in \mathcal{S}} \gamma F^*_{p,x} \leq \gamma$OPT. □

We now concentrate on non-self-chargeable events. Notice that for a non-self-chargeable event $E_{p,x}$, the optimal solution must broadcast page p during $E_{p,x}$, formally during $(b_{p,x}, e_{p,x})$. Otherwise, $F^*_{p,x} \geq F_{p,x}$ and the event is self-chargeable. We further partition non-self-chargeable events into two classes. Consider a non-self-chargeable event $E_{p,x}$. Let α and k be constants to be fixed later such that $\alpha < 1$, $k > 1$ and $\beta k < 1$. $E_{p,x}$ is in the set $\mathcal{N}_1$ if for some $\beta \leq \rho \leq \beta k$ it is the case that at least $\lceil \alpha s(e_{p,x} - \tau_p^\rho(e_{p,x})) \rceil$ self-chargeable events end on the interval $[\tau_p^\rho(e_{p,x}), e_{p,x})$. Notice that the time $\tau_p^\rho(e_{p,x})$ exists because $\rho < 1$. A non-self-chargeable event not in $\mathcal{N}_1$ is in $\mathcal{N}_2$.

The sets $\mathcal{N}_1$ and $\mathcal{N}_2$ are similar to how [12] partitions non-self-chargeable events. Bounding the flowtime of events in $\mathcal{N}_1$ by OPT is not too difficult and follows easily by combining the analysis given in [12] and the definition of τ. We do this by bounding $F(\mathcal{N}_1)$ by the flowtime of the self chargeable events ending during the events in $\mathcal{N}_1$. Knowing that $F(\mathcal{S}) \leq \gamma$OPT we will be able to bound $F(\mathcal{N}_1)$ by OPT. The proof follows easily from [12] and for better readability, we defer the proof of the following lemma to Appendix B.

Lemma 3.3 $F(\mathcal{N}_1) \leq O(\frac{1}{\epsilon^{11}})$OPT.

The most interesting events are those which are in $\mathcal{N}_2$. Since each event $E_{p,x}$ in $\mathcal{N}_2$ has a relatively small number of self-chargeable events ending during $E_{p,x}$, we cannot directly bound $F(\mathcal{N}_2)$ by OPT. Instead, we will show that the total flowtime of events in $\mathcal{N}_2$ accounts for only a fraction of $\mathbf{LA\text{-}W}_{1+\epsilon}$'s total flowtime, i.e. $F(\mathcal{N}_2) \leq \delta \mathbf{LA\text{-}W}_{1+\epsilon}$ for some constant $\delta < 1$ which is independent of ϵ. In [13] and [20] speed greater than 3.4 was needed to bound $F(\mathcal{N}_2)$. Our goal is to ensure $\delta < 1$ with only $(1+\epsilon)$ speed. Showing this will complete our analysis as follows. Using this, Lemma 3.2 and Lemma 3.3, we have that $\mathbf{LA\text{-}W}_{1+\epsilon} = F(\mathcal{S}) + F(\mathcal{N}) = F(\mathcal{S}) + F(\mathcal{N}_1) + F(\mathcal{N}_2) \leq \gamma\text{OPT} + O(\frac{1}{\epsilon^{11}})\text{OPT} + \delta\mathbf{LA\text{-}W}_{1+\epsilon}$, which simplifies to $\mathbf{LA\text{-}W}_{1+\epsilon} \leq \frac{\gamma + O(\frac{1}{\epsilon^{11}})}{1-\delta}$OPT. This will imply the following theorem.

Theorem 3.4 *For $0 < \epsilon \leq 1$, the algorithm* **LA-W** *is $(1+\epsilon)$-speed $O(\frac{1}{\epsilon^{11}})$-competitive for minimizing average flow time in broadcast scheduling with unit sized pages.*

Before continuing, we show some properties of events in $\mathcal{N}_2$. Say that we set $\gamma \geq \frac{1}{\beta}$. Then it is not hard to show that OPT must broadcast page p during $I = [\tau_p^\beta(e_{p,x}), e_{p,x})$ for any non-self-chargeable event $E_{p,x}$.

Indeed, the requests for page p that arrive during the interval I have total flowtime at least $\beta F_{p,x}$ in $\textbf{LA-W}_{1+\epsilon}$'s schedule by definition of τ^{β}. If OPT does not broadcast page p during I this implies that these requests also have total flowtime $\beta F_{p,x}$ in OPT's schedule. However, then $F^*_{p,x} \geq \beta F_{p,x} \geq \frac{1}{\gamma} F_{p,x}$, contradicting the fact that $E_{p,x}$ is non-sef-chargeable.

Lemma 3.5 *Suppose that $\gamma \geq \frac{1}{\beta}$. Then, for any non-self-chargeable event $E_{p,x}$, the optimal solution must broadcast page p during the interval $[\tau_p^{\beta}(e_{p,x}), e_{p,x})$.*

Proof: For the sake of contradiction assume the lemma is false. The event $E_{p,x}$ is non-self-chargeable therefore the optimal solution must broadcast page p at some time during $(b_{p,x}, \tau_p^{\beta}(e_{p,x}))$. Let t be the latest broadcasting time of page p by the optimal solution during $(b_{p,x}, \tau_p^{\beta}(e_{p,x}))$. Let $S_{[b_{p,x},t]}$ and $S_{(t,e_{p,x})}$ be the set of requests for page p which arrive during $[b_{p,x}, t]$ and $(t, e_{p,x})$, respectively. We know that $F(S_{[b_{p,x},t]}) < (1-\beta)F_{p,x}$ by definition of $\tau_p^{\beta}(e_{p,x})$ and $t < \tau_p^{\beta}(e_{p,x})$. Thus $F(S_{(t,e_{p,x})}) = F(R_{p,x} \setminus S_{[b_{p,x},t]}) > \beta F_{p,x}$. Since the optimal solution does not broadcast page p during $(t, e_{p,x})$, it follows that $F^*_{p,x} \geq F^*(S_{(t,e_{p,x})}) > \beta F_{p,x} \geq \frac{1}{\gamma} F_{p,x}$, which is a contradiction to $E_{p,x}$ being a non-self-chargeable event. □

Now say that we set $\gamma \geq \frac{10000}{\beta\epsilon^2}$. Using similar ideas as in Lemma 3.5, we will be able to show that $|[\tau_p^{\beta}(e_{p,x}), e_{p,x})| \geq \frac{10000}{\epsilon^2}$. This will be used to ensure that the intervals considered in our remaining arguments are sufficiently long.

Lemma 3.6 *Suppose $\gamma \geq \frac{10000}{\epsilon^2\beta}$. Then, for any non-self-chargeable event $E_{p,x}$, $|[\tau_p^{\beta}(e_{p,x}), e_{p,x}]| \geq \frac{10000}{\epsilon^2}$.*

Proof: For the sake of contradiction, assume that there exists a non-self-chargeable event $E_{p,x}$ such that $|[\tau_p^{\beta}(e_{p,x}), e_{p,x}]| < \frac{10000}{\epsilon^2}$. Let S be the set of requests for page p which arrive on the interval $[\tau_p^{\beta}(e_{p,x}), e_{p,x})$. By definition of $\tau_p^{\beta}(e_{p,x})$ it must be the case that $F(S) > \beta F_{p,x}$. We now want to bound the number of requests in S. Since each request in S can accumulate flow time at most $|[\tau_p^{\beta}(e_{p,x}), e_{p,x}]| < \frac{10000}{\epsilon^2}$, we have that $F(S) < |S|\frac{10000}{\epsilon^2}$, thus $\beta F_{p,x} < |S|\frac{10000}{\epsilon^2}$. Hence we have that $|S| > \frac{\epsilon^2}{10000}\beta F_{p,x}$. The optimal solution must accumulate at least $|S|$ flowtime for the requests in S, therefore $F^*_{p,x} \geq |S| > \frac{\epsilon^2}{10000}\beta F_{p,x} \geq \frac{1}{\gamma} F_{p,x}$. This is a contradiction to $E_{p,x}$ being non-self-chargeable. □

We start by giving intuition on why $F(\mathcal{N}_2) \leq \delta \textbf{LA-W}_{1+\epsilon}$. As in [20, 12], we use a global charging scheme built on Hall's theorem. We generalize charging techniques used in [20, 12] in the following lemma. This lemma shows how to charge the flowtime of some events to the total flowtime $\textbf{LA-W}_{1+\epsilon}$ accumulates. The proof is technical and is not to difficult to prove given the previous ideas shown in [20, 12], and we defer it to Appendix A.

Lemma 3.7 *Let $\mathcal{A}$ be a set of events. Let $\mu, \kappa > 0$ be some constants. Let $\lambda \geq 1$ be an integer. For each event $E_{p,x} \in \mathcal{A}$, suppose there exists an interval $I_{p,x}$ and a set of events $\mathcal{B}_{p,x}$ such that*

- *The optimal solution broadcasts page p at least λ times during the interval $I_{p,x}$. Further, $I_{p,x}$ is disjoint with $I_{p,x'}$ for any $E_{p,x'} \in \mathcal{A}$ s.t. $x' \neq x$.*
- *$|\mathcal{B}_{p,x}| \geq \mu |I_{p,x}|$ and $E_{q,y} \in \mathcal{B}_{p,x}$ only if $e_{q,y} \in I_{p,x}$* and *$F_{q,y} \geq \kappa F_{p,x}$.*

Let $\mathcal{B} = \bigcup_{(p,x):E_{p,x}\in\mathcal{A}} \mathcal{B}_{p,x}$ and $d = \min_{E_{p,x}\in\mathcal{A}} |I_{p,x}|$. Then, $F(\mathcal{A}) \leq (\frac{2}{\lambda\kappa\mu})(\frac{d+1}{d})F(\mathcal{B}) \leq (\frac{2}{\lambda\kappa\mu})(\frac{d+1}{d})\textbf{LA-W}_{1+\epsilon}$.

This lemma can be interpreted as follows. For a set of events $\mathcal{A} \subseteq \mathcal{N}_2$, we charge the flowtime of each event $E_{p,x} \in \mathcal{A}$ to some events ending during $I_{p,x}$. In our analysis, $I_{p,x}$ will always be a subinterval of $E_{p,x}$; thus for any fixed page p, $\{I_{p,x} \mid E_{p,x} \in \mathcal{A}\}$ are disjoint. If the following conditions hold for each event $E_{p,x} \in \mathcal{A}$, then $F(\mathcal{A}) \ll$ **LA-W**$_{1+\epsilon}$. (1) There are at least λ broadcasts by OPT of page p during $I_{p,x}$. (2) We can find a sufficiently large fraction of events ending during $I_{p,x}$, denoted by μ, such that each of these events have flowtime at least $\kappa F_{p,x}$. (3) $I_{p,x}$ is sufficiently long for all $E_{p,x} \in \mathcal{A}$. The bound we get on $F(\mathcal{A})$ improves by either finding many broadcasts of page p by OPT during $I_{p,x}$ or by finding sufficiently many events with very large flowtime ending during $I_{p,x}$.

To exploit Lemma 3.7, $\mathcal{N}_2$ is partitioned into three disjoint sets $\mathbb{T}_1$, $\mathbb{T}_2$ and $\mathbb{T}_3$. To discuss the high level interpretation of the sets $\mathbb{T}_1$, $\mathbb{T}_2$ and $\mathbb{T}_3$ we fix an event $E_{p,x} \in \mathcal{N}_2$ and page p and drop the subscript p, x. For the event E we will consider different subintervals of E defined by τ. Let $I^i = [\tau^{\beta(\frac{10}{\epsilon}i+1)}(e), e)$ for $i \in \mathbb{N}$. Notice that I^i is a subinterval of I^{i+1} for all i. We will concentrate on the intervals I^i for different values of i. Concentrating on these intervals will allow us to break up the event E so that we can better understand when the requests for page p arrived during E and how the optimal solution and **LA-W**$_{1+\epsilon}$ behaved during E.

The event E will be in the set $\mathbb{T}_1$ if for some i it is the case that page p is not in the queue Q for a sufficiently large number of broadcasts by **LA-W**$_{1+\epsilon}$ during I^i. By definition of Q, if p is not in $Q(t)$ then there exists another page q such that $F_q(t) > cF_p(t)$. `Rule 2` of **LA-W** broadcasts a page with the highest flowtime every $\lfloor \frac{10}{\epsilon} \rfloor$ broadcasts. Using this, we will be able to find sufficiently many events ending during E with flowtime much larger than the flowtime of event E. Then Lemma 3.7 can be used to show that $F(\mathbb{T}_1) \ll$ **LA-W**$_{1+\epsilon}$. Intuitively, the requests in $\mathbb{T}_1$ cannot account for most of **LA-W**$_{1+\epsilon}$'s flowtime since there exists other events with flowtime much larger than those in $\mathbb{T}_1$.

If the event E is not in the set $\mathbb{T}_1$ and if the length of I^{i+1} is sufficiently longer than the length of I^i for many different values of i then the event E will be in the set $\mathbb{T}_2$. For such an event E, the requests for page p that arrive during E will be grouped according to when they arrived. We will show that each of these groups contributes to a substantial amount of event E's flowtime. Knowing that E is non-self-chargeable, we will show that OPT must perform a unique broadcast of page p for each of these groups during E. This allows us to show that $F(\mathbb{T}_2) \ll$ **LA-W**$_{1+\epsilon}$ using Lemma 3.7. Intuitively, since the optimal solution has to perform a lot of broadcasts for each event in $\mathbb{T}_2$, there cannot be many events in $\mathbb{T}_2$. Therefore the events in $\mathbb{T}_2$ do not account for a large portion of **LA-W**$_{1+\epsilon}$'s flowtime.

Finally $\mathbb{T}_3$ will consist of all events in $\mathcal{N}_2$ that are not in $\mathbb{T}_1$ or $\mathbb{T}_2$. Using the definitions of $\mathbb{T}_1$, $\mathbb{T}_2$ and τ we will be able to show that no events can be in $\mathbb{T}_3$ and this will complete our analysis. Showing that $\mathbb{T}_3 = \emptyset$ is the most difficult part of the analysis and this is where `Rule 1` and resource augmentation plays a crucial role. We now formally define the sets $\mathbb{T}_1$, $\mathbb{T}_2$ and $\mathbb{T}_3$. For simplicity of notation, let $\tau_{p,x}^{\beta,i} = \tau_p^{\beta(\frac{10}{\epsilon}i+1)}(e_{p,x})$. A $\mathcal{N}_2$ event $E_{p,x}$ is in

- $\mathbb{T}_1$ if and only if for some $0 \leq i \leq \lceil 1000c \rceil + 2$ the page p is not in Q for at least $\lceil \frac{\epsilon s}{10} |[\tau_{p,x}^{\beta,i}, e_{p,x})| \rceil$ broadcasts by our algorithm on the interval $[\tau_{p,x}^{\beta,i}, e_{p,x})$.
- $\mathbb{T}_2$ if and only if $E_{p,x} \notin \mathbb{T}_1$ and for all $0 \leq i \leq \lceil 1000c \rceil$, $\tau_{p,x}^{\beta,i} - \tau_{p,x}^{\beta,i+1} \geq \frac{\epsilon}{10}(e_{p,x} - \tau_{p,x}^{\beta,i})$
- $\mathbb{T}_3$ otherwise.

We note that if β and c are chosen such that $\beta(\frac{10}{\epsilon}(\lceil 1000c \rceil + 2) + 1) < 1$, then the time $\tau_{p,x}^{\beta,i}$ must exist for all $0 \leq i \leq \lceil 1000c \rceil + 2$. The rest of the paper is organized as follows. In Section 3.1 we will show that $F(\mathbb{T}_1) \ll$ **LA-W**$_{1+\epsilon}$. Then in Section 3.2 we will show that $F(\mathbb{T}_2) \ll$ **LA-W**$_{1+\epsilon}$. Finally we will show that $\mathbb{T}_3 = \emptyset$ in Section 3.3. Before continuing, we fix our constants, so that our arguments can be verified. Let $\beta = (\frac{\epsilon}{1000})^4$, $c = \frac{10000}{\epsilon^3}$, $\gamma = \frac{10000}{\epsilon^2 \beta}$, $\alpha = \frac{\epsilon}{100}$ and $k = \frac{10}{\epsilon}(\lceil 1000c \rceil + 2) + 1$. Note that $\tau_{p,x}^{\beta,\lceil 1000c \rceil + 2} = \tau_p^{\beta k}(e_{p,x})$ for any page p by definition of k and $\tau_{p,x}^{\beta,i}$. Recall that our algorithm is parameterized by β and c. Here we have chosen c and β so that the analysis is readable and easy to verify and not to optimize the analysis.

3.1 Bounding $\mathbb{T}_1$ events.

In this section we bound $F(\mathbb{T}_1)$. By definition of $\mathbb{T}_1$, for each event $E_{p,x} \in \mathbb{T}_1$ the page p is not in Q for at least $\lceil \frac{\epsilon s}{10} |[t_{p,x}, e_{p,x})| \rceil$ broadcasts by **LA-W**$_{1+\epsilon}$ on the interval $[t_{p,x}, e_{p,x})$ where $t_{p,x} = \tau_{p,x}^{\beta,i}$ for some fixed $0 \leq i \leq \lceil 1000c \rceil + 2$. Recall that our goal is to show that there are many events ending during $E_{p,x}$ with flowtime much larger than $F_{p,x}$. After finding these events, we will charge $F_{p,x}$ to these events. We begin by actually finding such events in the next lemma.

Lemma 3.8 *For an event $E_{p,x} \in \mathbb{T}_1$ there exist at least $(\frac{\epsilon^2 s}{205})|[t_{p,x}, e_{p,x})|$ events ending on the interval $[t_{p,x}, e_{p,x})$ with flowtime at least $\frac{c\epsilon}{20}(1-\beta k)F_{p,x}$.*

Proof: Let $S_{[b_{p,x}, t_{p,x}]}$ be the requests for page p which arrive during $[b_{p,x}, t_{p,x}]$. By the definition of $t_{p,x}$ and τ, we have $F(S_{[b_{p,x}, t_{p,x}]}) \geq (1 - \beta(\frac{10}{\epsilon} i + 1))F_{p,x} \geq (1-\beta k)F_{p,x}$. Let $I = [t_{p,x} + \frac{\epsilon}{20}(e_{p,x} - t_{p,x}), e_{p,x})$. For any time $t \in I$, by Lemma 3.1,

$$F(S_{[b_{p,x}, t_{p,x}]}, t) \geq \frac{\epsilon}{20}(1-\beta k)F_{p,x}. \tag{1}$$

By definition of $\mathbb{T}_1$, there are at least $\lceil \frac{\epsilon s}{10}(e_{p,x} - t_{p,x}) \rceil$ broadcasts by our algorithm on the interval $[t_{p,x}, e_{p,x})$ where page p is not in Q. At most $\lceil \frac{\epsilon s}{20}(e_{p,x} - t_{p,x}) \rceil$ of these broadcasts end on the interval $[t_{p,x}, t_{p,x} + \frac{\epsilon}{20}(e_{p,x} - t_{p,x}))$. Therefore, there are at least $\lceil \frac{\epsilon s}{10}(e_{p,x} - t_{p,x}) \rceil - \lceil \frac{\epsilon s}{20}(e_{p,x} - t_{p,x}) \rceil \geq \lfloor \frac{\epsilon s}{20}(e_{p,x} - t_{p,x}) \rfloor$ broadcasts by our algorithm on the interval I where page p is not in Q when these broadcasts were scheduled.

Now consider a time $t \in I$ where page p is not in $Q(t)$. By definition of Q, at time t there must exist some page q such that $F_q(t) \geq cF_p(t)$. Our algorithm schedules the page with the largest flow time every $\lfloor \frac{10}{\epsilon} \rfloor$ broadcasts according to `Rule 2`. Using this and (1), there exists an event $E_{q,y}$ with flowtime at least $F_{q,y} > \frac{c\epsilon}{20}(1-\beta k)F_{p,x}$ such that $e_{q,y} \in [t, t + \frac{1}{s}\lfloor \frac{10}{\epsilon} \rfloor)$. Using Lemma 3.6 to ensure the interval $[t_{p,x}, e_{p,x})$ is sufficiently long, we conclude that there exist at least $\lfloor (\lfloor \frac{\epsilon s}{20}(e_{p,x} - t_{p,x}) \rfloor / \lfloor \frac{10}{\epsilon} \rfloor) \rfloor \geq (\frac{\epsilon^2 s}{205})|[t_{p,x}, e_{p,x})|$ events ending during I with flowtime at least $\frac{c\epsilon}{20}(1-\beta k)F_{p,x}$. □

We can now easily bound $F(\mathbb{T}_1)$ by **LA-W**$_{1+\epsilon}$ using lemmas 3.7, 3.8, 3.5 and 3.6.

Lemma 3.9 $F(\mathbb{T}_1) < \frac{83}{100}$**LA-W**$_{1+\epsilon}$.

Proof: We apply Lemma 3.7 using the notation given in the lemma. Consider any $E_{p,x} \in \mathbb{T}_1$. Let $I_{p,x} = [t_{p,x}, e_{p,x})$. We know that the optimal solution must broadcast page p at least once on the interval $[t_{p,x}, e_{p,x})$ by Lemma 3.5, since $[\tau_p^\beta(e_{p,x}), e_{p,x})$ is a subinterval of $[t_{p,x}, e_{p,x})$. So we can set $\lambda = 1$. By Lemma 3.8 we have that for any event $E_{p,x} \in \mathbb{T}_1$ there exist at least $\frac{\epsilon^2 s}{205}|[t_{p,x}, e_{p,x})|$ events ending on the interval $[t_{p,x}, e_{p,x})$ of flowtime at least $\frac{c\epsilon}{20}(1-\beta k)F_{p,x}$. If we let the set $\mathcal{B}_{p,x}$ consist of these events, we can set $\mu = \frac{\epsilon^2 s}{205}$ and $\kappa = \frac{c\epsilon}{20}(1-\beta k)$. Using Lemma 3.6 we know that $|I_{p,x}| \geq 10000/\epsilon^2$ and therefore $d = \min_{E_{p,x} \in \mathcal{A}} |I_{p,x}| \geq 10000/\epsilon^2$. Thus we have

$$F(\mathbb{T}_2) \leq \frac{2}{\kappa\mu\lambda}\frac{d+1}{d}\textbf{LA-W}_{1+\epsilon} = (\frac{41}{50(1+\epsilon)})(\frac{1}{1-\beta k})\frac{d+1}{d}\textbf{LA-W}_{1+\epsilon} < \frac{83}{100}\textbf{LA-W}_{1+\epsilon}.$$

□

3.2 Bounding $\mathbb{T}_2$ events.

In this section, we bound $F(\mathbb{T}_2)$. Recall that our goal is to show that for any event $E_{p,x} \in \mathbb{T}_2$, the optimal solution must broadcast page p many times during $E_{p,x}$. To find these broadcasts by the optimal solution, we break up each event $E_{p,x} \in \mathbb{T}_2$ into the time intervals $[\tau_{p,x}^{\beta,i+2}, \tau_{p,x}^{\beta,i})$. By definition of τ, we know that the requests for page p that arrive during $[\tau_{p,x}^{\beta,i+2}, \tau_{p,x}^{\beta,i+1}]$ account for a substantial portion of the flowtime of event $E_{p,x}$. Knowing this and that the length of $[\tau_{p,x}^{\beta,i+1}, \tau_{p,x}^{\beta,i})$ is sufficiently long by definition of events in $\mathbb{T}_2$, we will be able to show that the optimal solution must broadcast page p during $[\tau_{p,x}^{\beta,i+2}, \tau_{p,x}^{\beta,i})$. Otherwise, these requests wait for a sufficiently long time to be scheduled by OPT and, therefore, OPT must accumulate flowtime at least $\frac{1}{\gamma}F_{p,x}$ for these requests. This contradicts the fact that events in $\mathbb{T}_2$ are non-self-chargeable.

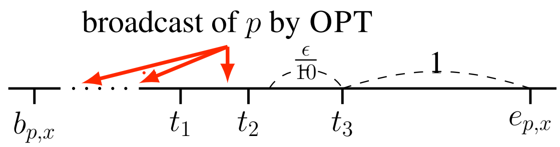

Figure 3: For any event $E_{p,x}$ in $\mathbb{T}_2$, OPT must broadcast page p during $[t_1, t_3)$.

Lemma 3.10 *Let $E_{p,x}$ be an event in $\mathbb{T}_2$. For any integer i s.t. $0 \leq i \leq \lceil 1000c \rceil$, the optimal solution must broadcast page p during the interval $[\tau_{p,x}^{\beta,i+2}, \tau_{p,x}^{\beta,i})$.*

Proof: For any fixed integer i such that $0 \leq i \leq \lceil 1000c \rceil$, let $t_1 = \tau_{p,x}^{\beta,i+2}$, $t_2 = \tau_{p,x}^{\beta,i+1}$, and $t_3 = \tau_{p,x}^{\beta,i}$. Note that $t_3 - t_2 \geq \frac{\epsilon}{10}(e_{p,x} - t_3)$ and $t_1 < t_2 < t_3$, since $E_{p,x} \in \mathbb{T}_2$. See Figure 3. Let $S_{[t_1, e_{p,x})}$, $S_{(t_2, e_{p,x})}$ and $S_{[t_1,t_2]}$ be the set of requests for page p which arrive on the intervals $[t_1, e_{p,x})$, $(t_2, e_{p,x})$ and $[t_1, t_2]$, respectively. By definition of t_1 and t_2, we have that $F(S_{[t_1,e_{p,x})}) > \beta(\frac{10}{\epsilon}(i+2)+1)F_{p,x}$ and $F(S_{(t_2,e_{p,x})}) \leq \beta(\frac{10}{\epsilon}(i+1)+1)F_{p,x}$. Thus we have,

$$F(S_{[t_1,t_2]}) = F(S_{[t_1,e_{p,x})}) - F(S_{(t_2,e_{p,x})}) > \frac{10}{\epsilon}\beta F_{p,x}. \tag{2}$$

With the fact $t_3 - t_2 \geq \frac{\epsilon}{10}(e_{p,x} - t_3)$, the fact that the requests in $S_{[t_1,t_2]}$ arrive by time t_2, and (2), by applying Lemma 3.1 we have

$$F(S_{[t_1,t_2]}, t_3) \geq (\frac{\epsilon}{10})(\frac{10}{\epsilon})\beta F_{p,x} = \beta F_{p,x}. \tag{3}$$

For the sake of contradiction, suppose that the optimal solution does not broadcast page p on the interval $[t_1, t_3)$. Then

$$F_{p,x}^* \geq F(S_{[t_1,t_2]}, t_3) \geq \beta F_{p,x} \geq \frac{1}{\gamma}F_{p,x}. \tag{4}$$

This is a contradiction to $E_{p,x}$ being non-self-chargeable. □

Corollary 3.11 *For each event $E_{p,x} \in \mathbb{T}_2$, the optimal solution broadcasts page p at least $\lceil 500c \rceil$ times during the interval $[\tau_{p,x}^{\beta k}, e_{p,x})$.*

At this point, we have shown the most interesting property of events in $\mathbb{T}_2$ and we are almost ready to bound $F(\mathbb{T}_2)$. Before bounding $F(\mathbb{T}_2)$, we first find events to charge to. For each event $E_{p,x} \in \mathbb{T}_2$, we want to charge $F_{p,x}$ to some events ending during $[\tau_{p,x}^{\beta k}, e_{p,x})$ because we know OPT broadcasts page p many times during this interval. Knowing that **LA-W** always broadcasts the page with flowtime close to the highest flowtime, we can easily find events ending during $[\tau_{p,x}^{\beta k}, e_{p,x})$ with sufficiently large flowtime.

Lemma 3.12 *Consider any event $E_{p,x} \in \mathbb{T}_2$. Let $I_{p,x} = [\tau_{p,x}^{\beta k}, e_{p,x})$. There exist at least $\frac{49}{100}(1+\epsilon)|I_{p,x}|$ events ending during $I_{p,x}$ with flowtime at least $\frac{1}{2c}(1-\beta k)F_{p,x}$.*

Proof: Let $I'_{p,x} = [\tau_{p,x}^{\beta k} + \frac{1}{2}(e_{p,x} - \tau_{p,x}^{\beta k}), e_{p,x})$. Note that there are at least $\lfloor (1+\epsilon)\frac{1}{2}|I_{p,x}| \rfloor \geq (1+\epsilon)\frac{49}{100}|I_{p,x}|$ events ending during $I'_{p,x}$; the inequality is due to Lemma 3.6 to ensure $|I_{p,x}|$ is sufficiently long. Let $E_{q,y}$ be an event such that $e_{q,y} \in I'_{p,x}$. We now show that $F_{q,y} \geq \frac{1}{2c}(1-\beta k)F_{p,x}$. By Lemma 3.1 and the definition of $\tau_{p,x}^{\beta k}$ we have $F_p(e_{q,y}) \geq \frac{1}{2}(1-\beta k)F_{p,x}$. Since our algorithm chose page q over page p at time t, according to either `Rule 1` or `Rule 2`, $F_{q,y} \geq \frac{1}{c}F_p(e_{q,y})$. Hence we conclude that $F_{q,y} \geq \frac{1}{2c}(1-\beta k)F_{p,x}$. □

Finally we bound the flowtime of $\mathbb{T}_2$ events by charging an event $E_{p,x} \in \mathbb{T}_2$ to the events we found in Lemma 3.12. Notice that the events we are charging to can have flowtime less that $F_{p,x}$, but we counter this by finding many broadcasts of page p by OPT during $E_{p,x}$.

Lemma 3.13 *For* $0 < \epsilon \leq 1$, $F(\mathbb{T}_2) < \frac{2}{100}$**LA-W**$_{1+\epsilon}$.

Proof: We apply Lemma 3.7. Let $E_{p,x} \in \mathbb{T}_2$ and $I_{p,x} = [\tau_{p,x}^{\beta k}, e_{p,x})$. By Corollary 3.11 we can set $\lambda = 500c$. By letting $\mathcal{B}_{p,x}$ be the set of events found for $E_{p,x}$ in Lemma 3.12, we can set $\kappa = \frac{1}{2c}(1 - \beta k)$ and $\mu = \frac{49}{100}(1 + \epsilon)$. Using Lemma 3.6 we know that $|I_{p,x}| \geq 10000/\epsilon^2$ and therefore $d = \min_{E_{p,x} \in \mathcal{A}} |I_{p,x}| \geq 10000/\epsilon^2$. The desired result follows by simple calculation. □

3.3 There are no events in $\mathbb{T}_3$.

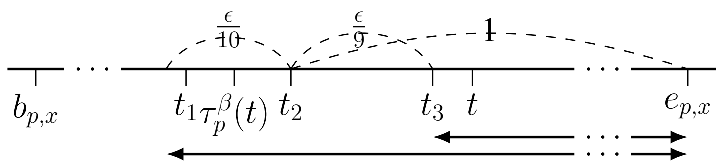

Figure 4: For an event $E_{p,x}$ in $\mathbb{T}_3$, during $[t_1, e_{p,x})$ OPT must make a unique broadcast for most events which end during $[t_3, e_{p,x})$.

In this section we show $\mathbb{T}_3 = \emptyset$. For the sake of contradiction suppose that $\mathbb{T}_3$ is non-empty. Fix an event $E_{p,x} \in \mathbb{T}_3$. For some *fixed* $0 \leq i \leq \lceil 1000c \rceil$ we have that $\tau_{p,x}^{\beta,i} - \tau_{p,x}^{\beta,i+1} < \frac{\epsilon}{10}(e_{p,x} - \tau_{p,x}^{\beta,i})$ because $E_{p,x} \notin \mathbb{T}_2$. Let $t_1 = \tau_{p,x}^{\beta,i+1}$ and $t_2 = \tau_{p,x}^{\beta,i}$. Let $t_3 = t_2 + \frac{\epsilon}{9}(e_{p,x} - t_2)$. Let $\mathcal{E}$ be all the non-self-chargeable events ending during $[t_3, e_{p,x})$ which were scheduled by `Rule 1` when page p was in Q. Our goal is to show that OPT must make a unique broadcast for each event in $\mathcal{E}$ on the interval $[t_1, e_{p,x})$. Then it will be shown that $|\mathcal{E}| > |[t_1, e_{p,x})| + 1$ by showing $|\mathcal{E}| \simeq (1 + \epsilon)|[t_3, e_{p,x})| > |[t_1, e_{p,x})| + 1$. Since OPT has 1 speed, this will show that OPT cannot complete these broadcasts on the interval $[t_1, e_{p,x})$. This contradiction will imply that $\mathbb{T}_3 = \emptyset$. See Figure 4.

Recall that by Lemma 3.5, for any $E_{q,y} \in \mathcal{E}$, the optimal solution must broadcast page q on the interval $[\tau_q^\beta(e_{q,y}), e_{q,y})$ because $E_{q,y}$ is non-self-chargeable. Further, note that such broadcasts are unique to $E_{q,y}$, i.e. not contained in $E_{q,y'}$ for any $y' \neq y$ because $E_{q,y'}$ and $E_{q,y}$ are disjoint by definition. For any $E_{q,y} \in \mathcal{E}$, if we show that $\tau_q^\beta(e_{q,y}) \in [t_1, e_{p,x})$ then we will know that OPT performs these broadcasts on $[t_1, e_{p,x})$. This is where `Rule 1` will play a crucial role in our analysis. We will first show that $\tau_p^\beta(t) \geq t_1$ for all times $t \in [t_3, e_{p,x})$. By definition, if page q was scheduled by `Rule 1` and page p was in $Q(t)$ then $\tau_p^\beta(t) \leq \tau_q^\beta(t)$. Hence, for any $E_{q,y} \in \mathcal{E}$ we will have that $t_1 \leq \tau_p^\beta(e_{q,y}) \leq \tau_q^\beta(e_{q,y})$ and OPT broadcasts page q on $[t_1, e_{p,x})$.

Lemma 3.14 *For the event* $E_{p,x} \in \mathbb{T}_3$*, at any time* $t \in [t_3, e_{p,x})$*,* $\tau_p^\beta(t) \geq t_1$.

Proof: For the sake of contradiction assume that $\tau_p^\beta(t) < t_1$. Let $t' = \tau_p^\beta(t)$. Note that $t' < t_1 \leq t_2 < t < e_{p,x}$. Let $S_{[t_1,e_{p,x})}$, $S_{(t_2,e_{p,x})}$ and $S_{[t_1,t_2]}$ be the set of requests which arrive for page p on the intervals $[t_1, e_{p,x})$, $(t_2, e_{p,x})$, and $[t_1, t_2]$, respectively. By definition of t_1 and t_2, we have $F(S_{[t_1,e_{p,x})}) > \beta(\frac{10}{\epsilon}(i+1) + 1)F_{p,x}$ and $F(S_{(t_2,e_{p,x})}) \leq \beta(\frac{10}{\epsilon}i + 1)F_{p,x}$. Hence,

$$F(S_{[t_1,t_2]}) = F(S_{[t_1,e_{p,x})}) - F(S_{(t_2,e_{p,x})}) > \frac{10}{\epsilon}\beta F_{p,x}. \tag{5}$$

By the definition of $t' = \tau_p^\beta(t)$, we have $F(S_{(t',t_2]}, t) \leq F(S_{(t',t]}, t) \leq \beta F_p(t) \leq \beta F_{p,x}$. Since $t \geq t_2 + \frac{\epsilon}{9}(e_{p,x} - t_2)$, by Lemma 3.7, $\frac{\epsilon}{9}F(S_{(t',t_2]}, e_{p,x}) \leq F(S_{(t',t_2]}, t)$. Thus we have,

$$F(S_{(t',t_2]}) = F(S_{(t',t_2]}, e_{p,x}) \leq \frac{9}{\epsilon}F(S_{(t',t_2]}, t) \leq \frac{9}{\epsilon}\beta F_{p,x}. \tag{6}$$

Knowing that $F(S_{(t',t_2]}) \geq F(S_{[t_1,t_2]})$, this is a contradiction to (5). □

Finally we are ready to show that $\mathbb{T}_3 = \emptyset$. This lemma follows by using the previous lemma and counting the number of broadcasts the optimal solution must do on the interval $[t_1, e_{p,x})$. It is in the next lemma that we rely strongly on resource augmentation.

Lemma 3.15 *It must be the case that* $\mathbb{T}_3 = \emptyset$.

Proof: Recall that $\mathcal{E}$ is the set of all the non-self-chargeable events ending during $[t_3, e_{p,x})$ which were scheduled by `Rule 1` when page p was in Q. We first show $|\mathcal{E}| > s(1 - \frac{34}{100}\epsilon)(e_{p,x} - t_2)$ by a simple counting argument. We know that at least $\lfloor s(1 - \frac{\epsilon}{9})(e_{p,x} - t_2)\rfloor$ events end during $[t_3, e_{p,x})$ by definition of t_3 and t_2. Among these events we know that at most $\alpha s(e_{p,x} - t_2)$ events are self-chargeable, since $E_{p,x} \in \mathcal{N}_2$; at most $\lceil s(e_{p,x} - t_2)\rceil / \lfloor \frac{10}{\epsilon} \rfloor + 1 \leq \epsilon s \frac{101}{900}(e_{p,x} - t_2)$ broadcasts are scheduled by `Rule 2`, since our algorithm performs according to `Rule 2` every $\lfloor \frac{10}{\epsilon} \rfloor$ broadcasts (the inequality is due to Lemma 3.6); and at most $\frac{\epsilon s}{10}(e_{p,x} - t_2)$ events were scheduled when p is not in the queue Q, since $E_{p,x} \notin \mathbb{T}_1$. By subtracting these numbers from the number of events ending during $I'_{p,x}$ and knowing that $(e_{p,x} - t_2) \geq \frac{10000}{\epsilon^2}$ by Lemma 3.6, we have

$$(1 - \frac{34}{100}\epsilon)(1 + \epsilon)(e_{p,x} - t_2) \leq |\mathcal{E}|. \tag{7}$$

Knowing that $t_2 - t_1 < \frac{\epsilon}{10}(e_{p,x} - t_2)$, we have

$$|[t_1, e_{p,x}]| < (1 + \frac{\epsilon}{10})(e_{p,x} - t_2). \tag{8}$$

As discussed previously, Lemma 3.14 implies that OPT must make a unique broadcast for each event in $\mathcal{E}$ during $[t_1, e_{p,x})$. Since the optimal solution has 1 speed, with Lemma 3.6, it must be the case that

$$|\mathcal{E}| \leq |[t_1, e_{p,x})| + 1 \leq (1 + \frac{\epsilon}{10000})|[t_1, e_{p,x})|. \tag{9}$$

By combining (7), (8), and (9), we have that $(1 - \frac{34}{100}\epsilon)(1 + \epsilon) < (1 + \frac{\epsilon}{10})(1 + \frac{\epsilon}{10000})$. For any $0 < \epsilon \leq 1$ this is not true, so we obtain a contradiction. □

This completes our analysis. By lemmas 3.9, 3.13 and 3.15 we have that $F(\mathcal{N}_2) \leq \frac{85}{100}$**LA-W**$_{1+\epsilon}$. The proof of Theorem 3.4 follows easily by combining this and lemmas 3.2 and 3.3.

4 Conclusion

In this paper we have given the first $(1 + \epsilon)$-speed $O(1)$-competitive algorithm for the objective of minimizing the total flowtime in broadcast scheduling with unit sized pages. Recently, Bansal et al. gave another scalable algorithm with better competitive ratio which works for varying sized pages [5].

It is also important to note that the algorithm **LA-W** is parameterized by ϵ and so is the algorithm given in [5]. It would be interesting to show a $(1 + \epsilon)$-speed $O(1)$-competitive algorithm which scales with ϵ without knowledge of ϵ. It would be worth exploring a simpler algorithm with better competitive ratio.

Acknowledgements: We thank Chandra Chekuri for many helpful discussions and his 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] 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.

[4] 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.

[5] Nikhil Bansal, Ravishankar Krishnaswamy, and Viswanath Nagarajan. Better scalable algorithms for broadcast scheduling. 2009.

[6] 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.

[7] 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.

[8] Luca Becchetti, Stefano Leonardi, Alberto Marchetti-Spaccamela, and Kirk Pruhs. Online weighted flow time and deadline scheduling. *J. Discrete Algorithms*, 4(3):339–352, 2006.

[9] 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.

[10] 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.

[11] 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.

[12] Chandra Chekuri, Sungjin Im, and Benjamin Moseley. Longest wait first for broadcast scheduling. In *WAOA '09: Proceedings of 7th Workshop on Approximation and Online Algorithms (to appear)*, 2009.

[13] Chandra Chekuri, Sungjin Im, and Benjamin Moseley. Minimizing maximum response time and delay factor in broadcasting scheduling. In *ESA '09: Proceedings of the seventeenth annual European symposium on algorithms*, pages 444–455, 2009.

[14] Chandra Chekuri and Benjamin Moseley. Online scheduling to minimize the maximum delay factor. In *SODA '09: Proceedings of the Nineteenth Annual ACM -SIAM Symposium on Discrete Algorithms*, pages 1116–1125, Philadelphia, PA, USA, 2009. Society for Industrial and Applied Mathematics.

[15] 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.

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

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

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

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

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

[21] Jeff Edmonds and Kirk Pruhs. Scalably scheduling processes with arbitrary speedup curves. In *SODA '09: Proceedings of the Nineteenth Annual ACM -SIAM Symposium on Discrete Algorithms*, pages 685–692, Philadelphia, PA, USA, 2009. Society for Industrial and Applied Mathematics.

[22] 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.

[23] 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.

[24] Rajiv Gandhi, Samir Khuller, Srinivasan Parthasarathy, and Aravind Srinivasan. Dependent rounding in bipartite graphs. In *FOCS '02: Proceedings of the 43rd Symposium on Foundations of Computer Science*, pages 323–332, 2002.

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

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

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

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

[33] 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 Proof of Lemma 3.7

Here we prove Lemma 3.7. The proof of this lemma relies on a generalization of Hall's theorem. This generalization of Hall's theorem was implicitly used in [20], and later formalized in [12].

Definition A.1 *[12][g-covering] Let $G = (X \cup Y, E)$ be a bipartite graph whose two parts are X and Y, respectively. Let $\ell : E \to [0, 1]$. We say $\{\ell_{u,v}\}$ is a g-covering if $\sum_{v \in Y} \ell_{u,v} = 1$ and $\sum_{u \in X} \ell_{u,v} \leq g$.*

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

Lemma A.2 *[12][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 $N_G(S) = \{v \in Y \mid uv \in E,\ u \in S\}$, be the neighborhood of S. For every $S \subseteq X$, if $|N_G(S)| \geq \frac{1}{g}|S|$, then there exists a g-covering for X.*

Now we are ready to prove Lemma 3.7.

Proof of [Lemma 3.7] We start by creating a bipartite graph $G = (X \cup Y, E)$. There is one vertex $u_{p,x} \in X$ for each event $E_{p,x} \in \mathcal{A}$ and there is a vertex $v_{q,y} \in Y$ for each event in $E_{q,y} \in \mathcal{B}$. Let $u_{p,x} \in X$ and $v_{q,y} \in Y$. There is an edge connecting $u_{p,x}$ and $v_{q,y}$ if and only if $E_{q,y} \in \mathcal{B}_{p,x}$. For any set $Z \subseteq X$, let $\mathcal{I}(Z)$ be the set of intervals corresponding to events in Z, i.e. $\mathcal{I}(Z) = \{\ I_{p,x} \mid u_{p,x} \in Z\}$. We let $\bigcup \mathcal{I}(Z)$ denote the union of intervals in $\mathcal{I}(Z)$. We denote the sum of length of maximal subintervals in $\bigcup \mathcal{I}(Z)$ by $|\bigcup \mathcal{I}(Z)|$. We will now show that G has a $((\frac{2}{\lambda\mu})(\frac{d+1}{d}))$-covering for X.

Consider any fixed set $Z \subseteq X$. Notice that

$$\lambda|Z| \leq \frac{d+1}{d}|\bigcup \mathcal{I}(Z)|. \tag{10}$$

This is because the optimal solution must perform λ unique broadcasts for each event in Z during $\mathcal{I}(Z)$, the optimal solution has 1 speed, and there are at most $\frac{d+1}{d}|\bigcup \mathcal{I}(Z)|$ integral time steps during $\mathcal{I}(Z)$ where OPT can broadcast.

From now on, for simplicity, we assume that $\bigcup \mathcal{I}(Z)$ is one continuous interval; otherwise our argument can be applied to each maximal subinterval in $\bigcup \mathcal{I}(Z)$. Let $\mathcal{I}' \subseteq \mathcal{I}(Z)$ be such that for any two intervals $I_{p,x}, I_{q,y} \in \mathcal{I}'$ it is the case that $I_{p,x}$ is not completely contained in $I_{q,y}$, and also $\bigcup \mathcal{I}' = \bigcup \mathcal{I}(Z)$. By definition,

$$|\bigcup \mathcal{I}'| = |\bigcup \mathcal{I}(Z)|. \tag{11}$$

We order all intervals in $\mathcal{I}'$ in the increasing order of starting points. We pick intervals from $\mathcal{I}'$ one by one and label them by the order they are picked; the i_{th} selected interval is denoted by I_i. Starting with I_1, we pick I_{i+1} so that I_{i+1} the least overlaps with I_i; here we will say I_{i+1} overlaps with I_i even when I_{i+1} starts exactly where I_i ends. Let $\mathcal{I}'_{odd}$ and $\mathcal{I}'_{even}$ be the odd indexed and even indexed intervals, respectively. WLOG, assume that $|\bigcup \mathcal{I}'_{odd}| \geq |\bigcup \mathcal{I}'_{even}|$. Since $\mathcal{I}'_{odd}$ and $\mathcal{I}'_{even}$ are a partition of $\mathcal{I}'$, we know that $|\bigcup \mathcal{I}'_{odd}| + |\bigcup \mathcal{I}'_{even}| \geq |\bigcup \mathcal{I}'|$. Thus we have

$$|\bigcup \mathcal{I}'_{odd}| \geq \frac{1}{2}|\bigcup \mathcal{I}'|. \tag{12}$$

Let $N_G(Z)$ be the neighborhood of Z. We now show that $|N_G(Z)| \geq \mu|\bigcup \mathcal{I}'_{odd}|$. Note that $u_{p,x}$, corresponding to $I_{p,x}$ in $\mathcal{I}'_{odd}$, has at least $\mu|I_{p,x}|$ neighbors. Also note that all intervals in $\mathcal{I}'_{odd}$ are disjoint by construction of $\mathcal{I}'_{odd}$. Hence, by summing up all neighbors of vertices corresponding to intervals in $\mathcal{I}'_{odd}$, we have

$$|N_G(Z)| \geq \mu |\bigcup \mathcal{I}'_{odd}|. \tag{13}$$

From (10), (11), (12) and (13), We have $|N_G(Z)| \geq (\frac{\lambda\mu}{2})(\frac{d}{d+1})|Z|$ and G has a $((\frac{2}{\lambda\mu})(\frac{d+1}{d}))$-covering using Lemma A.2. Let ℓ be such a covering.

$$\begin{aligned}
F(\mathcal{A}) &= \sum_{u_{p,x} \in X} F_{p,x} \\
&= \sum_{u_{p,x} v_{q,y} \in E} \ell_{u_{p,x} v_{q,y}} F_{p,x} \quad \text{[By definition of the covering]} \\
&\leq \sum_{u_{p,x} v_{q,y} \in E} \ell_{u_{p,x} v_{q,y}} \frac{F_{q,y}}{\kappa} \quad \text{[By } F_{q,y} \geq \kappa F_{p,x}\text{]} \\
&\leq (\frac{2}{\kappa\lambda\mu})(\frac{d+1}{d}) \sum_{v_{q,y} \in Y} F_{q,y} \quad \text{[Change order of the summation and } \ell \text{ is a } ((\tfrac{2}{\lambda\mu})(\tfrac{d+1}{d}))\text{-covering]} \\
&= (\frac{2}{\kappa\lambda\mu})(\frac{d+1}{d}) F(\mathcal{B}) \quad \text{[Since } Y \text{ is the set of vertices corresponding to events in } \mathcal{B}\text{]} \\
&\leq (\frac{2}{\kappa\lambda\mu})(\frac{d+1}{d}) \textbf{LA-W}_{1+\epsilon} \quad \text{[Since } \mathcal{B} \text{ is a subset of all events]}
\end{aligned}$$

□

B Omitted Proofs

Proof of [Lemma 3.3] We apply Lemma 3.7 using the notation given in the lemma. Let $\mathcal{A}$ be the set of all $\mathcal{N}_1$ events. Consider any event $E_{p,x} \in \mathcal{A}$. Let $I_{p,x} = [\tau_p^\rho(e_{p,x}), e_{p,x})$ for some fixed $\beta \leq \rho \leq \beta(\frac{10}{\epsilon}(\lceil 1000c \rceil + 2) + 1)$ such that at least $\lceil \alpha s(e_{p,x} - \tau^\rho(e_{p,x})) \rceil$ self-chargeable events end on $I_{p,x}$. Note that ρ exists by definition of $\mathcal{N}_1$ events. By Lemma 3.5, the optimal solution must broadcast page p during $I_{p,x}$. Due to this we set $\lambda = 1$. Since $|I_{p,x}| \geq \frac{10000}{\epsilon^2}$ by Lemma 3.6, we have $d = \min_{E_{p,x} \in \mathcal{A}} |I_{p,x}| \geq \frac{10000}{\epsilon^2}$.

Let $\mathcal{B}_{p,x}$ be the self-chargeable events ending during $I'_{p,x} = [\tau_p^\rho(e_{p,x}) + \frac{\alpha}{2}(e_{p,x} - \tau_p^\rho(e_{p,x})), e_{p,x})$. Note that there are at most $\lceil \frac{\alpha s}{2}|I_{p,x}| \rceil$ events ending during $I_{p,x} \setminus I'_{p,x}$. Therefore there exist at least $\lceil \alpha s |I_{p,x}| \rceil - \lceil \frac{\alpha s}{2}|I_{p,x}| \rceil \geq \lfloor \frac{\alpha s}{2}|I_{p,x}| \rfloor \geq \frac{\alpha s}{4}|I_{p,x}|$ self-chargeable events ending during $I'_{p,x}$. Hence, $|\mathcal{B}_{p,x}| \geq \frac{\alpha s}{4}|I_{p,x}|$ and we can set $\mu = \frac{\alpha s}{4}$.

Let $E_{q,y} \in \mathcal{B}_{p,x}$. By Lemma 3.1 and the definition of $\tau_p^\rho(e_{p,x})$ we know that at anytime $t \in I'_{p,x}$ it is the case that $F_{p,x}(t) \geq \frac{\alpha}{2}(1-\rho)F_{p,x}$. Since our algorithm chose to broadcast page q at time $e_{p,x} \in I'_{p,x}$ over page p, we have $F_{q,y} \geq \frac{\alpha}{2c}(1-\rho)F_{p,x}$. Therefore we can set $\kappa = \frac{\alpha}{2c}(1-\rho)$.

In sum, by Lemma 3.7,

$$F(\mathcal{N}_1) \leq \frac{2}{\lambda\kappa\mu}\frac{d+1}{d}F(\mathcal{S}) = (\frac{16c}{\alpha^2 s})(\frac{1}{1-\rho})\frac{d+1}{d}(\gamma\text{OPT}) = O(\frac{1}{\epsilon^{11}})\text{OPT}.$$

□