The Loss of Serving in The Dark

Yossi Azar*
Tel-Aviv University
Email: azar@tau.ac.il

Ilan Cohen†
Tel-Aviv University
Email: ilanrcohen@gmail.com

Iftah Gamzu
Tel-Aviv University
Email: iftah.gamzu@cs.tau.ac.il

Abstract

We study the following balls and bins stochastic process: There is a buffer with B bins, and there is a stream of balls $X = \langle X_1, X_2, \ldots, X_T \rangle$ such that X_i is the number of balls that arrive before time i but after time $i-1$. Once a ball arrives, it is stored in one of the unoccupied bins. If all the bins are occupied then the ball is thrown away. In each time step, we select a bin uniformly at random, clear it, and gain its content. Once the stream of balls ends, all the remaining balls in the buffer are cleared and added to our gain. We are interested in analyzing the expected gain of this randomized process with respect to that of an optimal gain-maximizing strategy, which gets the same online stream of balls, and clears a ball from a bin, if exists, at any step. We name this gain ratio the loss of serving in the dark.

In this paper, we determine the exact loss of serving in the dark. We prove that the expected gain of the randomized process is worse by a factor of $\rho + \varepsilon$ from that of the optimal gain-maximizing strategy for any $\varepsilon > 0$, where $\rho = \max_{\alpha>1} \alpha e^{\alpha}/((\alpha-1)e^{\alpha} + e - 1) \approx 1.69996$ and $B = \Omega(1/\varepsilon^3)$. We also demonstrate that this bound is essentially tight as there are specific ball streams for which the above-mentioned gain ratio tends to ρ. Our stochastic process occurs naturally in many applications. We present a prompt and truthful mechanism for bounded capacity auctions, and an application relating to packets scheduling.

1 Introduction

Consider the fundamental packets scheduling scenario in which there is an online stream of packets with arbitrary values arriving to a network device that can accommodate B packets. The device can transmit one packet in each time-step. The goal is to maximize the overall value of transmitted packets. A trivial greedy algorithm for this scenario keeps the B packets with the highest values at any point in time, and transmits the packet with the highest value when possible. This algorithm is optimal. However, it inspects the values of the packets prior to their transmission. We are interested in algorithms whose transmission decisions are *value-oblivious*. Such algorithms have the property that if one focuses on any single packet then for all possible values it is either transmitted in the same time-step or rejected. Value-oblivious algorithms are beneficial in game-theoretic settings and when fairness is required (for example in *prompt mechanism*). Informally, one would not like a high value packet to have a higher priority in transmission compared to a low value packet if both packets are to be transmitted. We note that value-oblivious algorithms may inspect the values of packets on their arrival. Therefore, one can assume without loss of generality that any value-oblivious algorithm keeps the B packets with the highest values at any point in time. One example of a

*Supported in part by the Israel Science Foundation (grant No. 1404/10). Part of this work was done while the author was visiting Microsoft Research, Redmond.

†Supported in part by the Google Inter-university center.

value-oblivious algorithm is the FIFO algorithm, which transmits the earliest packet in the buffer. This algorithm is known to be 2-competitive against the absolute optimum [24]. A natural question is whether one can design a value-oblivious algorithm with a better competitive ratio.

We consider a simple randomized algorithm that transmits a packet from the buffer uniformly at random. The core of analyzing this algorithm can be reduced using the zero-one principle [6], that is described later, to the following natural *balls and bins stochastic process*: There is a buffer with B bins, and there is a stream of balls $X = \langle X_1, X_2, \ldots, X_T \rangle$ such that X_i is the number of balls that arrive before time i but after time $i-1$. Once a ball arrives, it is stored in one of the unoccupied bins, i.e., a bin that does not hold a ball. If all the bins are occupied then the ball is thrown away. In each time step, we select a bin uniformly at random, clear it, and gain its content. In particular, if that bin is occupied with a ball then our gain is one; otherwise, our gain is zero. Once the stream of balls ends, all the remaining balls in the buffer are cleared and added to our gain. We are interested in analyzing the expected gain of this randomized process with respect to that of an *optimal gain-maximizing strategy*, which gets the same online stream of balls, and clears a ball from a bin, if exists, at any step. We name this gain ratio the *loss of serving in the dark* since the bins are selected without knowledge about their content.

1.1 Our results

Determining the exact loss of serving in the dark. We prove that the expected gain of the randomized process is worse by a factor of $\rho + \varepsilon$ from that of the optimal gain-maximizing strategy for any $\varepsilon > 0$, where ρ is defined by the following algebraic expression

$$\rho = \max_{\alpha > 1} \frac{\alpha e^{\alpha}}{(\alpha - 1)e^{\alpha} + e - 1} \approx 1.69996$$

and $B = \Omega(1/\varepsilon^3)$. We also demonstrate that this bound is essentially tight as there are specific ball streams for which the above-mentioned gain ratio tends to ρ. As a corollary, we attain that the asymptotic loss of serving in the dark is *exactly* ρ. These findings are presented in Section 2.

Application 1: Value-oblivious packets scheduling. The stochastic process occurs naturally in many applications. As described before, one such example is value-oblivious packets scheduling. The above result implies that the random transmission algorithm has a competitive ratio of $\rho + \varepsilon$. Note that in the randomized algorithm, a packet might remain in the buffer for a long time. Nevertheless, one can easily validate that with high probability, the delay of a packet is at most logarithmic more than its delay in the FIFO algorithm. Furthermore, one can reject a packet after it stayed in the buffer for $O(B \log B)$ steps without degrading the competitive ratio. For further details see Appendix 3.1.

Application 2: Prompt mechanisms for bounded capacity auctions. We use the stochastic process to establish a natural randomized selection mechanism for the bounded capacity auctions. A bounded capacity auction is a single-item periodic auction for bidders that arrive online, and the number of participating bidders is bounded , e.g., when the auction room has a limited size. We show that the random selection mechanism is truthful, support prompt payments and achieves an expected competitive ratio of $\rho + \varepsilon$. This finding surpasses a 2-competitive algorithm for the problem. Detailed description on the bounded capacity auction, truthful and prompt mechanisms, and related work described in Appendix 3.2.

1.2 Our approach and techniques

An essential component in our approach is to utilize a deterministic fractional process, designed in a natural way to correspond to the randomized process, as a proxy for the analysis of the loss of serving in the dark.

As we do not know how to analyze the loss of serving in the dark directly, we make the following two steps which combine together to yield the desired result:

(1) *Analyzing the fractional process against the optimal one* – We characterize the ball stream with the worst gain ratio between the fractional process and the optimal one. This characterization defines the stream uniquely (i.e., depending only on its length), and reduces the problem of finding the worst gain ratio between the two previously-mentioned processes to that of analyzing a specific algebraic expression, which was previously identified with ρ.

(2) *Analyzing the randomized process against the fractional one* – Ideally, we would have liked to show that the expected gain of the randomized process and the gain of the fractional one are essentially equal. Kurtz's theorem [27] informally says that the solutions of a stochastic process behave similar to the solutions to the differential equation of its fractional counterpart (see, e.g., [23]). Unfortunately, we cannot apply this theorem in our setting due to the hard constraint on number of bins that induces overflows. Specifically, one can demonstrate that there is a drift between the randomized and fractional processes. We define a modified fractional process that enables us to bound this ε-drift. We then bound the gain difference between our randomized process and the modified fractional process. This is achieved by applying Azuma's inequality to a super-martingale process defined with respect to the two previously-mentioned processes.

1.3 Related work

A classical and well-known balls and bins scenario is when B balls are placed into B bins, where the optimization criteria is the fraction of full bins, namely, bins that got at least one ball. A simple result demonstrates that if the balls are placed independently and uniformly at random then the expected fraction of full bins is $1-1/e$. This result has a similar flavor to our result in the sense that if this process could have been done in the light, i.e., one could deterministically place each ball in any bin, then the fraction of full bins would have been 1; however, since this process is done in the dark, i.e., the balls are placed in a random way, then there is a loss of gain.

There are other randomized ball and bins stochastic processes that have been analyzed using various techniques such as martingales and Azuma's inequality. Due to the ever-growing line of work in this context, it is beyond the scope of this writing to do justice and present an exhaustive survey of previous work. We refer the reader to directly related papers [22, 26, 31, 5, 2, 32, 33, 16] and to the references therein for a more comprehensive review of the literature.

2 The Stochastic Process and its Analysis

In this section, we prove the next theorem that determines the loss of serving in the dark.

Theorem 2.1. *The expected gain of the randomized process is worse by a factor of $\rho+\varepsilon$ from that of the optimal gain-maximizing strategy for $B=\Omega(1/\varepsilon^3)$.*

We also show that the above gain ratio is essentially tight, resulting in the following corollary.

Corollary 2.2. *The loss of serving in the dark is asymptomatically $\rho \approx 1.69996$.*

Note that all proofs omitted from the main part of the paper can be found in the full paper.

2.1 Notation and terminology

Given a buffer with B bins, and a stream of balls $X = \langle X_1, X_2, \ldots X_T \rangle$, we use the following notation with respect to some strategy ALG for clearing the balls:

- Let $G_i^{\mathrm{ALG}}(X)$ be the *gain* of ALG at time i, and let $L_i^{\mathrm{ALG}}(X)$ be the *load* of the buffer at time i, namely, the number of balls in the buffer just before ALG clears some bin at time i. Notice that $L_i^{\mathrm{ALG}}(X) = \min\{L_{i-1}^{\mathrm{ALG}}(X) - G_{i-1}^{\mathrm{ALG}}(X) + X_i, B\}$.
- Let $O_i^{\mathrm{ALG}}(X)$ be the *overflow* at time i, that is, the number of balls thrown away at time i. Specifically, $O_i^{\mathrm{ALG}}(X) = \max\{0, L_{i-1}^{\mathrm{ALG}}(X) - G_{i-1}^{\mathrm{ALG}}(X) + X_i - B\}$.
- Let $G^{\mathrm{ALG}}(X) = \sum_{i=1}^{T-1} G_i^{\mathrm{ALG}}(X) + L_T^{\mathrm{ALG}}(X)$ be the *overall gain* of ALG. In particular, notice that once the stream ends, all the remaining balls in the buffer are cleared and added to the gain. Also note that we can alternatively define
$G^{\mathrm{ALG}}(X) = \sum_{i=1}^{T} X_i - \sum_{i=1}^{T} O_i^{\mathrm{ALG}}(X)$.

It is easy to see that for the optimal gain-maximizing strategy OPT,

$$G_i^{\mathrm{OPT}}(X) = \begin{cases} 1 & \text{if } L_i^{\mathrm{OPT}}(X) > 0, \\ 0 & \text{otherwise.} \end{cases}$$

Turning to our randomized process RND, we denote by $Y = \langle Y_1, \ldots, Y_T \rangle$ the random choices that the process takes during the T time steps. Specifically, $Y_i \in \{1, \ldots, B\}$ denotes the bin that is uniformly selected at time i. Now, the gain of RND at step i (conditioned on Y) is

$$G_i^{\mathrm{RND}}(X|Y) = \begin{cases} 1 & \text{if } Y_i \le L_i^{\mathrm{RND}}(X|Y), \\ 0 & \text{otherwise.} \end{cases}$$

Here and later, we assume without loss of generality that if the buffer is loaded with L balls then all these balls reside in bins $1, \ldots, L$. Similarly to before, the overall gain of the randomized process (conditioned on Y) is $G^{\mathrm{RND}}(X|Y) = \sum_{i=1}^{T-1} G_i^{\mathrm{RND}}(X|Y) + L_T^{\mathrm{RND}}(X|Y)$, and $G^{\mathrm{RND}}(X)$ is the expected value of $G_{\mathrm{RND}}(X|Y)$ over all possible choices of Y. With these definitions in mind, our goal is to determine the exact *loss of serving in the dark* defined as

$$\hat{\rho} = \max_X \hat{\rho}(X) = \max_X \frac{G^{\mathrm{OPT}}(X)}{G^{\mathrm{RND}}(X)} \,.$$

2.2 Valid ball streams

We begin by showing that it is sufficient to consider *valid* ball streams for which the optimal strategy has no overflow nor subflow. An *overflow* is a situation in which OPT cannot store all arriving balls in the buffer and therefore has to throw some of them away, while a *subflow* is a situation in which there are no balls in OPT's buffer to be cleared.

Lemma 2.3. *Given any ball stream $X = \langle X_1, \ldots, X_T \rangle$ there is a valid ball stream $X' = \langle X'_1, \ldots, X'_{T'} \rangle$ for which the optimal strategy does not have an overflow nor a subflow and $\hat{\rho}(X') \ge \hat{\rho}(X)$.*

Proof. We first prove that given a ball stream X there is a ball stream X' for which OPT does not have a subflow and $\hat{\rho}(X') \ge \hat{\rho}(X)$. Suppose X has a subflow; otherwise, taking $X' = X$ is sufficient. Let i be the index of a subflow, namely, $L_i^{\mathrm{OPT}}(X) = 0$ and $X_i = 0$. We focus on the ball stream $\langle X_1,\dots,X_{i-1},X_{i+1},\dots,X_T\rangle$. Clearly, OPT has the same gain for this stream. On the other hand, RND has at most the same expected gain. Specifically, consider any $\langle Y_1,\dots,Y_T\rangle$. If $Y_i > L_i^{\mathrm{RND}}(X|Y)$ then nothing has changed since RND did not clear a ball given X. If $Y_i \le L_i^{\mathrm{RND}}(X|Y)$, namely, RND cleared a ball and the buffer load decreased by one, then there are two possible scenarios with respect to the modified ball stream: (1) the rest of the gain series along the time is the same, and thus, the total gain decreased by one. (2) the gain at some future step increased by one, but then, the state of the randomized process is identical to its state when applied to X, and hence, the gain series continues identically and the total gain is identical. In any case, the total gain of RND may not increase as a result of the stream modification. By applying this modification step repeatedly, we can obtain a ball stream X' for which OPT does not have a subflow, and $\hat{\rho}(X') \ge \hat{\rho}(X)$.

We turn to consider the case that OPT has overflows given X. By the previous argument, we may assume that OPT does not have subflows. Since OPT does not have any subflows then each $G_i^{\mathrm{OPT}}(X) = 1$. Therefore, $G_i^{\mathrm{OPT}}(X) \ge G_i^{\mathrm{RND}}(X|Y)$, for any $Y = \langle Y_1,\dots,Y_T\rangle$ and i. This implies that each $L_i^{\mathrm{OPT}}(X) \le L_i^{\mathrm{RND}}(X|Y)$, which in turn suggests that if OPT has an overflow then also RND has an overflow. As a result, given the modified stream $\langle X_1,\dots,X_{i-1},X_i - 1,X_{i+1},\dots,X_T\rangle$, the gain of both OPT and RND does not change. Applying this modification step repeatedly, we can obtain a ball stream X' for which OPT does not have an overflow (nor a subflow) and $\hat{\rho}(X') \ge \hat{\rho}(X)$. □

The following two simple observations relating to valid ball streams will be utilized later.

Observation 2.4. *A ball stream $X = \langle X_1,\dots,X_T\rangle$ is valid if and only if $0 < \sum_{i=1}^{k} X_i - (k-1) \le B$, for any $k \in \{1,\dots,T\}$.*

Observation 2.5. *Given a valid ball stream $X = \langle X_1,\dots,X_T\rangle$, the gain of the optimal strategy is $G^{\mathrm{OPT}}(X) = \sum_{i=1}^{T} X_i$, and its load in each step k is $L_k^{\mathrm{OPT}}(X) = \sum_{i=1}^{k} X_i - (k-1)$.*

In what follows, we focus on upper bounding $\max_X \hat{\rho}(X) = \max_X G^{\mathrm{OPT}}(X)/G^{\mathrm{RND}}(X)$ under the assumption that X is valid. By Lemma 2.3, we know that this is sufficient to upper bound $\hat{\rho}$. We do not know how to do it directly, and therefore, we define a *deterministic* fractional process that will be used as a proxy for the analysis. We analyze the gain of this fractional process and prove that it is far by a factor of ρ from the gain of the optimal gain-maximizing strategy. This fractional process is designed in a natural way to correspond to the randomized process. Unfortunately, we observe that the gain of this process is not the expected gain of the randomized process, but rather dominates it. Nevertheless, we still establish that it is within a $1+\varepsilon$ factor away from the expected gain of randomized process. Combining these two results together enables us to prove the claimed $\hat{\rho}$.

2.3 Analyzing the fractional process against the optimal one

The *fractional process* is defined so it clears a ball-fraction that corresponds to the fraction of balls in the buffer. For example, if the buffer is loaded with L (fractional) balls then the fractional process clears an L/B ball-fraction from the buffer. Formally, the gain of the fractional process FRAC at time i is $G_i^{\mathrm{FRAC}}(X) = L_i^{\mathrm{FRAC}}(X)/B$, while its load is $L_i^{\mathrm{FRAC}}(X) = \min\{L_{i-1}^{\mathrm{FRAC}}(X)\cdot(1-1/B) + X_i, B\}$. Let $\rho(X) = G^{\mathrm{OPT}}(X)/G^{\mathrm{FRAC}}(X)$ denote the gain ratio between OPT and FRAC on X. In the remainder of this subsection, we establish the following theorem.

Theorem 2.6. *$\rho(X) \le \rho$ for any valid ball stream X.*

Proof. Consider a bounded length ball stream. It is clear that there is a ball stream with worse gain ratio for this length as the number of relevant ball streams is finite. We next characterize the bounded length ball stream with the worst gain ratio between OPT and FRAC. Our characterization defines the stream uniquely (i.e., depending only on its length). Subsequently, we compute the exact gain ratio for this stream. The next lemma identifies an important property of the stream under consideration.

Lemma 2.7. *Given a valid ball stream X with a maximal gain ratio $\rho(X)$, we may assume that if $L_i^{\text{FRAC}}(X) = B$ then also $L_i^{\text{OPT}}(X) = B$.*

Corollary 2.8. *Given a valid ball stream X with a maximal gain ratio $\rho(X)$, we may assume that if $L_i^{\text{OPT}}(X) < B$ then $L_i^{\text{FRAC}}(X) < B$, and thus, $O_i^{\text{FRAC}}(X) = 0$.*

For the purpose of characterizing the ball stream with the worst gain ratio between OPT and FRAC, we subsequently focus on analyzing valid ball streams that maximize the number of balls thrown away by FRAC, namely, the total sum of the overflows. By Corollary 2.8, we know that when an overflow of FRAC occurs then OPT must be full. We define a *block* as a substream between two consecutive occasions where FRAC is full. Let $block(X,k,d)$ be a block that starts on step k with a length of d. Note that replacing one block with another block does not influence the load states of FRAC before the beginning or after the end of the block. Similarly to before, we say that a block is *valid* if OPT does not have an overflow nor a subflow in that block. The following observation summarizes the properties of a $block(X,k,d)$.

Observation 2.9. *For a $block(X,k,d)$:*

1. $L_k^{\text{FRAC}}(X) = B$, $L_k^{\text{OPT}}(X) = B$,
   $L_{k+d}^{\text{FRAC}}(X) = B$, *and* $L_{k+d}^{\text{OPT}}(X) = B$.
2. $\sum_{i=1}^{k} X_i = B + (k-1)$, *and*
   $\sum_{i=k+1}^{k+d} X_i = d$.
3. $L_{k+j}^{\text{OPT}}(X) = B - j + \sum_{i=k+1}^{k+j} X_i$,
   for $0 \le j \le d$.

Note that the only overflow of FRAC in a block may occur in the last step of the block. Accordingly, and in conjunction with the definition of the fractional process, we observe the following.

Observation 2.10. *For a $block(X,k,d)$:*

1. $L_{k+j}^{\text{FRAC}}(X) = B \cdot (1-1/B)^j + \sum_{i=1}^{j} X_{k+i} \cdot (1-1/B)^{j-i}$, *for* $0 \le j < d$.
2. $O_{k+d}^{\text{FRAC}}(X) = B \cdot (1-1/B)^d + \sum_{i=1}^{d} X_{k+i} \cdot (1-1/B)^{d-i} - B$.

In order to characterize a valid block with a maximum overflow of FRAC, we first define a *forward shift* procedure. This procedure simply moves a ball inside the block from one step to the consecutive step. We next prove that the overflow of FRAC strictly increases after a forward shift. This implies that in a valid block with a maximum overflow, one may not apply any forward shift while keeping the block valid. This characterizes the block with the maximum overflow uniquely (given its length). Formally, a $fshift(X,k,j)$ is defined within a $block(X,k,d)$ such that $0 < j < d$ and $X_{k+j} > 0$, and results in a ball stream X' in which $X'_i = X_i$ for all $i \neq k+j, k+j+1$, $X'_{k+j} = X_{k+j} - 1$, and $X'_{k+j+1} = X_{k+j+1} + 1$. We say that a forward shift is *admissible* if it keeps the validity of the block. The following observation identifies a condition for the validity of a block after a forward shift.

Observation 2.11. *Given a valid block(X,k,d), the block continues to be valid after applying a $fshift(X,k,j)$ if $L^{\text{OPT}}_{k+j}(X) > 1$.*

Notice that we do not need to require that $L^{\text{OPT}}_{k+j+1}(X) < B$ in the above observation. This follows since if $L^{\text{OPT}}_{k+j}(X) > 1$ then $L^{\text{OPT}}_{k+j+1}(X') = L^{\text{OPT}}_{k+j+1}(X)$.

Lemma 2.12. *Applying a $fshift(X,k,j)$ increases the overflow of the block(X,k,d) in* FRAC.

Proof. Using Observation 2.10(2), if we apply a forward shift at step $0 < j < d$ then

$$\begin{aligned}
&O^{\text{FRAC}}_{k+d}(X') = \\
&= O^{\text{FRAC}}_{k+d}(X) + \left(1-\frac{1}{B}\right)^{d-j-1} - \left(1-\frac{1}{B}\right)^{d-j} \\
&> O^{\text{FRAC}}_{k+d}(X) \ .
\end{aligned}$$

□

Corollary 2.13. *Given a valid ball stream X, if we apply an admissible forward shift within some block of X then we get a valid ball stream X' for which $\rho(X') > \rho(X)$.*

As a result of the last lemma, we can now characterize the worst valid block in terms of overflow for every block length.

Lemma 2.14. *Given a valid ball stream X with a maximal gain ratio $\rho(X)$ that contains some block(X,k,d) then*

- *if $d \le B$ then $\langle X_{k+1},\ldots,X_{k+d}\rangle = \langle 0,\ldots,0,d\rangle$, where the number of 0's is $d-1$.*
- *if $d > B$ then $\langle X_{k+1},\ldots,X_{k+d}\rangle = \langle 0,\ldots,0,1,\ldots,1,B\rangle$, where the number of 0's is $B-1$ and the number of 1's is $d-B$.*

Proof. Assume for the purpose of contradiction that a block with different structure participates in the ball stream with the worst gain ratio. We consider the following two cases, and demonstrate that in each case, one can modify the ball stream and attain a worse gain ratio.

Case I: $d \le B$. Notice that there exists a $j = \min\{i : 0 < i < d \text{ and } X_{k+i} > 0\}$. By Observation 2.9(3), we know that $L^{\text{OPT}}_{k+j}(X,k) = B - j + X_{k+j} > 1$, and therefore, $fshift(X,k,j)$ is admissible by Observation 2.11. Together with Corollary 2.13, we get a contradiction.

Case II: $d > B$. If there is a $j = \min\{i : 0 < i < B \text{ and } X_{k+i} > 0\}$ then getting a contradiction is similar to the previous case. Otherwise, there must be $j = \min\{i : B \le i < d \text{ and } X_{k+i} > 1\}$. Notice that $\sum_{i=1}^{j-1} X_{k+i} = j - B$, and thus, $L^{\text{OPT}}_{k+j}(X) = X_{k+j}$ using Observation 2.9(3). This implies that indeed $X_{k+j} > 1$ since otherwise OPT has a subflow and the ball stream is not valid. Now, using Observation 2.11, a $fshift(X,k,j)$ is admissible, and then, by Corollary 2.13, we get a contradiction. □

We are now ready to complete the proof of Theorem 2.6. We first analyze the maximal gain ratio for a valid $block(X,k,d)$. Using Lemma 2.14 and Observation 2.10(2), one can derive that if $d \le B$ then

$$O^{\text{FRAC}}_{k+d}(X) = B \cdot \left(1-\frac{1}{B}\right)^{d} + d - B \ ,$$

and if $d > B$ then

$$\begin{aligned}
&O^{\text{FRAC}}_{k+d}(X) = \\
&= B \cdot \left(1 - \frac{1}{B}\right)^d + \sum_{i=1}^{d-B} \left(1 - \frac{1}{B}\right)^i - B \\
&= B \cdot \left(1 - \frac{1}{B}\right)^d + \\
&\left(1 - \frac{1}{B}\right) \cdot B \cdot \left(1 - \left(1 - \frac{1}{B}\right)^{d-B}\right) .
\end{aligned}$$

Recall that the gain of OPT in a valid block of length d is d. So the worst gain ratio for a valid $block(X,k,d)$ is $d/(d - O^{\text{FRAC}}_{k+d}(X))$. If we substitute the formulas above in the last expression then the worst gain ratio happens when $d > B$. Specifically, we get that

$$\begin{aligned}
&\frac{d}{d - O^{\text{FRAC}}_{k+d}(X)} = \\
&\frac{d}{d - B\left(1 - \frac{1}{B}\right)^d + \left(1 - \frac{1}{B}\right) B \left(1 - \left(1 - \frac{1}{B}\right)^{d-B}\right)}
\end{aligned}$$

The last expression tends to

$$\frac{\alpha e^{\alpha}}{(\alpha - 1)e^{\alpha} + e - 1} ,$$

for a sufficiently large B by substituting $\alpha = d/B$. One can analytically verify that the worst gain ratio is attained for $\alpha \approx 1.429$, and its value is $\rho \approx 1.69996$.

We turn to bound $\rho(X)$ for any valid ball stream X. Given a ball stream X, we divide it into three parts: the first part consists of all ball arrivals until the first overflow of FRAC, the second part consists all the complete (valid) blocks defined by FRAC, and the last part consists of the remainder of the ball stream. Similarly to Lemma 2.14, one can demonstrate that the first part of X must have the structure $\langle 1, 1, \ldots 1, B \rangle$ in order to maximize the number of balls thrown away. In particular, the number of balls that are thrown away in this overflow can be easily shown to be $O^{\text{FRAC}}_{d'}(X) = \sum_{i=1}^{d'-1} (1 - 1/B)^i = (1 - 1/B) \cdot B \cdot (1 - (1 - 1/B)^{d'-1})$, where d' is the length of the first part. Regarding the last part of the stream, we may assume that it is empty since we know that FRAC has no overflows within it, and thus, it does not lose any gain with respect to OPT. This implies that after the end of the last block, the gain of FRAC and OPT is exactly the content of their buffer, namely, B balls. To sum up, the gain of FRAC in the first and last parts is $d' - (1 - 1/B) \cdot B \cdot (1 - (1 - 1/B)^{d'-1}) + B$, while the gain of OPT is $d' + B$. One can analytically verify that the worst gain ratio happens when $d' \approx 1.146B$, and its value is roughly 1.466. Now, if there are K blocks in the ball stream with parameters k_i, d_i to indicate the starting index and length of each block i, respectively, then we get that

$$
\begin{aligned}
&G^{\mathrm{OPT}}(X) = d' + B + \sum_{i=1}^{K} d_i \ , \\
&G^{\mathrm{FRAC}}(X) = \\
&d' - O_{d'}^{\mathrm{FRAC}}(X) + B + \sum_{i=1}^{K} \left(d_i - O_{k_i+d_i}^{\mathrm{FRAC}}(X)\right) \\
&\rho(X) = \frac{G^{\mathrm{OPT}}(X)}{G^{\mathrm{FRAC}}(X)} \\
&\leq \max_i \left\{ \frac{d' + B}{d' - O_{d'}^{\mathrm{FRAC}}(X) + B}, \frac{d_i}{d_i - O_{k_i+d_i}^{\mathrm{FRAC}}(X)} \right\} \\
&\leq \rho \ .
\end{aligned}
$$

□

2.4 Analyzing the randomized process against the fractional one

In what follows, we make a connection between the expected gain of our randomized process and the gain of the fractional one. For this purpose, we first generalize the definition of a ball stream X from an integer one, in which $X_i \in \mathbb{N}_+$, to a fractional one, in which $X_i \in \mathbb{R}_+$. For our random process, X_i indicates that it gets $\lfloor X_i \rfloor$ balls deterministically and an additional ball with probability $X_i - \lfloor X_i \rfloor$ at step i. Note that the expectation of the number of arriving balls is X_i. On the other hand, a deterministic process (e.g., the fractional process) gets exactly X_i balls. Clearly, if we bound the worst gain ratio between our randomized process and the fractional one with respect to such fractional ball streams, it immediately implies the same bound for integral ball streams.

We begin by defining a *modified fractional process*. This modified process has a smaller buffer size than FRAC, but clears the same ball-fraction at each step as FRAC. Specifically, FRAC$'$ has a buffer with size of $(1-\varepsilon)B$, where $\varepsilon \geq 1/B$. The gain of FRAC$'$ at time i is $G_i^{\mathrm{FRAC}'}(X) = L_i^{\mathrm{FRAC}'}(X)/B$, while its load is $L_i^{\mathrm{FRAC}'}(X) = \min\{L_{i-1}^{\mathrm{FRAC}'}(X) \cdot (1-1/B) + X_i, (1-\varepsilon)B\}$. We next bound the ratio between the gains of FRAC and FRAC$'$. First, we make the following simple monotonicity observation regarding all the processes under consideration (i.e., OPT, RND, FRAC, and FRAC$'$).

Observation 2.15. *Given two fractional balls streams X and X', if $X'_i \leq X_i$ in any step i then $G^{\mathrm{ALG}}(X') \leq G^{\mathrm{ALG}}(X)$, where* ALG *can be any of the processes under consideration.*

Theorem 2.16. $G^{\mathrm{FRAC}'}(X) \geq (1-\varepsilon)G^{\mathrm{FRAC}}(X)$ *for any valid ball stream X.*

We now turn to bound the difference between the gains of FRAC$'$ and RND.

Theorem 2.17. *For any ball stream X of length T,*
$G^{\mathrm{RND}}(X) \geq G^{\mathrm{FRAC}'}(X) - T\varepsilon$.

Proof. We first make several observations that will be utilized later.

Observation 2.18. *Given a ball stream X, if* FRAC$'$ *never has an overflow given X then*

$$L_k^{\mathrm{FRAC}'}(X) = \sum_{i=1}^{k} X_i \cdot \left(1 - \frac{1}{B}\right)^{k-i} .$$

Observation 2.19. *Consider the following process manipulations:*

- *Suppose at some step of* FRAC′ *we remove some positive fraction of balls from its load and add them to its total gain. If we continue with the process from the resulting state then the total gain of* FRAC′ *can not decrease (with respect to the process without the modification).*
- *Suppose that at some step of* RND *we remove some positive fraction of balls from its load and throw them away. If we continue with the process then the total gain of* RND *can not increase (with respect to the process without the modification).*

Using similar reasonings to before, one can demonstrate that it is sufficient to prove this theorem for ball streams X for which FRAC′ has no overflows; otherwise, we can modify X to X' by removing ball-fractions that correspond to overflows of FRAC′, while making the gain ratio between FRAC′ and RND for X' no better. As a result, we can assume that $L_k^{\mathrm{FRAC}'}(X) = \sum_{i=1}^k X_i \cdot (1-1/B)^{k-i}$ by Observation 2.18. Notice that $L_k^{\mathrm{RND}}(X)$ is a random variable, and observe that these loads form a Markov chain. Specifically, the conditional expectation of the load of RND in any step $k+1$ depends only on the preceding load $L_k^{\mathrm{RND}}(X)$. Formally,

$$\begin{aligned}&\mathbb{E}\left(L_{k+1}^{\mathrm{RND}}(X) | L_1^{\mathrm{RND}}(X), \ldots, L_k^{\mathrm{RND}}(X)\right) = \\ &\mathbb{E}\left(L_{k+1}^{\mathrm{RND}}(X) | L_k^{\mathrm{RND}}(X)\right) \ .\end{aligned}$$

Lemma 2.20. *The conditional expectation of the load satisfies that*

$$\begin{aligned}&\mathbb{E}\left(L_{k+1}^{\mathrm{RND}}(X) | L_k^{\mathrm{RND}}(X)\right) \leq \\ &\min\left\{L_k^{\mathrm{RND}}(X) \cdot \left(1 - \frac{1}{B}\right) + X_{k+1}, B\right\} \ .\end{aligned}$$

We now define a process $Z_k(X)$ which is the absolute difference between $L_k^{\mathrm{RND}}(X)$ and $L_k^{\mathrm{FRAC}'}(X)$. Specifically, since it is sufficient to consider the case that FRAC′ does not have an overflow on X then we know by Observation 2.18 that

$$\begin{aligned} Z_k(X) &= \left| L_k^{\mathrm{RND}}(X) - L_k^{\mathrm{FRAC}'}(X) \right| \\ &= \left| L_k^{\mathrm{RND}}(X) - \sum_{i=1}^{k} X_i \cdot \left(1 - \frac{1}{B}\right)^{k-i} \right| \ . \end{aligned}$$

We prove that this process is a *super-martingale*, that is, $\mathbb{E}[Z_{k+1}(X) | Z_1(X), \ldots, Z_k(X)] \leq Z_k(X)$. Notice that $\mathbb{E}[Z_{k+1}(X) | Z_1(X), \ldots, Z_k(X)] = \mathbb{E}[Z_{k+1}(X) | L_k^{\mathrm{RND}}(X)]$ by the definition of $Z_k(X)$ and the property stated in Equation 1. We consider two cases. The first is when $L_k^{\mathrm{RND}}(X) + X_{k+1} \leq B$, namely, RND surely has no overflow at step $k+1$. Then,

$$\begin{aligned}
&\mathbb{E}[Z_{k+1}(X)|L_k^{\mathrm{RND}}(X)] = \\
&= \left| L_k^{\mathrm{RND}}(X)\left(1-\frac{1}{B}\right) + X_{k+1} - \sum_{i=1}^{k+1} X_i \left(1-\frac{1}{B}\right)^{k+1-i} \right| \\
&= \left| L_k^{\mathrm{RND}}(X)\left(1-\frac{1}{B}\right) - \sum_{i=1}^{k} X_i \left(1-\frac{1}{B}\right)^{k+1-i} \right| \\
&= \left(1-\frac{1}{B}\right)\left| L_k^{\mathrm{RND}}(X) - \sum_{i=1}^{k} X_i \left(1-\frac{1}{B}\right)^{k-i} \right| \\
&= \left(1-\frac{1}{B}\right) Z_k(X) \le Z_k(X) ,
\end{aligned}$$

where the first equality follows from Lemma 2.20 and our assumption that $L_k^{\mathrm{RND}}(X)+X_{k+1} \le B$. The second case is when $L_k^{\mathrm{RND}}(X)+X_{k+1} > B$. The proof is similar to first case and appears in the full version.

Now, notice that $|Z_{k+1}(X)-Z_k(X)| \le 2$. In particular, the difference is bounded by 2 since (1) we consider fractional ball streams, and hence, the fractional process may get an extra (fraction of a) ball that the randomized process may not obtain, and (2) each process may clear at most one ball. Accordingly, we can apply Azuma's inequality [7] to this super martingale process, namely,

$$\mathbb{P}\left(Z_{k+N}(X)-Z_k(X) \ge t\right) \le \exp\left(\frac{-t^2}{8N}\right) .$$

Observe that by choosing $t = \varepsilon B$, we get that $\mathbb{P}(Z_{k+i}(X)-Z_k(X) \ge \varepsilon B) \le \exp(-\varepsilon^2 B/8)$ for any $i \le B$. Using union bound, we conclude that $\mathbb{P}(\exists i \le B : Z_{k+n}(X)-Z_k(X) \ge \varepsilon B) \le B \cdot \exp(-\varepsilon^2 B/8)$. The following lemma summarizes the consequences of this last finding.

Lemma 2.21. *Assume that $B = \Omega(1/\varepsilon^3)$. If $Z_k(X) = 0$ then the maximum load difference between* FRAC′ *and* RND *in each of the next B steps is at most εB with high probability. In particular, this implies that* RND *has no overflows in those steps with high probability.*

We next employ the last lemma to complete the proof of the theorem. Given a ball stream of length T, partition it to T/B equal parts of length B. Intuitively, we would like to apply the above lemma to each of the parts of the partition, and attain that the gain of FRAC′ and RND is close to one another. Specifically, if we knew that in the beginning of each part the corresponding random variable $Z_k(X) = 0$ then we would get that the load difference in each part is at most εB with high probability by applying the above lemma. In particular, this would imply that RND has no overflow in any of the parts with high probability. As a result, we get that the expected gain of RND is essentially equal to the gain of FRAC′, completing the proof. Unfortunately, this is not the case since the loads in the beginning of each part may not be the same.

We use the modifications exhibited in Observation 2.19 to manipulate the loads of FRAC′ and RND to be equal after each part. We inductively assume that in the beginning of a part under consideration the loads of RND and FRAC′ are the same. Then, we apply Lemma 2.21 that implies that load difference between FRAC′ and RND in the next B steps is at most εB with high probability. This implies that RND has no overflow in that part with high probability. At the end of each part, we manipulate the processes as described in Observation 2.19 by either decreasing the load of FRAC′ (to that of RND) or decreasing the load of RND (to that of FRAC′). In any case, this manipulation may decrease the gain of RND by at most εB.

This enables us to apply the inductive step while having an additional gain difference loss of at most εB in each part. In conclusion, by summing over all the parts, we attain that $G^{\mathrm{RND}}(X) \geq G^{\mathrm{FRAC}'}(X) - T/B \cdot \varepsilon B = G^{\mathrm{FRAC}'}(X) - T\varepsilon$. $\square$

2.5 Putting everything together

We are now ready to complete the proof of main theorem of the paper.

Proof of Theorem 2.1. We know from Lemma 2.3 that it is sufficient to consider valid ball streams X for the purpose of bounding $\hat{\rho} = \max_X G^{\mathrm{OPT}}(X)/G^{\mathrm{RND}}(X)$. By Observation 2.4, we know that $G^{\mathrm{OPT}}(X) \geq T$. Now, notice that

$$
\begin{aligned}
G^{\mathrm{RND}}(X) &\geq G^{\mathrm{FRAC}'}(X) - T\varepsilon \\
&\geq (1-\varepsilon)G^{\mathrm{FRAC}}(X) - T\varepsilon \\
&\geq (1-\varepsilon)\frac{G^{\mathrm{OPT}}(X)}{\rho} - T\varepsilon \\
&\geq (1-\varepsilon)\frac{G^{\mathrm{OPT}}(X)}{\rho} - G^{\mathrm{OPT}}(X)\varepsilon \\
&\geq G^{\mathrm{OPT}}(X)\left(\frac{1-\varepsilon}{\rho} - \varepsilon\right) ,
\end{aligned}
$$

where the first inequality is due to Theorem 2.17, the second inequality is by Theorem 2.16, and the third inequality holds by Theorem 2.6. Hence, $\hat{\rho} \leq \rho + O(\varepsilon)$. $\square$

2.6 Tightness of the analysis

In the previous subsections, we have established that the loss of serving in the dark $\rho + \varepsilon$. A natural question to ask is whether one can find a better bound for this ratio between the expected gain of the randomized process and that of the optimal gain-maximizing strategy. It turns out that our analysis is tight up to a difference of $O(\varepsilon)$, i.e., the bound that we have obtained is asymptotically the exact loss of serving in the dark. This finding can be proved by noticing the following:

(1) The value of the ratio $\rho(X) = G^{\mathrm{OPT}}(X)/G^{\mathrm{FRAC}}(X)$ presented in Theorem 2.6 is essentially tight. This can be proved by considering the $block(X,k,d)$ with worst gain ratio (i.e., a block with $d \approx 1.429B$ whose gain ratio is ρ), and creating a ball stream X in which this block appears over and over again. It is easy to validate that $\rho(X) \geq \rho - \varepsilon$, where ε depends on the number of blocks and B, as the gain of the processes under consideration in the non-block parts of the stream is negligible.

(2) The gain of the fractional process $G^{\mathrm{FRAC}}(X)$ dominates the expected gain of the randomized process $G^{\mathrm{RND}}(X)$ for the above-mentioned ball stream X. Specifically, one should notice that the gain of the fractional process in each $block(X,k,d)$ is equal to the expected gain of the randomized process as long as there are no overflows by any of the processes. The fractional process has an overflow only at the last step of the block, while it is not hard to validate that the randomized process may have overflows even before that. In such cases, the gain of the randomized process becomes smaller than that of the fractional one.

3 Applications:

3.1 Application 1: Value-oblivious packets scheduling.

We show that the competitiveness of value-oblivious transmission algorithm, that described in the introduction, can be analyzed by the stochastic process described. The value-oblivious transmission algorithm selects uniformly at random a packet to transmit among the packets in the buffer. We compare it to an unconstrained optimal algorithm which transmits the highest valued packet in the buffer. We claim that it is sufficient to analyze our random transmission algorithm with respect to packets whose values are restricted to the set $\{0,1\}$. This follows from the zero-one principle below, while observing that our algorithm is indeed a comparison-based algorithm. We note that an algorithm is *comparison-based* if all its decisions are based on the relative order between the values of the bids with no regard to their actual values.

Theorem 3.1. (The zero-one principle [6]) *Let $\mathscr{A}$ be a (deterministic or randomized) comparison-based algorithm for the packet scheduling problem. $\mathscr{A}$ is a c-approximation algorithm if and only if A achieves c-approximation for all packet streams whose values are restricted to* $\{0,1\}$ *for every possible way of breaking ties between equal values.*

Observe that the expected gain of our algorithm in each transmission is the number of 1-valued packets divided by the overall number of packets. Hence, for the sake of analysis, we may assume that there are exactly B packets in every transmission. Moreover, since our algorithm keeps the packets with the highest values at any step, we may assume that the input stream is $X = \langle X_1, X_2, \dots X_T \rangle$, where X_i represents the number of 1-valued packets that arrived before transmission i but after transmission $i-1$. One can easily validate that there is a correspondence between the competitive ratio of our randomized transmission algorithm and the loss of serving in the dark ratio. As a result, we obtain the following theorem.

Theorem 3.2. *The competitive ratio of the randomized transmission algorithm is $\rho + \varepsilon$ for $B = \Omega(1/\varepsilon^3)$.*

3.2 Application 2: Prompt mechanisms for bounded capacity auctions

We consider the problem of developing *prompt truthful mechanisms* for *periodic bounded capacity auctions*. In the underlying scenario, there is a single item with unlimited supply, and a stream of buyers, arriving dynamically over time, each of which is interested in purchasing one instance of the item. An instance of the item is offered for sale in a bounded capacity auction *periodically*, that is, over and over again. A *bounded capacity auction* is a single-item auction in which the number of participating bidders is bounded by a fixed $B \in \mathbb{N}_+$, e.g., when the auction room has a limited size. Since the auction has bounded capacity, it is common that the auction cannot accommodate all the buyers. In such a case, some of the buyers must be indefinitely rejected. These buyers cannot participate in any auction after their rejection. We remark that the auction events continue even if the stream of buyers ceases. This implies that one can sell items to all pending buyers once the stream ends. Our goal is to design prompt truthful mechanisms that maximizes the social welfare, i.e., the sum of the values of the buyers that purchase an item. In a *prompt mechanism* [15], a buyer that purchases an item learns her payment immediately after she wins the auction.

One practical motivation for studying the above problem relates to buffer management issues arising in context of network devices such as switches and routers. In this application domain, there is an incoming stream of (strategic) packets with intrinsic values, and there is a network device that can accommodate a bounded number of packets at any time. The device can transmit one packet in each time-slot. The goal is to design a truthful mechanism maximizing the overall value of transmitted packets, while charging any packet for the given service once it is transmitted.

In the above scenario, the private information of each buyer is her positive value for purchasing an item. Each buyer declares her bid for purchasing the item once she arrives. This model falls within the scope of online single-parameter setting (see, e.g., [35]). It is well-known that developing a truthful mechanism in this setting is roughly equivalent to designing a monotone algorithm. An algorithm is monotone if a winning buyer, namely, a buyer that purchases an item, remains a winner if she raises her bid. An algorithm for our problem can be logically split into two parts: (1) Once a buyer arrives, the algorithm needs to decide whether to reject that buyer or to keep her active. Note that there must be at most B active buyers at any time, and therefore, it may happen that in order to keep a new buyer active, the algorithm has to reject another active buyer. (2) When an auction occurs, the algorithm needs to decide which of the active bidders wins the item.

Example 1: the Greedy algorithm. The greedy algorithm keeps the buyers with highest bids at any step. Specifically, once a new buyer arrives, if there are less than B active buyers then the new buyer is kept as active; if the bid of the new buyer is higher than the minimal bid of the current B active buyers then she is kept as active and the active buyer with minimal bid is rejected; otherwise, she is rejected. On the other hand, when an auction occurs, the item is sold to an active buyer with a maximum bid.

One can easily verify that this simple algorithm achieves optimal outcome and that it is monotone. Therefore, the algorithm can underlie an optimal truthful mechanism. However, this algorithm does not support prompt payments. In particular, there are simple input instances for which a buyer may have to wait indefinitely to learn her payment.

Example 2: the FIFO algorithm. This algorithm activates and rejects buyers in an identical way to the greedy algorithm, that is, it keeps the buyers with highest bids at any step. However, when an auction occurs, the item is sold to the active buyer that arrived earliest.

One can easily verify that this algorithm is monotone. Moreover, it is 2-competitive [24], and supports prompt payments. Specifically, it is easy to prove that the payment of each winning buyer only depends on the bids of buyers that arrived before she won. In fact, computing these prices is simple. Essentially, the price that a winning buyer has to pay is the maximum over all the bids of bidders rejected between the arrival of that buyer and the time she won the auction.

3.2.1 Our algorithm

We suggest a natural *randomized selection* algorithm. Our algorithm activates and rejects buyers in an identical way to the greedy algorithm, that is, it keeps the buyers with highest bids at any step. When an auction occurs, one buyer is selected uniformly at random from all active buyers, and the item is sold to her. We prove that this algorithm is universally truthful (i.e. it is truthful for any possible coins flip), supports prompt payments, and achieves an expected competitive ratio that is strictly better than 2. Recall that $\rho \approx 1.69996$.

Theorem 3.3. *The randomized selection algorithm can underlie a prompt truthful mechanism whose competitive ratio is $\rho + \varepsilon$ for $B = \Omega(1/\varepsilon^3)$.*

As a side note, we remark that one can utilize standard techniques and prove that no prompt deterministic truthful mechanism can attain an optimal outcome for this problem. This implies a separation between prompt and truthful mechanisms as an optimal truthful mechanism is achievable. To the best of our knowledge, this kind of separation has not been exhibited before.

3.2.2 Related work

One problem which is closely related to ours is the *dynamic auction with expiring items* problem. In the underlying scenario, there is a single item with unlimited supply, and there is a stream of buyers arriving and departing dynamically over time. An instance of the item is offered for sale in a single-item auction over and over again. Each buyer is interested in purchasing one instance of the item between her arrival and departure times. The objective is to design a truthful mechanism that maximizes the social welfare, that is, the sum of the values of the buyers that purchase an item within their time window. We note that there is no bound to the number of pending buyers in this model, but rather, each buyer has an individual departure time.

The dynamic auction with expiring items problem was introduced by Hajiaghayi et al. [21]. They presented a truthful 2-competitive mechanism (see also [11]). They also established that this algorithm is best possible. Cole, Dobzinski, and Fleischer [15] concentrated on developing prompt mechanisms for this problem. They developed a different truthful 2-competitive mechanism which is also prompt. Lavi and Nisan [28] considered the scenario in which buyers may misreport their arrival and departure times, and proved that it is impossible to attain bounded competitive ratio in this case. Neglecting all strategic considerations, the dynamic auction with expiring items problem is equivalent to online scheduling of unit-length jobs on a single machine to maximize weighted throughput. The best known deterministic online algorithm for this problem has a competitive ratio of about 1.828 [18] (see also [30]), while it is known that no deterministic online algorithm can achieve a competitive ratio better than $\phi \approx 1.618$ [20, 13, 4]. Turning to the randomized setting, the best online algorithm attains a ratio of $e/(e-1)$ [9, 12], while no randomized online algorithm can attain a ratio better than 1.25 [13]. Some additional papers studying this model and other variants are [3, 8, 24, 1, 25, 29, 36, 17, 14, 19, 10].

3.2.3 Preliminaries

We present the notion of *monotonicity* and describe a characterization that links monotone algorithms with truthful mechanisms. Note that the illustrated terms are presented in the context of the problem under consideration, and thus, the keen reader may refer to [34, 35] for a more comprehensive overview of the underlying concepts. We later formalize the notion of *promptness* [15].

Definition 1. *An online algorithm $\mathscr{A}$ is said to be* monotone *with respect to the bid of a buyer if it satisfies the following property: if algorithm $\mathscr{A}$ selects a buyer as a winner when her bid is v then it selects that buyer as a winner when her bid is $\tilde{v}$, where $\tilde{v} \geq v$, and the bids of all the other buyers are fixed.*

Theorem 3.4. *If online algorithm $\mathscr{A}$ is monotone with respect to the bid of every buyer then there exists a corresponding truthful mechanism which can be efficiently computed using algorithm $\mathscr{A}$.*

Without delving too deeply into formalities, a mechanism is a pair consisting of an allocation algorithm and a payments scheme. A mechanism is called *truthful* if its payments motivate truthful behavior of the buyers, that is, no buyer has incentive to be dishonest when placing her bid. It is well-known that in a single-parameter setting, the payments of each winning buyer must be equal to her *critical value*, namely, the minimum value she could bid and still win. On the other hand, the payments of all losing buyer are strictly zero.

Definition 2. *An online mechanism is* prompt *if a buyer that wins an item learns his payment immediately after winning the item; a mechanism is* tardy *otherwise.*

Notice that by the discussion above, a mechanism is prompt if and only if the critical value of a winning buyer can be calculated by the time she wins an item. In particular, the critical value of a winning buyer should only depend on the bids of the buyers that arrived before she won.

3.2.4 Analysis of the algorithm

We begin by proving that the randomized selection algorithm is truthful and that it supports prompt payments. Later on, we analyze the competitive ratio of the algorithm.

Lemma 3.5. *The randomized selection algorithm is monotone and supports prompt payments.*

Proof. Consider two input instances that are identical with the exception that in the first instance the bid of buyer i is v and in the second instance her bid is $\tilde{v}$, where $\tilde{v} \geq v$. For the sake of monotonicity, we need to prove that if our algorithm selects i as a winner when her bid is v then it also selects her as a winner in the latter instance. This clearly happens. Specifically, notice that buyer i is kept active when she arrives since $\tilde{v} \geq v$. Moreover, she cannot be rejected by buyers arriving later since she was not rejected when her bid was v. This implies that she must win the same auction in the latter instance.

As for the promptness of the corresponding mechanism, observe that the selection of a buyer as a winner only depends on the bids provided by buyers arriving before she wins. In particular, a buyer is kept active as long as her bid is sufficiently large with respect to the other buyers, and the selection of the winner in each auction is determined by the random selection process which is independent of the bids of the buyers. Thus, we can calculate the payment of a winning buyer immediately after she wins the item. □

We now turn to analyze the competitive ratio of our algorithm. By the same arguments described in Section 3.1 this ratio is equivalent to the loss of serving in the dark ratio presented in Section 2. As a result, we obtain the following lemma:

Theorem 3.6. *The competitive ratio of the randomized selection algorithm is $\rho + \varepsilon$ for $B = \Omega(1/\varepsilon^3)$.*

References

[1] William Aiello, Yishay Mansour, S. Rajagopolan, and Adi Rosén. Competitive queue policies for differentiated services. *J. Algorithms*, 55(2):113–141, 2005.

[2] Noga Alon and Joel H. Spencer. *The Probabilistic Method*. Wiley, New York, second edition, 2000.

[3] Nir Andelman and Yishay Mansour. Competitive management of non-preemptive queues with multiple values. In *Proceedings 17th International Conference on Distributed Computing*, pages 166–180, 2003.

[4] Nir Andelman, Yishay Mansour, and An Zhu. Competitive queueing policies for qos switches. In *Proceedings 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 761–770, 2003.

[5] Yossi Azar, Andrei Z. Broder, Anna R. Karlin, and Eli Upfal. Balanced allocations. *SIAM J. Comput.*, 29(1):180–200, 1999.

[6] Yossi Azar and Yossi Richter. The zero-one principle for switching networks. In *Proceedings 36th Annual ACM Symposium on Theory of Computing*, pages 64–71, 2004.

[7] Kazuoki Azuma. Weighted sums of certain dependent random variables. *Tohoku Math. J.*, 19(3):357–367, 1967.

[8] Nikhil Bansal, Lisa Fleischer, Tracy Kimbrel, Mohammad Mahdian, Baruch Schieber, and Maxim Sviridenko. Further improvements in competitive guarantees for qos buffering. In *Proceedings 31st International Colloquium on Automata, Languages and Programming*, pages 196–207, 2004.

[9] Yair Bartal, Francis Y. L. Chin, Marek Chrobak, Stanley P. Y. Fung, Wojciech Jawor, Ron Lavi, Jiri Sgall, and Tomás Tichý. Online competitive algorithms for maximizing weighted throughput of unit jobs. In *Proceedings 21st Annual Symposium on Theoretical Aspects of Computer Science*, pages 187–198, 2004.

[10] Marcin Bienkowski, Marek Chrobak, Christoph Dürr, Mathilde Hurand, Artur Jez, Lukasz Jez, and Grzegorz Stachowiak. Collecting weighted items from a dynamic queue. In *Proceedings 20th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1126–1135, 2009.

[11] Chandra Chekuri and Iftah Gamzu. Truthful mechanisms via greedy iterative packing. In *Proceedings 12th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, pages 56–69, 2009.

[12] Francis Y. L. Chin, Marek Chrobak, Stanley P. Y. Fung, Wojciech Jawor, Jiri Sgall, and Tomás Tichý. Online competitive algorithms for maximizing weighted throughput of unit jobs. *J. Discrete Algorithms*, 4(2):255–276, 2006.

[13] Francis Y. L. Chin and Stanley P. Y. Fung. Online scheduling with partial job values: Does timesharing or randomization help? *Algorithmica*, 37(3):149–164, 2003.

[14] Marek Chrobak, Wojciech Jawor, Jiri Sgall, and Tomás Tichý. Improved online algorithms for buffer management in qos switches. *ACM Transactions on Algorithms*, 3(4), 2007.

[15] Richard Cole, Shahar Dobzinski, and Lisa Fleischer. Prompt mechanisms for online auctions. In *Proceedings 1st International Symposium on Algorithmic Game Theory*, pages 170–181, 2008.

[16] Devdatt P. Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, 2009.

[17] Matthias Englert and Matthias Westermann. Lower and upper bounds on fifo buffer management in qos switches. In *Proceedings 14th Annual European Symposium on Algorithms*, pages 352–363, 2006.

[18] Matthias Englert and Matthias Westermann. Considering suppressed packets improves buffer management in qos switches. In *Proceedings 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 209–218, 2007.

[19] Amos Fiat, Yishay Mansour, and Uri Nadav. Competitive queue management for latency sensitive packets. In *Proceedings 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 228–237, 2008.

[20] Bruce Hajek. On the competitiveness of online scheduling of unit-length packets with hard deadlines in slotted time. In *Proceedings Conference on Information Sciences and Systems*, pages 434–438, 2001.

[21] Mohammad Taghi Hajiaghayi, Robert D. Kleinberg, Mohammad Mahdian, and David C. Parkes. Online auctions with re-usable goods. In *Proceedings 6th ACM Conference on Electronic Commerce*, pages 165–174, 2005.

[22] Norman L. Johnson and Samuel Kotz. *Urn Models and Their Applications*. John Wiley & Sons, 1977.

[23] Richard M. Karp, Umesh V. Vazirani, and Vijay V. Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings 22nd Annual ACM Symposium on Theory of Computing*, pages 352–358, 1990.

[24] Alexander Kesselman, Zvi Lotker, Yishay Mansour, Boaz Patt-Shamir, Baruch Schieber, and Maxim Sviridenko. Buffer overflow management in qos switches. *SIAM J. Comput.*, 33(3):563–583, 2004.

[25] Alexander Kesselman, Yishay Mansour, and Rob van Stee. Improved competitive guarantees for qos buffering. *Algorithmica*, 43(1-2):63–80, 2005.

[26] Valentin F. Kolchin, Boris A. Sevastyanov, and Vladimir P. Chistyakov. *Random Allocations*. John Wiley & Sons, 1978.

[27] Thomas G. Kurtz. Solutions of ordinary differential equations as limits of pure jump markov processes. *Journal of Applied Probability*, 7:49–58, 1970.

[28] Ron Lavi and Noam Nisan. Online ascending auctions for gradually expiring items. In *Proceedings 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1146–1155, 2005.

[29] Fei Li, Jay Sethuraman, and Clifford Stein. An optimal online algorithm for packet scheduling with agreeable deadlines. In *Proceedings 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 801–802, 2005.

[30] Fei Li, Jay Sethuraman, and Clifford Stein. Better online buffer management. In *Proceedings 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 199–208, 2007.

[31] Colin McDiarmid. Concentration. In *Probabilistic Methods for Algorithmic Discrete Mathematics*. Springer, 1998.

[32] Michael Mitzenmacher, Andréa W. Richa, and Ramesh Sitaraman. The power of two random choices: A survey of techniques and results. In *Handbook of Randomized Computing*. Springer.

[33] Michael Mitzenmacher and Eli Upfal. *Probability and computing - randomized algorithms and probabilistic analysis*. Cambridge University Press, 2005.

[34] Noam Nisan. Introduction to mechanism design (for computer scientists). In Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani, editors, *Algorithmic Game Theory*, chapter 9. Cambridge University Press, 2007.

[35] David C. Parkes. Online mechanisms. In Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani, editors, *Algorithmic Game Theory*, chapter 16. Cambridge University Press, 2007.

[36] Markus Schmidt. Packet buffering: Randomization beats deterministic algorithms. In *Proceedings 22nd Annual Symposium on Theoretical Aspects of Computer Science*, pages 293–304, 2005.