Randomized Allocation Processes

Artur Czumaj,[1,*] **Volker Stemann**[2,†]
[1] *Department of Computer and Information Science, New Jersey Institute of Technology, Newark, NJ 07102-1982; e-mail: czumaj@cis.njit.edu; www: http://www.cis.njit.edu/~czumaj*
[2] *Deutsche Morgan Grenfell*

Received 7 October 1999; accepted 4 September 2000

ABSTRACT: Many dynamic resource allocation and on-line load balancing problems can be modeled by processes that sequentially allocate balls into bins. The balls arrive one by one and are to be placed into bins on-line without using a centralized controller. If n balls are sequentially placed into n bins by placing each ball in a randomly chosen bin, then it is widely known that the maximum load in bins is $\ln n / \ln\ln n \cdot (1 + o(1))$ with high probability. Azar, Broder, Karlin, and Upfal extended this scheme, so that each ball is placed sequentially into the least full of d randomly chosen bins. They showed that the maximum load of the bins reduces exponentially and is $\ln\ln n / \ln d + \Theta(1)$ with high probability, provided $d \geq 2$. In this paper we investigate various extensions of these schemes that arise in applications in dynamic resource allocation and on-line load balancing. Traditionally, the main aim of allocation processes is to place balls into bins to minimize the maximum load in bins. However, in many applications it is equally important to minimize the number of choices performed (the allocation time). We study adaptive allocation schemes that achieve optimal tradeoffs between the maximum load, the maximum allocation time, and the average allocation time. We also investigate allocation processes that may reallocate the balls. We provide a tight

Correspondence to: Artur Czumaj.
An extended abstract reporting some of the results presented in this paper appeared in [15].
*Research done while at Heinz Nixdorf Institute and Department of Mathematics & Computer Science, University of Paderborn, Germany.
†Research done while at the International Computer Science Institute, USA, Berkeley, CA 94704-1198.
Contract grant sponsor: EU ESPIRT.
Contract grant number: Long Term Research Project 20244 (ALCOM-IT).
Contract grant sponsor: DFG.
Contract grant numbers: Sonderforschungsbereich 376 "Massive Parallelität: Algorithmen, Entwurfsmethoden, Anwendungen" and Leibniz Grant Me872/6-1.

analysis of a natural class of processes that each time a ball is placed in one of d randomly chosen bins may move balls among these d bins arbitrarily. Finally, we provide a tight analysis of the maximum load of the off-line process in which each ball may be placed into one of d randomly chosen bins. We apply this result to competitive analysis of on-line load balancing processes. Random Struct. Alg., 18, 297–331, 2001

Key Words: urn models, occupancy problems, resource allocation, load balancing

1. INTRODUCTION

Consider the following *dynamic resource allocation* problem. There is a system in which processes arrive one by one and each process has to choose on-line between a number of identical resources (for example, the process chooses a server to use among the servers in a network or chooses a disk to store a directory). The system is not centralized and in order to test the state of the resource the process must submit a request to it. There are various parameters of such a system: the most important is the maximum load of the resource, another is the number of submitted requests. The dynamic resource allocation problem may be modeled as the problem of sequentially allocating *balls* into *bins*. In that case, the balls correspond to the allocated processes and the bins correspond to the resources. The main aim of this paper is to analyze various strategies of placing balls into bins that arise in applications in dynamic resource allocation and on-line load balancing.

We investigate the problem of sequentially allocating m balls into n bins, $m \geq n$; we assume that n and m are "sufficiently large." Different allocation rules may use different resources in various ways and may result in different distributions of the balls in the bins. In this paper we study *allocation processes* that sequentially allocate the m balls in a *noncentralized* and *time-homogeneous* way. Thus, for example, we exclude the deterministic scheme that places the first ball into the first bin, the second ball into the second bin, and so on, as this needs a global controller.

There are two main parameters measuring the performance of an allocation process: the *load of the bin*, which is the number of balls in the bin and the *allocation time*, which is the number of bins chosen as candidates for the ball to be allocated. In the study of allocation processes our main aim is to minimize the following parameters:

- the *maximum load* in any bin,
- the *average allocation time* of the balls, and
- the *maximum allocation time* of any ball.

One of the most effective ways to avoid global control of an allocation process is to use randomization. For instance, the very simple randomized process that assigns each ball into a bin chosen *i.u.r.* (*independently and uniformly at random*) does not need a centralized controller, its maximum and average allocation time is 1 and the maximum load is

$$\Theta\left(\frac{\ln n}{\ln\lceil 1 + \frac{n}{m} \cdot \ln n\rceil} + \frac{m}{n}\right),$$

w.h.p.[1] We shall refer to this process as the *classical allocation process*, CAP (see [21, 23, 35] for a general exposition of this process and its applications). In a very influential paper [6], Azar et al. extended this process and showed that if each ball chooses i.u.r. with replacements $d > 1$ bins and is then placed into the least loaded bin among those chosen, the maximum load decreases dramatically to $(1 + o(1)) \ln \ln n / \ln d + \Theta(m/n)$, w.h.p. We will refer to the process of Azar et al. as ABKU[d]. They also proved that, informally, each randomized on-line process that does not reassign balls and has the maximum allocation time d must have a bin with load $(1 + o(1)) \ln \ln n / \ln d + \Omega(m/n)$, w.h.p. In the case $m = n$, Azar et al. [6] were able to tighten these bounds up to an additive constant term. They showed that the maximum load in ABKU[d] is $\ln \ln n / \ln d + \Theta(1)$ w.h.p. and that, informally, every randomized on-line process that has the maximum allocation time d has a bin with load $\ln \ln n / \ln d + \Omega(1)$ w.h.p. (Very recently, Vöcking [37] presented an ingenious modification of ABKU[d] which allows one to obtain even slightly better leading constants in the maximum load, and thus, in particular, for $d = 2$ the maximum load is $\approx 0.69 \log \log n + \Theta(1)$ w.h.p. Vöcking's protocols introduce certain bias in the bin choices which makes them different from the protocols investigated by Azar et al. [6], and thus Vöcking's results do not contradict to the aforementioned lower bound.)

Allocation schemes similar to those studied in our paper have also been analyzed in parallel and distributed settings. MacKenzie et al. [25] and Karp et al. [22] analyzed allocation schemes that appear in contention resolution protocols and shared memory simulations. Adler et al. [2] and Stemann [36] were concerned with the tradeoff between the maximum load and the number of parallel communication rounds one has to perform in order to place all balls. Berenbrink et al. [9] extended the approach of Stemann [36] to weighted balls.

Motivated by applications in dynamic load balancing Azar et al. [6] also analyzed *infinite processes*, where in each "round" a single ball is removed from the system and another ball is injected into the system. Related infinite processes have also been investigated by Adler et al. [1], Ajtai et al. [3], Berenbrink et al. [8], Cole et al. [11, 12], Czumaj [14], Mitzenmacher [27–32], and Vvedenskaya et al. [38].

All of the above allocation processes do not *reallocate* the balls during the run. It has been observed that the performance of many allocation processes may be significantly improved if the process can *reassign* some of the balls (see e.g., [5, 24, 39]). Notice that if the number of reallocations is not limited, it is trivial to maintain the ideally balanced load. However, since the reassignments are typically very expensive, their usage should be limited. Therefore in most of the existing load balancing algorithms the reassignments are usually performed in a very restricted way (see e.g., [24, 39]).

1.1. New Results

1.1.1. Adaptive Allocation Processes. Although the ABKU[d] process significantly decreases the maximum load when compared to CAP, it increases to d the allocation

[1] Throughout the paper *w.h.p.* will denote that a given event holds *with high probability*, that is, with probability $1 - N^{-c}$ for some positive constant c and sufficiently large N, where N is a parameter measuring the input size.

time of a ball. The firm assumption about the allocation time of ABKU[d] implies, for example, that even if the first bin chosen is empty, another $d-1$ bins are to be chosen for a given ball. Since in many applications an additional cost is charged for each access to a "bin," this is a clear waste of resources. In order to avoid this additional cost we investigate *adaptive allocation processes*. In an adaptive process the number of choices made in order to place a ball depends on the loads of the bins previously chosen by the ball. We shall apply this approach to obtain a smooth, optimal tradeoff between the maximum load, the maximum allocation time, and the average allocation time. Adaptive processes (but for infinite processes) have been independently studied by Mitzenmacher (see, e.g., [29]).

We classify the adaptive allocation processes by an infinite nondecreasing sequence of positive integers $x_0, x_1, \ldots$; we allow some of x_i be infinite. The meaning of each x_l is that we forbid it to place a ball into a bin with load l after less than x_l choices made by the ball. More formally, to each such a sequence $x_0, x_1, \ldots$ we assign process $P^{(x_0, x_1, \ldots)}$ modeled by the following algorithm:

Adaptive-Allocation-Process $(n, m, x_0, x_1, \ldots)$
Sequentially for each of m balls:
Let $M = 1$.
Repeat
Choose a bin b_M i.u.r. from $\{1, \ldots, n\}$.
Let b be the bin of the minimum load among bins $\{b_1, b_2, \ldots, b_M\}$ and let l be the load of b.
If $x_l \leq M$ then place the ball into bin b.
Else $M = M + 1$.
Until the ball is placed.

Let us note that, for example, $P^{(1,1,\ldots)}$ denotes our process CAP and $P^{(d,d,\ldots)}$ denotes the process ABKU[d].

Observe that according to our discussion above it is reasonable to require that $x_0 = 1$. This corresponds to the situation where every time an empty bin is chosen, the ball is placed into it.

We provide a thorough study of the maximum load of bins in adaptive allocation processes $P^{(x_0, x_1, \ldots)}$. We begin with the following result. (We remark that we have not tried to optimize constants, and thus, for example, the constant 8 in the definition of $\phi(n)$ can be easily improved.)

Theorem 1. *Let $m = n$ and let $x_0, x_1, \ldots$ be a fixed nondecreasing sequence of positive integers (we allow some of them to be $+\infty$). Let $s = \min\{i \geq 4 : x_i > 1\}$ and let $i^+ = \max\{i : ((3-e)/e)^2 n \geq 6\, i! \ln n\} > 1$. Define also $\phi(n) = \min\{i \geq s : \frac{2}{3} n / \ln n < (s!/8)^{\prod_{j=s}^{i} x_j}\}$. Then the maximum load of $P^{(x_0, x_1, \ldots)}$ is upper-bounded by $\phi(n) + 1$ w.h.p.*

[Notice that Theorem 1 cannot be applied to CAP, that is, to the process $P^{(1,1,\ldots)}$ — the condition on i^+ requires that not too many (roughly, less than $c \ln n / \ln \ln n$ for certain constant c) x_i are equal to 1.]

The next theorem is a special case of Theorem 1 that is more transparent and is sufficient in many applications.

Theorem 2. *Let $n = m$ and let $x_0, x_1, \ldots$ be a fixed nondecreasing sequence of positive integers with $x_3 > 1$. Let $k \geq 3$ be the minimum integer such that*

$$(3/2)^{\prod_{j=3}^{k} x_j} > n.$$

Then the maximum load of any bin in $P^{(x_0, x_1, \ldots)}$ is at most $k + 2$ w.h.p.

Let us observe the very special role s plays in Theorem 1. This value can be viewed as a separation between the behavior of the process $P^{(x_0, x_1, \ldots)}$ similar to the CAP process (where the maximum load is roughly the minimum integer l such that $l! > n$) and similar to the ABKU[d] process (where the maximum load is roughly the minimum integer l such that $2^{d^l} > n$). And so, for example, the maximum load in $P^{(x_0, x_1, \ldots)}$ with $x_0 = x_1 = \cdots = x_{s-1} = 1$ and $x_s = x_{s+1} = \cdots = d$ is roughly the minimum integer l such that $(s!)^{d^{l-s}} > n$.

The approach used in the proofs of Theorems 1 and 2 is an extension of that in [6]. However, in comparison with [6], some subtle details make their proofs technically more complicated and strenuous.

We prove also lower bounds for adaptive processes, and in particular, we show that the bound in Theorems 1 and 2 is tight up to an additive constant term.

Theorem 3. *Let $n = m$ and let $x_0, x_1, \ldots$ be a fixed nondecreasing sequence of positive integers with s such that $x_0 = \cdots = x_{s-1} = 1$ and $x_s > 1$. There exists a positive constant a such that if $2^{s-1} \cdot (s-1)! \leq (a\, n/\ln n)$, then the maximum load of any bin in $P^{(x_0, x_1, \ldots)}$ is $k + \Theta(1)$ w.h.p., where k is the minimum integer such that $((s-1)!) \exp \prod_{t=s-1}^{k-1} x_t > n$.*

We notice also that one could extend our analysis to the case $m > n$; however in this case we were unable to provide tight bounds for $m = \omega(n)$.

1.1.2. Allocation Time of Adaptive Processes. In the sequential case most of the previous work concerned the issue of minimization of the maximum load. One main reason to study the adaptive strategies is that they may achieve tradeoff between different performance measures. In certain applications it is desirable to minimize not only the maximum load of the bins but also other parameters of the system. We study adaptive schemes that achieve tradeoff between the *maximum load*, the *maximum allocation time*, and the *average allocation time*.

We first analyze a very simple process M-Threshold(m, n, M), which is the adaptive process $P^{(x_0, x_1, \ldots)}$ that places m balls into n bins with $x_0 = x_1 = \cdots = x_{M-1} = 1$ and $x_M = +\infty$. That is, in order to place a ball we choose bins i.u.r. with replacement until a bin with a load less than M is selected; then the ball is allocated into the bin chosen.

M-Threshold(m, n, M)
    Sequentially for each of m balls :
        Repeat
            Choose a bin i.u.r. from $\{1, \ldots, n\}$.
            If the chosen bin has load less than M then place the ball into the bin.
        Until the ball is placed.

We provide tight upper and lower bounds for the average allocation time for M-Threshold. In particular, we show that already the very simple process M-Threshold$(n, n, 2)$ performs very well when only the maximum load and the average allocation time are concerned.

Theorem 4. *M-Threshold$(n, n, 2)$ has the maximum load at most 2 (with certainty) and the average allocation time at most $1.146194 + o(1)$ w.h.p.*

Actually, one can show that this tradeoff between the maximum load and the average allocation time is optimal (among all on-line random allocation processes) up to lower-order terms.

The only disadvantage of M-Threshold$(n, n, 2)$ is that its maximum allocation time is $\Theta(\ln n)$ w.h.p. We can obtain the full trade-off between the maximum load, the maximum allocation time, and the average allocation time if we couple M-Threshold with ABKU[d] and investigate adaptive processes $P^{(x_0, x_1, \ldots)}$ with $x_0 = x_1 = \cdots = x_{M-1} = 1$ and $x_M = x_{M+1} = \cdots = d$.

Theorem 5. *Let $n = m$ and $d \geq 2$ be integers. Process $P^{(1,1,d,d,\ldots)}$ (that is, the process $P^{(x_0, x_1, \ldots)}$ with $x_0 = x_1 = 1$ and $x_i = d$ for $i \geq 2$) has the maximum load $\ln \ln n / \ln d + \mathcal{O}(1)$, the maximum allocation time at most d, and the average allocation time at most $1.146194 + o(1)$, w.h.p.*

Because every process with the maximum allocation time at most d must have the maximum load $\ln \ln n / \ln d + \Omega(1)$ [6], process $P^{(1,1,d,d,\ldots)}$ achieves the *best possible* (up to an additive constant term) tradeoff between the maximum load, the maximum allocation time, and the average allocation time.

We notice that it is not very difficult to show that the average allocation time of $P^{(1,1,d,d,d,\ldots)}$ is constant. Our main concern is, however, in determining the exact value of the average allocation time of $P^{(1,1,d,d,d,\ldots)}$.

We can further extend the result of Theorem 5.

Theorem 6. *Let $n = m$ and $d > 1$ be any integer.*

- *For every constant $\varepsilon > 0$ there exists an adaptive allocation process that has the maximum load $\ln \ln n / \ln d + \mathcal{O}(1)$, the maximum allocation time at most d, and the average allocation time at most $1 + \varepsilon$, w.h.p.*
- *If $d = \ln^{o(1)} n$ then there exists an adaptive allocation process that has the maximum load $(1 + o(1)) \ln \ln n / \ln d$, the maximum allocation time at most d, and the average allocation time $1 + o(1)$, w.h.p.*

Because every process with the maximum allocation time at most d must have the maximum load $\ln \ln n / \ln d + \Omega(1)$ w.h.p. [6], these processes achieve the best possible tradeoffs between the maximum load, the maximum allocation time, and the average allocation time.

We can also provide some extensions of Theorems 4 and 6 and analyze the case $m \neq n$.

1.1.3. Applications of Adaptive Processes. Our analysis of adaptive processes can be used to improve results obtained by Azar et al. [6] for the dynamic resource allocation problem.

Dynamic Resource Allocation. Consider a distributed system in which a user has to place on-line a task in one of identical and nondistinguishable servers (or to store a file on a disk, etc.). The first possible approach to this problem is to allow the user to check the load of all servers in order to allocate the task into the least loaded server. Although this process leads to ideally balanced distribution of the tasks, it is expensive, since it requires sending a message to each server in the system and interrupting its work. The ABKU[2] scheme can be used to obtain a more efficient solution. In a system with n tasks and n servers, if each user samples the load of two resources chosen i.u.r. and submits the task to the least loaded one, then the total overhead caused by communication with the servers is $2n$, and the load of the n servers varies by only at most $\log\log n + \mathcal{O}(1)$ w.h.p. Using adaptive processes we can design more efficient schemes. For example, if each user samples the load of resources chosen i.u.r. until it finds a server with at most one task and then submits the task to that server, then Theorem 4 implies that the maximum load is 2 and the total number of messages sent is at most $(1.146194 + o(1))n$ w.h.p. This result compares very favorably with our first protocol. By increasing the maximum load from 1 to 2, we can decrease the number of messages sent from $\Theta(n^2)$ to slightly more than n.

If one also wants to keep small the number of messages sent for each task, then we could use the adaptive processes from Theorem 6. For example, let us choose arbitrarily $d \geq 2$ and a positive constant ε. If each user samples the load according to the adaptive process from Theorem 6 then, with high probability, the total number of messages sent is not more than $(1 + \varepsilon)n$, each user sends at most d messages, and the load of the n servers varies by at most $\ln\ln n/\ln d + \mathcal{O}(1)$. Notice, for example, that if d is chosen such that $d \geq \ln^c n$ for some positive constant c, then the difference in the loads is constant w.h.p. Our approach that uses Theorem 6 can also be applied (after some minor modifications) to more realistic systems, in which the number of tasks differs from the number of servers and/or when the number of tasks is not known in advance.

1.1.4. Reassignments. One of the main applications of the ABKU[d] scheme is in on-line load balancing. The balls may be viewed as nondistinguishable tasks that arrive sequentially and are to be assigned to servers (corresponding to bins). In many applications it is possible that some of the existing tasks (balls) may be *reassigned*. As it turns out, the performance of many load balancing algorithms may be significantly improved if we allow reallocations of the balls (see, e.g., [5, 24, 39]). If the number of reallocations is not limited, it is trivial to maintain the ideally balanced load. However, since the reassignments are usually expensive, we would like to perform only a limited number of reassignments and make them only locally.

We investigate how much can be gained by using allocation processes with reassignments. We consider a scenario where during the allocation of a new ball one may ask for some number of possible locations (called the *allocation time*) and then arbitrarily reassign the balls among the bins chosen. The lower bound of $\ln\ln n/\ln d + \Omega(1)$ for the maximum load of any process with maximum allocation time d due to Azar et al. [6] does not hold for processes with reallocations. We study the question of whether allocation processes with reassignments may substantially decrease the maximum load.

We begin with the analysis of a simple and very natural modification of ABKU[d], which we call d-Balance: upon arrival of a task, choose d servers i.u.r. with replacement, assign the task to any of the servers chosen, and then balance the load of all the servers chosen. Here, by balancing we mean that the total load of the bins chosen remains unchanged and the loads of any two of the bins chosen differ by at most one. We provide a detailed analysis of the maximum load of d-Balance.

Theorem 7. *Let $d \geq 2$. The maximum load of d-Balance is $\Theta(\ln\ln n/\ln d + m/n)$ w.h.p. For $m = n$, the maximum load of d-Balance is $\ln\ln n/\ln d + \Theta(1)$ w.h.p.*

Actually, our lower bound holds for a much larger class of allocation processes that may perform reassignments. We prove a tight lower bound that holds for all on-line allocation processes that may perform reassignments.

Theorem 8. *Let $\mathscr{A}$ be any (on-line) allocation process that assigns m tasks to n servers according to the following scheme: Upon arrival of a task, $\mathscr{A}$ chooses i.u.r. with replacement up to d servers, assigns the task to any of the chosen servers, and arbitrarily reassigns the tasks among the chosen servers, possibly using complete information about the distribution of the current load of all servers in the system. Then the maximum load of any server is $\Omega(\ln\ln n/\ln d + m/n)$ w.h.p. In particular, if $m = n$ then the maximum load of any server is $\ln\ln n/\ln d + \Omega(1)$ w.h.p.*

Observe that this theorem states that even though there are allocation processes that reassign the balls and have the maximum load (stochastically) not bigger than any process that does not allow reassignments (d-Balance is one such a process), the benefit of using reallocations is insignificant. Theorem 8 implies that ball reallocations do not help too much, since ABKU[d] achieves optimal (up to an additive constant term in the case $n = m$) maximum load even if reassignments of balls are allowed.

1.1.5. Competitive On-Line Load Balancing and Off-Line Allocations. In the competitive analysis of on-line load balancing processes (cf. [10, 20]) it is important to study the behavior of optimal off-line solutions. Let us suppose that all random choices made by ABKU[2] are given in advance. In that case, when the full information about the random choices is known, Azar et al. [6, Lemma 6.1] showed that one can assign the n balls into the n bins with the maximum load $\mathscr{O}(1)$ w.h.p. We strengthen their result and prove the following theorem.

Theorem 9. *Let ϵ be any positive constant and let $\gamma_3 = \min_{\lambda>0}\{\lambda/(1-(1+\lambda)\cdot e^{-\lambda})\} \approx 3.35091887$. Let $c \leq \frac{1}{2}\cdot(\gamma_3 - n^{-((1/2)-\epsilon)})$. Suppose that cn balls are to be distributed, each having two out of n possible locations chosen i.u.r. Then one can place the balls (off-line) to achieve the maximum load of 2 w.h.p.*

The proof of the theorem is based on the reduction of this problem to a certain problem on random multigraphs and then on the threshold function of the existence of a k-core in a random multigraph. One can also easily prove that this result is optimal in the sense that for $d = o(\ln n)$ and $c \geq 1$ no assignment with maximum load 1 exists w.h.p.

This theorem immediately tightens competitive analysis of an on-line load balancing process given in the preliminary version of [6, Section 6.1.1]. Theorem 6.3 in [6] concerns scheduling permanent tasks against distribution $\mathcal{P}_d$ (see [6] for the definition of distribution $\mathcal{P}_d$ and for more details). Azar et al. showed that the ABKU[d] algorithm achieves the competitive ratio $\mathcal{O}(\ln\ln n/\ln d)$ w.h.p, and no algorithm can do better. Our bound strengthens this result to the competitive ratio $(\ln\ln n/2\ln d) + \Theta(1)$ w.h.p., which is tight up to an additive constant term.

1.2. Organization of the Paper

Section 2 contains basic definitions and notation used in the paper. In Section 3 we analyze the upper bound for the maximum load in adaptive processes and prove Theorems 1 and 2. Section 4 provides a proof of Theorem 5 and discusses its extensions. In Section 5, we analyze lower bounds for the maximum load in adaptive processes and prove Theorem 3. Section 6 contains an analysis of processes with reassignments. Section 7 investigates the off-line process.

2. BASIC NOTATION

We analyze processes that allocate $m = \alpha n$ balls in n bins, where α is any real number greater than or equal to 1 unless stated otherwise. The balls and bins are numbered consecutively $1, 2, \ldots, m$ and $1, 2, \ldots, n$, respectively. The balls are placed sequentially one by one; that is, ball t is placed only after all balls $1, 2, \ldots, t-1$ are already placed into bins. We say we are at *time* t when t balls were already placed. In order to find a bin in which ball t is placed, the process performs a sequence of random *choices* or *trials*. In each trial the ball *chooses* i.u.r. with replacement a bin i, and then bin i is said to be *chosen by ball* t. If a ball t is *placed into a bin* i then we say *bin* i *is selected by ball* t. The total number of trials made in order to allocate a ball is called the *allocation time of the ball* and the *maximum allocation time* is the maximum over the allocation times of all the balls. The total number of trials in the process is called the *(total) allocation time* of the process, and the *average allocation time* is equal to the allocation time divided by the number of balls.

In the paper we are generally following the notation of [6] where feasible. The *load of bin* j at time t, denoted by $\lambda_j(t)$, is the number of balls in bin j at time t. The *height of ball* t, denoted by h_t, is the number of balls at time t at the bin where ball t is placed. The number of balls that have height k at time t is denoted by $\mu_k(t)$ and the number of balls that have height at least k at time t is denoted by $\mu_{\geq k}(t)$. The number of bins that have load k at time t is denoted by $\nu_k(t)$ and the number of bins of load at least k at time t is denoted by $\nu_{\geq k}(t)$. Clearly, $\mu_{\geq k}(t) = \sum_{i\geq k} \mu_i(t)$ and $\nu_{\geq k}(t) = \sum_{i\geq k} \nu_i(t)$.

Recall that a real valued random variable X is *stochastically dominated* by a real valued random variable Y (or equivalently, Y *stochastically dominates* X) if $\mathbf{Pr}[X \leq x] \geq \mathbf{Pr}[Y \leq x]$ for every $x \in \mathbb{R}$.

Let us define $\mathbb{B}(n, p)$ to be a binomially distributed random variable with parameters n and p; that is, $\mathbf{Pr}[\mathbb{B}(n, p) = k] = \binom{n}{k} p^k (1-p)^{n-k}$ for every $0 \leq k \leq n$.

We will frequently use (usually without referring to it) the following simple proposition.

Proposition 2.1. *Let $x_0, x_1, \ldots$ and $x_0^*, x_1^*, \ldots$ be two arbitrary nondecreasing sequences of positive integers with $x_i \le x_i^*$ for each i.*

- *The maximum load of bins in $P^{(x_0^*, x_1^*, \ldots)}$ is stochastically dominated by the maximum load of bins in $P^{(x_0, x_1, \ldots)}$.*
- *The average (maximum) allocation time of $P^{(x_0, x_1, \ldots)}$ is stochastically dominated by the average (maximum) allocation time of $P^{(x_0^*, x_1^*, \ldots)}$.*

In this paper we shall also use the following three "concentration bounds."

Lemma 2.2 (Chernoff bound) (see, e.g., [17]).

(1) *For every positive integer n and real p, $0 < p < 1$,*

$$\mathbf{Pr}[\mathbb{B}(n, p) \ge enp] \le e^{-np}. \tag{1}$$

(2) *Let $X_1, \ldots, X_n$ be independent 0–1 random variables and let $X = \sum_{i=1}^n X_i$. Then*

$$\mathbf{Pr}[X \ge (1+\varepsilon)\cdot \mathbf{E}[X]] \le \exp(-\varepsilon^2 \mathbf{E}[X]/3) \qquad \text{for any } 0 \le \varepsilon \le 1. \tag{2}$$

(3) *Let $X_1, \ldots, X_n$ be independent 0–1 random variables and let $X = \sum_{i=1}^n X_i$. Then*

$$\mathbf{Pr}[X < \mathbf{E}[X]/2] \le \exp(-\mathbf{E}[X]/8). \tag{3}$$

Lemma 2.3 (Hoeffding bound) [18]. *Let the variables $X_1, \ldots, X_n$ be independent, with $a_i \le X_i \le b_i$ for each i, for suitable constants a_i and b_i. Let $X = \sum_i X_i$. Then for $t > 0$*

$$\mathbf{Pr}[X \le \mathbf{E}[X] - t] \le \exp\left(\frac{-2t^2}{\sum_i (b_i - a_i)^2}\right).$$

Lemma 2.4 (Method of bounded differences) (see, e.g., [26]). *Let $X_1, \cdots, X_n$ be independent random variables, with $X_i \in A_i$ for each i. Suppose that the (measurable) function $f\colon \prod_{i=1}^n A_i \to \mathbb{R}$ satisfies $|f(\bar{x}) - f(\bar{x'})| \le c_i$ whenever the vectors $\bar{x}$ and $\bar{x'}$ differ only in the ith coordinate. Let Y be a random variable $f(X_1, \ldots, X_n)$. Then for any $t > 0$,*

$$\mathbf{Pr}[Y \le \mathbf{E}[Y] - t] \le \exp\left(\frac{-2t^2}{\sum_{i=1}^n c_i^2}\right), \tag{4}$$

$$\mathbf{Pr}[Y \ge \mathbf{E}[Y] + t] \le \exp\left(\frac{-2t^2}{\sum_{i=1}^n c_i^2}\right). \tag{5}$$

3. MINIMIZING THE MAXIMUM LOAD IN ADAPTIVE PROCESSES

In this section we prove Theorems 1 and 2. We begin with presentation of the main ideas behind the proofs.

Consider a process $P^{(x_0,x_1,\dots)}$ that allocates n balls into n bins. The proofs follow an approach introduced by Azar et al. [6] that uses "layered" induction. We first prove (Claims 3.2 and 3.3) that with high probability there are at most β_s bins with load greater than or equal to s, where β_s is a suitably chosen function. Then, we proceed inductively: given that there are at most β_i bins with load greater than or equal to i (where $s \le i < i^*$ with i^* being suitably chosen), we show that with high probability the number of bins with load greater than or equal to $i+1$ is upper-bounded by β_{i+1} (where β_{i+1} is a suitably chosen function depending on β_i and x_i). This procedure works only for $i < i^*$, and thus the final step of our proof is to show that conditioned that the number of bins with load greater than or equal to i^* is upper-bounded by β_{i^*}, with high probability there is no bin with load greater than i^*+1. It is easy to see that all these arguments imply that with high probability there is no bin with load greater than i^*+1.

The proofs of Theorems 1 and 2 are very similar, and therefore we present the proof of Theorem 1 in detail and then briefly sketch the required modification in the proof of Theorem 2.

We begin with an auxiliary claim whose proof uses ideas developed later in Section 4; it is needed in the proof of Claim 3.2.

Claim 3.1. *Let $\mathcal{U}_j(t)$ and $\mathcal{V}_j(t)$ ($j, t \in \mathbb{N}$) be the random variables denoting the number of bins of load j after allocating t balls in the process $P^{(x_0,x_1,\dots)}$ and CAP, respectively. Let i be any integer such that $3 \le i \le \ln n/(4 \ln\ln n)$ and $x_{i-1} = 1$. Then, for every $l \in \mathbb{N}$ it holds that*

$$\mathbf{Pr}\left[\sum_{j\ge i} \mathcal{U}_j(n) \ \ge\ l\right] \le \mathbf{Pr}\left[\sum_{j\ge i} \mathcal{V}_j(\kappa n) \ \ge\ l\right] - n^{-3},$$

where

$$\kappa = \begin{cases} 1 + 1/i! & \text{if } i \le 7 \\ 1 + 2\,(i/2)^{-i/2} & \text{if } i \ge 8. \end{cases}$$

Proof. We postpone the proof of this claim to Section 4.2, where we will be able to use a machinery developed in Section 4 that makes the proof of this claim more intuitive. ■

The following claim forms the basis for our analysis of the upper bound for the maximum load in the proof of Theorem 1.

Claim 3.2. *Let ε and c be arbitrary constants, $0 \le \varepsilon \le 1$ and $c > 0$. Let $i \ge 3$ be an integer. If $x_{i-1} = 1$ and $\varepsilon^2 \cdot n \ge 6 \cdot i! \cdot \ln n$, then $\mathbf{Pr}[\nu_{\ge i}(n) \ge 2\,(1+\varepsilon)\,n/i!] \le n^{-2}$.*

Proof. Let $\nu^*_{\ge l}(t)$ be the random variable denoting the number of bins with load at least l after placing t balls in the classical process CAP. By Claim 3.1, $\mathbf{Pr}[\nu_{\ge i}(n) \ge 2\,(1+\varepsilon)\,n/i!] \ \le\ \mathbf{Pr}[\nu^*_{\ge i}(\kappa n) \ge 2\,(1+\varepsilon)\,n/i!] - n^{-3}$, where κ is as in Claim 3.1. Observe that $\kappa^i \leq 2$. Therefore it is sufficient to prove that $\mathbf{Pr}[\nu^*_{\geq i}(\kappa n) \geq (1+\varepsilon)\, n\, \kappa^i/i!] \leq \frac{1}{2}\, n^{-2}$.

Consider the classical allocation process CAP with κn balls and n bins. Fix i. For $1 \leq j \leq n$, let $\mathscr{B}_j$ be the indicator random variable of the event $\lambda_j(\kappa n) \geq i$. Observe that $\mathbf{E}[\mathscr{B}_j] \leq \binom{\kappa n}{i}(1/n)^i \leq \kappa^i/i!$. Since $\nu^*_{\geq i}(\kappa n) = \sum_{j=1}^{n} \mathscr{B}_j$, we get $\mathbf{E}[\nu^*_{\geq i}] \leq n \cdot \kappa^i/i!$. Now we apply a result due to Dubhashi and Ranjan [16], which states that the $\mathscr{B}_j$s are *negatively associated* (cf. [16]), and hence, by [16, Proposition 5], we may use the Chernoff bound from Lemma 2.2 (2):

$$\mathbf{Pr}[\nu^*_{\geq i}(\kappa n) \geq (1+\varepsilon) n \cdot \kappa^i/i!] \leq \exp(-\varepsilon^2 \cdot n \cdot \kappa^i/3\, i!)$$
$$\leq \frac{1}{2}\exp(-2 \cdot \ln n) = \frac{1}{2}\, n^{-2}. \quad \blacksquare$$

Next, we present a very simple claim that forms the basis for our analysis of the upper bound for the maximum load in the proof of Theorem 2.

Claim 3.3. *For every* $i > 0$, $\mathbf{Pr}[\nu]_{\geq i}(n) \leq (n/i) \equiv 1$.

Proof. The proof uses the pigeon-hole principle. If we had more than n/i bins with load at least i, then we would have more than $i \cdot (n/i)$ balls in the system, which contradicts the fact that there are n balls in the system. ■

Given the results from Claims 3.2 and 3.3 we are now ready to prove Theorems 1 and 2.

Proof of Theorem 1: We prove this theorem only for sequences with $x_3 = 1$; the other case when $x_3 > 1$ is very similar and we discuss it in more details in the proof of Theorem 2.[2]

Let $\beta_s = (6 \cdot n)/(e \cdot s!)$. For each i, $s \leq i < i^*$, let $\beta_{i+1} = (4\, e\, n/3) \cdot (\beta_i/n)^{x_i}$, where i^* will be defined later. For each i, $s \leq i \leq i^*$, let $\mathscr{E}_i$ be the event that $\nu_{\geq i}(n) < \beta_i$. *Our aim is to show that if $\mathscr{E}_i$ holds then $\mathscr{E}_{i+1}$ holds w.h.p.* Since Claim 3.2 implies $\mathbf{Pr}[\mathscr{E}_s] \geq 1 - n^{-2}$, this will yield a chain of inequalities that will give us a bound for the maximum load that holds with high probability.

Fix an integer i, $s \leq i \leq i^*$. Observe that in order to place a ball t in a bin of load at least i at time $t-1$, one has to perform at least x_i trials, and each of the chosen bins must have the load at least i at time $t-1$. Hence, if $\nu_{\geq i}(t-1) < n$, then we obtain

$$\mathbf{Pr}[h_t \geq i+1 \mid \nu_{\geq i}(t-1)] \leq \sum_{k \geq x_i} \left(\frac{\nu_{\geq i}(t-1)}{n}\right)^k$$
$$= \left(\frac{\nu_{\geq i}(t-1)}{n}\right)^{x_i} \cdot \frac{1}{1 - \nu_{\geq i}(t-1)/n}\,.$$

Observe also that if $\nu_{\geq i}(t) \leq n/4$, then

$$\mathbf{Pr}[h_t \geq i+1 \mid \nu_{\geq i}(t-1)] \leq \frac{4}{3} \cdot \left(\frac{\nu_{\geq i}(t-1)}{n}\right)^{x_i}. \qquad (6)$$

[2] Indeed, it is easy to see that if $n < (3/2)^{\prod_{j=3}^{k} x_j}$, then $\phi(n) \leq k+1$. Therefore, if $x_3 > 1$, then Theorem 2 implies Theorem 1.

For any t, $2 \leq t \leq n$, let Y_t be the 0–1 random variable that is 1 if and only if the height of the tth ball is at least $i+1$ and there are fewer than β_i bins with load at least i at time $t-1$. Notice that if ω_j represents the bin selected by the jth ball, then

$$\begin{aligned}\mathbf{Pr}[Y_t = 1 \mid \omega_1, \omega_2, \ldots, \omega_{t-1}] &\leq \mathbf{Pr}[h_t \geq i+1 \mid \nu_{\geq i}(t-1) < \beta_i] \\ &\leq \frac{4}{3} \cdot \left(\frac{\beta_i}{n}\right)^{x_i}, \qquad (7)\end{aligned}$$

where the last inequality follows from (6) and from the fact that $\beta_i < n/4$ for each $i \geq s$.

Let us define

$$p_i = \frac{4}{3} \cdot \left(\frac{\beta_i}{n}\right)^{x_i} \qquad \text{for each } i \geq s.$$

Observe that inequality (7) implies that conditioned on $\mathscr{E}_i$, each Y_t is stochastically dominated by the random variable with the Bernoulli distribution with parameter p_i. Therefore the sum of Y_ts is stochastically dominated by the random variable with the binomial distribution with parameters n and p_i (see also [6, Lemma 3.1]), and hence

$$\mathbf{Pr}\left[\sum_{t=1}^{n} Y_t \geq k\right] \leq \mathbf{Pr}[\mathbb{B}(n, p_i) \geq k].$$

Observe that conditioned on $\mathscr{E}_i$ we have $\sum_{t=1}^{n} Y_t = \mu_{\geq i+1}(n)$. Therefore we obtain

$$\begin{aligned}\mathbf{Pr}[\nu_{i+1}(n) \geq k \mid \mathscr{E}_i] \leq \mathbf{Pr}[\mu_{\geq i+1}(n) \geq k \mid \mathscr{E}_i] &= \mathbf{Pr}\left[\sum_{t=1}^{n} Y_t \geq k \mid \mathscr{E}_i\right] \\ &\leq \frac{\mathbf{Pr}[\sum_{t=1}^{n} Y_t \geq k]}{\mathbf{Pr}[\mathscr{E}_i]} \leq \frac{\mathbf{Pr}[\mathbb{B}(n, p_i) \geq k]}{\mathbf{Pr}[\mathscr{E}_i]}. \qquad (8)\end{aligned}$$

Since our goal is to bound the probability of the event $\neg\mathscr{E}_{i+1}$ conditioned on $\mathscr{E}_i$, from now on we will use (8) with $k = \beta_{i+1}$. Thus, inequality (8) implies that

$$\mathbf{Pr}[\neg\mathscr{E}_{i+1} \mid \mathscr{E}_i] \leq \frac{\mathbf{Pr}[\mathbb{B}(n, p_i) \geq \beta_{i+1}]}{\mathbf{Pr}[\mathscr{E}_i]}. \qquad (9)$$

We have chosen $\beta_{i+1} = e\, n\, p_i$ to allow a simple application of the Chernoff bound from Lemma 2.2 (2.2) in the inequality above to obtain

$$\mathbf{Pr}[\neg\mathscr{E}_{i+1} \mid \mathscr{E}_i] \leq \frac{e^{-n\, p_i}}{\mathbf{Pr}[\mathscr{E}_i]}.$$

Therefore we define i^* to be the smallest $i \geq s$ such that $n\, p_i < 2 \ln n$ and obtain

$$\mathbf{Pr}[\neg\mathscr{E}_{i+1} \mid \mathscr{E}_i] \leq \frac{1}{n^2 \cdot \mathbf{Pr}[\mathscr{E}_i]} \qquad \text{for all } s \leq i < i^*. \qquad (10)$$

Set $\beta_{i^*+1} = 2\,e\,\ln n$ and let $\mathscr{E}_{i^*+1}$ be the event $\nu_{\geq i^*+1}(n) < \beta_{i^*+1}$. Using the same approach as above, we obtain

$$\mathbf{Pr}[\neg\mathscr{E}_{i^*+1} \mid \mathscr{E}_{i^*}] = \mathbf{Pr}[\nu_{\geq i^*+1}(n) \geq 2\,e\,\ln n \mid \mathscr{E}_{i^*}] \leq \frac{\mathbf{Pr}[\mathbb{B}(n, p_{i^*}) \geq 2\,e\,\ln n]}{\mathbf{Pr}[\mathscr{E}_{i^*}]}$$
$$\leq \frac{\mathbf{Pr}[\mathbb{B}(n, \frac{2\ln n}{n}) \geq 2\,e\,\ln n]}{\mathbf{Pr}[\mathscr{E}_{i^*}]} \leq \frac{e^{-2\ln n}}{\mathbf{Pr}[\mathscr{E}_{i^*}]} = \frac{1}{n^2 \cdot \mathbf{Pr}[\mathscr{E}_{i^*}]}. \tag{11}$$

Now, we estimate the probability that $\nu_{\geq i^*+2}(n) \geq 1$, or equivalently, that the maximum load is greater than or equal to $i^* + 2$. Similarly as above, we obtain the bound.

$$\mathbf{Pr}[\nu_{\geq i^*+2}(n) \geq 1 \mid \mathscr{E}_{i^*+1}] \leq \frac{\mathbf{Pr}[\mathbb{B}(n, \frac{4}{3}(2e\ln n/n)^{x_{i^*+1}}) \geq 1]}{\mathbf{Pr}[\mathscr{E}_{i^*+1}]} \leq \frac{\frac{4}{3}\cdot n \cdot (2e\ln n/n)^{x_{i^*+1}}}{\mathbf{Pr}[\mathscr{E}_{i^*+1}]},$$

where the last inequality follows from the Markov inequality. Hence

$$\mathbf{Pr}[\nu_{\geq i^*+2}(n) \geq 1] \leq \mathbf{Pr}[\nu_{\geq i^*+2}(n) \geq 1 \mid \mathscr{E}_{i^*+1}] \cdot \mathbf{Pr}[\mathscr{E}_{i^*+1}] + \mathbf{Pr}[\neg\mathscr{E}_{i^*+1}]$$
$$\leq \frac{4}{3} \cdot n \cdot \left(\frac{2\,e\,\ln n}{n}\right)^{x_{i^*+1}} + \mathbf{Pr}[\neg\mathscr{E}_{i^*+1}].$$

Now, because inequalities (10) and (11) imply that

$$\mathbf{Pr}[\neg\mathscr{E}_{i+1}] \leq \mathbf{Pr}[\neg\mathscr{E}_{i+1} \mid \mathscr{E}_i] \cdot \mathbf{Pr}[\mathscr{E}_i] + \mathbf{Pr}[\neg\mathscr{E}_i] \leq n^{-2} + \mathbf{Pr}[\neg\mathscr{E}_i] \qquad \text{for all } s \leq i \leq i^*,$$

we obtain

$$\mathbf{Pr}[\nu_{\geq i^*+2}(n) \geq 1] \leq \frac{4}{3} \cdot n \cdot \left(\frac{2\,e\,\ln n}{n}\right)^{x_{i^*+1}} + \frac{i^* + 1 - s}{n^2} + \mathbf{Pr}[\neg\mathscr{E}_s].$$

Since $x_{i^*+1} \geq 2$ and Claim 3.2 (with $\varepsilon = (3 - e)/e$) yields

$$\mathbf{Pr}[\neg\mathscr{E}_s] \equiv \mathbf{Pr}[\nu_{\geq s}(n) \geq (6\,n)/(e\,s!)] \leq n^{-2},$$

we can conclude that the maximum load is smaller than or equal to $i^* + 1$ w.h.p.

The remaining part of the proof is to show that $i^* \leq \phi(n)$ for $\phi(n) = \min\{i \geq s : \frac{2}{3}\,n/\ln n < (s!/8)^{\prod_{j=s}^{i} x_j}\}$. We consider two cases.

Suppose first that $s = \phi(n)$. Then, by the definition of $\phi(n)$ we must have $(2\,n/3\,\ln n) < (s!/8)^{x_s}$, or equivalently, $(4\,n/3) \cdot (8/s!)^{x_s} < 2\,\ln n$. Hence,

$$n \cdot p_s = \frac{4}{3} \cdot n \cdot (\beta_s/n)^{x_s} = \frac{4}{3} \cdot n \cdot \left(\frac{6}{e \cdot s!}\right)^{x_s} < \frac{4}{3} \cdot n \cdot (8/s!)^{x_s} < 2 \cdot \ln n.$$

This of course implies that $i^* = s = \phi(n)$.

Now, let us look at the case $s < \phi(n)$. Since we want to prove that $i^* \leq \phi(n)$, let us consider what would happen when $i^* \geq \phi(n)$; we show that in this case $i^* = \phi(n)$.

Basic calculations yield

$$\beta_{i+1} = n \cdot \left(\frac{4 \cdot e}{3}\right)^{1+\sum_{j=s+1}^{i}\prod_{t=j}^{i} x_t} \cdot (\beta_s/n)^{\prod_{t=s}^{i} x_t} \qquad \text{for } i^* > i \geq s,$$

and hence, since $x_s \geq 2$, we obtain

$$\beta_{i+1} \leq n \cdot \left(\frac{4 \cdot e \cdot \beta_s}{3 \cdot n}\right)^{\prod_{j=s}^{i} x_j} \qquad \text{for } i^* > i \geq s.$$

Therefore, by our choice of β_s, we have for $i^* > i \geq s$

$$n \cdot p_{i+1} = \frac{4}{3} \cdot n \cdot \left(\frac{\beta_{i+1}}{n}\right)^{x_{i+1}} \leq \frac{4}{3} \cdot n \cdot \left(\frac{4 \cdot e \cdot \beta_s}{3n}\right)^{\prod_{j=s}^{i+1} x_j} = \frac{4}{3} \cdot n \cdot (8/s!)^{\prod_{j=s}^{i+1} x_j}.$$

Since $s \leq \phi(n) - 1 < i^*$, by the inequality above we have $n \cdot p_{\phi(n)} \leq (4n/3) \cdot (8/s!)^{\prod_{j=s}^{\phi(n)} x_j}$. On the other hand, the definition of $\phi(n)$ implies that $(4n/3) \cdot (8/s!)^{\prod_{j=s}^{\phi(n)} x_j} < 2 \ln n$. Therefore, $n \cdot p_{\phi(n)} < 2 \ln n$, and thus, the definition of i^* yields $i^* \leq \phi(n)$. Hence we showed that if $i^* \leq \phi(n)$, then $i^* = \phi(n)$. ■

Proof of Theorem 2. Since the proof is almost the same as the proof of Theorem 1 for $x_3 = 1$, we only briefly sketch the required changes. We did not try to optimize constants.

We start with $s = 3$ and $\beta_3 = n/3$. Then, Claim 3.3 implies that $\mathbf{Pr}[\mathscr{E}_s] \geq 1 - n^{-2}$. Next, we define $\beta_{i+1} = 2n \cdot (\beta_i/n)^{x_i}$ for all i, $s \leq i < i^*$. Given this, in inequality (8) we have constant $\frac{3}{2}$ instead of $\frac{4}{3}$, and therefore we set $p_i = \frac{3}{2} \cdot (\beta_i/n)^{x_i}$. Observe that $\beta_{i+1} = (1 + \frac{1}{3}) \cdot n\, p_i$. Therefore, we bound (9) using the Chernoff bound from Lemma 2.2 (2) with $\varepsilon = \frac{1}{3}$ to obtain $\mathbf{Pr}[\neg\mathscr{E}_{i+1} \mid \mathscr{E}_i] \leq (\exp(-n\, p_i/27)/\mathbf{Pr}[\mathscr{E}_i])$. Further, we set i^* to be the smallest $i \geq s$ such that $n\, p_i < 54 \cdot \ln n$ to obtain (10). Next, we set $\beta_{i*+1} = 54 \cdot \ln n$ and show (11). The final part follows exactly the approach from the proof of Theorem 1. ■

Remark 1. It is possible to extend Theorem 1 to deal also with the case $x_{i^+} = 1$. Notice however that in that case the obtained bound is not as tight as the bound in Theorem 1. This is caused by the fact that then $P^{(x_0, x_1, \ldots)}$ becomes similar to CAP, in which the distribution of the maximum load is diffused.

Similarly, we can also extend Theorems 1 and 2 to deal with the case when $m > n$. Then, however, the obtained bounds would not be so sharp as in the case $m = n$.

4. MINIMIZING THE AVERAGE ALLOCATION TIME IN ADAPTIVE PROCESSES

In this section we prove Theorems 4, 5, and 6, and discuss some of their extensions. Our results are based on the analysis of process M-Threshold. We recall that M-Threshold(m, n, M) is the adaptive process $P^{(x_0, x_1, \ldots)}$ with $x_0 = x_1 = \cdots = x_{M-1} = 1$ and $x_M = \infty$.

It is not very difficult to prove that for $M \geq 2$ the average allocation time of M-Threshold(n, n, M) is constant w.h.p. Our aim here is to provide an accurate estimation of how big this constant is. We begin with a theorem that gives a tight, but implicit, bound.

Theorem 10. *Let $M > \alpha = m/n$, $m = \omega(\sqrt{n \ln n})$, and let χ be the solution of*

$$\alpha = \chi \sum_{k=0}^{M-2} \frac{\chi^k}{e^{\chi} k!} + M\left(1 - \sum_{k=0}^{M-1} \frac{\chi^k}{e^{\chi} k!}\right). \tag{12}$$

The average allocation time of M-Threshold(m, n, M) is at least $\chi - o(1)$ and at most $\chi + o(1)$ w.h.p.

Actually, our analysis leads to the bound for the average allocation time of at least $\chi - \mathcal{O}(1/(M\sqrt{n \ln n}))$ and of at most $\chi + \mathcal{O}(1/(M\sqrt{n \ln n}))$ w.h.p.

Proof. For a given nonnegative integer x let $\psi_M(x) = \min\{M, x\}$ and for given nonnegative integers $x_1, \ldots, x_n$ let $\Psi_M(x_1, \ldots, x_n) = \sum_{i=1}^n \psi_M(x_i)$.

Let $X_1, X_2, \ldots$ be the infinite sequence of independently and uniformly distributed random variables corresponding to the trials made in M-Threshold (m, n, M). That is, if the ith trial returns bin j then $X_i = j$. Let Υ denote the random variable defined as the number of trials required to place all m balls. [Notice that although M-Threshold(m, n, M) makes only Υ trials, it formally depends on the infinite sequence $X_1, X_2, \ldots$.] Let us recall that $\lambda_i(t)$ is the number of balls in bin i at time t in M-Threshold(m, n, M). For a given sequence $X_1, X_2, \ldots$, let $\lambda_i^*(\tau)$ denote the load of bin j after τ trials if we would run (an infinite) CAP with $X_1, X_2, \ldots$. That is, $\lambda_i^*(\tau) = |\{j \ : \ 1 \le j \le \tau \ \& \ X_j = i\}|$. The main motivation to study the sequence of $\lambda_i^*(\tau)$ is that if $t = \Psi_M(\lambda_1^*(\tau), \ldots, \lambda_n^*(\tau))$, then $\lambda_i(t) = \psi_M(\lambda_i^*(\tau))$ for each i, $1 \le i \le n$. Indeed, notice that if ball j which was placed into bin i has height $h \le M$ in process CAP, then the ball making the jth trial in M-Threshold(m, n, M) is placed in bin i and has height h in M-Threshold(m, n, M). Thus, Υ is equal to the smallest τ^* such that $m = \Psi_M(\lambda_1^*(\tau^*), \ldots, \lambda_n^*(\tau^*))$. Therefore, in order to find the value of τ^* it is sufficient to analyze process CAP.

For a given T we estimate the probability that $T \ge \tau^*$. We will bound this probability using the *Poisson approximation heuristic* (see, e.g., [2, 4, 7]). The main difficulty in analyzing $\mathbf{Pr}[T \ge \tau^*]$ is that the values $\lambda_i^*(T)$ [as well as $\psi_M(\lambda_i^*(T))$] are correlated. It is known however, that the distribution of the $\lambda_i^*(T)$s is *approximately* Poisson with parameter $\xi = T/n$. Therefore, we estimate the behavior of $\Psi_M(\lambda_1^*(T), \ldots, \lambda_n^*(T))$ by function Ψ_M taken on n *independent* random variables with Poisson distribution. Our main aim is to estimate values of ξ such that with high probability $\Psi_M(\lambda_1^*(T), \ldots, \lambda_n^*(T))$ will be larger or respectively smaller than $m = \alpha n$.

Fix ξ and let $Y_1, Y_2, \ldots, Y_n$ be independent Poisson random variables with parameter ξ. For each i, $1 \le i \le n$, let $Z_i = \psi_M(Y_i)$. Thus, for each i and k, $1 \le i \le n$, $0 \le k \le M$,

$$\mathbf{Pr}[Z_i = k] = \begin{cases} \frac{\xi^k}{e^{\xi} k!} & 0 \le k < M \\ 1 - \sum_{j=0}^{M-1} \frac{\xi^j}{e^{\xi} j!} & k = M \ . \end{cases}$$

Let us set $p_k = \mathbf{Pr}[Z_i = k]$. Let $Z = \sum_{i=1}^n Z_i$; that is, $Z = \Psi_M(Y_1, \ldots, Y_n)$. Since for each i, $1 \le i \le n$,

$$\mathbf{E}[Z_i] = \sum_{k=1}^{M} k p_k = \sum_{k=1}^{M-1} k \cdot \frac{\xi^k}{e^{\xi} k!} + M p_M = \xi \sum_{k=0}^{M-2} p_k + M p_M,$$

we obtain

$$\mathbf{E}[Z] = \sum_{i=1}^{n} \mathbf{E}[Z_i] = n\xi \sum_{k=0}^{M-2} p_k + nMp_M.$$

Observe that since the Y_is are independent, so are the Z_is. Let us scale Z_i to $Z_i^* = Z_i/M$ and Z to $Z^* = Z/M$. Then, since $0 \le Z_i^* \le 1$ for each i, we may apply Lemma 2.3 (the Hoeffding bound) to obtain

$$\mathbf{Pr}[Z \le \mathbf{E}[Z] - t] = \mathbf{Pr}[Z^* \le \mathbf{E}[Z^*] - t/M] \le \exp(-2(t/M)^2/n) \qquad \text{for every } t > 0.$$

Now we may set $t = M \cdot \sqrt{n \ln n}$ to obtain $\mathbf{Pr}[Z \le \mathbf{E}[Z] - t] \le 1/n^2$.

Let T^+ be the solution of

$$m = T^+ \sum_{k=0}^{M-2} p_k + nMp_M - M\sqrt{n \ln n}$$

with $\xi = T^+/n$. Notice that $T^+ = (\chi + o(1))m$. By our arguments given above $\mathbf{Pr}[\tau * \ge T^+] \le 1/n^2$. In order to prove the upper bound for the average allocation time we have to show that $\Upsilon < T^+$ w.h.p. We use Corollary 13 from [2] that implies that for some positive constant c

$$\mathbf{Pr}[\Upsilon \ge T^+] \le c \cdot \mathbf{Pr}[\tau * \ge T^+].$$

Since $\mathbf{Pr}[\tau * \ge T^+] \le 1/n^2$, this completes the proof of the upper bound.

The proof of the lower bound is similar. We define T^- as the solution of

$$m = T^- \sum_{k=0}^{M-2} p_k + nMp_M + M\sqrt{n \ln n}$$

with $\xi = T^-/n$ and show that $\mathbf{Pr}[\tau * < T^-] \le 1/n^2$. This yields $\Upsilon \ge T^-$ w.h.p. Because $T^- = (\chi - o(1))m$, this completes the proof.

Unfortunately, there is no closed form for χ defined by formula (12), even in the case $m = n$. The only easy case is when $M = 1$ (and hence we require $\alpha < 1$), and then we obtain

$$\chi = \ln \frac{1}{1-\alpha} = \ln\left(\frac{n}{n-m}\right) = \ln\left(1 + \frac{m}{n-m}\right).$$

Nevertheless, for $M = 2$ the solution of Eq. (12) can be obtained with the use of the Lambert W function (see, e.g., [13]), which is the inverse of the function $x \mapsto xe^x$. That is, $W(x)$ is the function satisfying

$$W(x) \cdot e^{W(x)} = x.$$

For real x, $-1/e \le x < 0$, there are possibly two real values of $W(x)$. Following [13], the value (branch) satisfying $W(x) \le -1$ is denoted by $W_{-1}(x)$.

Fact 4.1. *The solution of Eq.* (12) *with* $\alpha < 2$ *and* $M = 2$ *is*

$$\chi = -W_{-1}((\alpha - 2) \cdot e^{-2}) - 2.$$

Corless et al. [13] provide various estimations of the Lambert W function. Some asymptotic results for χ for a few basic cases (also for $M > 2$) are presented in Table 1.

Proof of Theorem 4: This proof follows from the fact that M-Threshold$(n, n, 2)$ always has the maximum load at most 2 and from asymptotics in Table 1. ∎

Notice that Theorem 10 implies that the average allocation time of M-Threshold$(n, n, 2)$ is at least $1.146193 + o(1)$, w.h.p. Using this observation one can easily show that every randomized allocation process with n balls and n bins that has the maximum load at most 2 with probability at least 1/2 must have the average allocation time at least $1.146193 + o(1)$ w.h.p.

One can extend the bound for the average allocation time of M-Threshold$(n, n, 2)$ to prove the following.

Theorem 11. *Let* $\alpha > \varepsilon > 0$. *Then there exists a constant* $c > 0$ *such that if* $M > \alpha \geq 1$ *is even*, $M \geq 8$, *and* M *satisfies*

$$M \geq c \cdot \frac{\ln(1 + \alpha/\varepsilon)}{\ln\left(\frac{1}{\alpha+\varepsilon} \cdot \ln(1 + \alpha/\varepsilon)\right)},$$

the average allocation time of M-Threshold$(\alpha n, n, M)$ *is at most* $\alpha + \varepsilon + \mathcal{O}(1/(M \cdot \sqrt{n \ln n}))$ *w.h.p.*

Proof. Let $\xi = \alpha + \varepsilon$. According to Theorem 10, we must show that

$$\alpha \leq \xi \cdot \sum_{k=0}^{M-2} \frac{\xi^k}{e^{\xi} \cdot k!} + M\left(1 - \sum_{k=0}^{M-1} \frac{\xi^k}{e^{\xi} \cdot k!}\right).$$

This is equivalent to the inequality

$$(M - \alpha) \cdot \left(1 - \sum_{k=0}^{M-2} \frac{\xi^k}{e^{\xi} \cdot k!}\right) \geq M \cdot \frac{\xi^{M-1}}{e^{\xi} \cdot (M-1)!} - \varepsilon \cdot \sum_{k=0}^{M-2} \frac{\xi^k}{e^{\xi} \cdot k!}.$$

TABLE 1 Asymptotic Solutions for χ

α	M	χ	α	M	χ	α	M	χ	α	M	χ
1	2	1.146193221	2	3	2.346024071	3	4	3.544307460	10	11	11.75253034
1	3	1.025439801	2	4	2.088502887	3	5	3.168088995	10	12	10.81771484
1	4	1.004433559	2	5	2.023764969	3	6	3.055518942	10	13	10.41813731
1	5	1.000691457	2	6	2.006024834	3	7	3.017798513	10	14	10.21829930
1	6	1.000094794	2	7	2.001396923	3	8	3.005353618	10	15	10.11327126
1	7	1.000011498	2	8	2.000294182	3	9	3.001492267	10	16	10.05760408
1	11	1.000000001	2	15	2.000000001	3	18	3.000000001	10	32	10.00000001
1	12	1.000000000	2	16	2.000000000	3	19	3.000000000	10	33	10.00000000

Notice that the left-hand side is positive. Therefore it is sufficient to show that the right-hand side is negative. For that observe that

$$\begin{aligned}\frac{M\cdot\xi^{M-1}}{e^{\xi}\cdot(M-1)!}-\varepsilon\cdot\sum_{k=0}^{M-2}\frac{\xi^k}{e^{\xi}\cdot k!} &\le \frac{M\cdot\xi^{M-1}}{e^{\xi}\cdot(M-1)!}-\varepsilon\cdot\frac{\xi^{M/2-2}}{e^{\xi}\cdot(M/2-2)!}\\ &= \frac{\xi^{M/2-2}}{e^{\xi}\cdot(M/2-2)!}\cdot\left(\frac{M\cdot\xi^{1+M/2}}{(M-1)!/(M/2-2)!}-\varepsilon\right)\\ &\le \frac{\xi^{M/2-2}}{e^{\xi}\cdot(M/2-2)!}\cdot\left(\xi\cdot\left(\frac{2\xi}{M}\right)^{M/2}-\varepsilon\right).\end{aligned}$$

Thus it is sufficient to show that

$$\xi\cdot\left(\frac{2\xi}{M}\right)^{M/2}<\varepsilon,$$

which is equivalent to

$$\left(\frac{M}{2(\alpha+\varepsilon)}\right)^{M/(2(\alpha+\varepsilon))}>\left(\frac{\alpha+\varepsilon}{\varepsilon}\right)^{1/(\alpha+\varepsilon)}. \tag{13}$$

Thus we have obtained the inequality of the form $x^x > y$. Therefore, since $\alpha > \varepsilon > 0$, one can show that there exists a constant $c' > 0$ ($c' = 2$ suffices) such that if

$$\frac{M}{2(\alpha+\varepsilon)}>c'\cdot\frac{\ln\left(\left(\frac{\alpha+\varepsilon}{\varepsilon}\right)^{1/(\alpha+\varepsilon)}\right)}{\ln\ln\left(\left(\frac{\alpha+\varepsilon}{\varepsilon}\right)^{1/(\alpha+\varepsilon)}\right)},$$

then the inequality (13) is satisfied. Since the inequality above is equivalent to

$$M>\frac{2\,c'\cdot\ln\left(\frac{\alpha+\varepsilon}{\varepsilon}\right)}{\ln\left(\ln\left(\frac{\alpha+\varepsilon}{\varepsilon}\right)/(\alpha+\varepsilon)\right)},$$

this completes the proof.

Corollary 4.2. *For every constant ε, $\varepsilon > 0$, there exists a constant M_ε such that the average allocation time of M-Threshold(n, n, M_ε) is at most $1+\varepsilon$ w.h.p.*

Corollary 4.3. *Let $\varepsilon = o(1)$. There exists a constant $c > 0$ such that if*

$$M \ge c\cdot\frac{\ln\varepsilon^{-1}}{\ln\ln\varepsilon^{-1}},$$

then the average allocation time of M-Threshold(n, n, M_ε) is at most $1+\varepsilon+\mathcal{O}(1/(M\cdot\sqrt{n\ln n}))$ w.h.p.

4.1. Maximum Allocation Time

Notice that the maximum load of M-Threshold(m, n, M) is at most M. However the disadvantage of this process is its large maximum allocation time.

Proposition 4.4. *If M is constant, then the maximum allocation time of M-Threshold(n, n, M) is $\Theta(\log n)$ w.h.p.*

We can improve the maximum allocation time when we combine M-Threshold with ABKU[d].

Proof of Theorem 5: It is obvious that the maximum allocation time of any ball in $P^{(1,1,d,d,\ldots)}$ is at most d. Using Theorem 1 we can obtain that the maximum load of any bin in $P^{(1,1,d,d,\ldots)}$ is $\ln\ln n/\ln d + \Theta(1)$ w.h.p. Thus we only have to prove that the average allocation time is bounded by $1.146194 + o(1)$ w.h.p. For that observe that by Proposition 2.1, the average allocation time of any ball in $P^{(1,1,d,d,\ldots)}$ is stochastically dominated by the average allocation time of any ball in M-THRESHOLD$(n, n, 2)$. Hence we may apply Theorem 4 to complete the proof. ■

Proof of Theorem 6: The first part follows by combining Corollary 4.2 with the arguments from the proof of Theorem 5.

Now we prove the second part of the theorem. Fix $d \geq 2$ and let us consider process $P^{(x_0,x_1,\ldots)}$ with $x_0 = x_1 = \cdots = x_{M-1} = 1$ and $x_M = x_{M+1} = \cdots = d$, where M will be exposed later. By Theorems 1 and 3 the maximum load of $P^{(x_0,x_1,\ldots)}$ is $k + \Theta(1)$ w.h.p, where $k = \min\{t \in \mathbb{N} : ((M-1)!)^{d^{k-M}} > n\}$. One may easily show that $k = \mathcal{O}(M) + \ln(\ln n/M)/\ln d + \Theta(1)$. Hence if $M = o(\ln\ln n/\ln d)$ then the maximum load of $P^{(x_0,x_1,\ldots)}$ is $(1 + o(1))\ln\ln n/\ln d$ w.h.p.

Let $c > 0$ be the constant from Corollary 4.3. Choose $\varepsilon = o(1)$ such that if we set $M = \lceil c \cdot \ln \varepsilon^{-1}/\ln\ln \varepsilon^{-1}\rceil$ then $M = o(\ln\ln n/\ln d)$. By coupling Corollary 4.3 with the arguments used in the proof of Theorem 5, the average allocation time of $P^{(x_0,x_1,\ldots)}$ is $1 + \varepsilon(1 + o(1)) = 1 + o(1)$ w.h.p. Since the maximum allocation time of $P^{(x_0,x_1,\ldots)}$ is at most d, this completes the proof. ■

We can also prove the following theorem which estimates the dependence of α from M and the average allocation time.

Theorem 12. *For*

$$\frac{m}{n} = \alpha \leq M - \sqrt{\frac{M}{2\pi}} \cdot \left(1 - \frac{1}{12M} + \frac{1}{288M^2}\right)$$

the average allocation time of M-Threshold(m, n, M) is at most $M + \mathcal{O}(M^{-5/2})$ w.h.p.

Proof. Let us take $\chi = M$ and ask how big can α be chosen so that the following inequality does hold (cf. Theorem 10):

$$\alpha \leq \chi \sum_{k=0}^{M-2} \frac{\chi^k}{e^\chi \cdot k!} + M\left(1 - \sum_{k=0}^{M-1} \frac{\chi^k}{e^\chi \cdot k!}\right).$$

Notice that because $M = \chi$, we can simplify this formula to

$$\alpha \leq \chi \sum_{k=0}^{\chi-2} \frac{\chi^k}{e^\chi \cdot k!} + \chi \left(1 - \sum_{k=0}^{\chi-1} \frac{\chi^k}{e^\chi \cdot k!}\right),$$

which in turn is equivalent to

$$\alpha \leq \chi \left(1 - \frac{\chi^{\chi-1}}{e^\chi \cdot (\chi-1)!}\right). \tag{14}$$

Now we use the following formula for $1/r!$ (see, e.g., p. 334 in [19])

$$\frac{1}{r!} = \frac{e^{r+1}}{(r+1)^r \sqrt{2\pi(r+1)}} \cdot \left(1 - \frac{1}{12(r+1)} + \frac{1}{288(r+1)^2} + \mathcal{O}(r^{-3})\right). \tag{15}$$

Hence (14) implies that

$$\begin{aligned}\alpha &\leq \chi \cdot \left(1 - \frac{\chi^{\chi-1}}{e^\chi} \cdot \frac{e^\chi}{\chi^{\chi-1}\sqrt{2\pi\chi}} \cdot \left(1 - \frac{1}{12\chi} + \frac{1}{288\chi^2} + \mathcal{O}(\chi^{-3})\right)\right)\\ &= \chi \cdot \left(1 - \frac{1}{\sqrt{2\pi\chi}} \cdot \left(1 - \frac{1}{12\chi} + \frac{1}{288\chi^2} + \mathcal{O}(\chi^{-3})\right)\right).\end{aligned}$$

We may combine this bound with Theorem 10 to complete the proof.

How tight is the bound in Theorem 12 for α when M is constant? Although this bound is tight up to a constant factor, it is not as precise as that in Fact 4.1 or Theorem 4. On the other hand, since the bound (15) is very tight even for a small constant r, the estimation in Theorem 12 is very accurate even for $M = 2$. In that case the tight bound for $\chi = M = 2$ requires that $\alpha \leq 2(1 - 2e^{-2}) = 2 - 4/e^2 \approx 1.45866$ and the estimation in Theorem 12 gives $M - \sqrt{M/2\pi} \cdot \left(1 - (1/12M) + (1/288M^2)\right) \approx 1.45883$. Thus the difference is ≈ 0.00017. For $M = 10$ this difference is $\approx 0.33 \cdot 10^{-5}$, for $M = 100$ the difference is less than 10^{-7}, and for $M = 1000$ the difference does not exceed 10^{-10}.

Finally, from our discussion above [cf. inequality (14)] one can derive the following:

Corollary 4.5. *If $\alpha \leq M(1 - (M^{M-1}/e^M \cdot (M-1)!))$, then the average allocation time of M-Threshold$(\alpha n, n, M)$ is $\Theta(\alpha)$ w.h.p.*

4.2. A Proof of an Auxiliary Claim from Section 3

In this subsection we present a proof of Claim 3.1.

Proof of Claim 3.1. Let $\mathcal{U}_{\geq r}(t) = \sum_{j \geq r} \mathcal{U}_j(t)$ and let $\mathcal{V}_{\geq r}(t) = \sum_{j \geq r} \mathcal{V}_j(t)$. We relate the values $\mathcal{U}_{\geq i}(n)$ and $\mathcal{V}_{\geq i}(\kappa n)$ using the ideas developed in the proofs of Theorems 10 and 11. We again emphasize that the proofs of these theorems are independent of the results presented in Section 3.

Let $X_1, X_2, \ldots$ be the infinite sequence of independently and uniformly distributed random variables, $X_l \in [n]$ for every $t \in \mathbb{N}$. We shall consider this sequence as the sequence underlying processes $P^{(x_0,x_1,\ldots)}$ and CAP, such that the jth trial in the corresponding process returns bin X_j.

Let us first notice that the load of bin k at time t in CAP corresponding to the sequence $X_1, X_2, \ldots$ is equal to $|\{X_j = k \ : \ 1 \le j \le t\}|$. Further, the difference between CAP and $P^{(x_0,x_1,\ldots)}$ is that the former may not always locate a ball into a chosen bin with load larger than or equal to $i-1$.

Now, suppose that after t trials a bin b has load l in CAP. Then, if $l \le i-1$, bin b has load l also after t trials in $P^{(x_0,x_1,\ldots)}$. On the other hand, if $l \ge i$, then bin b has load at least $i-1$ after t trials in $P^{(x_0,x_1,\ldots)}$. Therefore, if after τ trials there have been n balls allocated into bins in process $P^{(x_0,x_1,\ldots)}$, then $\mathcal{U}_j(n) = \mathcal{V}_j(\tau)$ for all $j \le i-1$, and $\mathcal{U}_{\ge i}(n) \le \mathcal{V}_{\ge i}(\tau)$.

Suppose first that $i \le 7$. Then, by the results presented in Table 1 it holds that for $\tau = n(1 + 1/i!)$, after τ trials at least n balls will be allocated in $P^{(x_0,x_1,\ldots)}$, with probability at least $1 - n^{-3}$. Therefore we have (stochastically) $\mathcal{U}_{\ge i}(n) > \mathcal{V}_{\ge i}(\tau)$ with probability upper bounded by n^{-3}.

If $i \ge 8$, then by Theorem 11, we see that if $i \ge 4 \cdot (\ln(1+\alpha)/\ln\ln(1+\alpha))$, then after

$$\tau = \left(1 + \frac{1}{\alpha} + \frac{1}{\Omega(i\sqrt{n}\,\ln n)}\right) \cdot n$$

trials at least n balls will be allocated in $P^{(x_0,x_1,\ldots)}$, with probability at least $1 - n^{-3}$. Therefore we have (stochastically) $\mathcal{U}_{\ge i}(n) > \mathcal{V}_{\ge i}(\tau)$ with probability upper-bounded by n^{-3}. Simple calculations yield that if $i \le (\ln n/4 \ln\ln n)$, then $\tau \le n(1 + 2(i/2)^{-i/2})$. ■

5. LOWER BOUND FOR THE MAXIMUM LOAD IN ADAPTIVE PROCESSES

In this section we prove Theorem 3 and show that the bound for the maximum load from Theorem 1 is tight up to an additive constant term. We begin with a theorem that shows that this bound is tight for $s = \Theta(1)$ and next extend this result to larger s.

Theorem 13. *Let $m = n$ and let $x_0, x_1, \ldots$ be a fixed nondecreasing sequence of positive integers (we allow some of them to be $+\infty$).*

(1) *Let $x_0 > 1$ and let k be such that*

$$\frac{n}{8 \ln n} \ge \left(2^9\right)^{\prod_{t=1}^{k} x_t}.$$

Then the maximum load of $P^{(x_0,x_1,\ldots)}$ is at least k with probability at least $1 - 1/n$.

(2) *Let $x_{s-1} = 1$ and $x_s > 1$, and let k be such that*

$$\frac{n}{8 \ln n} \ge 2^{\left(\frac{s(s+10)}{2}\right)\prod_{t=s}^{k} x_t}.$$

Then there exists a bin in $P^{(x_0,x_1,\ldots)}$ *with load at least* k *with probability at least* $1 - 1/n$.

Proof. The proof is similar in spirit to the proof of Theorem 3.3 from [6].

We start with case (1), that is, with $x_0 > 1$. Let $\gamma_0 = n$ and let us recursively set $\gamma_{i+1} = (\gamma_i^{x_i}/(n^{x_i-1} \cdot 2^{i+3}))$ for $i \geq 0$. Easy calculations lead to the formula

$$\gamma_{i+1} = \frac{n}{2^{(i+3)+\sum_{j=1}^{i}(j+2)\prod_{t=j}^{i} x_t}} \qquad \text{for } i \geq 0.$$

Since $x_i \geq x_{i-1} \geq \cdots \geq x_0 \geq 2$, one can show that

$$\gamma_{i+1} \geq \frac{n}{2^{9 \cdot \prod_{t=1}^{i} x_t}}. \tag{16}$$

Let $\mathcal{F}_i$ be the event that

$$\nu_{\geq i}(n - n/2^i) \geq \gamma_i.$$

Observe that $\mathcal{F}_0$ holds with certainty. Our aim is to estimate $\mathbf{Pr}[\mathcal{F}_{i+1} \mid \mathcal{F}_i]$. Let $\mathcal{I}_i = \{t \in \mathbb{N} \,:\, n(1 - 1/2^{i-1}) < t \leq n(1 - 1/2^i)\}$. Fix i. For $t \in \mathcal{I}_{i+1}$, let Z_t be the 0–1 random variable defined by

$$Z_t = 1 \qquad \text{if and only if } h_t = i+1 \text{ or } \nu_{\geq i+1}(t-1) \geq \gamma_{i+1}.$$

Now suppose for a moment that $\nu_{\geq i+1}(t-1) < \gamma_{i+1}$. In that case, if among the first x_i chosen bins at least one has load i and the remaining bins have load at least i, then clearly $Z_t = 1$. Therefore, if ω_j represents the bin selected by ball j, then we may use the fact that $\mathbf{Pr}[A \vee B] \geq \mathbf{Pr}[A \mid \neg B]$ to obtain that

$$\mathbf{Pr}[Z_t = 1 \mid \omega_1, \omega_2, \ldots, \omega_{t-1}, \mathcal{F}_i] \geq \frac{\gamma_i - \gamma_{i+1}}{n} \cdot (\gamma_i/n)^{x_i - 1} \geq \frac{1}{2}(\gamma_i/n)^{x_i}.$$

Let $p_i = 1/2 \cdot (\gamma_i/n)^{x_i}$. Similarly as in [6], we notice that the inequality above implies that conditioned on $\mathcal{F}_i$, $\sum_{t \in \mathcal{I}_{i+1}} Z_t$ stochastically dominates binomially distributed random variable $\mathbb{B}(|\mathcal{I}_{i+1}|, p_i)$. Hence,

$$\mathbf{Pr}\left[\sum_{t \in \mathcal{I}_{i+1}} Z_t < k \;\middle|\; \mathcal{F}_i\right] \leq \mathbf{Pr}[\mathbb{B}(n/2^{i+1}, p_i) < k]. \tag{17}$$

Now we use the Chernoff bound from Lemma 2.2 (3) and obtain that whenever

$$n \cdot p_i/2^{i+1} \geq 16 \ln n, \tag{18}$$

then it holds that

$$\mathbf{Pr}[\mathbb{B}(n/2^{i+1}, p_i) < np_i/2^{i+2}] \leq 1/n^2.$$

Since we have set $\gamma_{i+1} = np_i/2^{i+2}$, (17) and (18) yield

$$\mathbf{Pr}\left[\sum_{t \in \mathcal{I}_{i+1}} Z_t < \gamma_{i+1} \;\middle|\; \mathcal{F}_i\right] \leq 1/n^2.$$

Let i^* be the smallest integer for which (18) does not hold. Then we obtain that for each i, $0 \le i < i^*$,

$$\mathbf{Pr}[\neg \mathcal{F}_{i+1} \mid \mathcal{F}_i] \le \mathbf{Pr}\left[\sum_{t \in \mathcal{I}_{i+1}} Z_t < \gamma_{i+1} \,\middle|\, \mathcal{F}_i \right] \le 1/n^2.$$

Therefore

$$\mathbf{Pr}[\nu_{i^*-1} \ge 1] \ge \mathbf{Pr}[\mathcal{F}_{i^*-1}] \ge \prod_{i=0}^{i^*-2} \mathbf{Pr}[\mathcal{F}_{i+1} \mid \mathcal{F}_i] \cdot \mathbf{Pr}[\mathcal{F}_0] \ge (1 - 1/n^2)^{i^*-1} \ge 1 - 1/n.$$

In order to estimate the value of i^* we use (16) to complete the proof of (1).

If $x_0 = 1$, then one can show that

$$\gamma_{i+1} \ge \frac{n}{2^{(s(s+10)/2)\prod_{t=s}^{i} x_t}},$$

where s satisfies $x_s > x_{s-1} = 1$. Thus using exactly the same calculations as in the case $x_0 > 1$ one can prove the second part of the theorem. ■

Observe that Theorem 13 implies that the bound for the maximum load given in Theorem 1 with $s = \Theta(1)$ (and hence also the bound in Theorem 2) is tight up to a constant additive term. Now we extend this tight result to $s = \omega(1)$.

We begin with a lemma concerning the number of bins with the maximum load in the M-Threshold process.

Lemma 5.1. *There exist constants $c_1, c_2 > 0$, such that if $n\alpha^M/M! \ge c_1 \ln n$, $M^2 \le \alpha n$, $0 < \alpha \le M(1 - (M^{M-1}/e^M(M-1)!))$, and $\alpha = \Theta(1)$, then in M-Threshold$(\alpha n, n, M)$ there are at least $c_2 \cdot n \cdot \alpha^M/M!$ bins with load M w.h.p.*

Proof. Similar to the proof of Theorem 10, we compare the behavior of M-Threshold with the behavior of CAP.

Let us consider CAP in which αn balls are allocated into n bins. For $1 \le j \le n$, let $\mathcal{B}_j$ be the indicator random variable of the event that bin j has load at least M (in CAP). Observe that

$$\mathbf{Pr}[\mathcal{B}_j = 1] \ge \binom{\alpha n}{M} \cdot \left(\frac{1}{n}\right)^M \cdot \left(1 - \frac{1}{n}\right)^{\alpha n - M} \ge \frac{(\alpha n - M)^M}{M!} \cdot \frac{1}{n^M} \cdot \left(1 - \frac{1}{n}\right)^{\alpha n}.$$

Further, the assumption $M^2 \le \alpha n$ yields

$$\begin{aligned}\mathbf{Pr}[\mathcal{B}_j = 1] &\ge \frac{(\alpha n - M)^M}{M!} \cdot \frac{1}{n^M} \cdot \left(1 - \frac{1}{n}\right)^{\alpha n} \ge \left(\frac{\alpha n - M}{n}\right)^M \cdot \frac{1}{M!} \cdot \left(1 - \frac{1}{n}\right)^{\alpha n} \\ &\ge \alpha^M \cdot \left(1 - \frac{M}{\alpha n}\right)^M \cdot \frac{1}{M!} \cdot e^{-\alpha/2} \ge \alpha^M \cdot e^{-1/2} \cdot \frac{1}{M!} \cdot e^{-\alpha/2} \ge \frac{c \cdot \alpha^M}{M!}\end{aligned}$$

for a positive constant c that depends only on α. Hence we obtain that

$$\mathbf{E}\left[\sum_{j=1}^{n} \mathcal{B}_j\right] \ge c \cdot n \cdot \frac{\alpha^M}{M!}.$$

Now, similar to the proof of Claim 3.2, we can show that the $\mathcal{B}_j$s are negatively associated (cf. [16]) and hence we may use the Chernoff bound from Lemma 2.2 (2.2) to obtain

$$\mathbf{Pr}\left[\sum_{j=1}^{n} \mathcal{B}_j < \frac{1}{2} \cdot c \cdot n \cdot \frac{\alpha^M}{M!}\right] \leq \exp\left(-c \cdot n \cdot \alpha^M/(8M!)\right).$$

Therefore, if we set $c_1 = c_3 \cdot 8/c$ and $c_2 = c/2$, then $n\,\alpha^M/M! \geq c_1 \ln n$ implies that

$$\mathbf{Pr}\left[\sum_{j=1}^{n} \mathcal{B}_j < \frac{c_2\, n\, \alpha^M}{M!}\right] \leq 1/n^{c_3}.$$

To conclude the proof, we just observe that the number (random variable) of the bins with load M in M-Threshold(m, n, M) stochastically dominates $\sum_{j=1}^{n} \mathcal{B}_j$, and therefore with high probability there are at least $(c_2\, n\, \alpha^M)/M!$ bins with load M in M-Threshold(m, n, M). ■

Using this lemma we can prove the following theorem.

Theorem 14. *Let $n = m$ and let $x_0, x_1, \ldots$ be a fixed nondecreasing sequence of positive integers with $x_0 = \cdots = x_{s-1} = 1$ and $x_s > 1$, for $s > 0$. There exist positive constants c_1 and c_3 such that if*

$$2^{s-1} \cdot (s-1)! \leq \frac{n}{c_1 \cdot \ln n} \qquad \textit{and}$$

$$\left(\frac{2^{s-1} \cdot (s-1)!}{c_3}\right)^{\prod_{t=s-1}^{k} x_t} \leq \frac{n}{2^{k+6-s} \cdot \ln n},$$

then the maximum load of any bin in $P^{(x_0, x_1, \ldots)}$ is at least k w.h.p.

Proof. We proceed along the line of the proof of Theorem 13. The main difference is the starting point, that is, the smallest i for which the bound for $\nu_{\geq i}(n/2)$ is handled.

Recall that $x_{s-1} = 1$. Therefore, after throwing $n/2$ balls in $P^{(x_0, x_1, \ldots)}$ the random variable $\nu_{\geq s-1}(n/2)$ stochastically dominates the random variable ν_{s-1}^* that denotes the number of bins with load $s-1$ in M-Threshold$(n/2, n, s-1)$. By Lemma 5.1 there exist positive constants c_1 and c_2 such that if $2^{s-1} \cdot (s-1)! \leq n/(c_1 \cdot \ln n)$, then $\nu_{s-1}^* \geq c_2 \cdot n/(2^{s-1} \cdot (s-1)!)$ w.h.p. Therefore $\nu_{\geq s-1}(n/2) \geq c_2 \cdot n/(2^{s-1} \cdot (s-1)!)$ w.h.p. This motivates us to set

$$\gamma_{s-1} = c_2 \cdot n/(2^{s-1} \cdot (s-1)!).$$

Similar to the proof of Theorem 13, we set

$$\gamma_{i+1} = \frac{\gamma_i^{x_i}}{n^{x_i - 1} \cdot 2^{i+3-s}} \qquad \text{for } i \geq s-1.$$

Now, one can show that there exists a positive constant c_3 such that

$$\gamma_i \geq n \cdot (c_3/(2^{s-1} \cdot (s-1)!))^{\prod_{t=s-1}^{i-1} x_t}. \tag{19}$$

Let us define $\mathcal{F}_i$ as the event that $\nu_{\geq i}(n - n/2^{i+1-(s-1)}) \geq \gamma_i$ for $i \geq s-1$. From our arguments above $\mathcal{F}_{s-1}$ holds with high probability. Let $p_i = 1/2 \cdot (\gamma_i/n)^{x_i}$ for $i \geq s-1$. We can proceed as in the proof of Theorem 13 and show that if

$$n \cdot p_i/2^{i+1-(s-1)} \geq 16 \ln n \qquad \text{for } i \geq s-1,$$

then

$$\mathbf{Pr}[\mathcal{F}_{i+1} \mid \mathcal{F}_i] \leq 1/n^2,$$

which, in turn, clearly yields

$$\mathbf{Pr}[\mathcal{F}_{i+1}] \geq 1 - \frac{1}{n}.$$

Therefore, if we set i^* to be the smallest integer for which $n \cdot p_{i^*} < 2^{i^*+6-s} \ln n$, then

$$\mathbf{Pr}[\nu_{\geq i^*-1} \geq 1] \geq 1 - \frac{1}{n}.$$

Now we can use inequality (19) to complete the proof of the theorem. ■

Proof of Theorem 3: This follows immediately from proofs of Theorems 1 and 14. ■

6. REASSIGNMENTS

In this section we study processes that may reallocate balls. We first define formally process d-Balance which is a simple extension of ABKU[d] that allows reassignments. Then we show that the maximum load of any bin in ABKU[d] stochastically dominates the maximum load of any bin in d-Balance. Afterward we present a general lower bound for processes that may change the locations of balls.

Process d-Balance places balls into bins sequentially one by one. For each ball it chooses i.u.r. with replacement d bins. It first places the ball into one of the chosen bins and then *balances* the load of the bins chosen. That is, if there were initially $L-1$ balls in the chosen d bins, then afterwards $L \bmod d$ of the bins will have load $\lfloor L/d \rfloor + 1$ and the remaining bins will have load $\lfloor L/d \rfloor$.

We show that the maximum load of any bin in ABKU[d] stochastically dominates the maximum load of any bin in d-Balance. We start with some basic definitions.

For a vector $\mathbf{v} = (v_1, \ldots, v_n)$ let $\pi_{\mathbf{v}}$ be a permutation of n elements such that $v_{\pi_{\mathbf{v}}(1)} \geq v_{\pi_{\mathbf{v}}(2)} \geq \cdots \geq v_{\pi_{\mathbf{v}}(n)}$. Let $\mathbf{v}$ and $\mathbf{u}$ be two n-vectors, $\mathbf{v} = (v_1, \ldots, v_n)$ and $\mathbf{u} = (u_1, \ldots, u_n)$. We say that $\mathbf{v}$ *majorizes* $\mathbf{u}$: what is denoted by $\mathbf{v} \succeq \mathbf{u}$, if $\sum_{j=1}^{i} v_{\pi_{\mathbf{v}}(j)} \geq \sum_{j=1}^{i} u_{\pi_{\mathbf{u}}(j)}$ for every i, $1 \leq i \leq n$. Let $\mathbf{e}_i$ be the n-vector $\mathbf{e}_i = (e_{i,1}, e_{i,2}, \ldots, e_{i,n})$ such that $e_{i,j} = \delta_{i,j}$.

Azar et al. [6] proved the following proposition.

Proposition 6.1. [6] *Let* $\mathbf{v}$ *and* $\mathbf{u}$ *be two n-vectors such that* $v_1 \geq v_2 \geq \cdots \geq v_n$ *and* $u_1 \geq u_2 \geq \cdots \geq u_n$. *Let i be an arbitrary integer,* $1 \leq i \leq n$. *If* $\mathbf{v} \succeq \mathbf{u}$, *then* $\mathbf{v} + \mathbf{e}_i \succeq \mathbf{u} + \mathbf{e}_i$.

We prove two extensions of this result.

Proposition 6.2. *Let* $\mathbf{v}$ *and* $\mathbf{u}$ *be two n-vectors such that* $v_1 \geq v_2 \geq \cdots \geq v_n$ *and* $u_1 \geq u_2 \geq \cdots \geq u_n$. *Let i and j be arbitrary distinct integers,* $1 \leq i < j \leq n$. *If* $\mathbf{v} \succeq \mathbf{u}$ *and* $u_i > u_j + 2(s-1)$ *for an integer s, then* $\mathbf{v} \succeq \mathbf{u} - s(\mathbf{e}_i - \mathbf{e}_j)$.

Proof. It is enough to prove the proposition for $s = 1$. Since $i < j$, we have $u_i \geq u_j$. If $u_i = u_j$ then clearly $\mathbf{u} = \mathbf{u} - s(\mathbf{e}_i - \mathbf{e}_j)$, and hence the proposition trivially holds. If $u_i - u_j = 1$ then similarly, $\mathbf{v} \succeq \mathbf{u}$ if and only if $\mathbf{v} \succeq \mathbf{u} - (\mathbf{e}_i - \mathbf{e}_j)$. For that, notice that if $\mathbf{u}^* = \mathbf{u} - (\mathbf{e}_i - \mathbf{e}_j)$ then $u_i^* = u_j$ and $u_j^* = u_i$.

Now consider the case when $u_i - u_j \geq 2$. Let $a = \max\{t \ : \ u_t = u_i\}$ and $b = \min\{t \ : \ u_t = u_j\}$. Now, clearly $\mathbf{v} \succeq \mathbf{u} - (\mathbf{e}_i - \mathbf{e}_j)$ if and only if $\mathbf{v} \succeq \mathbf{u} - (\mathbf{e}_a - \mathbf{e}_b)$. Let

$$u_t^* = \begin{cases} u_t & t \notin \{a, b\} \\ u_t - 1 & t = a \\ u_t + 1 & t = b. \end{cases}$$

Then $\mathbf{v} \succeq \mathbf{u} - (\mathbf{e}_i - \mathbf{e}_j)$ if and only if for each k, $1 \leq k \leq n$,

$$\sum_{t=1}^{k} v_t \geq \sum_{t=1}^{k} u_t^*.$$

Notice that

$$\sum_{t=1}^{k} u_t^* = \begin{cases} \sum_{t=1}^{k} u_t & \text{if } 1 \leq k < a \text{ or } b \leq k \leq n \\ \sum_{t=1}^{k} u_t - 1 & \text{if } a \leq k < b. \end{cases}$$

Therefore

$$\sum_{t=1}^{k} v_t \geq \sum_{t=1}^{k} u_t \geq \sum_{t=1}^{k} u_t^*.$$

Proposition 6.3. *Let* $\mathbf{v}$ *and* $\mathbf{u}$ *be two n-vectors such that* $v_1 \geq v_2 \geq \cdots \geq v_n$ *and* $u_1 \geq u_2 \geq \cdots \geq u_n$. *Let i and j be arbitrary distinct integers,* $1 \leq i < j \leq n$. *If* $u_i > u_j + 2k - 1$ *for some integer* $k \geq 1$, *then* $\mathbf{v} + \mathbf{e}_j \succeq \mathbf{u} - k\mathbf{e}_i + (k+1)\mathbf{e}_j$.

Proof. Let $\mathbf{v}^* = \mathbf{v} + \mathbf{e}_j$ and $\mathbf{u}^* = \mathbf{u} + \mathbf{e}_j$. By Proposition 6.1 we get $\mathbf{v}^* \succeq \mathbf{u}^*$. Now, apply Proposition 6.2 to $\mathbf{v}^*$ and $\mathbf{u}^*$ to get the claimed inequality.

Using these propositions we can prove the following.

Lemma 6.4. *The maximum load of any bin in ABKU*$[d]$ *stochastically dominates the maximum load of any bin in d-Balance.*

Proof. We present the proof only for $d = 2$; the proof for larger d is analogous. The proof is based on the coupling technique and it mimics the proof of Theorem 3.5 of Azar et al. [6].

Let $\lambda_i(t)$ be the load of bin i after time t in ABKU[2], and let $\lambda_i^*(t)$ be the load of bin i after time t in d-Balance. Let $\boldsymbol{\lambda}(t)$ and $\boldsymbol{\lambda}^*(t)$ be the bin-load vectors of ABKU[2] and 2-Balance, respectively, at time t; that is, $\boldsymbol{\lambda}(t) = (\lambda_1(t), \lambda_2(t), \ldots, \lambda_n(t))$ and $\boldsymbol{\lambda}^*(t) = (\lambda_1^*(t), \lambda_2^*(t), \ldots, \lambda_n^*(t))$.

Fix t. Let Ω_t be the set of all vectors of length $2t$ of elements in $\{1, \dots, n\}$. Any vector $\boldsymbol{\omega} = (\omega_1, \dots, \omega_{2t}) \in \Omega_t$ determines uniquely the $2t$ choices made by t balls, so that bins ω_{2r-1} and ω_{2r} were chosen by ball r. Our aim now is to show that there exists a 1–1 function $f_t \colon \Omega_t \to \Omega_t$, such that if $\boldsymbol{\omega}$ represents the choices made in ABKU[2] and $f_t(\boldsymbol{\omega})$ represents the choices made in 2-Balance, then the bin-load vector of ABKU[2] majorizes the bin-load vector of 2-Balance. This would complete the proof.

Our construction of f_t is by induction on t. For $t = 0$ we can take the identity function. Now assume that we have such a function f_t and we show how to construct function f_{t+1}. f_{t+1} restricted to the first $2t$ elements is equal to f_t. Thus we only have to determine f_{t+1} on the last two elements.

Fix $\boldsymbol{\omega} = (\omega_1, \dots, \omega_{2t})$. Let $\boldsymbol{\lambda}(t)$ be the bin-load vector of ABKU[2] for bins $\boldsymbol{\omega}$ and let $\boldsymbol{\lambda}^*(t)$ be the bin-load vector of 2-Balance for bins $f_t(\boldsymbol{\omega})$. To simplify the notation, let us assume that $\lambda_1(t) \geq \cdots \geq \lambda_n(t)$. Let $\pi = \pi_{\boldsymbol{\lambda}^*(t)}$ be a permutation such that $\lambda^*_{\pi(1)}(t) \geq \cdots \geq \lambda^*_{\pi(n)}(t)$. We set f_{t+1} so that, when restricted to the last two elements, f_{t+1} maps $(\omega_{2t+1}, \omega_{2t+2})$ into $(\pi(\omega_{2t+1}), \pi(\omega_{2t+2}))$. What we have to show now is that $\boldsymbol{\lambda}(t) \succeq \boldsymbol{\lambda}^*(t)$ implies that $\boldsymbol{\lambda}(t+1) \succeq \boldsymbol{\lambda}^*(t+1)$. Here $\boldsymbol{\lambda}(t+1)$ and $\boldsymbol{\lambda}^*(t+1)$ are the bin-load vectors of ABKU[2] and 2-Balance for bins $\boldsymbol{\omega}^* = (\omega_1, \dots, \omega_{2t+2})$ and $f_{t+1}(\boldsymbol{\omega}^*)$, respectively.

Assume, without loss of generality, that $i \leq j$. Then $\boldsymbol{\lambda}(t+1) = \boldsymbol{\lambda}(t) + \mathbf{e}_j$. Because $i \leq j$, we obtain that $\lambda^*_{\pi(i)}(t) - \lambda^*_{\pi(j)}(t) = r \geq 0$. Hence

$$\boldsymbol{\lambda}^*(t+1) = \boldsymbol{\lambda}^*(t) - \left\lfloor \frac{r}{2} \right\rfloor \cdot \mathbf{e}_{\pi(i)} + \left(\left\lfloor \frac{r}{2} \right\rfloor + 1 \right) \cdot \mathbf{e}_{\pi(j)}.$$

Now we can apply Proposition 6.3 to complete the proof. ■

Lemma 6.4 immediately yields the following.

Lemma 6.5. *Lemma* 6.5 *Let $d \geq 2$. The maximum load of bins in d-Balance is at most* $(1 + o(1)) \ln \ln n / \ln d + \mathcal{O}(m/n)$ *w.h.p. If $m = n$, then the maximum load of d-Balance is at most* $\ln \ln n / \ln d + \mathcal{O}(1)$ *w.h.p.*

Now we study the lower bound of d-Balance. We will show a rather surprising result that the maximum load of any bin in d-Balance is stochastically the same as in ABKU[d] within an additive constant term w.h.p. Actually, we prove a stronger result.

Let us consider a class of on-line allocation algorithms that we call "*d-reallocation-on-line-algorithms*," d-ROLA(m, n). Any d-ROLA(m, n) algorithm $\mathcal{A}$ places sequentially, possibly with reallocations, m balls into n bins one by one. Each placement of a ball consists of three steps:

(1) $\mathcal{A}$ chooses *at most d* bins i.u.r. with replacement.
(2) $\mathcal{A}$ places the ball into one of the chosen bins.
(3) $\mathcal{A}$ reassigns the balls from the chosen bins among these chosen bins.

In step (6) we require that $\mathcal{A}$ makes its decision on how to move the balls on-line without any knowledge about future choices of balls. Nevertheless, apart from that restriction $\mathscr{A}$ may reassign the balls arbitrarily (so it can also not reassign the balls at all), and we assume that $\mathscr{A}$ may know the current load of each bin.

Notice that ABKU[d] and d-Balance are trivially d-ROLA(m, n). We prove a general lower bound for the maximum load that holds for any d-ROLA(m, n) algorithm. (Observe that the following theorem implies Theorem 8 for $m = n$.)

Theorem 15. *Let $\mathscr{A}$ be a d–ROLA(n, n) algorithm. The maximum load of any bin in $\mathscr{A}$ is $\ln\ln n/\ln d + \Omega(1)$ w.h.p.*

Proof. Because the theorem is trivial for $d = \Omega(\ln n)$, we will proceed on with $d = o(\ln n)$. Notice also that we may assume, without loss of generality, that $\mathscr{A}$ chooses exactly d bins in step (2).

Let $\mathscr{B}_i$ be the set of bins with load at least i at time $n(1 - 2^{-i})$. Hence $|\mathscr{B}_i| = \nu_{\geq i}(n(1 - 2^{-i}))$. Denote by $\mathscr{F}_i$ the event that $|\mathscr{B}_i| \geq \gamma_i$ [or equivalently, that $\nu_{\geq i}(n(1 - 2^{-i})) \geq \gamma_i$], where γ_i will be exposed later. Let $\mathscr{I}_i = \{t \in \mathbb{N} : n(1 - 2^{-(i-1)}) < t \leq n(1 - 2^{-i})\}$. Denote by $\mathscr{L}_i$ the set of bins that are chosen exactly once during the $d \cdot |\mathscr{I}_i|$ trials of balls from $\mathscr{I}_i$. Let $\mathscr{C}_{i+1} = \mathscr{B}_i \cap \mathscr{L}_{i+1}$.

Suppose that $\mathscr{F}_i$ holds; that is, that after $n(1-2^{-i})$ steps at least γ_i bins have load at least i. Let us consider the bins in $\mathscr{C}_{i+1}$. Each such a bin is chosen exactly once during the interval time $\mathscr{I}_{i+1}$, and therefore it changes its load in $\mathscr{I}_{i+1}$ at most once. Suppose that a ball t in $\mathscr{I}_{i+1}$ chooses d bins from $\mathscr{C}_{i+1}$. Because the load of each chosen bin is at least i, at least one of the chosen bin will have load at least $i + 1$ at time $t + 1$. In that case, let the bin with the smallest index among the chosen d bins with the load at least $i + 1$ at time $t + 1$ be called the *representative*. Because the representative is chosen only once in $\mathscr{I}_{i+1}$, it belongs to $\mathscr{B}_{i+1}$. Let $\mathscr{D}_{i+1}$ be the set of representatives in the interval time $\mathscr{I}_{i+1}$. Clearly, $\mathscr{D}_{i+1} \subseteq \mathscr{B}_{i+1}$. Our aim is to estimate a lower bound for the size of $\mathscr{D}_{i+1}$. We will set the value of i^* and the sequence $\gamma_0, \gamma_1, \ldots, \gamma_i^*$ so that for every $i < i^*$, conditioned on $\mathscr{F}_i$, $|\mathscr{D}_{i+1}| \geq \gamma_{i+1}$ holds with high probability.

Fix i and let us consider all the balls from $\mathscr{I}_{i+1}$. For each bin j

$$\mathbf{Pr}[j \in \mathscr{L}_{i+1}] = d|\mathscr{I}_{i+1}|\frac{1}{n}\left(1-\frac{1}{n}\right)^{d|\mathscr{I}_{i+1}|-1} = d2^{-(i+1)}\left(1-\frac{1}{n}\right)^{nd2^{-(i+1)}-1} \geq d2^{-(i+1)}e^{-d}.$$

Therefore we let $q_i = d \cdot 2^{-(i+1)} \cdot e^{-d}$. Since $\mathbf{Pr}[j \in \mathscr{L}_{i+1}] \geq q_i$, we obtain

$$\mathbf{E}[|\mathscr{C}_{i+1}| \mid \mathscr{F}_i] \geq \gamma_i q_i.$$

We use the method of bounded differences (Lemma 2.4) to estimate deviations from the expected size of $\mathscr{C}_{i+1}$. Let $\omega_1, \omega_2, \ldots, \omega_{d\cdot|\mathscr{I}_{i+1}|}$ be the random choices made by the balls in $\mathscr{I}_{i+1}$. Then $|\mathscr{C}_{i+1}|$ is a function of $\omega_1, \omega_2, \ldots, \omega_{d\cdot|\mathscr{I}_{i+1}|}$. Since change of any ω_j may cause $|\mathscr{C}_{i+1}|$ to change by at most 2, Lemma 2.4 implies that

$$\mathbf{Pr}[|\mathscr{C}_{i+1}| \leq \gamma_i q_i/2 \mid \mathscr{F}_i] \leq \exp\left(-\frac{2\cdot(\gamma_i \cdot q_i/2)^2}{4\cdot d\cdot|\mathscr{I}_{i+1}|}\right) = \exp\left(-\frac{\gamma_i^2 \cdot d}{n\cdot e^{2d}\cdot 2^{i+4}}\right).$$

Notice now, that for any $c > 0$, if

$$\gamma_i \geq \sqrt{\frac{c\cdot n\cdot \ln n\cdot e^{2d}\cdot 2^{i+4}}{d}}, \tag{20}$$

then $\mathbf{Pr}[|\mathscr{C}_{i+1}| \leq \gamma_i q_i/2 \mid \mathscr{F}_i] \leq n^{-c}$.

Now we show that conditioned on $\mathscr{F}_i$, the size of $\mathscr{D}_{i+1}$ is larger than that of γ_{i+1} (which is to be set momentarily) w.h.p. We can estimate the size of $\mathscr{D}_{i+1}$ as

$$\mathbf{E}[|\mathscr{D}_{i+1}| \mid |\mathscr{C}_{i+1}| > \gamma_i q_i/2 \ \&\ \mathscr{F}_i] \geq |\mathscr{I}_{i+1}| \cdot \left(\frac{\gamma_i q_i/2}{n}\right)^d = n \cdot 2^{-(i+1)} \cdot \left(\frac{\gamma_i \cdot d}{2^{i+2} \cdot e^d \cdot n}\right)^d.$$

This motivates us to set $\gamma_0 = n$ and

$$\gamma_{i+1} = \frac{1}{2} \cdot n \cdot 2^{-(i+1)} \cdot \left(\frac{\gamma_i \cdot d}{2^{i+2} \cdot e^d \cdot n}\right)^d = \frac{\gamma_i^d \cdot d^d}{n^{d-1} \cdot e^{d^2} \cdot 2^{(i+2)(d+1)}}. \tag{21}$$

Hence $\mathbf{E}[|\mathscr{D}_{i+1}| \mid |\mathscr{C}_{i+1}| > \gamma_i q_i/2 \ \&\ \mathscr{F}_i] \geq 2\gamma_{i+1}$.

Now, one can show that

$$\gamma_{i+1} = n \cdot (d/e^d)(d/(d-1))(d^{i+1}-1) \cdot 2^{-((d+1)/(d-1)^2))(2d^{i+2}-d^{i+1}-d-(d-1)(i+2))}.$$

Provided that $d \geq 2$, simple estimations yield

$$\gamma_{i+1} \geq n/e^{d^{i+6}}. \tag{22}$$

Let, as before, $\omega_1, \omega_2, \ldots, \omega_{d \cdot |\mathscr{I}_{i+1}|}$ be the random choices made by the balls in $\mathscr{I}_{i+1}$. Then the change of any ω_j may cause $|\mathscr{D}_{i+1}|$ to change by at most 2. Hence we may use Lemma 2.4 to obtain

$$\mathbf{Pr}[|\mathscr{D}_{i+1}| \leq \gamma_{i+1} \mid |\mathscr{C}_{i+1}| > \gamma_i q_i/2 \,\&\, \mathscr{F}_i] \leq \exp\left(-\frac{2\gamma_{i+1}^2}{4 \cdot d \cdot |\mathscr{I}_{i+1}|}\right) = \exp\left(-\frac{\gamma_{i+1}^2 \cdot 2^i}{d \cdot n}\right).$$

Therefore we obtain that for any $c > 0$, if

$$\gamma_{i+1} \geq \sqrt{c \cdot d \cdot n \cdot \ln n/2^i}\ , \tag{23}$$

then $\mathbf{Pr}[|\mathscr{D}_{i+1}| \leq \gamma_{i+1} \mid |\mathscr{C}_{i+1}| > \gamma_i q_i/2 \ \&\ \mathscr{F}_i] \leq n^{-c}$.

Fix c. Let i^* be the largest i that satisfies (20) and (23). Inequality (22) implies that $i^* = \ln\ln n/\ln d - \Theta(1)$. Therefore we obtain that for each $i \leq i^*$, $\mathbf{Pr}[|\mathscr{C}_{i+1}| \leq \gamma_i q_i/2 \mid \mathscr{F}_i] < n^{-c}$ and $\mathbf{Pr}[\neg\mathscr{F}_{i+1} \mid |\mathscr{C}_{i+1}| \leq \gamma_i q_i/2 \ \&\ \mathscr{F}_i] < n^{-c}$. Combining these bounds for all $i < i^*$, we obtain

$$\mathbf{Pr}[\neg\mathscr{F}_{i^*}] = n^{-(c-1)}.$$

Now we shall show that, conditioned on $\mathscr{F}_{i^*}$, at least one bin from $\mathscr{B}_{i^*}$ will not be chosen by the last $n/2^{i^*}$ balls with high probability. Since such a bin has at time $n - n/2^{i^*}$ the load greater than or equal to i^* and since it is not chosen by the last $n/2^{i^*}$ balls, such a bin is never rebalanced again and it has the load of at least i^* at the end of the process. This would complete the proof of the theorem.

Notice that condition (23) combined with (21) yields

$$\frac{\gamma_{i^*}^d \cdot d^d}{n^{d-1} \cdot e^{d^2} \cdot 2^{(i^*+2)(d+1)}} \geq \sqrt{c \cdot d \cdot n \cdot \ln n/2^{i^*}},$$

which is equivalent to

$$\begin{aligned}\gamma_{i^*}^{2d} &\geq n^{2d-1} \cdot \frac{c \cdot d \cdot \ln n \cdot e^{2d^2} \cdot 2^{2(i^*+2)(d+1)}}{d^{2d} \cdot 2^{i^*}} \\ &= n^{2d-1} \cdot \left(c \cdot d \cdot \ln n \cdot (e^d/d)^{2d} \cdot 2^{2di^*+i^*+4d+4}\right) \\ &\geq n^{2d-1}.\end{aligned}$$

Hence $\gamma_{i^*} \geq n^{1-1/2d} \geq n^{3/4}$ for $d \geq 2$. Therefore, conditioned on $\mathcal{F}_{i^*}$, $|\mathcal{B}_{i^*+1}| \geq n^{3/4}$. The last $n/2^{i^*}$ balls perform $dn/2^{i^*} = o(n \ln n)$ random choices. Hence, the standard calculations as, for example, in the coupon's collector problem (see, e.g., [21, 23, 33]) can be used to show that with high probability at least one element from $\mathcal{B}_{i^*}$ will be not chosen by any of the last $n/2^{i^*}$ balls. This completes the proof of the theorem. ■

We can extend Theorem 15 to deal with $\alpha > 1$. This clearly implies Theorem 8.

Theorem 16. *Let $\mathcal{A}$ be a d-*ROLA*(m, n) algorithm. The maximum load of any bin in $\mathcal{A}$ is $\Omega(\ln \ln n/\ln d + m/n)$ w.h.p.*

Proof. Since the theorem is obvious for $m/n = \Omega(\ln \ln n/\ln d)$, we only must show that if $m = \mathcal{O}(n \ln \ln n/\ln d)$, then the maximum load is $\Omega(\ln \ln n/\ln d)$ w.h.p.

Using arguments similar to those in the proof of Theorem 15 we may assume that $d = o(\sqrt{\ln n})$ w.h.p. Then we use exactly the same calculations as in the proof of Theorem 15 to get that for $i^* = \ln \ln n/\ln d - \Theta(1)$ we have $|\mathcal{B}_{i^*+1}| \geq n^{3/4}$ w.h.p. The last $m - (n(1 - 2^{-i^*}))$ balls will made $\mathcal{O}(md) = o(n \ln n)$ trials. Therefore one can easily show that the last $m - (n(1 - 2^{-i^*}))$ balls will not choose all the bins in $\mathcal{B}_{i^*+1}$ w.h.p., which completes the proof. ■

Proof of Theorem 7: Combine the bounds for the maximum loads of ABKU[d] from [6] with Lemma 6.4, Theorem 15, and Theorem 16. ■

7. OFF-LINE ALLOCATIONS

A *random (m, n, d)-problem* is the allocation problem in which each (out of m) ball must be placed into one of d bins that are chosen (among n bins) i.u.r. with replacement. Azar et al. [6, Lemma 6.1] showed that if we know in advance the random choices performed in the $(n, n, 2)$-problem (i.e., we consider the off-line problem), then with high probability, one may allocate n balls with maximum constant load (this result can be also derived from Lemma 6.3 in [22]). In this section we extend this bound and prove Theorem 9 which states that the maximum load in the $(m, n, 2)$-problem is 2 w.h.p., for all $m \leq 1.675 \cdot n$. We first consider an *orientation problem* in multigraphs and then relate it to the statement of Theorem 9.

Let us define a *k-core* (cf. [34]) of a multigraph as the unique maximal subgraph with minimum degree at least k. If the k-core is edgeless, then we say the graph has no k-core. We will use the following simple fact.

Proposition 7.1. *If a multigraph G does not have a k-core then one may orient the edges of G such that each vertex will have the out-degree bounded by $k-1$.*

Proof. It is easy to see (cf. also Section 3 in [34]) that the following algorithm, when it starts with a multigraph G finds the k-core of G:

Repeat until possible:

- Pick any vertex of degree deg $\in \{0, 1, \ldots, k-1\}$ and remove it from the graph.

Therefore, if G does not have a k-core, then the procedure above will end up with the empty graph. Now, note that we can use this algorithm to orient the edges of G. Each time a vertex v is removed, the edges incident to v at the given time are oriented out of v. Therefore, if G does not have a k-core, then this procedure orients the edges of G such that each vertex is of out-degree at most $k-1$.

Now we shall estimate the probability that a multigraph does not have a k-core. Let $M_{n,m}$ be a *random multigraph* on n vertices with m ordered edges, where the endpoints of each edge are chosen independently with replacement ($M_{m,n}$ may contain self-loops and parallel edges). Pittel et al. [34] proved the following lemma.

Lemma 7.2. [34] *Let $k \geq 3$ be a fixed integer and let*

$$\gamma_k = \inf_{\lambda > 0} \left\{ \frac{\lambda}{1 - \sum_{i=0}^{k-2} \frac{\lambda^i}{e^\lambda i!}} \right\}.$$

Let $0 < \epsilon < \frac{1}{2}$ and let $c \leq \gamma_k - n^{-\epsilon}$. Then the probability that $M_{n,cn/2}$ has a k-core is $\mathcal{O}(n^{-(k-1)(k+2)/2})$.

In particular, if $m \leq 1.675459 \cdot n$, then the probability that $M_{n,m}$ has a 3-core is $\mathcal{O}(n^{-2})$.

Remark 2. This lemma is presented in [34] only implicitly, since the main interest of the authors of [34] was in studying random graphs $\mathcal{G}_{n,m}$.

Corollary 7.3. *Let M be a random multigraph with $m \leq (\gamma_3 - n^{-((1/2)-\epsilon)})n/2$ edges. Then, with probability $1 - \mathcal{O}(n^{-2})$, one may orient the edges of M so that each vertex will have out-degree at most 2.*

Remark 3. In the preliminary version of this paper [15], we presented a weaker result (with a more complicated proof) that held only for $c = 1$. Afterward we found that the results from the paper of Pittel et al. [34] concerning the threshold for the existence of the k-core in random graphs can be applied to our problem.

Proof of Theorem 9. One can model the $(m, n, 2)$-problem as a multigraph M with n vertices and m edges. The vertex set of M corresponds to the set of bins. If (x, y) is the pair of bins chosen for the ith ball, then $\{x, y\}$ is the ith undirected edge of M. Observe that if the bins are generated i.u.r., then the obtained multigraph M is random, according to our definition above. Now we use Corollary 7.3 to orient the edges of M, so that each vertex has out-degree at most 2 w.h.p. We assign the balls to the bins using this orientation. If the ith edge, (x, y), is directed from x to y, then we place ball i into bin x. Because there are at most two edges outgoing from x, the load of bin x is at most 2. ■

We can also easily show that even for the off-line processes the maximum load cannot be 1 w.h.p.

Theorem 17. *Suppose that n balls are to be distributed, each having d out of n possible locations chosen i.u.r. If $d = \ln n - \omega(\ln \ln n)$, then with probability at least $1 - 1/n$ there is no placement of n balls in n bins such that the maximum load is* 1.

Proof. Observe that in order to place n balls into n bins each bin must be chosen in at least one of dn random choices of the balls' destinations. An analysis similar to that of the coupon collector's problem (see, e.g., [33, p. 63, 21, 23]) implies that if $d = \ln n - \omega(\ln \ln n)$ then the probability that all bins have been chosen is smaller than $1/n$. ■

REFERENCES

[1] M. Adler, P. Berenbrink, and K. Schröder, Analyzing an infinite parallel job allocation process, Proc 6th Annual European Symp Algorithms, Venice, Italy, Vol. 1461 of Lecture Notes in Computer Science, (G. Bilardi, G. F. Italiano, A. Pietracaprina, and G. Pucci, Editors), Springer-Verlag, Berlin, 1998, pp. 417–428.

[2] M. Adler, S. Chakrabarti, M. Mitzenmacher, and L. Rasmussen, Parallel randomized load balancing, Random Struc Algorithms, 13 (1998), 159–188. A preliminary version appeared in Proc 27th Annual ACM Symp Theory of Computing, Las Vegas, NV, ACM Press, New York, 1995, pp. 238–247.

[3] M. Ajtai, J. Aspnes, M. Naor, Y. Rabani, L.J. Schulman, and O. Waarts, Fairness in scheduling, J Algorithms 29 (1998), 306–357. A preliminary version appeared in Proc 6th Annual ACM-SIAM Symp Discrete Algorithms, San Francisco, SIAM, Philadelphia, 1995, pp. 477–485.

[4] D.J. Aldous, Probability approximations via the Poisson clumping heuristic, Springer-Verlag, New York, 1989.

[5] B. Awerbuch, Y. Azar, S. Plotkin, and O. Waarts, Competitive routing of virtual circuits with unknown duration, Proc 5th Annual ACM-SIAM Symp Discrete Algorithms, Philadelphia, SIAM, Philadelphia, 1994, pp. 321–327.

[6] Y. Azar, A.Z. Broder, A.R. Karlin, and E. Upfal, Balanced allocations, SIAM Comput, 29 (1999), 180–200. A preliminary version appeared in Proc 26th Annual ACM Symp Theory of Computing, Montreal, ACM Press, New York, 1994, pp. 593–602.

[7] A.D. Barbour, L. Holst, and S. Janson, Poisson approximation, Oxford University Press, New York, 1992.

[8] P. Berenbrink, A. Czumaj, T. Friedetzky, and N.D. Vvedenskaya, On the analysis of infinite parallel job allocation processes via differential equations, Proc 12th Annual ACM Symp Parallel Algorithms and Architectures, Bar Harbor, ME, ACM Press, New York, 2000, pp. 99–108

[9] P. Berenbrink, F. Meyer auf der Heide, and K. Schröder, Allocating weighted jobs in parallel, Theory Comput Syst, 32 (1999) 281–300. A preliminary version appeared in Proc 9th Annual ACM Symp Parallel Algorithms and Architectures, Newport, RI, ACM Press, New York, 1997, pp. 302–310.

[10] A. Borodin and R. El-Yaniv, Online computation and competitive analysis, Cambridge University Press, New York, 1998.

[11] R. Cole, A. Frieze, B.M. Maggs, M. Mitzenmacher, A.W. Richa, R.K. Sitaraman, and E. Upfal, On balls and bins with deletions, Proc 2nd Int Workshop on Randomization and Approximation Techniques in Computer Science, Barcelona, Spain, Vol. 1518 of Lecture Notes in Computer Science, (M. Luby, J. Rolim, and M. Serna, Editors), Springer-Verlag, Berlin, 1998, pp. 145–158.

[12] R. Cole, B.M. Maggs, F. Meyer auf der Heide, M. Mitzenmacher, A.W. Richa, K. Schröder, R.K. Sitaraman, and B. Vöcking, Randomized protocols for low-congestion circuit routing in multistage interconnection networks, Proc 30th Annual ACM Symp Theory of Computing, Dallas, ACM Press, New York, 1998, pp. 378–388.

[13] R.M. Corless, G.H. Gonnet, D.E.G. Hare, D.J. Jeffrey, and D.E. Knuth, On the Lambert W function, Adv Comput Math 5 (1996), 329–359.

[14] A. Czumaj, Recovery time of dynamic allocation processes, Proc 10th Annual ACM Symp Parallel Algorithms and Architectures, Puerto Vallarta, Mexico, ACM Press, New York, 1998, pp 202–211.

[15] A. Czumaj and V. Stemann, Randomized allocation processes, Proc 38th IEEE Symp Foundations of Computer Science, Miami Beach, IEEE Computer Society Press, Los Alamitos, CA, 1997, pp. 194–203.

[16] D. Dubhashi and D. Ranjan, Balls and bins: a study in negative dependence, Random Struc Algorithms, 13 (1998) 99–124.

[17] T. Hagerup and C. Rüb, A guided tour of Chernoff bounds, Inform Process Lett, 33 (1990), 305–308.

[18] W. Hoeffding, Probability inequalities for sums of bounded random variables, J Amer Statist Assoc, 58(301) (1963), 13–30.

[19] M. Hofri, Analysis of algorithms: computational methods and mathematical tools, Oxford University Press, New York, 1995.

[20] S. Irani and A.R. Karlin, Online computation, Approximation algorithms for $\mathscr{NP}$-hard problems (D.S. Hochbaum, Editor), Chap. 13, PWS Publishing Company, Boston, 1996, pp. 521–564.

[21] N.L. Johnson and S. Kotz, Urn models and their application: an approach to modern discrete probability theory, John Wiley & Sons, New York, 1977.

[22] R.M. Karp, M. Luby, and F. Meyer auf der Heide, Efficient PRAM simulation on a distributed memory machine, Algorithmica 16 (1996) 517–542. A preliminary version appeared in Proc 24th Annual ACM Symp Theory of Computing, Victoria, British Columbia, ACM Press, New York, 1992, pp. 318–326.

[23] V.F. Kolchin, B.A. Sevast'yanov, and V.P. Chistyakov, Random allocations, VH Winston & Sons, Washington, DC, 1978.

[24] R. Lüling and B. Monien, A dynamic distributed load balancing algorithm with provable good performance, Proc 5th Annual ACM Symp Parallel Algorithms and Architectures, Velen, Germany, ACM Press, New York, 1993, pp. 164–172,

[25] P.D. MacKenzie, C.G. Plaxton, and R. Rajaraman, On contention resolution protocols and associated probabilistic phenomena, J ACM, 45 (1998) 324-378. A preliminary version appeared in Proc 26th Annual ACM Symp Theory of Computing, Montreal, ACM Press, New York, 1994, pp. 153–162.

[26] C. McDiarmid, On the method of bounded differences, Surveys in Combinatorics, (J. Siemons, Editor), London Mathematical Society Lecture Note Series 141, Cambridge University Press, Cambridge, UK, 1989, pp. 148–188.

[27] M. Mitzenmacher, Load balancing and density dependent jump Markov processes, Proc 37th IEEE Symp Foundations of Computer Science, Burlington, VT, IEEE Computer Society Press, Los Alamitos, CA, 1996, pp. 213–222.

[28] M. Mitzenmacher, How useful is old information, Proc 16th ACM Symp Principles of Distributed Computing, Santa Barbara, CA, ACM Press, New York, 1997, pp. 83–91.

[29] M. Mitzenmacher, On the analysis of randomized load balancing schemes, Proc 9th Annual ACM Symp Parallel Algorithms and Architectures, Newport, RI, ACM Press, New York, 1997, pp. 292–301.

[30] M. Mitzenmacher, Studying balanced allocations with differential equations, Combin, Probab Comput 8 (1999) 473–482.

[31] M. Mitzenmacher, Analyses of load stealing models based on differential equations, Proc 10th Annual ACM Symp Parallel Algorithms and Architectures, Puerto Vallarta, Mexico, ACM Press, New York, 1998, pp. 212–221.

[32] M.D. Mitzenmacher, The power of two choices in randomized load balancing, PhD thesis, Computer Science Department, University of California at Berkeley, September 1996.

[33] R. Motwani and P. Raghavan, Randomized algorithms, Cambridge University Press, New York, 1995.

[34] B. Pittel, J. Spencer, and N. Wormald, Sudden emergence of a giant k-core in a random graph, J Combin Theory Ser A, (1996), 111–151.

[35] M. Raab and A. Steger, "Balls into bins"—a simple and tight analysis, Proc 2nd Int Workshop on Randomization and Approximation Techniques in Computer Science, Barcelona, Spain, Vol. 1518 of Lecture Notes in Computer Science, (M. Luby, J. Rolim, and M. Serna, Editors), Springer-Verlag, Berlin, 1998, pp. 159–170.

[36] V. Stemann, Parallel balanced allocations, Proc 8th Annual ACM Symp Parallel Algorithms and Architectures, Padua, Italy, ACM Press, New York, 1996, pp. 261–269.

[37] B. Vöcking, How asymmetry helps load balancing, Proc 40th IEEE Symp Foundations of Computer Science, New York City, IEEE Computer Society Press, Los Alamitos, CA, 1999, pp. 131–141.

[38] N.D. Vvedenskaya, R.L. Dobrushin, and F.I. Karpelevich, Queueing system with selection of the shortest of two queues: An asymptotic approach, Problems Inform Transmission 32(1) (1996), 15–27.

[39] C. Xu, S. Tschöke, and B. Monien, Performance evaluation of load distribution strategies in parallel branch and bound computations, Proc 7th IEEE Symp Parallel and Distributed Processing, 1995, pp. 402–405.