Variable-Processor Cup Games

William Kuszmaul*[1], Alek Westover†[1]

[1]MIT
kuszmaul@mit.edu, alek.westover@gmail.com

Abstract

In the ***cup game*** two players, the ***filler*** and the ***emptier***, take turns adding and removing water from a set of cups, subject to certain constraints. In the classic p-processor cup game the filler distributes p units of water among the n cups, giving at most 1 unit of water to any particular cup, and the emptier chooses p cups to remove at most 1 unit of water from. Analysis of the cup game is important for applications in processor scheduling, buffer management in networks, quality of service guarantees, and deamortization.

We consider a new variant of the classic p-processor cup game in which the resources of the emptier and filler are variable: in our variant of the game, the ***variable-processor cup game***, the filler is allowed to *change* p at the beginning of each round. Surprisingly, we found that the variable-processor cup game is fundamentally different from the classic fixed-resources version of the game.

We give an adaptive filling strategy that achieves backlog $\Omega(n^{1-\varepsilon})$ for any constant $\varepsilon > 0$ of our choice in running time $2^{O(\log^2 n)}$. This is enormous compared to in the p-processor cup game where an emptier can prevent backlog from exceeding $O(\log n)$. We also present an adaptive filling strategy that is able to achieve backlog $\Omega(n)$ in running time $O(n!)$.

We demonstrate, using a novel set of invariants, that a greedy emptier never lets backlog exceed $O(n)$; this matches our lower bound, so our analysis is tight.

We also give an oblivious filling strategy that achieves backlog $\Omega(n^{1-\varepsilon})$ for $\varepsilon > 0$ constant of our choice in time $2^{\operatorname{polylog} n}$ against any "greedy-like" emptier with probability at least $1 - 2^{-\operatorname{polylog}(n)}$. Being oblivious, i.e. not being able to observe the game state, seems a large disadvantage, but in the variable-processor cup game the lower bound is the same as in the adaptive case!

1 Introduction

Definition and Motivation. The ***cup game*** is a multi-round game in which the two players, the ***filler*** and the ***emptier***, take turns adding and removing water from cups. The ***backlog*** at a state is the fill in the fullest cup; the emptier tries to minimize backlog while the filler tries to maximize backlog. On each round of the classic ***p-processor cup game*** on n cups, the filler first distributes p units of water among the n cups with at most 1 unit to any particular cup (without this restriction the filler can trivially achieve unbounded backlog by placing all of its fill in a single cup every round), and then the emptier removes at most 1 unit of water from each of p cups.[1] The game has been studied for ***adaptive*** fillers, i.e. fillers that can observe the emptier's actions, and for ***oblivious*** fillers, i.e. fillers that cannot observe the emptier's actions.

The cup game naturally arises in the study of processor-scheduling. The incoming water added by the filler represents work added to the system at time steps. At each time step after the new work comes in, each of p processors must be allocated to a task which they will achieve 1 unit of progress on before the next time step. The assignment of processors to tasks is modeled by the emptier deciding which cups to empty from. The backlog of the system is the largest amount of work left on any given task; in the cup game the ***backlog*** of the cups is the fill of the fullest cup at a given state. In analyzing a cup game we aim to prove upper and lower bounds on backlog.

*Supported by a Hertz fellowship and a NSF GRFP fellowship
†Supported by MIT
[1]Note that negative fill is not allowed, so if the emptier empties from a cup with fill below 1 that cup's fill becomes 0.

Previous Work. The bounds on backlog are well known for the case where $p = 1$, i.e. the ***single-processor cup game***. In the single-processor cup game an adaptive filler can achieve backlog $\Omega(\log n)$ and a greedy emptier never lets backlog exceed $O(\log n)$. In the randomized version of the single-processor cup game, i.e. when the filler is oblivious, which can be interpreted as a smoothed analysis of the deterministic version, the emptier never lets backlog exceed $O(\log \log n)$, and a filler can achieve backlog $\Omega(\log \log n)$.

Recently Kuszmaul has established bounds on the case where $p > 1$, i.e. the ***multi-processor cup game*** [4]. Kuszmaul showed that a greedy emptier never lets backlog exceed $O(\log n)$. He also proved a lower bound of $\Omega(\log(n - p))$ on backlog. Recently we showed a lower bound of $\Omega(\log n - \log(n - p))$. Combined, these lower bounds bounds imply a lower bound of $\Omega(\log n)$. Kuszmaul also established an upper bound of $O(\log \log n + \log p)$ against oblivious fillers, and a lower bound of $\Omega(\log \log n)$. Tight bounds on backlog against an oblivious filler are not yet known for large p.

The Variable-Processor Cup Game. We investigate a new variant of the classic p-processor cup game which we call the ***variable-processor cup game***. In the variable-processor cup game the filler is allowed to change p (the total amount of water that the filler adds, and the emptier removes, from the cups per round) at the beginning of each round. Note that we do not allow the resources of the filler and emptier to vary separately; just like in the classic cup game we take the resources of the filler and emptier to be identical. This restriction is crucial; if the filler has more resources than the emptier, then the filler could trivially achieve unbounded backlog, as average fill will increase by at least some positive constant at each round. Taking the resources of the players to be identical makes the game balanced, and hence interesting.

The variable-processor cup game models the natural situation where many users are all on a server, and the number of processors allocated to each user is variable as other users get some portion of the processors.

A priori having variable resources offers neither player a clear advantage: lower values of p mean that the emptier is at more of a discretization disadvantage but also mean that the filler can "anchor" fewer cups. [2] Furthermore, at any fixed value of p upper bounds have been proven. For instance, regardless of p a greedy emptier prevents an adaptive filler from having backlog greater than $O(\log n)$ [4]. Switching between different values of p, all of which the filler cannot individually use to get backlog larger than $O(\log n)$ is not obviously going to help the filler achieve larger backlog. We hoped that the variable-processor cup game could be simulated in the classic multi-processor cup game, because the extra ability given to the filler does not seem very strong.

However, we show that attempts at simulating the variable-processor cup game are futile because the variable-processor cup game is vastly different from the classic multi-processor cup game.

Outline and Results. In Section 2 we establish the conventions and notations we will use to discuss the variable-processor cup game.

Many of the proofs in this paper are quite complicated. In Section 3 we provide proof sketches with the main ideas of the proofs; this is helpful for understanding the main ideas of the paper without all the details.

In Section 4 we provide an inductive proof of a lower bound on backlog with an adaptive filler. Theorem 5 states that a filler can achieve backlog $\Omega(n^{1-\varepsilon})$ for any constant $\varepsilon > 0$ in quasi-polynomial running time. Proposition 4 also provides an extremal strategy that achieves backlog $\Omega(n)$ in incredibly long games: it has $O(n!)$ running time.

In Section 5 we prove a novel invariant maintained by the greedy emptier. In particular Theorem 6 establishes that a greedy emptier keeps the average fill of the k fullest cups at most $2n - k$. In particular this implies (setting $k = 1$) that a greedy emptier prevents backlog from exceeding $O(n)$.

The lower bound and upper bound agree; our analysis is tight for adaptive fillers!

In Section 6 we prove a lower bound on backlog with an oblivious filler. Theorem 7 states that an oblivious filler can achieve backlog $\Omega(n^{1-\varepsilon})$ for any constant $\varepsilon > 0$ in quasi-polynomial time with probability at least $1 - 2^{-\operatorname{polylog}(n)}$. Theorem 7 only applies to a certain class of emptiers: "greedy-like emptiers". Nonetheless, this class of emptiers is very interesting; it contains the emptiers that are used in upper bound proofs. It is shocking that randomization doesn't help the emptier in this game; being oblivious seems like a large disadvantage for the filler!

[2] A useful part of many filling algorithms is maintaining an "anchor" set of "anchored" cups. The filler always places 1 unit of water in each anchored cup. This ensures that the fill of an anchored cup never decreases after it is placed in the anchor set.

2 Preliminaries

The cup game consists of a sequence of rounds. On the t-th round, the state starts as S_t. To start, the filler chooses the number of processors p_t for the round. Next, the filler distributes p_t units of water among the cups (with at most 1 unit of water to any particular cup). Now the game is in an intermediate state on round t, which we call state I_t. Finally the emptier chooses p_t cups to empty at most 1 unit of water from, which marks the conclusion of round t. The state is then S_{t+1}.

Note that if the emptier empties from a cup c on round t with fill at I_t less than 1, then c now has 0 fill (not negative fill); we say that the emptier ***zeroes out*** c on round t. Note that on any round where the emptier zeroes out a cup the emptier has removed less total fill from the cups than the filler has added to the cups; hence the average fill of the cups has increased.

We denote the fill of a cup c at state S by $\text{fill}_S(c)$. Let the ***mass*** of a set of cups X at state S be $m_S(X) = \sum_{c\in X} \text{fill}_S(c)$. Denote the average fill of a set of cups X at state S by $\mu_S(X)$. Note that $\mu_S(X)|X| = m_S(X)$. Let the ***backlog*** at state S be $\max_c \text{fill}_S(c)$, let the ***anti-backlog*** at state S be $\min_c \text{fill}_S(c)$. Let the ***rank*** of a cup at a given state be its position in a list of the cups sorted by fill at the given state, breaking ties arbitrarily but consistently. For example, the fullest cup at a state has rank 1, and the least full cup has rank n. Let $[n] = \{1, 2, \dots, n\}$, let $i+[n] = \{i+1, i+2, \dots, i+n\}$. For a state S, let $S(r)$ denote the rank r cup at state S, and let $S(\{r_1, r_2, \dots, r_m\})$ denote the set of cups of ranks $r_1, r_2, \dots r_m$.

As a tool in the analysis we define a new variant of the cup game: the ***negative-fill*** cup game. In the negative-fill cup game, when the emptier empties from a cup its fill always decreases by exactly 1, i.e. there is no zeroing out. We refer to the standard version of the cup game where cups can zero out as the ***standard-fill*** cup game when necessary for clarity. Negative fill can be interpreted as fill below average fill. Negative fill is especially useful in our recursive lower bound proofs in which we build on the average fill already achieved. Note that it is strictly easier for the filler to achieve high backlog in the standard-fill cup game than in the negative-fill cup game; hence a lower bound on backlog in the negative-fill cup game also serves as a lower bound on backlog in the standard-fill cup game. On the other hand, during the upper bound proof we use the standard-fill cup game: this makes it harder for the emptier to guarantee its upper bound.

3 Technical Overview

In this section we provide sketches of the proofs in the paper, omitting details.

3.1 Adaptive Lower Bound

In Section 4 we provide filling strategies that an adaptive filler can use to achieve backlog $\text{poly}(n)$; in this subsection we sketch the proofs of these results.

First we note that there is a trivial algorithm, that we call **trivalg**, for achieving backlog at least $1/2$ on at least 2 cups in time $O(1)$.

The essential ingredient in our polynomial lower bound proof is the ***Amplification Lemma*** which gives a way to improve a function for achieving a certain backlog.

Lemma 1. *Let $\text{alg}(f)$ be a filling strategy that achieves backlog $f(n)$ on n cups. There exists a filling strategy $\text{alg}(f')$, the **amplification** of $\text{alg}(f)$, that achieves backlog at least*

$$f'(n) \geq (1-\delta)f(\lfloor(1-\delta)n\rfloor) + f(\lceil\delta n\rceil).$$

Amplification Lemma Proof Sketch. The filler designates an ***anchor set*** A of size $\lceil\delta n\rceil$ and a ***non-anchor set*** B of size $\lfloor(1-\delta)n\rfloor$. The filler's strategy is as follows:

Step 1: Make $\mu(A) \geq (1-\delta)f(|B|)$ by using $\text{alg}(f)$ repeatedly on B while placing 1 unit of fill into each anchor cup each round; if the emptier does not place extra resources into B this results in the filler achieving a cup with large fill in B which it can swap into A causing $\mu(A)$ to increase, if the emptier does place extra resources into B to prevent a cup in B from gaining high fill, then the emptier is correspondingly placing fewer resources into A causing $\mu(A)$ to increase.

Step 2: Use $\text{alg}(f)$ once on A to obtain some cup with fill $\mu(A) + f(|A|)$.

Note that in order to use $\text{alg}(f)$ on subsets of the cups the filler will need to vary p.

Consider Step 1. It can be shown that by using $\text{alg}(f)$ at most $|A|$ times on B the average fill of A will have increased to at least $(1-\delta)f(|B|)$ due either to the filler swapping many cups with high fill into

A, or to the emptier repeatedly using more resources in B and less in A than the filler. Step 2 trivially succeeds. Thus the filler achieves the desired backlog:

$$(1-\delta)f(|B|)+f(|A|).$$

$\square$

We use the Amplification Lemma to give two lower bounds on backlog: one with reasonable running time, the other with slightly better backlog.

Theorem 1. *There is an adaptive filling strategy for achieving backlog $\Omega(n^{1-\varepsilon})$ for constant $\varepsilon \in (0,1/2)$ in running time $2^{O(\log^2 n)}$.*

Theorem 5 Proof Sketch. We construct a sequence of filling strategies with $\text{alg}(f_{i+1})$ the amplification of $\text{alg}(f_i)$ using $\delta = \Theta(1)$ determined as a function of ε, and $\text{alg}(f_0) = \text{trivalg}$. Choosing δ appropriately, we show by induction on i that $\text{alg}(f_{\Theta(\lg n)})$ achieves backlog $\Omega(n^{1-\varepsilon})$ in running time $2^{O(\log^2 n)}$. $\square$

Theorem 2. *There is an adaptive filling strategy for achieving backlog $\Omega(n)$ in running time $O(n!)$.*

Proposition 4 Proof Sketch. We construct a sequence of filling strategies with $\text{alg}(f_{i+1})$ the amplification of $\text{alg}(f_i)$ using $\delta = 1/(i+1)$, and $\text{alg}(f_0)$ a filling strategy for achieving backlog 1 on $O(1)$ cups in $O(1)$ time (this is a slight modification of trivalg). We show by induction on i that $\text{alg}(f_{\Theta(n)})$ achieves backlog $\Omega(n)$ in running time $O(n!)$. $\square$

3.2 Upper Bound

In Section 5 we prove that a greedy emptier, i.e. an emptier that always empties from the p fullest cups, never lets backlog exceed $O(n)$; in this subsection we sketch the proof of this result.

The upper bound on backlog is derived by setting $k = 1$ in Theorem 3.

Theorem 3. *The average fill of the k fullest cups never exceeds $2n - k$.*

Theorem 6 Proof Sketch. The proof is by induction on the round. Fix some round t and assume that all invariants hold on round t. Fix some k; we aim to prove that the average fill of the k fullest cups is at most $2n - k$ at the start of round $t + 1$.

Let A be the cups that are among the k fullest cups in I_t, are emptied from, and are among the k fullest cups in S_{t+1}. Let B be the cups that are among the k fullest cups in state I_t, are emptied from, and are not among the k fullest cups in S_{t+1}. Let C be the cups with ranks $|A| + |B| + 1, \ldots, k + |B|$ in state I_t. The set C is defined so that the k fullest cups in state S_{t+1} are AC, since once the cups in B are emptied from, the cups in B are not among the k fullest cups, so cups in C take their places among the k fullest cups.

It can be shown that we may assume without loss of generality that the rank r cup at state S_t is also the rank r cup at state I_t for all ranks $r \in [n]$, by changing the labels of the cups; intuitively this is true because if a cup c changes ranks from S_t to I_t, then some other cup must have fill very close to c's fill.

We prove the invariant by considering several cases.

Case 1: Some cup in A zeroes out in round t.
Analysis: The fill of all cups in C must be at most 1 at state I_t to be less than the fill of the cup in A that zeroed out. Further, A has average fill at most $2n - (a - 1)$ due to the cup with zero fill. Combined, with some algebra, these facts imply that the average fill in AC is not too large, in particular not larger than $2n - k$.
Case 2: No cups in A zero out in round t and $b = 0$.
Analysis: In this case the set of cups of ranks in $[k]$ at state S_t is the same as the set of cups of ranks in $[k]$ at state S_{t+1}, and these are both A. During round t the emptier removes a units of fill from the cups A. The filler cannot have added more than k fill to these cups, because it can add at most 1 fill to any given cup. Also, the filler cannot have added more than p_t fill to the cups because this is the total amount of fill that the filler is allowed to add. Hence the filler adds at most $\min(p_t, k) = a+b = a+0 = a$ fill to these cups. The emptier thus is removing at least as much water as the filler is adding to these cups, so the average fill has not increased, and is still at most $2n - k$.
Case 3: No cups in A zero out on round t and $b > 0$.
Analysis: Consider $m_{S_{t+1}}(AC)$, which is the mass of the k fullest cups at state S_{t+1}. Each cup in A was emptied from. The filler adds at most $\min(k, p_t) = a + b$ fill to these cups. Hence,

$$m_{S_{t+1}}(AC) \le m_{S_t}(AC) + b. \tag{1}$$

The key insight necessary to bound $m_{S_t}(AC)$ is to notice that larger values for $m_{S_t}(A)$ correspond to smaller values for $m_{S_t}(C)$ because the invariants are satisfied at state S_t. In particular, because

$$m_{S_t}(C) \le \frac{c}{b+c} m_{S_t}(BC) = \frac{c}{b+c}(m_{S_t}(ABC) - m_{S_t}(A)),$$

we have

$$m_{S_t}(AC) \le \frac{c}{b+c} m_{S_t}(ABC) + \frac{b}{b+c} m_{S_t}(A). \tag{2}$$

As (2) is monotonically increasing in both $m_{S_t}(A)$ and $m_{S_t}(ABC)$ we can upper bound (2) by substituting the extremal values of $m_{S_t}(A)$ and $m_{S_t}(ABC)$ in, namely $|A|(2n-|A|)$ and $|ABC|(2n-|ABC|)$. After some algebra (or via an elegant combinatorial argument) it can be shown that

$$\frac{c}{b+c}|ABC|(2n-|ABC|) + \frac{b}{b+c}|A|(2n-|A|) \le k(2n-k) - cb. \tag{3}$$

Combined with (1), and the fact $b > 0$, (3) implies that the average fill of the k fullest cups in state S_{t+1} is at most $2n-k$, as desired.

We have shown the invariant holds for arbitrary k, so given that the invariants all hold at state S_t they also must all hold at state S_{t+1}. Thus, by induction we have the invariant for all rounds $t \in \mathbb{N}$. □

3.3 Oblivious Lower Bound

In Section 6 we provide filling strategies that an oblivious filler can use to achieve backlog $n^{1-\varepsilon}$ for $\varepsilon \in (0, 1/2)$ constant against "greedy-like" emptiers with probability at least $1 - 2^{-\operatorname{polylog}(n)}$ in running time $2^{\operatorname{polylog}(n)}$; in this subsection we sketch the proofs of these results. We remark that this proof is by far our most technically difficult result; however, interestingly, many of the ideas driving the oblivious lower bound are similar to those driving the adaptive lower bound.

First we make some definitions. The ***fill-range*** of a set of cups at state S is $\max_c \text{fill}_S(c) - \min_c \text{fill}_S(c)$. A cup configuration is ***R-flat*** if the fill-range is at most R. An emptier is called **$\boldsymbol{\Delta}$*-greedy-like*** if whenever there are two cups c_1, c_2 with $\text{fill}_{I_t}(c_1) > \text{fill}_{I_t}(c_2) + \Delta$ the emptier doesn't empty from c_2 on round t unless it also empties from c_1 on round t. For an oblivious filler we only prove lower bounds against $O(1)$-greedy-like emptiers; however this is a very interesting class of emptiers because all known randomized algorithms for the cup game are $O(1)$-greedy-like [1, 4]. We define a new variant of the cup game: In the ***E-extra-emptyings S-skip-emptyings*** cup game on n cups, the filler distributes p units of water amongst the cups, and then the emptier empties from *p or more, or less* cups. In particular the emptier is allowed to do E extra emptyings and is also allowed to skip S emptyings over the course of the game. Let the ***regular*** cup game be the 0-extra-emptyings ∞-skip-emptyings cup game: this is the regular cup game. Allowing for some extra emptyings, and bounding the number of skip emptyings is sometimes necessary when analyzing an algorithm that is a subroutine of a larger algorithm however, hence it sometimes makes sense to consider games with different values of E, S. Unless explicitly stated otherwise however we are considering the regular cup game.

Let $R_\Delta = 2(2+\Delta)$. We now prove an important fact about Δ-greedy-like emptiers.

Lemma 2 (Sketch of proof of Lemma 6)**.** *There is an oblivious filling strategy* **flatalg** *that, given an R-flat configuration of cups, achieves an R_Δ-flat configuration of cups against a Δ-greedy-like emptier in the p-processor, E-extra-emptyings, S-skip-emptyings negative-fill cup game on $n = 2p$ cups in running time $O(R+E+S)$. Throughout the duration of* flatalg *the cups are always R-flat.*

Proof. The filler's strategy is to place 1/2 units of fill into each cup on every round. Intuitively, if the fill-range of the cups is large then by greediness the emptier should be forced to empty from the fullest cups and not empty from the least full cups, thus decreasing the fill-range. □

Now we describe a well-known simple oblivious filling strategy that will be used as a subroutine later.

Proposition 1. *Consider an R-flat configuration of cups in the regular single-processor negative-fill cup game on n cups with initial average fill μ_0. Let $k \in [n]$. Let $d = \sum_{i=2}^{k} 1/i$.*

There is an oblivious filling strategy **randalg($\boldsymbol{k}$)** *that achieves backlog $\mu_0 - R + d$ with probability at least $1/k!$ in time $O(k)$.*

Furthermore, when applied against a Δ-greedy-like emptier with $R = R_\Delta$, even if the emptier is allowed arbitrarily many extra emptyings (i.e. if the game occurs in the ∞-extra-emptyings ∞-skip-emptyings cup game), randalg(k) *guarantees that the cup configuration is $(R+d)$-flat on every round.*

Proof. The filler maintains an ***active set***, initialized to being an arbitrary subset of k of the cups. Every round the filler distributes 1 unit of fill equally among all cups in the active set. Next the emptier removes 1 unit of fill from some cup, or skips its emptying. Then the filler removes a random cup from the active set (chosen uniformly at random from the active set). This continues until a single cup c remains in the active set. If c has never been emptied from, then its fill has increased by at least $1/k + 1/(k-1) + \cdots + 1/2 = d$ from its starting value which was at least $\mu_0 - R$. Thus c has at least the desired fill if it has not been emptied from. By randomly removing cups from the active set the filler guarantees that, with probability at least $1/(k-1)!$, c is not emptied from.

Now we consider a greedy-like emptier that is allowed extra-emptyings. Let $\mathcal{A}_t$ be the event that the anti-backlog is smaller in S_{t+1} than in S_t, let $\mathcal{B}_t$ be the event that some cup with fill equal to the backlog in S_{t+1} was emptied from on round t. If $\mathcal{A}_t$ and $\mathcal{B}_t$ are both true on round t, then by greediness the cups are quite flat, in particular R_Δ-flat. Consider a round t_1 where the cups are not R_Δ-flat. Let t_0 be the last round that the cups were R_Δ-flat. On all rounds $t \in (t_0, t_1)$ at least one of $\mathcal{A}_t$ or $\mathcal{B}_t$ must not hold. On a round where $\mathcal{A}_t$ does not hold, anti-backlog does not decrease and backlog increases by at most $1/(k-t+1)$, so fill range increases by at most $1/(k-t+1)$. On a round where $\mathcal{B}_t$ does not hold, anti-backlog decreases by at most 1 and backlog decreases by at least $1 - 1/(k-t+1)$, as all cups with fill equal to the backlog in state S_{t+1} were emptied from on round t, so fill-range increases by at most $1/(k-t+1)$. Hence in total fill-range increases by at most $\sum_{i=2}^{k} 1/i$ from R, i.e. the cups are $(R+d)$-flat on round t_1.

□

We now give a method for transforming a filling strategy for achieving large backlog into a filling strategy for achieving high fill in many cups.

Definition 1. Let alg_0 be an oblivious filling strategy, that can get high fill (for some definition of high) in some cup against greedy-like emptiers with some probability. We construct a new filling strategy $\mathbf{rep_\delta(alg_0)}$ as follows:

Say we have some configuration of n cups. Let $n_A = \lceil \delta n \rceil, n_B = \lfloor (1-\delta)n \rfloor$. Let $N \gg n$ be large, let $M = 2^{\text{polylog}(N)}$ be a chosen parameter. Initialize A to $\varnothing$ and B to being all of the cups. We call A the ***anchor set*** and B the ***non-anchor set***. The filler always places 1 unit of fill in each anchor cup on each round. The filling strategy consists of n_A ***donation-processes***, which are procedures that result in a cup being ***donated*** from B to A (i.e. removed from B and added to A). At the start of each donation-processes the filler chooses a value m_0 uniformly at random from $[M]$. We say that the filler ***applies*** a filling strategy alg to B if the filler uses alg on B while placing 1 unit of fill in each anchor cup. During the donation-process the filler applies alg_0 to B m_0 times, and flattens B by applying flatalg to B for $\Theta(N^2)$ rounds before each application of alg_0. At the end of each donation process the filler takes the cup given by the final application of alg_0 (i.e. the cup that alg_0 guarantees with some probability against a certain class of emptiers to have a certain high fill), and donates this cup to A.

We say that the emptier ***neglects*** the anchor set on a round if it does not empty from each anchor cup. We say that an application of alg_0 to B is ***non-emptier-wasted*** if the emptier does not neglect the anchor set during any round of the application of alg_0.

We use rep in two distinct places: first to get constant backlog, and second to prove the Oblivious counterpart of the Adaptive Amplification Lemma. For the rest of the section our goal is eventually to get backlog $\text{poly}(N)$ in N cups for a value of N that we fix now; this value of N will be used throughout all of the proofs.

First we analyze rep(randalg).

Lemma 3. *Let $\Delta \leq O(1)$, let $h \leq O(1)$, with $h \geq 16(1+\Delta)$, let $k = \lceil e^{2h+1} \rceil$, let $\delta = \Theta(e^{-2h})$, let $n \ll N$ be sufficiently large. Consider an R_Δ-flat cup configuration in the variable-processor cup game on n cups with initial average fill μ_0.*

Against a Δ-greedy-like emptier, $\text{rep}_\delta(\text{randalg}(k))$ using $M = \Theta(N^2)$ either achieves mass N^2 in the cups, or with probability at least $1 - 2^{-\Omega(n)}$ makes an (unknown) set of $\Theta(n)$ cups in A have fill at least $h + \mu_0$ while also guaranteeing that $\mu(B) \geq -h/2 + \mu_0$, where A, B are the sets defined in Definition 1. The running time of $\text{rep}_\delta(\text{randalg}(k))$ is poly(N).

Lemma 3 Proof Sketch. We use the definitions given in Definition 1.

Note that if the emptier neglects the anchor set N^2 times, or skips N^2 emptyings, then the mass of the cups will be at least N^2, so the filler is done. For the rest of the proof we consider the case where the emptier chooses to neglect the anchor set fewer than N^2 times, and choose to skip fewer than N^2 emptyings.

First we show that $\mu(B)$ never sinks too low, which is one of the guarantees of our algorithm, and necessary to ensure that we get large fill in many cups in A. Let $d = \sum_{i=2}^{k} 1/i = \Theta(h)$. Because extra-emptyings and skip-emptyings of B are limited, using Lemma 2 and Proposition 1, we can inductively show that B is always $(R_\Delta + d)$-flat, and that all applications of flatalg make B be R_Δ-flat. Now we claim that $\mu(B) \le \mu(AB) + 2$. There are two ways $\mu(B) - \mu(AB)$ can increase:
Case 1: The emptier could empty from 0 cups in B while emptying from every cup in A.
By greediness this means that $\mu(A) \ge \mu(B) - \Delta$. Since $|B| \gg |A|$, this implies $\mu(B) \le \mu(AB) + 1$.
Case 2: The filler could evict a cup with fill lower than $\mu(B)$ from B at the end of a donation-process. Because B starts each application of randalg(k) being $(R_\Delta + d)$-flat, and the running time of the application is $k-1$ the cup donated from B cannot have fill lower than $\mu(B) - R_\Delta - (k-1)$. But because $n_B \gg n_A$ even if B donated n_A cups that were maximally empty the difference $\mu(B) - \mu(AB)$ would only increase by at most 1.
Combining the analysis of Case 1 and Case 2 we have that $\mu(B) \le \mu(AB) + 2$.

Combining the upper bound on $\mu(B)$ with the fact that B is always flat and that the emptier is greedy-like, we have that no cup in A ever has its fill exceed $u_A = \mu(AB) + 2 + R_\Delta + d + 1$. Obviously $\mu(A) \le u_A$. Hence we have an upper bound on $\mu(A) - \mu(AB)$; plugging this upper bound into the relation $m(A) + m(B) = m(AB)$, and using the fact that $|B| \gg |A|$ we get the desired bound on $\mu(B)$:

$$\mu(B) \ge -h/2 + \mu(AB) \ge -h/2 + \mu_0.$$

Now we show that we get a constant-fraction of the cups in A to have fill $\mu_0 + h$. If the emptier were not able to neglect the anchor set, then by a Chernoff bound a constant fraction of the applications of randalg(k) in a donation-process succeed with exponentially good probability in M. A successful application of randalg(k) yields a cup with fill at least $\mu(B) - R_\Delta + d \ge h + \mu_0$ by our lower bound on $\mu(B)$. In order to mitigate the problem that the emptier can do extra-emptying we do many, in particular $M = \Theta(N^2)$, applications of randalg(k) and choose randomly how many to do before donating a cup; this guarantees that with constant probability each donation-process succeeds. Taking a Chernoff bound over the donation-processes gives that with exponentially good probability in n a constant fraction of the donation-processes succeed. Taking a union bound over all of our Chernoff bounds gives the desired probability of success.

The running time of the filling strategy is clearly $n_A O(M)(O(N^2) + O(1)) = \text{poly}(N)$, as each of the n_A donation-processes consists of $O(M)$ applications of randalg(k) and $O(M)$ applications of flatalg. □

Using Lemma 3 we show that an oblivious filler can achieve constant fill in a known cup.

Proposition 2. *Let $H \le O(1)$, let $\Delta \le O(1)$, let $n \ll N$ be at least a sufficiently large constant determined by H and Δ. Consider an R_Δ-flat cup configuration in the variable-processor cup game on n cups with average fill μ_0. There is an oblivious filling strategy that either achieves mass N^2 among the cups, or achieves fill at least $\mu_0 + H$ in a chosen cup in running time poly(N) against a Δ-greedy-like emptier with probability at least $1 - 2^{-\Omega(n)}$.*

Proof. The filler starts by using rep$_\delta$(randalg(k)) with parameter settings as in Lemma 7 where $h = H \cdot 16(1+\Delta)$, i.e. $k = \lceil e^{2h+1} \rceil$, $\delta = \Theta(e^{-2h})$. If this results in mass N^2 among the cups we are done; we assume this is not the case for the rest of the proof. Let the number of cups which, with exponentially good probability in n, now exist by Lemma 7 with fill at least $h + \mu_0$ be of size $nc = \Theta(n)$.

The filler sets $p = 1$, i.e. uses a single processor. Now the filler exploits the emptier's greedy-like nature to to get fill H in a chosen cup c_0. Specifically, for $(5/8)h$ rounds the filler places 1 unit of fill into c_0. Because the emptier is Δ-greedy-like it must empty from the nc cups in A with fill at least $h + \mu_0$ until c_0 has large fill. Over $(5/8)h$ rounds the cups in A cannot have their fill decrease below $(3/8)h \ge h/8 + \Delta + \mu_0$. Hence, any cups with fills less than $h/8 + \mu_0$ must not be emptied from during these rounds. The fill of c_0 started as at least $-h/2 + \mu_0$ as $\mu(B) \ge -h/2 + \mu_0$. After $(5/8)h$ rounds c_0 has fill at least $h/8 + \mu_0$, because the emptier cannot have emptied c_0 until it attained fill $h/8 + \mu_0$, and if c_0 is never emptied from then it achieves fill $h/8 + \mu_0$. Thus the filling strategy achieves backlog $h/8 + \mu_0 \ge H + \mu_0$ in c_0, a known cup, as desired.

The running time is of course still poly(N) by Lemma 3. □

Next we prove the Oblivious Amplification Lemma.

Lemma 4. *Let $\delta \in (0, 1/2)$ be a constant parameter. Let $\Delta \le O(1)$. Consider a cup configuration in the variable-processor cup game on $n \le N, n > \Omega(1/\delta^2)$ cups with average fill μ_0 that is R_Δ-flat. Let* alg(f) *be an oblivious filling strategy that either achieves mass N^2 or, with failure probability at most*

$p \geq 2^{-\lg^8 N}$, *achieves backlog* $\mu_0 + f(n)$ *on such cups in running time* $T(n)$ *against a* Δ*-greedy-like emptier. Let* $M = 2^{polylog(N)}$.

Consider a cup configuration in the variable-processor cup game on $n \leq N, n > \Omega(1/\delta^2)$ *cups with average fill* μ_0 *that is* R_Δ*-flat. There exists an oblivious filling strategy* $\text{alg}(f')$ *that either achieves mass* N^2 *or with failure probability at most*

$$p' \leq np + 2^{-\lg^8 N}$$

achieves backlog $f'(n)$ *satisfying*

$$f'(n) \geq (1-\delta)^2 f(\lfloor (1-\delta)n \rfloor) + f(\lceil \delta n \rceil) + \mu_0$$

and $f'(n) \geq f(n)$, *in running time*

$$T'(n) \leq Mn \cdot T(\lfloor (1-\delta)n \rfloor) + T(\lceil \delta n \rceil)$$

against a Δ*-greedy-like emptier.*

Proof. We use the definitions and notation given in Definition 2.

Note that if the emptier neglects the anchor set N^2 times, or skips N^2 emptyings, then the mass of the cups will be at least N^2, so the filler is done. For the rest of the proof we consider the case where the emptier chooses to neglect the anchor set fewer than N^2 times, and choose to skip fewer than N^2 emptyings.

The filler simply uses $\text{alg}(f)$ on all the cups if this results in sufficient backlog, i.e. if

$$f(n) \geq (1-\delta)^2 f(n_B) + f(n_A).$$

In this case our strategy trivially has the desired guarantees. In the rest of the proof we consider the case where $\text{alg}(f)$ does not achieve sufficient backlog.

The filler's strategy is as follows:

Step 1: Make $\mu(A) \geq (1-\delta)^2 f(n_B)$ by using $\text{rep}_\delta(\text{alg}(f))$ on all the cups, i.e. applying $\text{alg}(f)$ repeatedly to B, flattening B before each application, and then donating a cup from B to A.

Step 2: Flatten A using flatalg, and then use $\text{alg}(f)$ on A.

We now analyze Step 1. For this proof we need all donation-processes to succeed, as opposed to in the proof of Lemma 3 in which we only needed a constant fraction of the donation-processes to succeed. This necessitates choosing M very large. In particular we choose $M = 2^{\log^{24} N}$ —recall that $[M]$ is the set from which we randomly choose how many times to apply $\text{alg}(f)$ in a donation-process. By choosing M this large we cannot hope to guarantee that every application of $\text{alg}(f)$ succeeds: there are far too many applications. On the other hand, having M so large allows us to have a very tight concentration bound on how many applications of $\text{alg}(f)$ succeed. By a Chernoff bound with probability at least $1 - e^{-2Mp^2}$ at least $M(1-2p)$ of M applications of $\text{alg}(f)$ would succeed if the emptier did not ***interfere***, i.e. neglect the anchor set and do an extra emptying in the non-anchor set. The emptier can interfere with at most N^2 of the $M(1-2p)$ applications that would otherwise be successful. Let $1-q$ be the probability that a donation-process succeeds, i.e. the final application of $\text{alg}(f)$ is not emptier-wasted and succeeds. We have

$$1 - q \geq (1 - e^{-2Mp^2}) \left(\frac{M \cdot (1-2p) - N^2}{M} \right).$$

Rearranging, simplifying by loosening the bound, and using the assumption $p \geq 2^{-\lg^8 N}$, we can show

$$q \leq 2p + 2^{-\lg^8 N}.$$

Taking a union bound, we have that with probability at least $1 - q \cdot n_A$ all donation-process successfully achieve a cup with fill at least $\mu_{S_0}(B) + f(n_B)$ where $\mu_{S_0}(B)$ refers to the average fill of B measured at the start of the application of $\text{alg}(f)$; now we assume all donation-processes are successful, and demonstrate that this translates into the desired average fill in A.

Let $\textbf{skips}_t$ denote the number of times that the emptier has skipped the anchor set by round t. Consider how $\mu(B) - \text{skips}/n_B$ changes over the course of the donation processes. As noted above, at the end of each donation-process $\mu(B)$ decreases due to B donating a cup with fill at least $\mu(B) + f(n_B)$. In particular, if S denotes the cup state immediately before a cup is donated on the i-th donation-process, B_0 denotes the set B before the donation and B_1 denotes the set B after the donation, then $\mu_S(B_1) = \mu_S(B_0) - f(n_B)/(n-i)$. Now we claim that $t \mapsto \mu_{S_t}(B) - \text{skips}_t / n_B$ is monotonically decreasing. Clearly donation decreases $\mu(B) - \text{skips}/n_B$. If the anchor set is neglected then $\mu(B)$

decreases, causing $\mu(B) - \text{skips}/n_B$ to decrease. If a skip occurs, then skips/n_B increases by more than $\mu(B)$ increases, causing $\mu(B) - \text{skips}/n_B$ to decrease. Let t_* be the cup state at the end of all the donation-processes. We have that

$$\mu_{S_{t_*}}(B) - \frac{\text{skips}_{t_*}}{n_B} \le \mu_0 - \sum_{i=1}^{n_A} \frac{f(n_B)}{n-i}. \tag{4}$$

By conservation of mass we have

$$n_A \cdot \mu_{S_{t_*}}(A) + n_B \cdot \mu_{S_{t_*}}(B) = n\mu_0 + \text{skips}_{t_*}.$$

Rearranging,

$$\mu_{S_{t_*}}(A) = \mu_0 + \frac{n_B}{n_A}\left(\mu_0 + \frac{\text{skips}_{t_*}}{n_B} - \mu_{S_{t_*}}(B)\right). \tag{5}$$

Now we obtain a simpler form of Inequality (4). Recalling that harmonic numbers grow like ln we have

$$\sum_{i=1}^{n_A} \frac{1}{n-i} \approx \ln n/(n-n_A) \approx \ln \frac{1}{1-\delta} > \delta.$$

Then using Inequality (4) in Equation 5 we essentially have

$$\mu_{S_{t_*}}(A) \ge \mu_0 + \frac{n_B}{n_A}\delta f(n_B) \approx \mu_0 + (1-\delta) f(n_B).$$

To handle the imprecisions indicated in the approximations above the bound becomes slightly worse:

$$\mu_{S_{t_*}}(A) \ge \mu_0 + (1-\delta)^2 f(n_B).$$

We have shown that in Step 1 the filler achieves average fill $\mu_0 + (1-\delta)f(n_B)$ in A with failure probability at most $q \cdot n_A$. Now the filler flattens A and uses $\text{alg}(f)$ on A. It is clear that this is possible, and succeeds with probability at least p. This gets a cup with fill

$$\mu_0 + (1-\delta)^2 f(n_B) + f(n_A)$$

in A, as desired.

Taking a union bound over the probabilities of Step 1 and Step 2 succeeding gives that the entire procedure fails with probability at most

$$p' \le p + q \cdot n_A \le np + 2^{-\lg^8 N}.$$

The running time of Step 1 is clearly $M \cdot n \cdot T(\lfloor (1-\delta)n \rfloor)$ and the running time of Step 2 is clearly $T(\lceil \delta n \rceil)$; summing these yields the desired upper bound on running time. $\square$

Finally we prove that an oblivious filler can achieve backlog $N^{1-\varepsilon}$.

Theorem 4. *There is an oblivious filling strategy for the variable-processor cup game on N cups that achieves backlog at least $\Omega(N^{1-\varepsilon})$ for any constant $\varepsilon > 0$ in running time $2^{polylog(n)}$ with probability at least $1 - 2^{-polylog(n)}$ against a Δ-greedy-like emptier with $\Delta \le O(1)$.*

Proof. We show how to achieve backlog $(N/n_b)^{1-\varepsilon} - 1$ for some $n_b \le \text{polylog}(N)$ on N cups; note that this implies that the filler can achieve backlog $\Omega(N^{1-\varepsilon'})$ for any constant $\varepsilon' \in (0, 1/2)$ on N cups. Let δ be a constant, chosen as a function of ε.

By Proposition 6 there is an oblivious filling strategy that achieves backlog $\Omega(1)$ on n cups with exponentially good probability in n; we call this algorithm $\text{alg}\, f_0$. Let $n_b = \log^8(N)$. We can make $\text{alg}\, f_0$ achieve backlog $f_0(k) \ge H \ge \Omega(1)$ for all $k \ge n_b$, for constant $H \ge \Omega(1)$ to be determined as a function of δ. We construct f_{i+1} as the amplification of f_i using Lemma 8.

One can inductively show that $f_{\Theta(\log N)}$ achieves backlog $(N/n_b)^{1-\varepsilon} - 1$, as desired. Analysis of the running time recurrence from the Oblivious Amplification Lemma gives that $\text{alg}(f_{\Theta(\log N)})$ has running time $2^{\text{polylog}(N)}$, while analysis of the probability recurrence shows that $\text{alg}(f_{\Theta(\log N)})$ succeeds with probability at least $1 - 2^{-\text{polylog}(N)}$. $\square$

4 Adaptive Filler Lower Bound

In this section we give a $2^{\text{polylog}\, n}$-time filling strategy that achieves backlog $n^{1-\varepsilon}$ for any positive constant ε. We also give a $O(n!)$-time filling strategy that achieves backlog $\Omega(n)$.

We begin with a trivial filling strategy that we call **trivalg** that gives backlog at least 1/2 when applied to at least 2 cups.

Proposition 3. *Consider an instance of the negative-fill* 1*-processor cup game on* n *cups, and let the cups start in any state with average fill is* 0*. If* $n \geq 2$*, there is an* $O(1)$*-step adaptive filling strategy* trivalg *that achieves backlog at least* $1/2$*. If* $n = 1$*,* trivalg *achieves backlog* 0 *in running time* 0*.*

Proof. If $n = 1$, trivalg does nothing and acheives backlog 0; for thre rest of the proof we consider the case $n \geq 2$.

Let a and b be the fullest and second fullest cups in the in the starting configuration, and let their initial fills be $\text{fill}(a) = \alpha, \text{fill}(b) = \beta$. If $\alpha \geq 1/2$ the filler need not do anything, the desired backlog is already achieved. Otherwise, if $\alpha \in [0, 1/2]$, the filler places $1/2 - \alpha$ fill into a and $1/2 + \alpha$ fill into b (which is possible as both fills are in $[0, 1]$, and they sum to 1). Since $\alpha + \beta \geq 0$ we have $\beta \geq -\alpha$. Clearly a and b now both have fill at least $1/2$. The emptier cannot empty from both a and b as $p = 1$, so even after the emptier empties from a cup we still have backlog $1/2$, as desired. □

Next we prove the ***Amplification Lemma***, which takes as input a filling strategy $\text{alg}(f)$ and outputs a new filling strategy $\text{alg}(f')$ that we call the ***amplification*** of $\text{alg}(f)$. $\text{alg}(f')$ is able to achieve higher fill than $\text{alg}(f)$; in particular, we will show that by starting with a filling strategy $\text{alg}(f_0)$ for achieving constant backlog and then forming a sufficiently long sequence of filling strategies $\text{alg}(f_0), \text{alg}(f_1), \ldots, \text{alg}(f_{i_*})$ with $\text{alg}(f_{i+1})$ the amplification of $\text{alg}(f_i)$, we eventually get a filling strategy for achieving $\text{poly}(n)$ backlog.

Lemma 5 (Adaptive Amplification Lemma)**.** *Let* $\delta \in (0, 1/2]$ *be a parameter. Let* $\text{alg}(f)$ *be an adaptive filling strategy that achieves backlog* $f(n) < n$ *in the negative-fill variable-processor cup game on* n *cups in running time* $T(n)$ *starting from any initial cup state where the average fill is* 0*.*

Then there exists an adaptive filling strategy $\text{alg}(f')$ *that achieves backlog* $f'(n)$ *satisfying*

$$f'(n) \geq (1-\delta) f(\lfloor (1-\delta) n \rfloor) + f(\lceil \delta n \rceil)$$

and $f'(n) \geq f(n)$ *in the negative-fill variable-processor cup game on* n *cups in running time*

$$T'(n) \leq n \lceil \delta n \rceil \cdot T(\lfloor (1-\delta) n \rfloor) + T(\lceil \delta n \rceil)$$

starting from any initial cup state where the average fill is 0*.*

Proof. Let $n_A = \lceil \delta n \rceil, n_B = n - n_A = \lfloor (1-\delta) n \rfloor$.

The filler defaults to using $\text{alg}(f)$ if

$$f(n) \geq (1-\delta) f(n_B) + f(n_A).$$

In this case using $\text{alg}(f)$ achieves the desired backlog in the desired running time. In the rest of the proof, we describe our strategy in the case that we cannot simply use $\text{alg}(f)$ to achieve the desired backlog.

Let A, the ***anchor set***, be initialized to consist of the n_A fullest cups, and let B the ***non-anchor set*** be initialized to consist of the rest of the cups (so $|B| = n_B$). Let $h = (1-\delta) f(n_B)$.

The filler's strategy can be summarized as follows:

Step 1: Make $\mu(A) \geq h$ by using $\text{alg}(f)$ repeatedly on B to achieve cups with fill at least $\mu(B) + f(n_B)$ in B and then swapping these into A. While doing this the filler always places 1 unit of fill in each anchor cup.

Step 2: Use $\text{alg}(f)$ once on A to obtain some cup with fill $\mu(A) + f(n_A)$.

Note that in order to use $\text{alg}(f)$ on subsets of the cups the filler will need to vary p.

We now describe how to achieve Step 1, which is complicated by the fact that the emptier may attempt to prevent the filler from achieving high fill in a cup in B.

The filling strategy always places 1 unit of water in each anchor cup. This ensures that no cups in the anchor set ever have their fill decrease. If the emptier wishes to keep the average fill of the anchor cups from increasing, then emptier must empty from every anchor cup on each step. If the emptier fails to do this on a given round, then we say that the emptier has ***neglected*** the anchor cups.

We say that the filler ***applies*** $\text{alg}(f)$ to B if it follows the filling strategy $\text{alg}(f)$ on B while placing 1 unit of water in each anchor cup. An application of $\text{alg}(f)$ to B is said to be ***successful*** if A is never neglected during the application of $\text{alg}(f)$ to B. The filler uses a procedure that we call a ***swapping-process*** to achieve the desired average fill in A. In a swapping-process, the filler repeatedly applies $\text{alg}(f)$ to B until a successful application occurs, and then takes the cup generated by $\text{alg}(f)$ within B on this successful application with fill at least $\mu(B) + f(|B|)$ and swaps it with the least full cup in A so long as doing so would increase $\mu(A)$. If the average fill in A ever reaches h, then the algorithm immediately halts (even if it is in the middle of a swapping-process) and is complete.

We give pseudocode for the filling strategy in Algorithm 1.

Algorithm 1 Adaptive Amplification (Step 1)

Input: $\text{alg}(f), \delta$, set of n cups
Output: Guarantees that $\mu(A) \geq h$

```
A ← n_A fullest cups, B ← rest of the cups
Always place 1 fill in each cup in A
while μ(A) < h do                                    ▷ Swapping-Processes
    Immediately exit this loop if ever μ(A) ≥ h
    successful ← false
    while not successful do
        Apply alg(f) to B, alg(f) gives cup c
        if fill(c) ≥ h then
            successful ← true
    Swap c with least full cup in A
```

Note that

$$\mu(A) \cdot |A| + \mu(B) \cdot |B| = \mu(AB) \geq 0,$$

as $\mu(AB)$ starts as 0, but could become positive if the emptier skips emptyings. Thus we have

$$\mu(A) \geq -\mu(B) \cdot \frac{\lfloor (1-\delta)n \rfloor}{\lceil \delta n \rceil} \geq -\frac{1-\delta}{\delta}\mu(B).$$

Thus, if at any point B has average fill lower than $-h \cdot \delta/(1-\delta)$, then A has average fill at least h, so the algorithm is finished. Thus we can assume in our analysis that

$$\mu(B) \geq -h \cdot \delta/(1-\delta). \tag{6}$$

We will now show that during each swapping process, the filler applies $\text{alg}(f)$ to B at most hn_A times. Each time the emptier neglects the anchor set, the mass of the anchor set increases by 1. If the emptier neglects the anchor set hn_A times, then the average fill in the anchor set increases by h. Since $\mu(A)$ started as at least 0, and since $\mu(A)$ never decreases (note in particular that cups are only swapped into A if doing so will increase $\mu(A)$), an increase of h in $\mu(A)$ implies that $\mu(A) \geq h$, as desired. Thus the swapping process consists of at most hn_A applications of $\text{alg}(f)$.

Consider the fill of a cup c swapped into A at the end of a swapping-process. Cup c's fill is at least $\mu(B) + f(n_B)$, which by (6) is at least

$$-h \cdot \frac{\delta}{1-\delta} + f(n_B) = (1-\delta)f(n_B) = h.$$

Thus the algorithm for Step 1 succeeds within $|A|$ swapping-processes, since at the end of the $|A|$-th swapping process either every cup in A has fill at least h, or the algorithm halted before $|A|$ swapping-processes because it already achieved $\mu(A) \geq h$.

After achieving $\mu(A) \geq h$, the filler performs Step 2, i.e. the filler applies $\text{alg}(f)$ to A, and hence achieves a cup with fill at least

$$\mu(A) + f(|A|) \geq (1-\delta)f(n_B) + f(n_A),$$

as desired.

Now we analyze the running time of the filling strategy $\text{alg}(f')$. First, recall that in Step 1 $\text{alg}(f')$ calls $\text{alg}(f)$ on B, which has size n_B, as many as hn_A times. Because we mandate that $h < n$, Step 1 contributes no more than $(n \cdot n_A) \cdot T(n_B)$ to the running time. Step 2 requires applying alg(f) to A, which has size n_A, once, and hence contributes $T(n_A)$ to the running time. Summing these we have

$$T'(n) \leq n \cdot n_A \cdot T(n_B) + T(n_A).$$

□

We next show that by recursively using the Amplification Lemma we can achieve backlog $n^{1-\varepsilon}$.

Theorem 5. *There is an adaptive filling strategy for the variable-processor cup game on n cups that achieves backlog $\Omega(n^{1-\varepsilon})$ for any constant $\varepsilon > 0$ of our choice in running time $2^{O(\log^2 n)}$.*

Proof. Take constant $\varepsilon \in (0, 1/2)$. Let c, δ be constants that will be chosen (later) as functions of ε satisfying $c \in (0, 1), 0 < \delta \ll 1/2$. We show how to achieve backlog at least $cn^{1-\varepsilon} - 1$.

Let alg(f_0) = trivalg, the algorithm given by Proposition 3; recall trivalg achieves backlog $f_0(k) \geq 1/2$ for all $k \geq 2$, and $f_0(1) = 0$.

Next, using the Amplification Lemma we recursively construct alg(f_{i+1}) as the amplification of alg(f_i) for $i \geq 0$.

Define a sequence g_i with

$$g_i = \begin{cases} \lceil 16/\delta \rceil, & i = 0, \\ \lfloor g_{i-1}/(1-\delta) \rfloor & i \geq 1 \end{cases}$$

We claim the following regarding this construction:

Claim 1. *For all $i \geq 0$,*

$$f_i(k) \geq ck^{1-\varepsilon} - 1 \quad \text{for all} \quad k \in [g_i]. \tag{7}$$

Proof. We prove Claim 1 by induction on i. For $i = 0$, the base case, (7) can be made true by taking c sufficiently small; in particular, taking $c < 1$ makes (7) hold for $k = 1$, and, as $g_0 > 2$, taking c small enough to make $cg_0^{1-\varepsilon} - 1 \leq f_0(g_0) = 1/2$ makes (7) hold for $k \in [2, g_0]$ by monotonicity of $k \mapsto ck^{1-\varepsilon} - 1$. [3]

As our inductive hypothesis we assume (7) for f_i; we aim to show that (7) holds for f_{i+1}. Note that, by design of g_i, if $k \leq g_{i+1}$ then $\lfloor k \cdot (1-\delta) \rfloor \leq g_i$. Consider any $k \in [g_{i+1}]$. First we deal with the trivial case where $k \leq g_0$. In this case

$$f_{i+1}(k) \geq f_i(k) \geq \cdots \geq f_0(k) \geq ck^{1-\varepsilon} - 1.$$

Now we consider the case where $k \geq g_0$. Since f_{i+1} is the amplification of f_i we have

$$f_{i+1}(k) \geq (1-\delta) f_i(\lfloor (1-\delta)k \rfloor) + f_i(\lceil \delta k \rceil).$$

By our inductive hypothesis, which applies as $\lceil \delta k \rceil \leq g_i$, $\lfloor k \cdot (1-\delta) \rfloor \leq g_i$, we have

$$f_{i+1}(k) \geq (1-\delta)(c \cdot \lfloor (1-\delta)k \rfloor^{1-\varepsilon} - 1) + c \lceil \delta k \rceil^{1-\varepsilon} - 1.$$

Dropping the floor and ceiling, incurring a -1 for dropping the floor, we have

$$f_{i+1}(k) \geq (1-\delta)(c \cdot ((1-\delta)k - 1)^{1-\varepsilon} - 1) + c(\delta k)^{1-\varepsilon} - 1.$$

Because $(x-1)^{1-\varepsilon} \geq x^{1-\varepsilon} - 1$, due to the fact that $x \mapsto x^{1-\varepsilon}$ is a sub-linear sub-additive function, we have

$$f_{i+1}(k) \geq (1-\delta)c \cdot (((1-\delta)k)^{1-\varepsilon} - 2) + c(\delta k)^{1-\varepsilon} - 1.$$

Moving the $ck^{1-\varepsilon}$ to the front we have

$$f_{i+1}(k) \geq ck^{1-\varepsilon} \cdot \left((1-\delta)^{2-\varepsilon} + \delta^{1-\varepsilon} - \frac{2(1-\delta)}{k^{1-\varepsilon}} \right) - 1.$$

Because $(1-\delta)^{2-\varepsilon} \geq 1 - (2-\varepsilon)\delta$, a fact called Bernoulli's Identity, we have

$$f_{i+1}(k) \geq ck^{1-\varepsilon} \cdot \left(1 - (2-\varepsilon)\delta + \delta^{1-\varepsilon} - \frac{2(1-\delta)}{k^{1-\varepsilon}} \right) - 1.$$

[3]Note that it is important here that ε and δ are constants, that way c is also a constant.

Of course $-2(1-\delta) \geq -2$, so

$$f_{i+1}(k) \geq ck^{1-\varepsilon} \cdot \left(1 - (2-\varepsilon)\delta + \delta^{1-\varepsilon} - \frac{2}{k^{1-\varepsilon}}\right) - 1.$$

Because

$$\frac{-2}{k^{1-\varepsilon}} \geq \frac{-2}{g_0^{1-\varepsilon}} \geq -2(\delta/16)^{1-\varepsilon} \geq -\delta^{1-\varepsilon}/2,$$

which follows from our choice of $g_0 = \lceil 16/\delta \rceil$ and the restriction $\varepsilon < 1/2$, we have

$$f_{i+1}(k) \geq ck^{1-\varepsilon} \cdot \left(1 - (2-\varepsilon)\delta + \delta^{1-\varepsilon} - \delta^{1-\varepsilon}/2\right) - 1.$$

Finally, combining terms we have

$$f_{i+1}(k) \geq ck^{1-\varepsilon} \cdot \left(1 - (2-\varepsilon)\delta + \delta^{1-\varepsilon}/2\right) - 1.$$

Because $\delta^{1-\varepsilon}$ dominates δ for sufficiently small δ, there is a choice of $\delta = \Theta(1)$ such that

$$1 - (2-\varepsilon)\delta + \delta^{1-\varepsilon}/2 \geq 1.$$

Taking δ to be this small we have,

$$f_{i+1}(k) \geq ck^{1-\varepsilon} - 1,$$

completing the proof. We remark that the choices of c, δ are the same for every i in the inductive proof, and depend only on ε. □

To complete the proof, we will show that g_i grows exponentially in i. Thus, after there exists $i_* \leq O(\log n)$ such that $g_{i_*} \geq n$, and hence we have an algorithm $\text{alg}(f_{i_*})$ that achieves backlog $cn^{1-\varepsilon} - 1$ on n cups, as desired.

We lower bound the sequence g_i with another sequence g_i' defined as

$$g_i' = \begin{cases} 4/\delta, & i = 0 \\ g_{i-1}'/(1-\delta) - 1, & i > 0. \end{cases}$$

Solving this recurrence, we find

$$g_i' = \frac{4 - (1-\delta)^2}{\delta} \frac{1}{(1-\delta)^i} \geq \frac{1}{(1-\delta)^i},$$

which clearly exhibits exponential growth. In particular, let $i_* = \left\lceil \log_{1/(1-\delta)} n \right\rceil$. Then,

$$g_{i_*} \geq g_{i_*}' \geq n,$$

as desired.

Let the running time of $f_i(n)$ be $T_i(n)$. From the Amplification Lemma we have following recurrence bounding $T_i(n)$:

$$\begin{aligned} T_i(n) &\leq n \lceil \delta n \rceil \cdot T_{i-1}(\lfloor (1-\delta)n \rfloor) + T_{i-1}(\lceil \delta n \rceil) \\ &\leq 2n^2 T_{i-1}(\lfloor (1-\delta)n \rfloor). \end{aligned}$$

It follows that $\text{alg}(f_{i_*})$, recalling that $i_* \leq O(\log n)$, has running time

$$T_{i_*}(n) \leq (2n^2)^{O(\log n)} \leq 2^{O(\log^2 n)}$$

as desired.

□

Now we provide a very simple construction that can achieve backlog $\Omega(n)$ in very long games. The construction can be interpreted as the same argument as in Theorem 5 but with an extremal setting of δ to $\Theta(1/n)$. [4]

Proposition 4. *There is an adaptive filling strategy that achieves backlog $\Omega(n)$ in time $O(n!)$.*

[4] Or more precisely, setting δ in each level of recursion to be $\Theta(1/n)$, where n is the subproblem size; note in particular that δ changes between levels of recursion, which was not the case in the proof of Theorem 5.

Proof. First we construct a slightly stronger version of trivalg that achieves backlog 1 on $n \geq n_0 = 8$ cups, instead of just backlog 1/2; this simplifies the analysis.

Claim 2. *There is a filling algorithm* trivalg_2 *that achieves backlog at least* 1 *on* $n_0 = 8$ *cups.*

Proof. Let trivalg_1 be the amplification of trivalg using $\delta = 1/2$. On 4 cups trivalg_1 achieves backlog at least $(1/2)(1/2) + 1/2 = 3/4$. Let trivalg_2 be the amplification of trivalg_1 using $\delta = 1/2$. On 8 cups trivalg_2 achieves backlog at least $(1/2)(3/4) + 3/4 \geq 1$. □

Let $\text{alg}(f_0) = \text{trivalg}_2$; we have $f_0(k) \geq 1$ for all $k \geq n_0$. For $i > 0$ we construct $\text{alg}(f_i)$ as the amplification of $\text{alg}(f_{i-1})$ using the Amplification Lemma with parameter $\delta = 1/(i+1)$.

We claim the following regarding this construction:

Claim 3. *For all* $i \geq 0$,

$$f_i((i+1)n_0) \geq \sum_{j=0}^{i} \left(1 - \frac{j}{i+1}\right). \tag{8}$$

Proof. We prove Claim 3 by induction on i. When $i = 0$, the base case, (8) becomes $f_0(n_0) \geq 1$ which is true. Assuming (8) for f_{i-1}, we now show (8) holds for f_i. Because f_i is the amplification of f_{i-1} with $\delta = 1/(i+1)$, we have by the Amplification Lemma

$$f_i((i+1) \cdot n_0) \geq \left(1 - \frac{1}{i+1}\right) f_{i-1}(i \cdot n_0) + f_{i-1}(n_0).$$

Since $f_{i-1}(n_0) \geq f_0(n_0) \geq 1$ we have

$$f_i((i+1) \cdot n_0) \geq \left(1 - \frac{1}{i+1}\right) f_{i-1}(i \cdot n_0) + 1.$$

Using the inductive hypothesis we have

$$f_i((i+1) \cdot n_0) \geq \left(1 - \frac{1}{i+1}\right) \sum_{j=0}^{i-1} \left(1 - \frac{j}{i}\right) + 1.$$

Note that

$$\begin{aligned}\left(1 - \frac{1}{i+1}\right) \cdot \left(1 - \frac{j}{i}\right) &= \frac{i}{i+1} \cdot \frac{i-j}{i} \\ &= \frac{i-j}{i+1} \\ &= 1 - \frac{j+1}{i+1}.\end{aligned}$$

Thus we have

$$f_i((i+1) \cdot n_0) \geq \sum_{j=1}^{i} \left(1 - \frac{j}{i+1}\right) + 1 = \sum_{j=0}^{i} \left(1 - \frac{j}{i+1}\right),$$

as desired. □

Let $i_* = \lfloor n/n_0 \rfloor - 1$, which by design satisfies $(i_* + 1)n_0 \leq n$. By Claim 3 we have

$$f_{i_*}((i_* + 1) \cdot n_0) \geq \sum_{j=0}^{i_*} \left(1 - \frac{j}{i_* + 1}\right) = i_*/2 + 1.$$

As $i_* = \Theta(n)$, we have thus shown that $\text{alg}(f_{i_*})$ can achieve backlog $\Omega(n)$ on n cups.

Let T_i be the running time of $\text{alg}(f_i)$. The recurrence for the running running time of f_{i_*} is

$$T_i(n) \leq n \cdot n_0 T_{i-1}(n - n_0) + O(1).$$

Clearly $T_{i_*}(n) \leq O(n!)$.

□

5 Upper Bound

In this section we analyze the ***greedy emptier***, which always empties from the p fullest cups. We prove in Corollary 1 that the greedy emptier prevents backlog from exceeding $O(n)$.

In order to analyze the greedy emptier, we establish a system of invariants that hold at every step of the game.

Let $\mu_S(X)$ and $m_S(X)$ denote the average fill and the mass, respectively, of a set of cups X at state S (where $S = S_t$ or $S = I_t$).[5] We will use concatenation of sets to denote unions, i.e. $AB = A \cup B$.

The main result of the section is the following theorem.

Theorem 6. *In the variable-processor cup game on n cups, the greedy emptier maintains, at every step t, the invariants*

$$\mu_{S_t}(S_t([k])) \leq 2n - k \tag{9}$$

for all $k \in [n]$.

By applying Theorem 6 to the case of $k = 1$, we arrive at a bound on backlog:

Corollary 1. *In the variable-processor cup game on n cups, the greedy emptying strategy never lets backlog exceed $O(n)$.*

Proof of Theorem 6. We prove the invariants by induction on t. The invariants hold trivially for $t = 1$ (the base case for the inductive proof): the cups start empty so $\mu_{S_1}(S_1([k])) = 0 \leq 2n - k$ for all $k \in [n]$.

Fix a round $t \geq 1$, and any $k \in [n]$. We assume the invariants for all values of $k' \in [n]$ for state S_t (we will only explicitly use two of the invariants for each k, but the invariants that we need depend on the choice of p_t by the filler) and show that the invariant on the k fullest cups holds on round $t + 1$, i.e. that

$$\mu_{S_{t+1}}(S_{t+1}([k])) \leq 2n - k.$$

Note that because the emptier is greedy it always empties from the cups $I_t([p_t])$. Let A, with $a = |A|$, be $A = I_t([\min(k, p_t)]) \cap S_{t+1}([k])$; A consists of the cups that are among the k fullest cups in I_t, are emptied from, and are among the k fullest cups in S_{t+1}. Let B, with $b = |B|$, be $I_t([\min(k, p_t)]) \setminus A$; B consists of the cups that are among the k fullest cups in state I_t, are emptied from, and are not among the k fullest cups in S_{t+1}. Let $C = I_t(a + b + [k - a])$, with $c = k - a = |C|$; C consists of the cups with ranks $a + b + 1, \ldots, k + b$ in state I_t. The set C is defined so that $S_{t+1}([k]) = AC$, since once the cups in B are emptied from, the cups in B are not among the k fullest cups, so cups in C take their places among the k fullest cups.

Note that $k - a \geq 0$ as $a + b \leq k$, and also $|ABC| = k + b \leq n$, because by definition the b cups in B must not be among the k fullest cups in state S_{t+1} so there are at least $k + b$ cups. Note that $a + b = \min(k, p_t)$. We also have that $A = I_t([a])$ and $B = I_t(a + [b])$, as every cup in A must have higher fill than all cups in B in order to remain above the cups in B after 1 unit of water is removed from all cups in AB.

We now establish the following claim, which we call the ***interchangeability of cups***:

Claim 4. *There exists a cup state S'_t such that: (a) S'_t satisfies the invariants (9), (b) $S'_t(r) = I_t(r)$ for all ranks $r \in [n]$, and (c) the filler can legally place water into cups in order to transform S'_t into I_t.*

Proof. Fix $r \in [n]$. We will show that S_t can be transformed into a state S_t^r by relabelling only cups with ranks in $[r]$ such that (a) S_t^r satisfies the invariants (9), (b) $S_t^r([r]) = I_t([r])$ and (c) the filler can legally place water into cups in order to transform S_t^r into I_t.

Say there are cups x, y with $x \in S_t([r]) \setminus I_t([r]), y \in I_t([r]) \setminus S_t([r])$. Let the fills of cups x, y at state S_t be f_x, f_y; note that

$$f_x > f_y. \tag{10}$$

Let the amount of fill that the filler adds to these cups be $\Delta_x, \Delta_y \in [0, 1]$; note that

$$f_x + \Delta_x < f_y + \Delta_y. \tag{11}$$

Define a new state S'_t where cup x has fill f_y and cup y has fill f_x. Note that the filler can transform state S'_t into state I_t by placing water into cups as before, except changing the amount of water placed into cups x and y to be $f_x - f_y + \Delta_x$ and $f_y - f_x + \Delta_y$, respectively.

[5]Note that in the lower bound proofs (i.e. Section 4 and Section 6) when we use the notation m (for mass) and μ (for average fill), we omit the subscript indicating the state at which the properties are measured. In those proofs the state is implicitly clear. However, in this section it is necessary to make the state S explicit in the notation.

In order to verify that the transformation from S'_t to I_t is a valid step for the filler, one must check three conditions. First, the amount of water placed by the filler is unchanged: this is because $(f_x - f_y + \Delta_x) + (f_y - f_x + \Delta_y) = \Delta_x + \Delta_y$. Second, the fills placed in cups x and y are at most 1: this is because $f_x - f_y + \Delta_x < \Delta_y \leq 1$ (by (11)) and $f_y - f_x + \Delta_x < \Delta_x \leq 1$ (by (10)). Third, the fills placed in cups x and y are non-negative: this is because $f_x - f_y + \Delta_x > \Delta_x \geq 0$ (by (10)) and $f_y - f_x + \Delta_y > \Delta_x \geq 0$ (by (11)).

We can repeatedly apply this process to swap each cup in $I_t([r]) \setminus S_t([r])$ into being in $S'_t([r])$. At the end of this process we will have some state S_t^r for which $S_t^r([r]) = I_t([r])$. Note that S_t^r is simply a relabeling of S_t, hence it must satisfy the same invariants (9) satisfied by S_t. Further, S_t^r can be transformed into I_t by a valid filling step.

Now we repeatedly apply this process, in descending order of ranks. In particular, we have the following process: create a sequence of states by starting with S_t^{n-1}, and to get to state S_t^r from state S_t^{r+1} apply the process described above. Note that S_t^{n-1} satisfies $S_t^{n-1}([n-1]) = I_t([n-1])$ and thus also $S_t^{n-1}(n) = I_t(n)$. If S_t^{r+1} satisfies $S_t^{r+1}(r') = I_t(r')$ for all $r' > r+1$ then S_t^r satisfies $S_t^r(r') = I_t(r')$ for all $r > r$, because the transition from S_t^{r+1} to S_t^r has not changed the labels of any cups with ranks in $(r+1, n]$, but the transition does enforce $S_t^r([r]) = I_t([r])$, and consequently $S_t^r(r+1) = I_t(r+1)$. We continue with the sequential process until arriving at state S_t^1 in which we have $S_t^1(r) = I_t(r)$ for all r. Throughout the process each S_t^r has satisfied the invariants (9), so S_t^1 satisfies the invariants (9). Further, throughout the process from each S_t^r it is possible to legally place water into cups in order to transform S_t^r into I_t.

Hence S_t^1 satisfies all the properties desired, and the proof of Claim 4 is complete.

□

Claim 4 tells us that we may assume without loss of generality that $S_t(r) = I_t(r)$ for each rank $r \in [n]$. We will make this assumption for the rest of the proof.

In order to complete the proof of the theorem, we break it into three cases.

Claim 5. *If some cup in A zeroes out in round t, then the invariant $\mu_{S_{t+1}}(S_{t+1}([k])) \leq 2n - k$ holds.*

Proof. Say a cup in A zeroes out in step t. Of course

$$m_{S_{t+1}}(I_t([a-1])) \leq (a-1)(2n - (a-1))$$

because the $a-1$ fullest cups must have satisfied the invariant (with $k = a-1$) on round t. Moreover, because $\text{fill}_{S_{t+1}}(I_{t+1}(a)) = 0$

$$m_{S_{t+1}}(I_t([a])) = m_{S_{t+1}}(I_t([a-1])).$$

Combining the above equations, we get that

$$m_{S_{t+1}}(A) \leq (a-1)(2n - (a-1)).$$

Furthermore, the fill of all cups in C must be at most 1 at state I_t to be less than the fill of the cup in A that zeroed out. Thus,

$$\begin{aligned}
m_{S_{t+1}}(S_{t+1}([k])) &= m_{S_{t+1}}(AC) \\
&\leq (a-1)(2n - (a-1)) + k - a \\
&= a(2n-a) + a - 2n + a - 1 + k - a \\
&= a(2n-a) + (k-n) + (a-n) - 1 \\
&< a(2n-a)
\end{aligned}$$

as desired. As k increases from 1 to n, $k(2n-k)$ strictly increases (it is a quadratic in k that achieves its maximum value at $k = n$). Thus $a(2n-a) \leq k(2n-k)$ because $a \leq k$. Therefore,

$$m_{S_{t+1}}(S_{t+1}([k])) \leq k(2n-k).$$

□

Claim 6. *If no cups in A zero out in round t and $b = 0$, then the invariant $\mu_{S_{t+1}}(S_{t+1}([k])) \leq 2n - k$ holds.*

Proof. If $b = 0$, then $S_{t+1}([k]) = S_t([k])$. During round t the emptier removes a units of fill from the cups in $S_t([k])$, specifically the cups in A. The filler cannot have added more than k fill to these cups, because it can add at most 1 fill to any given cup. Also, the filler cannot have added more than p_t fill to the cups because this is the total amount of fill that the filler is allowed to add. Hence the filler adds at most $\min(p_t, k) = a + b = a + 0 = a$ fill to these cups. Thus the invariant holds:

$$m_{S_{t+1}}(S_{t+1}([k])) \leq m_{S_t}(S_t([k])) + a - a \leq k(2n - k).$$

□

The remaining case, in which no cups in A zero out and $b > 0$ is the most technically interesting.

Claim 7. *If no cups in A zero out on round t and $b > 0$, then the invariant $\mu_{S_{t+1}}(S_{t+1}([k])) \leq 2n - k$ holds.*

Proof. Because $b > 0$ and $a + b \leq k$ we have that $a < k$, and $c = k - a > 0$. Recall that $S_{t+1}([k]) = AC$, so the mass of the k fullest cups at S_{t+1} is the mass of AC at S_t plus any water added to cups in AC by the filler, minus any water removed from cups in AC by the emptier. The emptier removes exactly a units of water from AC. The filler adds no more than p_t units of water to AC (because the filler adds at most p_t total units of water per round) and the filler also adds no more than $k = |AC|$ units of water to AC (because the filler adds at most 1 unit of water to each of the k cups in AC). Thus, the filler adds no more than $a + b = \min(p_t, k)$ units of water to AC. Combining these observations we have:

$$m_{S_{t+1}}(S_{t+1}([k])) \leq m_{S_t}(AC) + b. \tag{12}$$

The key insight necessary to bound this is to notice that larger values for $m_{S_t}(A)$ correspond to smaller values for $m_{S_t}(C)$ because of the invariants; the higher fill in A ***pushes down*** the fill that C can have. By capturing the pushing-down relationship combinatorially we will achieve the desired inequality.

We can upper bound $m_{S_t}(C)$ by

$$\begin{aligned} m_{S_t}(C) &\leq \frac{c}{b+c} m_{S_t}(BC) \\ &= \frac{c}{b+c}(m_{S_t}(ABC) - m_{S_t}(A)) \end{aligned}$$

because $\mu_{S_t}(C) \leq \mu_{S_t}(B)$ without loss of generality by the interchangeability of cups. Thus we have

$$m_{S_t}(AC) \leq m_{S_t}(A) + \frac{c}{b+c} m_{S_t}(BC) \tag{13}$$

$$= \frac{c}{b+c} m_{S_t}(ABC) + \frac{b}{b+c} m_{S_t}(A). \tag{14}$$

Note that the expression in (14) is monotonically increasing in both $\mu_{S_t}(ABC)$ and $\mu_{S_t}(A)$. Thus, by numerically replacing both average fills with their extremal values, $2n - |ABC|$ and $2n - |A|$. At this point the claim can be verified by straightforward (but quite messy) algebra (and by combining (12) with (14)). We instead give a more intuitive argument, in which we examine the right side of (13) combinatorially.

Consider a new configuration of fills F achieved by starting with state S_t, and moving water from BC into A until $\mu_F(A) = 2n - |A|$. [6] This transformation increases (strictly increases if and only if we move a non-zero amount of water) the right side of (13). In particular, if mass $\Delta \geq 0$ fill is moved from BC to A, then the right side of (13) increases by $\frac{b}{b+c}\Delta \geq 0$. Note that the fact that moving water from BC into A increases the right side of (13) formally captures the way the system of invariants being proven forces a tradeoff between the fill in A and the fill in BC—that is, higher fill in A pushes down the fill that BC (and consequently C) can have.

Since $\mu_F(A)$ is above $\mu_F(ABC)$, the greater than average fill of A must be counter-balanced by the lower than average fill of BC. In particular we must have

$$(\mu_F(A) - \mu_F(ABC))|A| = (\mu_F(ABC) - \mu_F(BC))|BC|.$$

[6] Note that whether or not F satisfies the invariants is irrelevant.

Note that

$$\begin{aligned}
&\mu_F(A) - \mu_F(ABC) \\
&= (2n - |A|) - \mu_F(ABC) \\
&\geq (2n - |A|) - (2n - |ABC|) \\
&= |BC|.
\end{aligned}$$

Hence we must have

$$\mu_F(ABC) - \mu_F(BC) \geq |A|.$$

Thus

$$\mu_F(BC) \leq \mu_F(ABC) - |A| \leq 2n - |ABC| - |A|. \tag{15}$$

Combing (13) with the fact that the transformation from S_t to F only increases the right side of (13), along with (15), we have the following bound:

$$\begin{aligned}
m_{S_t}(AC) &\leq m_F(A) + c\mu_F(BC) \\
&\leq a(2n - a) + c(2n - |ABC| - a) \\
&\leq (a + c)(2n - a) - c(a + c + b) \\
&\leq (a + c)(2n - a - c) - cb.
\end{aligned} \tag{16}$$

By (12) and (16), we have that

$$\begin{aligned}
m_{S_{t+1}}(S_{t+1}([k])) &\leq m_{S_t}(AC) + b \\
&\leq (a + c)(2n - a - c) - cb + b \\
&= k(2n - k) - cb + b \\
&\leq k(2n - k),
\end{aligned}$$

where the final inequality uses the fact that $c \geq 1$. This completes the proof of the claim.

□

We have shown the invariant holds for arbitrary k, so given that the invariants all hold at state S_t they also must all hold at state S_{t+1}. Thus, by induction we have the invariant for all rounds $t \in \mathbb{N}$. □

6 Oblivious Filler Lower Bound

In this section we prove that, surprisingly, an oblivious filler can achieve backlog $n^{1-\varepsilon}$, although only against a certain class of "greedy-like" emptiers.

The ***fill-range*** of a set of cups at a state S is $\max_c \text{fill}_S(c) - \min_c \text{fill}_S(c)$. We call a cup configuration ***R-flat*** if the fill-range of the cups less than or equal to R; note that in an R-flat cup configuration with average fill 0 all cups have fills in $[-R, R]$. We say an emptier is ***Δ-greedy-like*** if, whenever there are two cups with fills that differ by at least Δ, the emptier never empties from the less full cup without also emptying from the more full cup. That is, if on some round t, there are cups c_1, c_2 with $\text{fill}_{I_t}(c_1) > \text{fill}_{I_t}(c_2) + \Delta$, then a Δ-greedy-like emptier doesn't empty from c_2 on round t unless it also empties from c_1 on round t. Note that a perfectly greedy emptier is 0-greedy-like. We call an emptier ***greedy-like*** if it is Δ-greedy-like for $\Delta \leq O(1)$.

With an oblivious filler we are only able to prove lower bounds on backlog against greedy-like emptiers; whether or not our results can be extended to a more general class of emptiers is an interesting open question. Nonetheless, greedy-like emptiers are of great interest because all the known randomized algorithms for the cup game are greedy-like [1, 4].

As a tool in our analysis we define a new variant of the cup game: In the p-processor ***E-extra-emptyings S-skip-emptyings*** negative-fill cup game on n cups, the filler distributes p units of water amongst the cups, and then the emptier empties from *p or more, or less* cups. In particular the emptier is allowed to do E extra emptyings and is also allowed to skip S emptyings over the course of the game. Note that the emptier still cannot empty from the same cup twice on a single round, and also that note that a Δ-greedy-like emptier must take into account extra emptyings and skip emptyings to determine valid moves. Further, note that the emptier is allowed to skip extra emptyings, although skipping extra emptyings looks the same as if the extra-emptyings had simply not been performed. Let the ***regular*** cup game be the 0-extra-emptyings ∞-skip-emptyings cup game: this is the regular cup game. Allowing for some extra emptyings, and bounding the number of skip emptyings is sometimes necessary when analyzing an algorithm that is a subroutine of a larger algorithm however, hence it sometimes makes sense to consider games with different values of E, S. Unless explicitly stated otherwise however we are considering the regular cup game.

For a Δ-greedy-like emptier let $R_\Delta = 2(2+\Delta)$; we now prove a key property of these emptiers: there is an oblivious filling strategy, which we term **flatalg**, that attains an R_Δ-flat cup configuration against a Δ-greedy-like emptier, given cups of a known starting fill-range.

Lemma 6. *Consider an R-flat cup configuration in the p-processor E-extra-emptyings S-skip-emptyings negative-fill cup game on $n = 2p$ cups. There is an oblivious filling strategy* **flatalg** *that achieves an R_Δ-flat configuration of cups against a Δ-greedy-like emptier in running time $2(R + \lceil (1+1/n)(E+S) \rceil)$. Furthermore,* flatalg *guarantees that the cup configuration is R-flat on every round.*

Proof. If $R \le R_\Delta$ the algorithm does nothing, since the desired fill-range is already achieved; for the rest of the proof we consider $R > R_\Delta$.

The filler's strategy is to distribute fill equally amongst all cups at every round, placing $p/n = 1/2$ fill in each cup. Let $\ell_t = \min_{c \in S_t} \text{fill}_{S_t}(c)$, $u_t = \max_{c\in S_t} \text{fill}_{S_t}(c)$.

First we show that the fill-range of the cups can only increase if the fill-range is very small.

Claim 8. *If $u_{t+1} - \ell_{t+1} > u_t - \ell_t$ then*

$$u_{t+1} - \ell_{t+1} \le R_\Delta.$$

Proof. First we remark that the fill of any cup changes by at most $1/2$ from round to round, and in particular $|u_{t+1} - u_t| \le 1/2$, $|\ell_{t+1} - \ell_t| \le 1/2$. In order for the fill-range to increase, the emptier must have emptied from some cup with fill in $[\ell_t, \ell_t+1]$ without emptying from some cup with fill in $[u_t-1, u_t]$; if the emptier had not emptied from every cup with fill in $[\ell_t, \ell_t+1]$ then we would have $\ell_{t+1} = \ell_t + 1/2$ so the fill-range could not have increased, and similarly if the emptier had emptied from every cup with fill in $[u_t - 1, u_t]$ then we would have $u_{t+1} = u_t - 1/2$ so again the fill-range could not have increased. Because the emptier is Δ-greedy-like emptying from a cup with fill at most $\ell_t + 1$ and not emptying from a cup with fill at least $u_t - 1$ implies that $u_t - 1$ and $\ell_t + 1$ differ by at most Δ. Thus,

$$u_{t+1} - \ell_{t+1} \le u_t + 1/2 - (\ell_t - 1/2) \le \Delta + 3 \le R_\Delta.$$

□

Because by Claim 8 whenever the fill-range of the cups increases it increases to a value bounded above by $R_\Delta \le R$, we have by induction that the fill-range of the cups never exceeds R, i.e. the cups are always R-flat. While Claim 8 does imply that the fill-range must decrease until the fill-range is at most R_Δ, and once the fill-range is at most R_Δ it is always at most R_Δ, Claim 8 does not preclude the possibility that the fill-range doesn't change for many rounds, or decreases by a very small amount. For this reason we actually do not use Claim 8 in the remainder of the proof; we proved this result because the fact that fill-range does not increase during flatalg is an important property of flatalg. In the rest of the proof we establish that the fill-range indeed must eventually be at most R_Δ.

Let L_t be the set of cups c with $\text{fill}_{S_t}(c) \le \ell_t + 2 + \Delta$, and let U_t be the set of cups c with $\text{fill}_{S_t}(c) \ge u_t - 2 - \Delta$.

Now we prove a key property of the sets U_t and L_t: if a cup is in U_t or L_t it is also in $U_{t'}, L_{t'}$ for all $t' > t$. This follows immediately from Claim 9.

Claim 9.

$$U_t \subseteq U_{t+1}, \quad L_t \subseteq L_{t+1}.$$

Proof. Consider a cup $c \in U_t$.

If c is not emptied from, i.e. $\text{fill}(c)$ has increased by $1/2$ from the previous round, then clearly $c \in U_{t+1}$, because backlog has increased by at most $1/2$, so $\text{fill}(c)$ must still be within $2 + \Delta$ of the backlog on round $t+1$.

On the other hand, if c is emptied from, i.e. $\text{fill}(c)$ has decreased by $1/2$, we consider two cases.
Case 1: If $\text{fill}_{S_t}(c) \ge u_t - \Delta - 1$, then $\text{fill}_{S_t}(c)$ is at least 1 above the bottom of the interval defining which cups belong to U_t. The backlog increases by at most $1/2$ and the fill of c decreases by $1/2$, so $\text{fill}_{S_{t+1}}(c)$ is at least $1 - 1/2 - 1/2 = 0$ above the bottom of the interval, i.e. still in the interval.
Case 2: On the other hand, if $\text{fill}_{S_t}(c) < u_t - \Delta - 1$, then every cup with fill in $[u_t - 1, u_t]$ must have been emptied from because the emptier is Δ-greedy-like. Therefore the fullest cup on round $t+1$ is the

same as the fullest cup on round t, because every cup with fill in $[u_t - 1, u_t]$ has had its fill decrease by $1/2$, and no cup with fill less than $u_t - 1$ had its fill increase by more than $1/2$. Hence $u_{t+1} = u_t - 1/2$. Because both $\text{fill}(c)$ and the backlog have decreased by $1/2$, the distance between them is still at most $\Delta + 2$, hence $c \in U_{t+1}$.
The argument for why $L_t \subseteq L_{t+1}$ is symmetric. □

Now we show that under certain conditions u_t decreases and ℓ_t increases.

Claim 10. *On any round t where the emptier empties from at least $n/2$ cups, if $|U_t| \leq n/2$ then $u_{t+1} = u_t - 1/2$. On any round t where the emptier empties from at most $n/2$ cups, if $|L_t| \leq n/2$ then $\ell_{t+1} = \ell_t + 1/2$.*

Proof. Consider a round t where the emptier empties from at least $n/2$ cups. If there are at least $n/2$ cups outside of U_t, i.e. cups with fills in $[\ell_t, u_t - 2 - \Delta]$, then all cups with fills in $[u_t - 2, u_t]$ must be emptied from; if one such cup was not emptied from then by the pigeon-hole principle some cup outside of U_t was emptied from, which is impossible as the emptier is Δ-greedy-like. This clearly implies that $u_{t+1} = u_t - 1/2$: no cup with fill less than $u_t - 2$ has gained enough fill to become the fullest cup, and the fullest cup from the previous round has lost $1/2$ unit of fill.

By a symmetric argument $\ell_{t+1} = \ell_t + 1/2$ if the emptier empties at most $n/2$ cups on a round t where $|L_t| \leq n/2$. □

Now we show that eventually $L_t \cap U_t \neq \varnothing$.

Claim 11. *There is a round $t_0 \leq 2(R + \lceil (1 + 1/n)(E + S) \rceil)$ such that $U_t \cap L_t \neq \varnothing$ for all $t \geq t_0$.*

Proof. We call a round where the emptier doesn't use $p = n/2$ resources, i.e. a round where the number of skipped emptyings and the number of extra emptyings are not equal, an ***unbalanced round***; we call a round that is not unbalanced a ***balanced*** round.

Note that there are clearly at most $E + S$ unbalanced rounds. We now associate some unbalanced rounds with balanced rounds; in particular we define what it means for a balanced round to ***cancel*** an unbalanced round. We define cancellation by a sequential process. For $i = 1, 2, \ldots, 2(R + \lceil (1 + 1/n)(E + S) \rceil)$ (iterating in ascending order of i), if round i is unbalanced then we say that the first balanced round $j > i$ that hasn't already been assigned (earlier in the sequential process) to cancel another unbalanced round $i' < i$, if any such round j exists, ***cancels*** round i. Note that cancellation is a one-to-one relation: each unbalanced round is cancelled by at most one balanced round and each balanced round cancels at most one unbalanced round.

Consider rounds of the form $2(R + \lceil (E + S)/n \rceil) + (E + S) + i$ for $i \in [E + S + 1] - 1$. We claim that there is some such i such that among rounds $[2(R + \lceil (E + S)/n \rceil) + (E + S) + i]$ every unbalanced round has been cancelled, and such that there are $2(R + \lceil (E + S)/n \rceil)$ balanced rounds not cancelling other rounds. Assume for contradiction that such an i does not exist. Note that there are at least $2(R + \lceil (E + S)/n \rceil)$ balanced rounds in the first $2(R + \lceil (E + S)/n \rceil) + (S + E)$ rounds. Thus every balanced round $2R + (E + S) + \lceil (E + S)/n \rceil + i - 1$ for $i \in [E + S + 1]$ is necessarily a cancelling round, or else there would be a round by which there are no uncancelled unbalanced rounds. Hence by round $2(R + \lceil (E + S)/n \rceil) + 2(E + S)$, there must have been $E + S$ cancelled rounds, so on round $2(R + \lceil (E + S)/n \rceil) + 2(E + S)$ all unbalanced rounds are cancelled, which leaves $2(R + \lceil (E + S)/n \rceil)$ balanced rounds that are not cancelling any rounds, as desired.

Let t_e be the first round by which there are $2(R + \lceil (E + S)/n \rceil)$ balanced non-cancelling rounds. Note that the average fill of the cups cannot have decreased by more than E/n from its starting value; similarly the average fill of the cups cannot have increased by more than S/n. Because the cups start R-flat, we have that u_t cannot have decreased by more than $R + E/n$ or else u_t would necessarily be below the average fill, and identically ℓ_t cannot have increased by more than $R + S/n$ or else it would be above the average fill. Now, by Claim 10 we have that eventually $|L_t| > n/2$: if $|L_t| \leq n/2$ were always true, then on every balanced round ℓ_t would have increased by $1/2$, and since ℓ_t increases by at most $1/2$ on unbalanced rounds, this implies that in total ℓ_t would have increased by at least $(1/2)2(R + \lceil (E + S)/n \rceil)$, which is impossible. By a symmetric argument it is impossible that $|U_t| \leq n/2$ for all rounds.

Since $|U_{t+1}| \geq |U_t|$ and $|L_{t+1}| \geq |L_t|$ by Claim 9, we have that there is some round $t_0 \in [2(R + \lceil (1 + 1/n)(E + S) \rceil)]$ such that for all $t \geq t_0$ we have $|U_t| > n/2$ and $|L_t| > n/2$. But then we have $U_t \cap L_t \neq \varnothing$, as desired. □

If there exists a cup $c \in L_t \cap U_t$, then

$$\text{fill}(c) \in [u_t - 2 - \Delta, u_t] \cap [\ell_t, \ell_t + 2 + \Delta].$$

Hence we have that

$$\ell_t + 2 + \Delta \geq u_t - 2 - \Delta.$$

Rearranging,

$$u_t - \ell_t \leq 2(2 + \Delta) = R_\Delta.$$

Thus the cup configuration is R_Δ-flat by the end of this flattening process.

□

Next we describe a simple oblivious filling strategy, that we call **randalg**, that will be used as a subroutine in Lemma 7; variants of this strategy are well-known, and similar versions of it can be found in [1, 2, 3, 4].

Proposition 5. *Consider an R-flat cup configuration in the regular single-processor ∞-extra-emptyings ∞-skip-emptyings negative-fill cup game on n cups with initial average fill μ_0. Let $k \in [n]$ be a parameter. Let $d = \sum_{i=2}^{k} 1/i$.*

There is an oblivious filling strategy **randalg(k)** *with running time $k - 1$ that achieves fill at least $\mu_0 - R + d$ in a known cup c with probability at least $1/k!$ if we condition on the emptier not performing extra emptyings.* randalg(k) *achieves fill at most $\mu_0 + R + d$ in this cup (unconditionally).*

Furthermore, when applied against a Δ-greedy-like emptier with $R = R_\Delta$, randalg(k) *guarantees that the cup configuration is $(R + d)$-flat on every round (unconditionally).*

Proof. First we condition on the emptier does not using extra emptying and show that in this case the filler has probability at least $1/(k-1)!$ (which we lower bound by $1/k!$ for sake of simplicity) of attaining a cup with fill at least $\mu_0 - R + d$. The filler maintains an ***active set***, initialized to being an arbitrary subset of k of the cups. Every round the filler distributes 1 unit of fill equally among all cups in the active set. Next the emptier removes 1 unit of fill from some cup, or skips its emptying. Then the filler removes a random cup from the active set (chosen uniformly at random from the active set). This continues until a single cup c remains in the active set.

We now bound the probability that c has never been emptied from. Assume that on the i-th step of this process, i.e. when the size of the active set is $n - i + 1$, no cups in the active set have ever been emptied from; consider the probability that after the filler removes a cup randomly from the active set there are still no cups in the active set that the emptier has emptied from. If the emptier skips its emptying on this round, or empties from a cup not in the active set then it is trivially still true that no cups in the active set have been emptied from. If the cup that the emptier empties from is in the active set then with probability $1/(k - i + 1)$ it is evicted from the active set, in which case we still have that no cup in the active set has ever been emptied from. Hence with probability at least $1/(k - 1)!$ the final cup in the active set, c, has never been emptied from. In this case, c will have gained fill $d = \sum_{i=2}^{k} 1/i$ as claimed. Because c started with fill at least $-R + \mu_0$, c now has fill at least $-R + d + \mu_0$.

Now note that regardless of if the emptier uses extra emptyings c has fill at most $\mu_0 + R + d$, as c starts with fill at most R, and c gains at most $1/(k - i + 1)$ fill on the i-th round of this process.

Now we analyze this algorithm specifically for a Δ-greedy-like emptier. Let $\mathcal{A}_t$ be the event that the anti-backlog is smaller in S_{t+1} than in S_t, let $\mathcal{B}_t$ be the event that some cup with fill equal to the backlog in S_{t+1} was emptied from on round t. If $\mathcal{A}_t$ and $\mathcal{B}_t$ are both true on round t, then by greediness the cups are quite flat. In particular, let a be a cup with fill equal to the anti-backlog in state S_{t+1} that was emptied from on round t, and let b be a cup with fill equal to the backlog in state S_{t+1} that was not emptied from on round t. By greediness $\text{fill}_{I_t}(a) + \Delta > \text{fill}_{I_t}(b)$. Of course $\text{fill}_{I_t}(b) = \text{fill}_{S_{t+1}}(b)$; for b to have fill equal to the backlog on round $t + 1$, b must have fill less than 1 below backlog on round t. Of course $\text{fill}_{I_t}(a) \leq \text{fill}_{S_t}(a) + 1$; for a to have fill equal to the anti-backlog on round $t + 1$, a must have fill less than 1 above the anti-backlog on round t. Thus we have that the backlog and anti-backlog differ by at most $\Delta + 3 \leq R_\Delta$ on round t, i.e. the cups are R_Δ-flat.

Consider a round t_1 where the cups are not R_Δ-flat. Let t_0 be the last round that the cups were R_Δ-flat. On all rounds $t \in (t_0, t_1)$ at least one of $\mathcal{A}_t$ or $\mathcal{B}_t$ must not hold. On a round where $\mathcal{A}_t$ does not hold, anti-backlog does not decrease and backlog increases by at most $1/(k-t+1)$, so fill range increases by at most $1/(k - t + 1)$. On a round where $\mathcal{B}_t$ does not hold, anti-backlog decreases by at most 1 and backlog decreases by at least $1 - 1/(k - t + 1)$, as all cups with fill equal to the backlog in state S_{t+1} were emptied from on round t, so fill-range increases by at most $1/(k - t + 1)$. Hence in total fill-range increases by at most $\sum_{i=2}^{k} 1/i$ from R, i.e. the cups are $(R + d)$-flat on round t_1.

□

We now give a method for transforming a filling strategy for achieving large backlog into a filling strategy for achieving high fill in many cups, or high average fill in a set of cups (which of these we guarantee depends on the original filling strategy). The idea of repeating an algorithm many times is also used in the proof of the Adaptive Amplification Lemma; the construction is slightly more complicated in the randomized case however, and is much harder to analyze.

Definition 2. Let alg_0 be an oblivious filling strategy, that can get high fill (for some definition of high) in some cup against greedy-like emptiers with some probability. We construct a new filling strategy $\mathbf{rep}_{\boldsymbol{\delta}}\mathbf{(alg_0)}$ (rep stands for "repetition") as follows:

Say we have some configuration of n cups. Let $n_A = \lceil \delta n \rceil, n_B = \lfloor (1-\delta)n \rfloor$. Let $N \gg n$ be large, let $M = 2^{\text{polylog}(N)}$ be a chosen parameter. Initialize A to $\varnothing$ and B to being all of the cups. We call A the ***anchor set*** and B the ***non-anchor set***. The filler always places 1 unit of fill in each anchor cup on each round. The filling strategy consists of n_A ***donation-processes***, which are procedures that result in a cup being ***donated*** from B to A (i.e. removed from B and added to A). At the start of each donation-processes the filler chooses a value m_0 uniformly at random from $[M]$. We say that the filler ***applies*** a filling strategy alg to B if the filler uses alg on B while placing 1 unit of fill in each anchor cup. During the donation-process the filler applies alg_0 to B m_0 times, and flattens B by applying flatalg to B for $\Theta(N^2)$ rounds before each application of alg_0. At the end of each donation process the filler takes the cup given by the final application of alg_0 (i.e. the cup that alg_0 guarantees with some probability against a certain class of emptiers to have a certain high fill), and donates this cup to A.

We give pseudocode for this algorithm in Algorithm 2.

Algorithm 2 $\text{rep}_\delta(\text{alg}_0)$

Input: $\text{alg}_0, \delta, N, M$, set of n cups
Output: Guarantees on the sets A, B (will vary based on alg_0)

```
n_A ← ⌈δn⌉ , n_B ← ⌊(1 − δ)n⌋
A ← ∅, B ← all the cups
Always place 1 fill in each cup in A
for i ∈ [n_A] do                                  ▷ Donation-Processes
    m_0 ← random([M])
    for j ∈ [m_0] do
        Apply flatalg to B for Θ(N^2) rounds
        Apply alg_0 to B
    Donate the cup given by alg_0 from B to A
```

We say that the emptier ***neglects*** the anchor set on a round if it does not empty from each anchor cup. We say that an application of alg_0 to B is ***non-emptier-wasted*** if the emptier does not neglect the anchor set during any round of the application of alg_0.

We use the idea of repeating an algorithm in two different contexts. First in Proposition 6 we prove a result analogous to that of Proposition 3: in particular, we show that we can achieve constant fill in a known cup by using $\text{rep}_{\Theta(1)}(\text{randalg}(\Theta(1))$ which achieves, by a Chernoff bound, $\Theta(n)$ unknown cups with constant fill, and then exploiting the emptier's greedy-like nature to achieve constant fill in a known cup. After doing this, we prove the ***Oblivious Amplification Lemma***, a result analogous to the Adaptive Amplification Lemma: in particular, we show how to take an algorithm for achieving some backlog, and then achieve higher backlog by repeating the algorithm many times. Although these results have deterministic analogues, their proofs are different and significantly more complex than the proofs for the deterministic cases.

In the rest of the section our aim is to achieve backlog $N^{1-\varepsilon}$ in N cups. We will use this value N within all of the following proofs. Many values implicitly depend on N. Note that we implicitly consider the cups to be part of a larger game in these results. Also note that we are happy if mass N^2 is achieved in the cups, because then backlog is always at least N.

Before proving Proposition 6 we analyze $\text{rep}_{\Theta(1)}(\text{randalg}(\Theta(1)))$ in Lemma 7.

Lemma 7. *Let $\Delta \le O(1)$, let $h \le O(1)$ with $h \ge 16 + 16\Delta$, let $k = \lceil e^{2h+1} \rceil$, let $\delta = \Theta(e^{-2h})$, let n be at least a sufficiently large constant determined by h and Δ. Consider an R_Δ-flat cup configuration in the variable-processor cup game on n cups with initial average fill μ_0.*

Against a Δ-greedy-like emptier, $\text{rep}_\delta(\text{randalg}(k))$ *using* $M = \Theta(N^2)$ *either achieves mass at least* N^2 *in the cups, or with probability at least* $1 - 2^{-\Omega(n)}$ *makes an (unknown) set of* $\Theta(n)$ *cups in* A *have fill at least* $h + \mu_0$ *while also guaranteeing that* $\mu(B) \geq -h/2 + \mu_0$*, where* A, B *are the sets defined in Definition 2. Furthermore,* $\text{rep}_\delta(\text{randalg}(k))$ *has running time poly*(N)*.*

Proof. We use the definitions given in Definition 2.

Without loss of generality we assume that the emptier does not neglect the anchor set more than N^2 in any particular donation-process; if the emptier chooses to neglect the anchor set this much then the anchor cups will have achieved mass N^2 so Lemma 7 is already fulfilled. Similarly we assume that the emptier does not choose to skip more than N^2 emptyings; doing so clearly would result in mass at least mass N^2 in the cups.

As in Proposition 5, we define $d = \sum_{i=2}^{k} 1/i$; we remark that, because harmonic numbers grow like $x \mapsto \ln x$, it is clear that $d = \Theta(h)$. We say that an application of randalg(k) to D is ***lucky*** if it achieves backlog at least $\mu_S(B) - R_\Delta + d$ where S denotes the state of the cups at the start of the application of randalg(k); note that by Proposition 5 if we condition on an application of randalg(k) where B started R_Δ-flat being non-emptier-wasted then the application has at least a $1/k!$ chance of being lucky.

Now we prove several important bounds satisfied by A and B.

Claim 12. *All applications of* flatalg *make* B *be* R_Δ*-flat and* B *is always* $(R_\Delta + d)$*-flat.*

Proof. Given that the application of flatalg immediately prior to an application of randalg(k) made B be R_Δ-flat, by Proposition 5 we have that B will stay $(R_\Delta + d)$-flat during the application of randalg(k). Given that the application of randalg(k) immediately prior to an application of flatalg resulted in B being $(R_\Delta + d)$-flat, we have that B remains $(R_\Delta + d)$-flat throughout the duration of the application of flatalg by Lemma 6. Given that B is $(R_\Delta + d)$-flat before a donation occurs B is clearly still $(R_\Delta + d)$-flat after the donation, because the only change to B during a donation is that a cup is removed from B which cannot increase the fill-range of B. Note that B started R_Δ-flat at the beginning of the first donation-process. Note that if an application of flatalg begins with B being $(R_\Delta + d)$-flat, then by considering the flattening to happen in the $(|B|/2)$-processor N^2-extra-emptyings N^2-skip-emptyings cup game we ensure that it makes B be R_Δ-flat. Hence we have by induction that B has always been $(R_\Delta + d)$-flat and that all flattening processes have made B be R_Δ-flat. □

Now we aim to show that $\mu(B)$ is never very low, which we need in order to establish that every non-emptier-wasted lucky application of randalg(k) gets a cup with high fill. Interestingly, in order to lower bound $\mu(B)$ we find it convenient to first upper bound $\mu(B)$, which by greediness and flatness of B gives an upper bound on $\mu(A)$ which we then use to get a lower bound on $\mu(B)$.

Claim 13. *We have always had*

$$\mu(B) \leq \mu(AB) + 2.$$

Proof. There are two ways that $\mu(B) - \mu(AB)$ can increase:

Case 1: The emptier could empty from 0 cups in B while emptying from every cup in A.

Case 2: The filler could evict a cup with fill lower than $\mu(B)$ from B at the end of a donation-process.

Note that cases are exhaustive, in particular note that if the emptier skips more than 1 emptying then $\mu(B) - \mu(AB)$ must decrease because $|B| > |AB|/2$, as opposed to in Case 1 where $\mu(B) - \mu(AB)$ increases.

In Case 1, because the emptier is Δ-greedy-like,

$$\min_{a \in A} \text{fill}(a) > \max_{b \in B} \text{fill}(b) - \Delta.$$

Thus $\mu(B) \leq \mu(A) + \Delta$. We can use this to get an upper bound on $\mu(B) - \mu(AB)$. We have,

$$\begin{aligned} \mu(B) &= \frac{\mu(AB)|AB| - \mu(A)|A|}{|B|} \\ &\leq \frac{\mu(AB)|AB| - (\mu(B) - \Delta)|A|}{|B|}. \end{aligned}$$

Rearranging terms:

$$\mu(B)\left(1 + \frac{|A|}{|B|}\right) \leq \frac{\mu(AB)|AB| + \Delta|A|}{|B|}.$$

Now, because $|A| \cdot \Delta \leq n_A \cdot \Delta < n$ (by our choice of δ to be a very small constant), we have

$$\mu(B)\frac{|AB|}{|B|} \leq \frac{\mu(AB)|AB| + n}{|B|}.$$

Isolating $\mu(B)$ we have

$$\mu(B) \leq \mu(AB) + 1.$$

Consider the final round on which B is skipped while A is not skipped (or consider the first round if there is no such round).

From this round onwards the only increase to $\mu(B) - \mu(AB)$ is due to B evicting cups with fill well below $\mu(B)$. We can upper bound the increase of $\mu(B) - \mu(AB)$ by the increase of $\mu(B)$ as $\mu(AB)$ is strictly increasing.

The cup that B evicts at the end of a donation-process has fill at least $\mu(B) - R_\Delta - (k-1)$, as the running time of randalg(k) is $k-1$, and because B starts R_Δ-flat by Claim 12. Evicting a cup with fill $\mu(B) - R_\Delta - (k-1)$ from B changes $\mu(B)$ by $(R_\Delta + k - 1)/(|B| - 1)$ where $|B|$ is the size of B before the cup is evicted from B. Even if this happens on each of the n_A donation-processes $\mu(B)$ cannot rise higher than $n_A(R_\Delta + k - 1)/(n - n_A)$ which by design in choosing $n_B \gg n_A$, as was done in choosing $\delta = \Theta(e^{-2h})$, is at most 1.

Thus $\mu(B) \leq \mu(AB) + 2$ is always true.

□

Now, the upper bound on $\mu(B) - \mu(AB)$ along with the guarantee that B is flat allows us to bound the highest that a cup in A could rise by greediness, which in turn upper bounds $\mu(A)$ which in turn lower bounds $\mu(B)$.

Claim 14. *We always have*

$$\mu(B) \geq -h/2 + \mu_0.$$

Proof. By Claim 13 and Claim 12 we have that no cup in B ever has fill greater than $u_B = \mu(AB) + 2 + R_\Delta + d$. Let $u_A = u_B + \Delta + 1$. We claim that the backlog in A never exceeds u_A. Note that $\mu(AB), u_A, u_B$ are implicitly functions of the round; $\mu(AB)$ can increase from μ_0 if the emptier skips emptyings.

Consider how high the fill of a cup $c \in A$ could be. If c came from B then when it is donated to A its fill is at most u_B; otherwise, c started with fill at most R_Δ. Both of these expressions are less than $u_A - 1$. Now consider how much the fill of c could increase while being in A. Because the emptier is Δ-greedy-like, if a cup $c \in A$ has fill more than Δ higher than the backlog in B then c must be emptied from, so any cup with fill at least $u_B + \Delta = u_A - 1$ must be emptied from, and hence u_A upper bounds the backlog in A.

Of course an upper bound on backlog in A also serves as an upper bound on the average fill of A as well, i.e. $\mu(A) \leq u_A$. Now we have

$$\begin{aligned}
&\mu(B) \\
&= -\frac{|A|}{|B|}\mu(A) + \frac{|AB|}{|B|}\mu(AB) \\
&\geq -(\mu(AB) + 3 + R_\Delta + d + \Delta)\frac{|A|}{|B|} + \frac{|AB|}{|B|}\mu(AB) \\
&= -(3 + R_\Delta + d + \Delta)\frac{|A|}{|B|} + \mu(AB) \\
&\geq -h/2 + \mu(AB)
\end{aligned}$$

where the final inequality follows because $\mu(AB) \geq 0$, and $|B| \gg |A|$, in particular by our choice of $\delta = \Theta(e^{-2h})$. Of course $\mu(AB) \geq \mu_0$ so we have

$$\mu(B) \geq -h/2 + \mu_0.$$

□

Now we show that at least a constant fraction of the donation-processes succeed with exponentially good probability.

Claim 15. *By choosing $M = \Theta(N^2)$ the filler can guarantee that with probability at least $1 - 2^{-\Omega(n)}$, the filler achieves fill at least $h + \mu_0$ in $\Theta(n)$ of the cups in A.*

Proof. If the emptier was not allowed to neglect the anchor set ever then the claim would clearly be true as each application of randalg(k) would simply succeed with constant probability, so a Chernoff bound would give that $\Theta(n)$ of the donation-processes donate a cup with fill at least $\mu(B) - R_\Delta + d \geq h + \mu_0$, where the inequality follows from Claim 14 which asserts that $\mu(B) \geq -h/2 + \mu_0$, and from the facts $d \geq 2h$ and $h \geq 16(1 + \Delta)$.

However, the emptier is allowed to neglect the anchor set, and worse, the emptier can choose to neglect the anchor set conditional on the filler's progress during randalg(k)! However, by applying randalg(k) a random number of times, chosen from $[M]$ (where $M = \Theta(N^2)$), we guarantee that with exponentially good probability in M the filler succeeds many times, in particular $\Theta(N^2)$ times. But since the emptier cannot neglect the anchor set more than N^2 times, by appropriately large choice of M we can make it so that the filler succeeds at least $2N^2$ times with exponentially good probability. Then the emptier would have at best a $1/2$ chance of preventing the donation-process from giving away a cup with fill $h + \mu_0$ whenever one such cup is achieved. We now formalize this reasoning.

We can lower bound the probability of getting $\Theta(n)$ cups with fills all at least $h + \mu_0$ by considering an augmented emptier that is allowed to ***interfere*** with N^2 applications of randalg(k) per donation-process that only interferes with applications of randalg(k) that would otherwise donate a cup with fill at least $h + \mu_0$ into A; if this (augmented) emptier interferes with an application of randalg(k) then the application is ***emptier-wasted***, i.e. we assume no guarantees on the fill it achieved. The optimal strategy for such an emptier, for the goal of maximizing the probability that the final round in a donation-process is interfered with, given our filler's strategy of randomly choosing how many times to apply randalg(k) before donating a cup, is obviously to interfere with the first N^2 applications of randalg(k) that would have achieved a cup with fill $h + \mu_0$ without interference.

Let $M = 4N^2k!$; note that as stated previously we choose $M = \Theta(N^2)$. Recall that conditional on the emptier not interfering, each of these applications of randalg(k) has at least a $1/k!$ chance of getting a cup with fill h. Hence, by a Chernoff bound with exponentially good probability in M at least $2N^2$ of the M applications of randalg(k) have the potential to donate a cup with fill $h + \mu_0$ to A, if the emptier does not interfere. The filler chooses an application uniformly at random from $[M]$ on which to donate a cup. With probability at least $1/k!$ this is on an application where the filler could get a cup with fill $h + \mu_0$ in A if the emptier does not interfere, and with probability at least $1/2$ the emptier does not interfere on this application of randalg(k), because the emptier can interfere on at most N^2 of the applications of randalg(k).

Against this augmented emptier whether or not donation-processes achieve a cup with fill $h + \mu_0$ in A are independent events. As each happens with at least constant probability, by a Chernoff bound there is exponentially high probability that at least a constant fraction of them succeed.

Note that we used a Chernoff bound in two distinct places: (a) in guaranteeing that each donation-process consists of at least $2N^2$ applications of randalg(k) that would donate a cup with fill $\mu_0 + h$ if the emptier did not interfere, and (b) in guaranteeing that a constant fraction of the donation-processes succeed given that their successes are independent and all happen with constant probability. The Chernoff bound in (a) is actually with exponentially good probability in $M \gg n$, but of course also holds with exponentially good probability in n. Then we can take a union bound over poly(n) events that all occur with exponentially good probability in n, which gives still gives exponentially good probability in n that all of the desired events occur.

The described augmented emptier is clearly strictly more powerful than the real emptier, so the result transfers over. □

We now analyze the running time of the filling strategy. There are n_A donation-processes. Each donation-process consists of $O(M)$ applications of randalg(k), which each take time $O(1)$, and $O(M)$ applications of flatalg, which each take $\Theta(N^2)$ time. Thus overall the algorithm takes time

$$n_A \cdot O(M)(O(1) + O(N^2)) = \text{poly}(N),$$

as desired.

□

Now, using Lemma 7 we show in Proposition 6 that an oblivious filler can achieve constant fill in a known cup.

Proposition 6. *Let $H \le O(1)$, let $\Delta \le O(1)$, let $n \ll N$ be at least a sufficiently large constant determined by H and Δ. Consider an R_Δ-flat cup configuration in the variable-processor cup game on n cups with average fill μ_0. There is an oblivious filling strategy that either achieves mass N^2 among the cups, or achieves fill at least $\mu_0 + H$ in a chosen cup in running time poly(N) against a Δ-greedy-like emptier with probability at least $1 - 2^{-\Omega(n)}$.*

Proof. The filler starts by using $\text{rep}_\delta(\text{randalg}(k))$ with parameter settings as in Lemma 7 where $h = H \cdot 16(1+\Delta)$, i.e. $k = \lceil e^{2h+1} \rceil$, $\delta = \Theta(e^{-2h})$. If this results in mass N^2 among the cups we are done; we assume this is not the case for the rest of the proof. Let the number of cups which, with exponentially good probability in n, now exist by Lemma 7 with fill at least $h + \mu_0$ be of size $nc = \Theta(n)$.

The filler sets $p = 1$, i.e. uses a single processor. Now the filler exploits the emptier's greedy-like nature to to get fill H in a chosen cup c_0. Specifically, for $(5/8)h$ rounds the filler places 1 unit of fill into c_0. Because the emptier is Δ-greedy-like it must empty from the nc cups in A with fill at least $h + \mu_0$ until c_0 has large fill. Over $(5/8)h$ rounds the cups in A cannot have their fill decrease below $(3/8)h \ge h/8 + \Delta + \mu_0$. Hence, any cups with fills less than $h/8 + \mu_0$ must not be emptied from during these rounds. The fill of c_0 started as at least $-h/2 + \mu_0$ as $\mu(B) \ge -h/2 + \mu_0$. After $(5/8)h$ rounds c_0 has fill at least $h/8 + \mu_0$, because the emptier cannot have emptied c_0 until it attained fill $h/8 + \mu_0$, and if c_0 is never emptied from then it achieves fill $h/8 + \mu_0$. Thus the filling strategy achieves backlog $h/8 + \mu_0 \ge H + \mu_0$ in c_0, a known cup, as desired.

The running time is of course still poly(N) by Lemma 7.

□

Next we prove the ***Oblivious Amplification Lemma***.

Lemma 8 (Oblivious Amplification Lemma)**.** *Let $\delta \in (0, 1/2)$ be a constant parameter. Let $\Delta \le O(1)$. Consider a cup configuration in the variable-processor cup game on $n \le N, n > \Omega(1/\delta^2)$ cups with average fill μ_0 that is R_Δ-flat. Let* $\text{alg}(f)$ *be an oblivious filling strategy that either achieves mass N^2 or, with failure probability at most $p \ge 2^{-\lg^8 N}$, achieves backlog $\mu_0 + f(n)$ on such cups in running time $T(n)$ against a Δ-greedy-like emptier. Let $M = 2^{polylog(N)}$.*

Consider a cup configuration in the variable-processor cup game on $n \le N, n > \Omega(1/\delta^2)$ cups with average fill μ_0 that is R_Δ-flat. There exists an oblivious filling strategy $\text{alg}(f')$ *that either achieves mass N^2 or with failure probability at most*

$$p' \le np + 2^{-\lg^8 N}$$

achieves backlog $f'(n)$ satisfying

$$f'(n) \ge (1-\delta)^2 f(\lfloor (1-\delta)n \rfloor) + f(\lceil \delta n \rceil) + \mu_0$$

and $f'(n) \ge f(n)$, in running time

$$T'(n) \le Mn \cdot T(\lfloor (1-\delta)n \rfloor) + T(\lceil \delta n \rceil)$$

against a Δ-greedy-like emptier.

Proof. We use the definitions and notation given in Definition 2.

Note that the emptier cannot neglect the anchor set more than N^2 times per donation-process, and the emptier cannot skip more than N^2 emptyings, without causing the mass of the cups to be at least N^2; we assume for the rest of the proof that the emptier chooses not to do this.

The filler defaults to using $\text{alg}(f)$ on all the cups if

$$f(n) \ge (1-\delta)^2 f(n_B) + f(n_A).$$

In this case our strategy trivially has the desired guarantees. In the rest of the proof we consider the case where we cannot simply fall back on $\text{alg}(f)$ to achieve the desired backlog.

The filler's strategy is roughly as follows:

Step 1: Make $\mu(A) \ge (1-\delta)^2 f(n_B)$ by using $\text{rep}_\delta(\text{alg}(f))$ on all the cups, i.e. applying $\text{alg}(f)$ repeatedly to B, flattening B before each application, and then donating a cup from B to A.

Step 2: Flatten A using flatalg, and then use $\text{alg}(f)$ on A.

Now we analyze Step 1, and show that by appropriately choosing parameters it can be made to succeed.

For this proof we need all donation-processes to succeed, as opposed to in the proof of Lemma 7 in which we only needed a constant fraction of the donation-processes to succeed. This necessitates choosing

M very large. In particular we choose $M = 2^{\log^{24} N}$ —recall that $[M]$ is the set from which we randomly choose how many times to apply $\text{alg}(f)$ in a donation-process. By choosing M this large we cannot hope to guarantee that every application of $\text{alg}(f)$ succeeds: there are far too many applications. On the other hand, having M so large allows us to have a very tight concentration bound on how many applications of $\text{alg}(f)$ succeed. Ignoring for a moment the fact that the emptier can choose to neglect the anchor set, i.e. assuming that no applications of $\text{alg}(f)$ are emptier-wasted, the probability that fewer than $M \cdot (1-2p)$ applications of $\text{alg}(f)$ succeed is at most

$$e^{-2Mp^2}$$

by a Chernoff bound. The emptier is allowed to interfere, i.e. neglect the anchor set and do extra emptying in the non-anchor set, with at most N^2 of the applications of $\text{alg}(f)$, thus if we condition on there being at least $M(1-2p)$ applications that would succeed if the emptier does not interfere, there are at least $M(1-2p)-N^2$ applications of $\text{alg}(f)$ that succeed. Let $\mathcal{W}$ be the event that the donation-process succeeds, i.e. the final application of $\text{alg}(f)$ is not emptier-wasted and succeeds, and let $\mathcal{D}$ be the even that the event that at least $M(1-2p)$ of the M applications of $\text{alg}(f)$ would succeed without interference by the emptier. Let $1-q = \Pr[\mathcal{W}]$. Obviously

$$\Pr[\mathcal{W}] \geq \Pr[\mathcal{W} \wedge \mathcal{D}] = \Pr[\mathcal{D}] \cdot \Pr[\mathcal{W}|\mathcal{D}].$$

Because the filler chooses which application of $\text{alg}(f)$ is the final application uniformly at random from $[M]$ we thus have

$$1 - q \geq (1 - e^{-2Mp^2}) \left(\frac{M \cdot (1-2p) - N^2}{M} \right).$$

Rearranging, and over-estimating (i.e. dropping unnecessary terms to simplify the expression, while maintaining the truth of the expression), we have

$$q \leq e^{-2Mp^2} + 2p + \frac{N^2}{M}.$$

By assumption $p \geq 2^{-\lg^8 N}$, so $Mp^2 \geq 2^{\lg^8 N}$, and we have the bound

$$q \leq 2p + 2^{-2 \cdot 2^{\lg^8 N}} + \frac{N^2}{2^{\lg^{24} N}}.$$

We choose to loosen this to

$$q \leq 2p + 2^{-\lg^8 N}.$$

Taking a union bound we have that with failure probability at most $q \cdot n_A$ all donation-process successfully achieve a cup with fill at least $\mu_{S_0}(B) + f(n_B)$ where $\mu_{S_0}(B)$ refers to the average fill of B measured at the start of the application of $\text{alg}(f)$; now we assume all donation-processes are successful, and demonstrate that this translates into the desired average fill in A.

Let $\mathbf{skips}_t$ denote the number of times that the emptier has skipped the anchor set by round t. Consider how $\mu(B) - \text{skips}/n_B$ changes over the course of the donation processes. As noted above, at the end of each donation-process $\mu(B)$ decreases due to B donating a cup with fill at least $\mu(B) + f(n_B)$. In particular, if S denotes the cup state immediately before a cup is donated on the i-th donation-process, B_0 denotes the set B before the donation and B_1 denotes the set B after the donation, then $\mu_S(B_1) = \mu_S(B_0) - f(n_B)/(n-i)$. Now we claim that $t \mapsto \mu_{S_t}(B) - \text{skips}_t / n_B$ is monotonically decreasing. Clearly donation decreases $\mu(B) - \text{skips}/n_B$. If the anchor set is neglected then $\mu(B)$ decreases, causing $\mu(B) - \text{skips}/n_B$ to decrease. If a skip occurs, then skips/n_B increases by more than $\mu(B)$ increases, causing $\mu(B) - \text{skips}/n_B$ to decrease. Let t_* be the cup state at the end of all the donation-processes. We have that

$$\mu_{S_{t_*}}(B) - \frac{\text{skips}_{t_*}}{n_B} \leq \mu_0 - \sum_{i=1}^{n_A} \frac{f(n_B)}{n-i}. \tag{17}$$

By conservation of mass we have

$$n_A \cdot \mu_{S_{t_*}}(A) + n_B \cdot \mu_{S_{t_*}}(B) = n\mu_0 + \text{skips}_{t_*}.$$

Rearranging,

$$\mu_{S_{t_*}}(A) = \mu_0 + \frac{n_B}{n_A} \left(\mu_0 + \frac{\text{skips}_{t_*}}{n_B} - \mu_{S_{t_*}}(B) \right). \tag{18}$$

Now we obtain a simpler form of Inequality (17). Let H_n denote the n-th harmonic number. We desire a simpler lower bound for

$$\sum_{i=1}^{n_A} \frac{1}{n-i} = H_{n-1} - H_{n_B - 1}.$$

We use the well known fact that

$$\frac{1}{2(n+1)} < H_n - \ln n - \gamma < \frac{1}{2n} \tag{19}$$

where $\gamma = \Theta(1)$ denotes the Euler-Mascheroni constant. Of course $H_{n-1} - H_{n_B-1} \geq H_n - H_{n_B}$. Now using Inequality (19) we have

$$\begin{aligned} H_n - H_{n_B} &> \left(\ln n + \gamma + \frac{1}{2(n+1)}\right) - \left(\ln n_B + \gamma + \frac{1}{2n_B}\right) \\ &> \ln \frac{1}{1-\delta} + \frac{1}{2}\left(\frac{n_B - n - 1}{(n+1)n_B}\right) \\ &> \delta - \Theta\left(\frac{\delta}{(1-\delta)n}\right). \end{aligned}$$

Now using this lower bound on $H_n - H_{n_B}$ in Inequality (18) we have:

$$\begin{aligned} \mu_{t_*}(A) &> \mu_0 + \frac{n_B}{n_A}\left(\delta - \Theta\left(\frac{\delta}{(1-\delta)n}\right)\right) f(n_B) \\ &= \mu_0 + \frac{\lfloor (1-\delta)n \rfloor}{\lceil \delta n \rceil}\left(\delta - \Theta\left(\frac{\delta}{(1-\delta)n}\right)\right) f(n_B) \\ &> \mu_0 + \left(\frac{1-\delta}{\delta} - \frac{1}{\delta^2 n}\right)\left(\delta - \Theta\left(\frac{\delta}{(1-\delta)n}\right)\right) f(n_B) \\ &> \mu_0 + \left((1-\delta) - \Theta(1/(\delta n))\right) f(n_B). \end{aligned}$$

Thus, by choosing $n > \Omega(1/\delta^2)$ we have

$$\mu_{t_*}(A) > \mu_0 + (1-\delta)^2 f(n_B).$$

We have shown that in Step 1 the filler achieves average fill $\mu_0 + (1-\delta)f(n_B)$ in A with failure probability at most $q \cdot n_A$. Now the filler flattens A and uses $\text{alg}(f)$ on A. It is clear that this is possible, and succeeds with probability at least p. This gets a cup with fill

$$\mu_0 + (1-\delta)^2 f(n_B) + f(n_A)$$

in A, as desired.

Taking a union bound over the probabilities of Step 1 and Step 2 succeeding gives that the entire procedure fails with probability at most

$$p' \leq p + q \cdot n_A \leq np + 2^{-\lg^8 N}.$$

The running time of Step 1 is clearly $M \cdot n \cdot T(\lfloor (1-\delta)n \rfloor)$ and the running time of Step 2 is clearly $T(\lceil \delta n \rceil)$; summing these yields the desired upper bound on running time.

□

Finally we prove that an oblivious filler can achieve backlog $N^{1-\varepsilon}$, just like an adaptive filler despite the oblivious filler's disadvantage. The proof is very similar to the proof of Theorem 5, but more complicated because in the oblivious case we must guarantee that the result holds with good probability, and also more complicated because the Oblivious Amplification Lemma is more complicated than the Adaptive Amplification Lemma.

Theorem 7. *There is an oblivious filling strategy for the variable-processor cup game on N cups that achieves backlog at least $\Omega(N^{1-\varepsilon})$ for any constant $\varepsilon > 0$ in running time $2^{polylog(n)}$ with probability at least $1 - 2^{-\,polylog(n)}$ against a Δ-greedy-like emptier with $\Delta \leq O(1)$.*

Proof. We aim to achieve backlog $(N/n_b)^{1-\varepsilon} - 1$ for some $n_b \le \text{polylog}(N)$ on N cups. Let δ be a constant, chosen as a function of ϵ.

By Proposition 6 there is an oblivious filling strategy that achieves backlog $\Omega(1)$ on n cups with exponentially good probability in n; we call this algorithm alg f_0. However, unlike in the proof of Theorem 5, we obviously cannot use the base case with a constant number of cups: doing so would completely destroy our probability of success. Because the running time of our algorithm will be $2^{\text{polylog}(N)}$, we will be required to take a union bound over $2^{\text{polylog}(N)}$ events. By making the size of our base case $n_b = \log^8(N)$ we get that the probability of the algorithm failing in the base case is at most $2^{-\log^8(N)}$. By Proposition 6 alg f_0 achieves backlog $f_0(k) \ge H \ge \Omega(1)$ for all $k \ge n_b$, for some constant $H \ge \Omega(1)$ to be determined (H is a function of δ).

We construct f_{i+1} as the amplification of f_i using Lemma 8.

Define a sequence g_i as

$$g_i = \begin{cases} n_b \lceil 16/\delta \rceil, & i = 0 \\ \lfloor g_{i-1}/(1-\delta) \rfloor, & i \ge 1 \end{cases}.$$

We claim the following regarding our construction:

Claim 16.

$$f_i(k) \ge (k/n_b)^{1-\varepsilon} - 1 \textit{ for all } k \le g_i. \tag{20}$$

Proof. We prove Claim 16 by induction on i.

When $i = 0$, the base case of our induction, (20) is trivially true as $(k/n_b)^{1-\epsilon} - 1 \le H$ by definition of H for $k \le g_0$.

Assume (20) for f_i, consider f_{i+1}.

Note that, by design of g_i, if $k \le g_{i+1}$ then $\lfloor k \cdot (1-\delta) \rfloor \le g_i$. Consider any $k \in [g_{i+1}]$.

First we deal with the trivial case where $k \le g_0$. In this case

$$f_{i+1}(k) \ge f_i(k) \ge \cdots \ge f_0(k) \ge (k/n_b)^{1-\varepsilon} - 1.$$

Now we consider $k \ge g_0$. Note that in this case $\lfloor (1-\delta)k \rfloor \ge n_b$. Since f_{i+1} is the amplification of f_i, and k is sufficiently large, we have by Lemma 8 that

$$f_{i+1}(k) \ge (1-\delta)^2 f_i(\lfloor (1-\delta)k \rfloor) + f_i(\lceil \delta k \rceil).$$

By our inductive hypothesis, which applies as $\lceil \delta k \rceil \le g_i$, $\lfloor k \cdot (1-\delta) \rfloor \le g_i$, we have

$$f_{i+1}(k) \ge (1-\delta)^2 (\lfloor (1-\delta)k/n_b \rfloor^{1-\varepsilon} - 1) + \lceil \delta k/n_b \rceil^{1-\varepsilon} - 1.$$

Dropping the floor and ceiling, incurring a -1 for dropping the floor, we have

$$f_{i+1}(k) \ge (1-\delta)^2 (((1-\delta)k/n_b - 1)^{1-\varepsilon} - 1) + (\delta k/n_b)^{1-\varepsilon} - 1.$$

Because $(x-1)^{1-\varepsilon} \ge x^{1-\varepsilon} - 1$, due to the fact that $x \mapsto x^{1-\varepsilon}$ is a sub-linear sub-additive function, we have

$$f_{i+1}(k) \ge (1-\delta)^2 (((1-\delta)k/n_b)^{1-\varepsilon} - 2) + (\delta k/n_b)^{1-\varepsilon} - 1.$$

Moving the $(k/n_b)^{1-\varepsilon}$ to the front we have

$$f_{i+1}(k) \ge (k/n_b)^{1-\varepsilon} \cdot \left((1-\delta)^{3-\varepsilon} + \delta^{1-\varepsilon} - \frac{2(1-\delta)^2}{(k/n_b)^{1-\varepsilon}} \right) - 1.$$

Because $(1-\delta)^{3-\varepsilon} \ge 1 - (3-\varepsilon)\delta$, a fact called Bernoulli's Identity, we have

$$f_{i+1}(k) \ge (k/n_b)^{1-\varepsilon} \cdot \left(1 - (3-\varepsilon)\delta + \delta^{1-\varepsilon} - \frac{2(1-\delta)^2}{(k/n_b)^{1-\varepsilon}} \right) - 1.$$

Of course $-2(1-\delta)^2 > -2$, so

$$f_{i+1}(k) \ge (k/n_b)^{1-\varepsilon} \cdot \left(1 - (3-\varepsilon)\delta + \delta^{1-\varepsilon} - 2/(k/n_b)^{1-\varepsilon} \right) - 1.$$

Because

$$\frac{-2}{(k/n_b)^{1-\varepsilon}} \ge \frac{-2}{(g_0/n_b)^{1-\varepsilon}} \ge -2(\delta/16)^{1-\varepsilon} \ge -\delta^{1-\varepsilon}/2,$$

which follows from our choice of $g_0 = \lceil 8/\delta \rceil n_b$ and the restriction $\varepsilon < 1/2$, we have

$$f_{i+1}(k) \geq (k/n_b)^{1-\varepsilon} \cdot \left(1 - (3-\varepsilon)\delta + \delta^{1-\varepsilon} - \delta^{1-\varepsilon}/2\right) - 1.$$

Finally, combining terms we have

$$f_{i+1}(k) \geq (k/n_b)^{1-\varepsilon} \cdot \left(1 - (3-\varepsilon)\delta + \delta^{1-\varepsilon}/2\right) - 1.$$

Because $\delta^{1-\varepsilon}$ dominates δ for sufficiently small δ, there is a choice of $\delta = \Theta(1)$ such that

$$1 - (3-\varepsilon)\delta + \delta^{1-\varepsilon}/2 \geq 1.$$

Taking δ to be this small we have,

$$f_{i+1}(k) \geq (k/n_b)^{1-\varepsilon} - 1,$$

completing the proof. □

The sequence g_i is n_b times the sequence g_i from the proof of Theorem 5; we thus have that $g_{i_*} \geq N$ for some $i_* \leq O(\log N)$. Hence alg f_{i_*} achieves backlog

$$f_{i_*}(N) \geq (N/n_b)^{1-\varepsilon} - 1.$$

Let $\varepsilon' = 2\varepsilon$. Of course $\Omega(N^{\varepsilon}) \geq \text{polylog}(N)$, so

$$(N/n_b)^{1-\varepsilon} - 1 \geq \Omega(N^{1-\varepsilon'}).$$

Let the running time of $f_i(N)$ be $T_i(N)$. From the Amplification Lemma we have following recurrence bounding $T_i(N)$:

$$\begin{aligned} T_i(n) &\leq 2^{\text{polylog}(N)} \cdot T_{i-1}(\lfloor (1-\delta)n \rfloor) + T_{i-1}(\lceil \delta n \rceil) \\ &\leq 2^{\text{polylog}(N)} T_{i-1}(\lfloor (1-\delta)n \rfloor). \end{aligned}$$

It follows that alg f_{i_*}, recalling that $i_* \leq O(\log N)$, has running time

$$T_{i_*}(n) \leq (2^{\text{polylog}(N)})^{O(\log N)} \leq 2^{\text{polylog}(N)}$$

as desired.

Now we analyze the probability that the construction fails. Consider the recurrence $a_{i+1} = \alpha a_i + \beta, a_0 = \gamma$; the recurrence bounding failure probability is a special case of this. Expanding, we see that the recurrence solves to $a_k = \Theta(\alpha^{k-1})\beta + a^k \gamma$. In our case we have

$$\alpha \leq N, \beta = 2^{-\lg^8 N}, \gamma = 2^{-\lg^8 N}.$$

Hence the recurrence solves to

$$p_{i_*} \leq 2^{-\text{polylog}(N)},$$

as desired.

□

7 Conclusion

We asked a natural question to extend understanding of the cup game: what if the resources of the players are variable? We found several shocking results, which combined demonstrate that having variable resources makes the cup game fundamentally changes the cup game.

More work remains to be done on the variable-processor cup game. Extending our oblivious lower bound on backlog to apply to a broader class of emptiers, rather than just greedy-like emptiers, is an interesting open question. Analyzing versions of the cup game with variable resources and resources augmentation is also an interesting open problem.

Our work suggests that looking at other scheduling problems in the context of variable resources is, surprisingly, a very interesting question.

References

[1] Michael A Bender, Martín Farach-Colton, and William Kuszmaul. Achieving optimal backlog in multi-processor cup games. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1148–1157, 2019.

[2] Michael A Bender, Sándor P Fekete, Alexander Kröller, Vincenzo Liberatore, Joseph SB Mitchell, Valentin Polishchuk, and Jukka Suomela. The minimum backlog problem. *Theoretical Computer Science*, 605:51–61, 2015.

[3] Paul Dietz and Rajeev Raman. Persistence, amortization and randomization. 1991.

[4] William Kuszmaul. Achieving optimal backlog in the vanilla multi-processor cup game. *SIAM*, 2020.