The Variable Processor Cup Game

Alek Westover

Belmont High School

June 7, 2020

What is the Cup Game?

Definition

p-processor cup-game on n cups:
multi-round game
on each round:

- The ***filler*** distributes p units of water among the cups (with at most 1 unit to any particular cup)
- Then the ***emptier*** chooses p cups to remove (at most) one unit of water from

Note:
Emptier must allocate resources discretely
Filler can allocate resources continuously

What is the Cup Game?

Definition

Backlog:
Amount of water in the fullest cup

Emptier tries to *minimize* backlog
Filler tries to *maximize* backlog

Why is the Cup Game Important?

Models ***work scheduling***:

- Cups represent tasks
- At each time step:
  - new work comes in, distributed arbitrarily among the tasks
  - p processors are allocated to work on a subset of the tasks

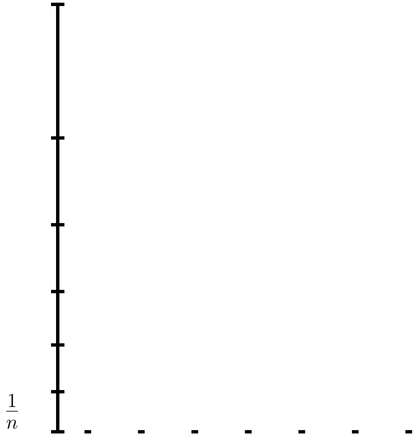
Also an interesting mathematical object

Analysis of the cup game

Prove ***upper bounds*** and ***lower bounds*** on backlog.

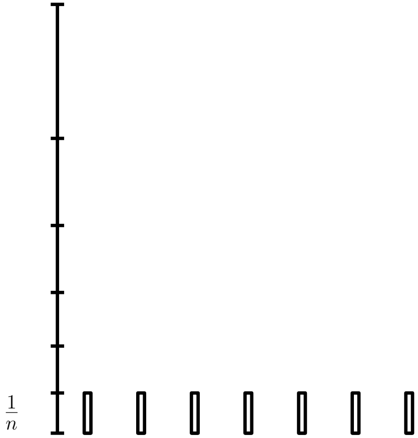
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



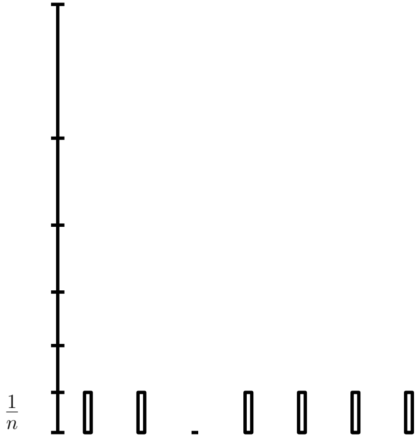
SINGLE-PROCESSOR LOWER BOUND

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



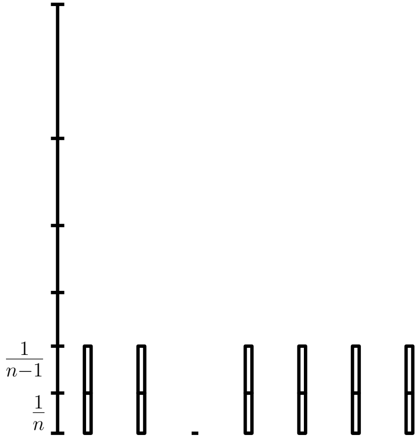
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



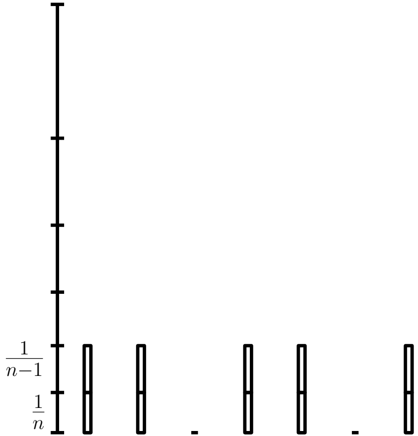
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



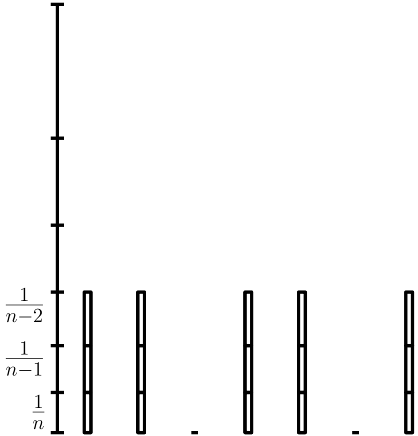
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



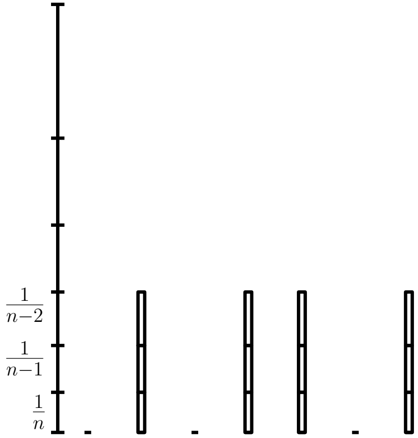
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



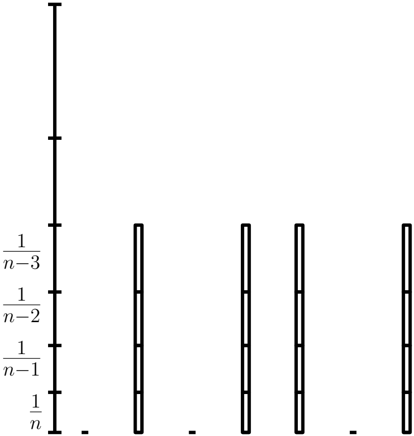
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



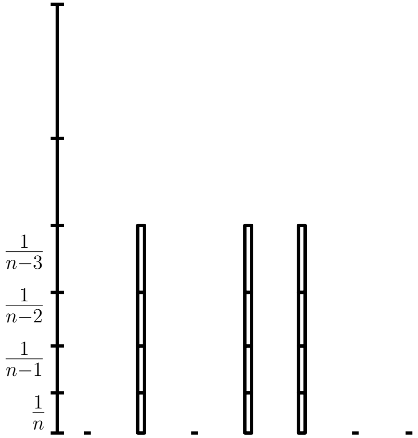
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



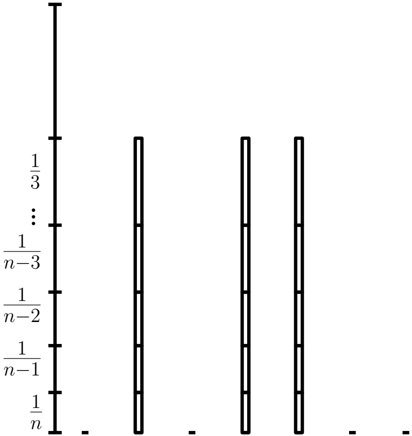
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



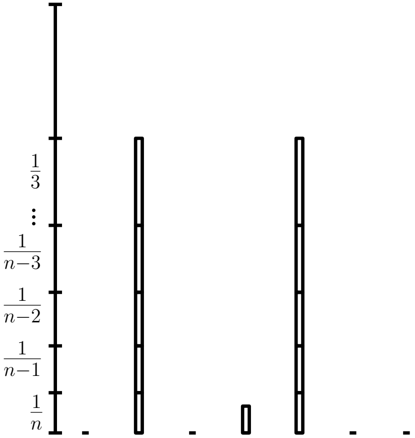
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



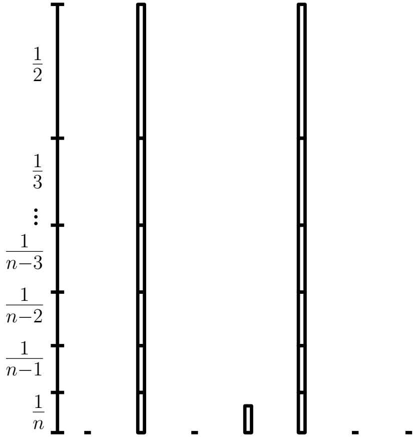
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



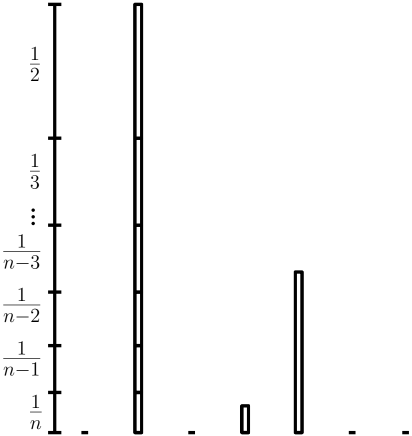
Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



SINGLE-PROCESSOR LOWER BOUND

Filling strategy:
Distribute water equally amongst cups not yet emptied from.



Single-Processor Lower Bound

Filling strategy:
Distribute water equally amongst cups not yet emptied from.

Achieves backlog:

$$\frac{1}{n} + \frac{1}{n-1} + \cdots + \frac{1}{2} = \Omega(\log n).$$

Single-Processor Upper Bound

A ***greedy emptier*** – an emptier that always empties from the fullest cup – never lets backlog exceed $O(\log n)$.

Definitions

- S_t: state at start of round t
- I_t: state after the filler adds water on round t, but before the emptier removes water
- $\mu_k(S)$: average fill of k fullest cups at state S.

Single-Processor Upper Bound Proof

Proof: Inductively prove a set of invariants:

$$\mu_k(S_t) \leq \frac{1}{k+1} + \ldots + \frac{1}{n}.$$

Let a be the cup that the emptier empties from on round t

If a is one of the k fullest cups in S_{t+1}:

$$\mu_k(S_{t+1}) \leq \mu_k(S_t).$$

Otherwise:

$$\mu_k(S_{t+1}) = \mu_{k+1}(I_t) \leq \mu_{k+1}(S_{t+1}) + \frac{1}{k+1}.$$

Our Variant of the Cup Game

We analyzed a new variant of the classic multi-processor cup game, the ***variable-processor cup game***, in which *the resources are variable*: the filler is allowed to change p.

Specifically we found upper bounds and lower bounds on backlog with oblivious and adaptive fillers.

The modification to allow variable resources may seem small. However, we show that it drastically changes the game.

Adaptive Filler
Lower Bound

Negative Fill

In lower bound proofs we allow ***negative fill***

- We measure fill relative to average fill
- Important for recursion
- Game is strictly easier for the filler if cups can zero out

Amplification Lemma

Lemma

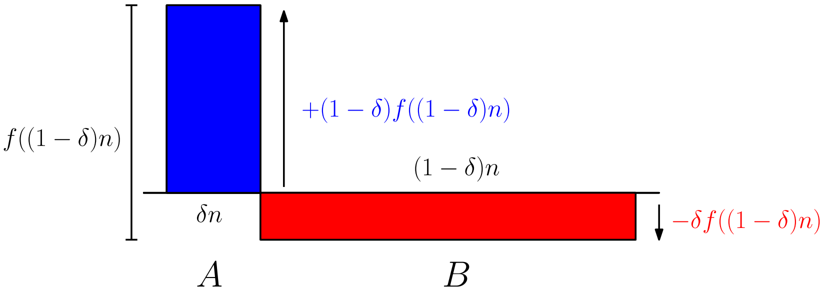
Given a strategy f, we can construct a new strategy that achieves backlog

$$f'(n) \geq (1-\delta) \sum_{\ell=0}^{L} f((1-\delta)\delta^{\ell} n)$$

for appropriate parameters $L \in \mathbb{N}, 0 < \delta \ll 1/2$.

Amplification Lemma Proof Sketch

- A starts as the δn fullest cups, B as the $(1-\delta)n$ other cups.
- Repeatedly apply f to B and swap the cup with fill increased by $f((1-\delta)n)$ into A.
- Recurse on A, filler will decrease p.

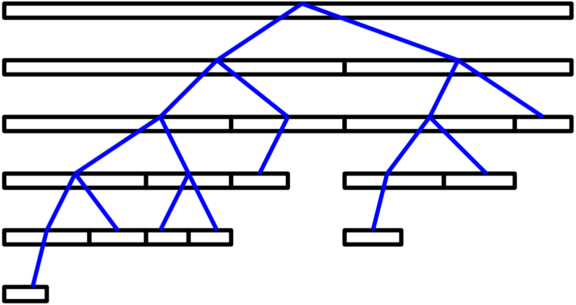


Adaptive Filler Lower Bound

By repeated amplification, using $\delta = \Theta(1)$, we get:

Theorem

There is an adaptive filling strategy that achieves backlog $\Omega(n^{1-\epsilon})$ for constant $\epsilon > 0$ of our choice in running-time $2^{O(\log^2 n)}$.

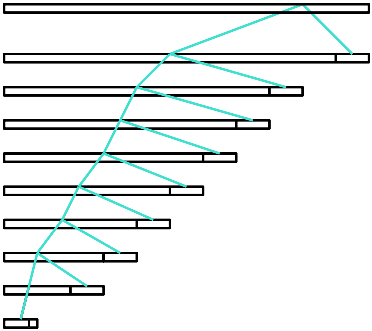


Adaptive Filler Lower Bound

By repeated amplification, using $\delta = \Theta(1/n)$, we get:

Theorem

There is an adaptive filling strategy that achieves backlog $\Omega(n)$ in running-time $2^{O(n)}$.



Upper Bound

Upper Bound

We prove a novel set of invariants:

Theorem

A greedy emptier maintains the invariant:

$$\mu_k(S_t) \le 2n - k.$$

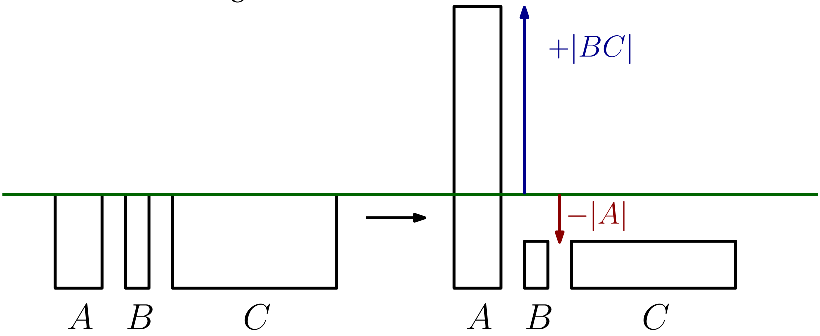
In particular this implies that backlog is

$$O(n).$$

Note: this matches our lower bound!

Upper Bound Proof Sketch

Extremal fill configuration:



Oblivious Filler Lower Bound

Oblivious Filler Lower Bound

Classically emptier does much better in the randomized setting.

But not in the variable-processor cup game!

Oblivious Filler: Constant Fill

Getting constant fill in a *known* cup is hard now. Strategy:

- Play lots of single-processor cup games on constant numbers of cups blindly. Each succeeds with some constant probability.
- By a Chernoff Bound, with *exponentially* good probability, i.e. $1 - 2^{-\Omega(n)}$, at least a constant fraction, say nc, of the single-processor cup games succeed.
- Set $p = nc$.
- Exploiting the greedy-like nature of the emptier fill a set of nc known cups while the emptier is forced to focus on the set of nc with high fill.
- Recurse for a constant number of levels on the nc cups with known high fill.

Oblivious Amplification Lemma

Almost identical to the Adaptive Amplification Lemma!

Lemma

Given a strategy f, we can construct a new strategy that achieves backlog

$$f'(n) \geq \phi \cdot (1-\delta) \sum_{\ell=0}^{L} f((1-\delta)\delta^{\ell} n)$$

for appropriate parameters $L \in \mathbb{N}, 0 < \delta \ll 1/2$ and constant $\phi \in (0,1)$ of our choice.

(Note: Lemma is actually more complicated than this.)

Oblivious Filler Lower Bound

Theorem

There is an oblivious filling strategy that achieves backlog

$$\Omega(n^{1-\epsilon})$$

for constant $\epsilon > 0$ *with probability at least* $1 - 2^{-\operatorname{polylog}(n)}$ *in running time* $2^{O(\log^2 n)}$.

Proof notes:

- Similar to adaptive filler proof
- need larger base case for union bound to work; this doesn't harm backlog though

OPEN QUESTIONS

- Can we extend the Oblivious lower-bound construction to work with arbitrary emptiers?
- Are there shorter more simple constructions?

Acknowledgements

- My mentor, William Kuszmaul!
- MIT PRIMES
- My Parents