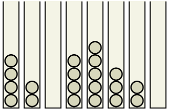
# Balanced Allocations: The Power of Choice versus Noise



Thomas Sauerwald

Department of Computer Science and Technology, University of Cambridge

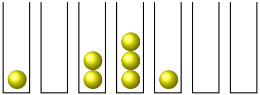
20 June 2024

(based on joint work with Dimitris Los and John Sylvester)

# Background

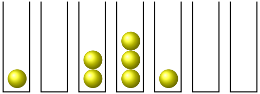
# Balanced Allocations a.k.a. Balls-into-Bins

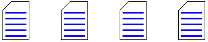
Balls $\to$ Bins



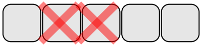
# Balanced Allocations a.k.a. Balls-into-Bins

Balls $\rightarrow$ Bins



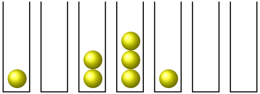


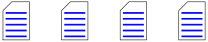
Items $\rightarrow$ Buckets



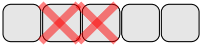
# Balanced Allocations a.k.a. Balls-into-Bins

Balls → Bins



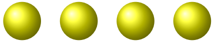


Items → Buckets

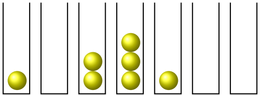


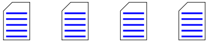
Balls → Baskets

# Balanced Allocations a.k.a. Balls-into-Bins

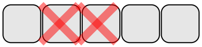


Balls $\rightarrow$ Bins



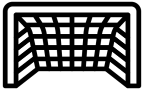


Items $\rightarrow$ Buckets



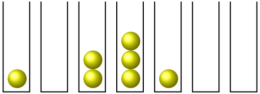
Balls $\rightarrow$ Baskets

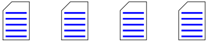
Football $\rightarrow$ Goal



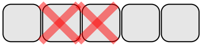
# Balanced Allocations a.k.a. Balls-into-Bins

Balls $\rightarrow$ Bins



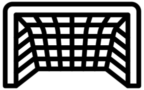


Items $\rightarrow$ Buckets



Balls $\rightarrow$ Baskets

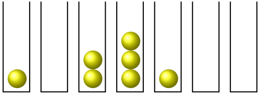
Football $\rightarrow$ Goal

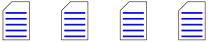


...

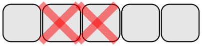
# Balanced Allocations a.k.a. Balls-into-Bins

Balls → Bins



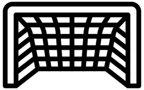


Items → Buckets



Balls → Baskets

Football → Goal



. . .

# Notation

Allocate $m$ tasks (balls) into $n$ machines (bins).

# Notation

Allocate $m$ tasks (balls) into $n$ machines (bins).

**Goal:** minimise the **maximum load** $\max_{i \in [n]} x_i^m$, where $x^t$ is the load vector after ball $t$.

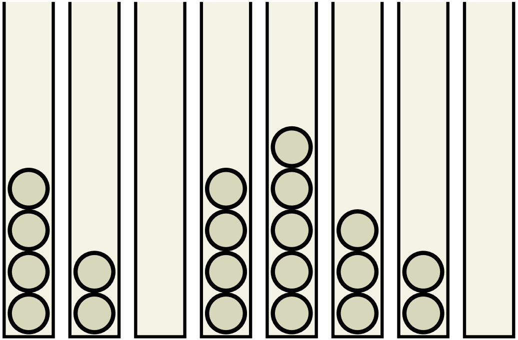
$\Leftrightarrow$ minimise the **gap**, where $\text{Gap}(m) = \max_{i \in [n]}(x_i^m - m/n)$.

# Notation

Allocate $m$ tasks (balls) into $n$ machines (bins).

**Goal:** minimise the **maximum load** $\max_{i \in [n]} x_i^m$, where $x^t$ is the load vector after ball $t$.

$\Leftrightarrow$ minimise the **gap**, where $\text{Gap}(m) = \max_{i \in [n]}(x_i^m - m/n)$.

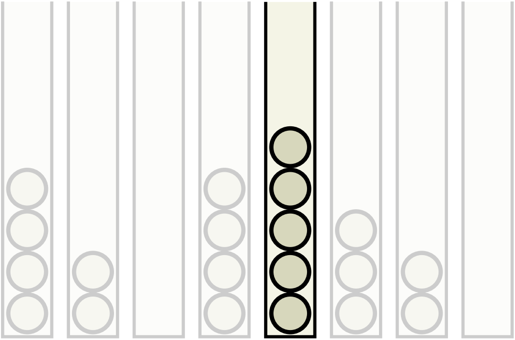


# Notation

Allocate $m$ tasks (balls) into $n$ machines (bins).

**Goal:** minimise the **maximum load** $\max_{i \in [n]} x_i^m$, where $x^t$ is the load vector after ball $t$.

$\Leftrightarrow$ minimise the **gap**, where $\text{Gap}(m) = \max_{i \in [n]}(x_i^m - m/n)$.

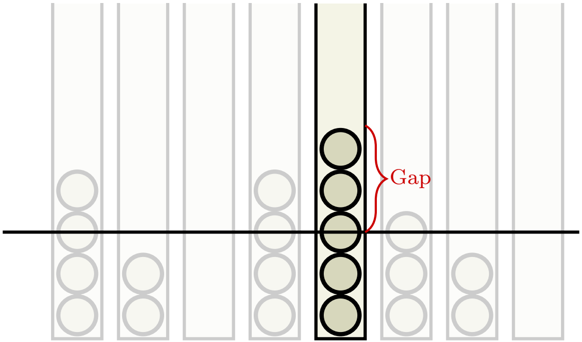


# Notation

Allocate $m$ tasks (balls) into $n$ machines (bins).

**Goal:** minimise the **maximum load** $\max_{i \in [n]} x_i^m$, where $x^t$ is the load vector after ball $t$.

$\Leftrightarrow$ minimise the **gap**, where $\text{Gap}(m) = \max_{i \in [n]}(x_i^m - m/n)$.



# ONE-CHOICE versus TWO-CHOICE

ONE-CHOICE:
**Iteration**: For each ball $1, 2, \ldots, m$ sample **one** bin uniformly at random (u.a.r.) and allocate ball there.

# One-Choice versus Two-Choice

**One-Choice:**
Iteration: For each ball $1, 2, \ldots, m$ sample **one** bin uniformly at random (u.a.r.) and allocate ball there.

- For $m \gg n$, w.h.p. $\mathrm{Gap}(m) = \Theta\left(\sqrt{\frac{m}{n} \cdot \log n}\right)$ (e.g. [RS98])

# One-Choice versus Two-Choice

**One-Choice:**
Iteration: For each ball $1, 2, \ldots, m$ sample **one** bin uniformly at random (u.a.r.) and allocate ball there.

- For $m \gg n$, w.h.p. $\text{Gap}(m) = \Theta\left(\sqrt{\frac{m}{n} \cdot \log n}\right)$ (e.g. [RS98])

Meaning with probability at least $1 - n^{-c}$ for constant $c > 0$.

# One-Choice versus Two-Choice

**One-Choice:**
Iteration: For each ball $1, 2, \ldots, m$ sample **one** bin uniformly at random (u.a.r.) and allocate ball there.

- For $m \gg n$, w.h.p. $\text{Gap}(m) = \Theta\left(\sqrt{\frac{m}{n} \cdot \log n}\right)$ (e.g. [RS98])

**Two-Choice:**
Iteration: For each ball $1, 2, \ldots, m$ sample **two** bins independently u.a.r. and allocate ball to the least loaded of the two.

# One-Choice versus Two-Choice

**One-Choice:**
Iteration: For each ball $1, 2, \ldots, m$ sample **one** bin uniformly at random (u.a.r.) and allocate ball there.

- For $m \gg n$, w.h.p. $\text{Gap}(m) = \Theta\left(\sqrt{\frac{m}{n} \cdot \log n}\right)$ (e.g. [RS98])

**Two-Choice:**
Iteration: For each ball $1, 2, \ldots, m$ sample **two** bins independently u.a.r. and allocate ball to the least loaded of the two.

- For any $m \geq n$, w.h.p. $\text{Gap}(m) = \log_2 \log n + \Theta(1)$ [BCSV06].

# ONE-CHOICE versus TWO-CHOICE

**ONE-CHOICE:**
Iteration: For each ball $1, 2, \ldots, m$ sample **one** bin uniformly at random (u.a.r.) and allocate ball there.

- For $m \gg n$, w.h.p. $\text{Gap}(m) = \Theta\left(\sqrt{\frac{m}{n} \cdot \log n}\right)$ (e.g. [RS98])

**TWO-CHOICE:**
Iteration: For each ball $1, 2, \ldots, m$ sample **two** bins independently u.a.r. and allocate ball to the least loaded of the two.

- For any $m \geq n$, w.h.p. $\text{Gap}(m) = \log_2 \log n + \Theta(1)$ [BCSV06].

Caen Hill Locks (main flight consists of **16** locks)

Source: Wikipedia

# Proof Idea: Layered Induction

# Proof Idea: Layered Induction

- Let $m = n$

# Proof Idea: Layered Induction

- Let $m = n$
- Let $b_k$ be the fraction of bins with load at least $k$ after $n$ balls are allocated.

# Proof Idea: Layered Induction

- Let $m = n$
- Let $b_k$ be the fraction of bins with load at least $k$ after $n$ balls are allocated. Then:

$$b_{k+1} \leq (b_k)^2$$

# Proof Idea: Layered Induction

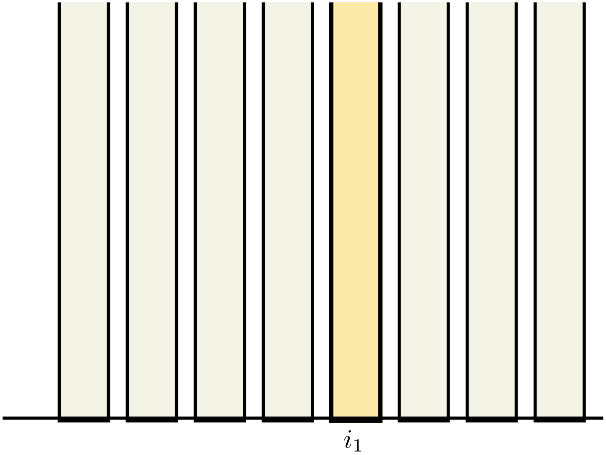
- Let $m = n$
- Let $b_k$ be the fraction of bins with load at least $k$ after $n$ balls are allocated. Then:

$$b_{k+1} \leq (b_k)^2$$

- $b_2 \leq 1/2 \rightsquigarrow b_{\log_2 \log_2 n+3} \leq n^{-2}$

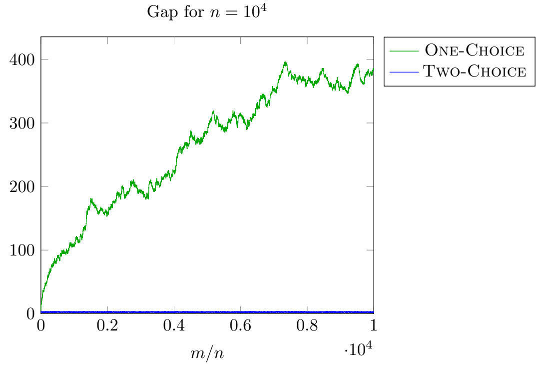
**This does not work in the heavily loaded case $m \gg n$!**

# Two-Choice: Visualization

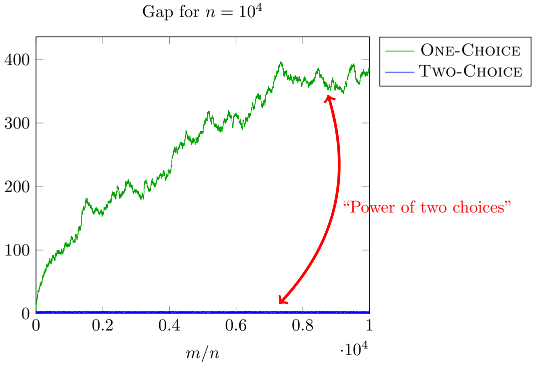


# Experiments

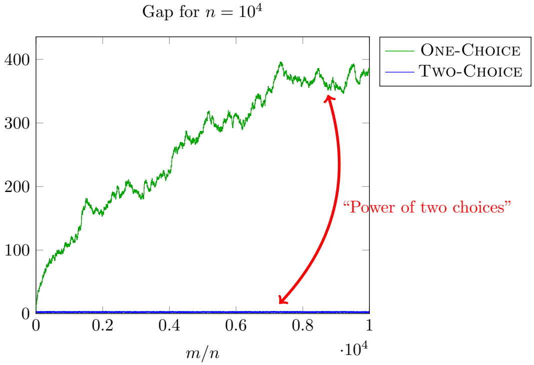
# Experiments



# Experiments



# Experiments




Distribution of Gap($m$), $m = 10^8$, $n = 10^4$ over 100 runs:

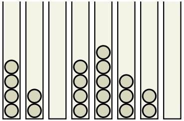
- One-Choice: gap values ranging from 328 to 520
- Two-Choice: 34 runs with gap 2; 66 runs with gap 3

# ACM Paris Kanellakis Theory and Practice Award 2020


*For "the discovery and analysis of balanced allocations, known as the power of two choices, and their extensive applications to practice."*

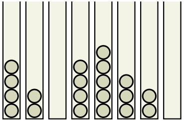
*"These include i-Google's web index, Akamai's overlay routing network, and highly reliable distributed data storage systems used by Microsoft and Dropbox, which are all based on variants of the power of two choices paradigm. There are many other software systems that use balanced allocations as an important ingredient."*

# Relaxations of the Original Model



Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

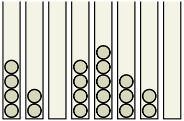
# Relaxations of the Original Model



Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins?

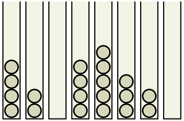
# Relaxations of the Original Model



Balls arrive sequentially and sample two bins independently and uniformly at random.
After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-Choice

# Relaxations of the Original Model

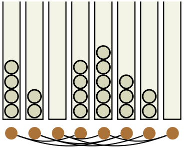


Balls arrive sequentially and sample two bins independently and uniformly at random.
After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-Choice

2. What if the two bin samples are correlated?

# Relaxations of the Original Model

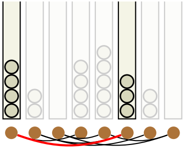


Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? ⇝ 1.5-Choice

2. What if the two bin samples are correlated? ⇝ Balanced Allocations on Graphs

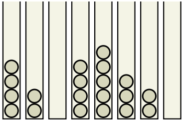
# Relaxations of the Original Model



Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? ⇝ 1.5-Choice
2. What if the two bin samples are correlated? ⇝ Balanced Allocations on Graphs

# Relaxations of the Original Model



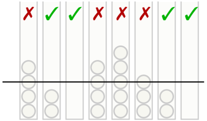
Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-Choice

2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried?

# Relaxations of the Original Model



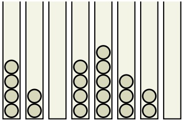
> Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? ⇝ 1.5-Choice

2. What if the two bin samples are correlated? ⇝ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? ⇝ Thinning

# Relaxations of the Original Model

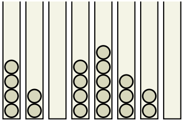


Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-Choice
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? $\rightsquigarrow$ Thinning

# Relaxations of the Original Model



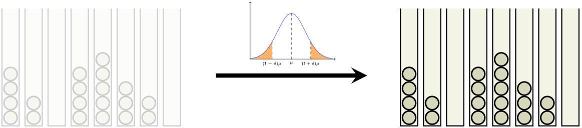
Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-Choice
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? $\rightsquigarrow$ Thinning

3b. What if the load information is outdated,

# Relaxations of the Original Model



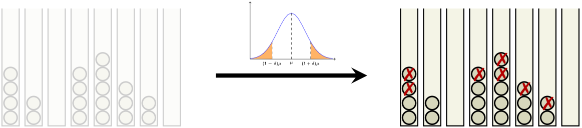
Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-Choice
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? $\rightsquigarrow$ Thinning

3b. What if the load information is outdated,

# Relaxations of the Original Model



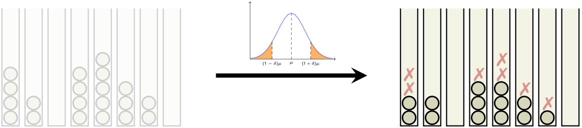

Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-Choice
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? $\rightsquigarrow$ Thinning

3b. What if the load information is outdated,

# Relaxations of the Original Model



Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

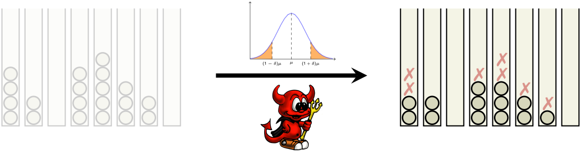
1. What if we are not (always) able to sample two bins? ⇝ 1.5-Choice

2. What if the two bin samples are correlated? ⇝ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? ⇝ Thinning

3b. What if the load information is outdated,

# Relaxations of the Original Model



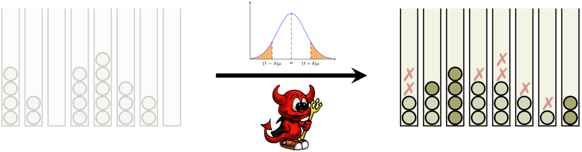
Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-Choice
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? $\rightsquigarrow$ Thinning

3b. What if the load information is outdated, or possibly manipulated by an adversary?

# Relaxations of the Original Model



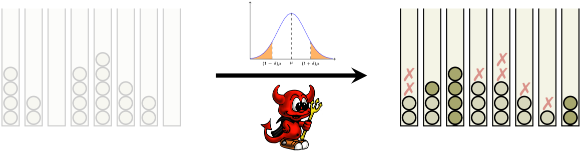

> Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? $\rightsquigarrow$ 1.5-CHOICE
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? $\rightsquigarrow$ THINNING

3b. What if the load information is outdated, or possibly manipulated by an adversary?

# Relaxations of the Original Model



Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

1. What if we are not (always) able to sample two bins? ⇝ 1.5-Choice
2. What if the two bin samples are correlated? ⇝ Balanced Allocations on Graphs

3a. What if the exact load information cannot be queried? ⇝ Thinning

3b. What if the load information is outdated, or possibly manipulated by an adversary? ⇝ Noise and Delay Models

# Between One-Choice and Two-Choice

- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$
- Two-Choice: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$

# Between One-Choice and Two-Choice

- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$
- Two-Choice: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- Three-Choice: slightly better than Two-Choice

# Between One-Choice and Two-Choice

- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n}\cdot\log n})$, especially when $m \gg n$

  $\vdots$ ??

- Two-Choice: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- Three-Choice: slightly better than Two-Choice

# Between One-Choice and Two-Choice

- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$

  $\vdots$ ??

- Two-Choice: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- Three-Choice: slightly better than Two-Choice

1.5-Choice (a.k.a. $(1+\beta)$-process):
Iteration: For each $t \geq 0$, with probability $1/2$ allocate one ball via the Two-Choice process, otherwise allocate one ball via the One-Choice process.

# Between ONE-CHOICE and TWO-CHOICE

- ONE-CHOICE: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$

  $\vdots$ ??
- TWO-CHOICE: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- THREE-CHOICE: slightly better than TWO-CHOICE

Of course 1/2 could be replaced by $\beta \in [0, 1]$

1.5-CHOICE (a.k.a. $(1 + \beta)$-process):
Iteration: For each $t \geq 0$, with probability $1/2$ allocate one ball via the TWO-CHOICE process, otherwise allocate one ball via the ONE-CHOICE process.

# Between One-Choice and Two-Choice

- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n}} \cdot \log n)$, especially when $m \gg n$

  $\vdots$ ??

- Two-Choice: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- Three-Choice: slightly better than Two-Choice

1.5-Choice (a.k.a. $(1+\beta)$-process):
Iteration: For each $t \geq 0$, with probability $1/2$ allocate one ball via the Two-Choice process, otherwise allocate one ball via the One-Choice process.

- The number of samples per ball is 1.5 (on average)

# Between One-Choice and Two-Choice

- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$

  $\vdots$ ??

- Two-Choice: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- Three-Choice: slightly better than Two-Choice

1.5-Choice (a.k.a. $(1+\beta)$-process):
Iteration: For each $t \geq 0$, with probability $1/2$ allocate one ball via the Two-Choice process, otherwise allocate one ball via the One-Choice process.

- The number of samples per ball is 1.5 (on average)
- It can be seen as making an erroneous comparison with probability $1/4$

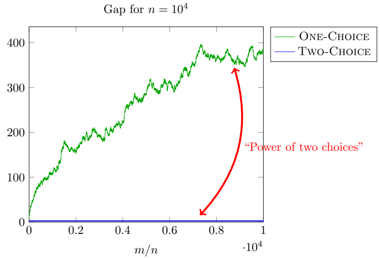
# Between One-Choice and Two-Choice

- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$

  $\vdots$ ??
- Two-Choice: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- Three-Choice: slightly better than Two-Choice

1.5-Choice (a.k.a. $(1+\beta)$-process):
Iteration: For each $t \geq 0$, with probability $1/2$ allocate one ball via the Two-Choice process, otherwise allocate one ball via the One-Choice process.

- The number of samples per ball is 1.5 (on average)
- It can be seen as making an erroneous comparison with probability $1/4$
- The gap is w.h.p. $\Theta(\log n)$ [PTW15]



# Between ONE-CHOICE and TWO-CHOICE

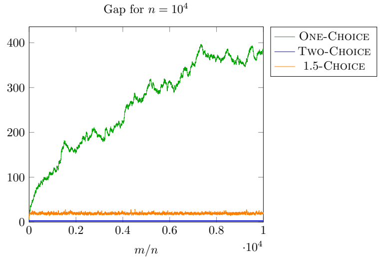
- ONE-CHOICE: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$

  $\vdots$ ??

- TWO-CHOICE: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- THREE-CHOICE: slightly better than TWO-CHOICE

1.5-CHOICE (a.k.a. $(1+\beta)$-process):
Iteration: For each $t \geq 0$, with probability $1/2$ allocate one ball via the TWO-CHOICE process, otherwise allocate one ball via the ONE-CHOICE process.

- The number of samples per ball is 1.5 (on average)
- It can be seen as making an erroneous comparison with probability $1/4$
- The gap is w.h.p. $\Theta(\log n)$ [PTW15]



# Between ONE-CHOICE and TWO-CHOICE

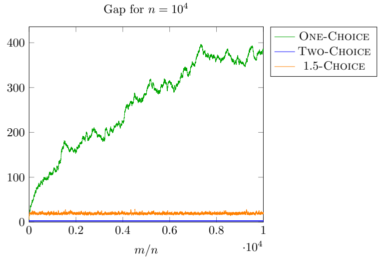
- ONE-CHOICE: large gap $\Theta(\sqrt{\frac{m}{n}} \cdot \log n)$, especially when $m \gg n$

  $\vdots$ ??

- TWO-CHOICE: very small gap $\log_2 \log n + \Theta(1)$ regardless of value of $m$
- THREE-CHOICE: slightly better than TWO-CHOICE

1.5-CHOICE (a.k.a. $(1+\beta)$-process):
Iteration: For each $t \geq 0$, with probability $1/2$ allocate one ball via the TWO-CHOICE process, otherwise allocate one ball via the ONE-CHOICE process.

- The number of samples per ball is 1.5 (on average)
- It can be seen as making an erroneous comparison with probability $1/4$
- The gap is w.h.p. $\Theta(\log n)$ [PTW15]



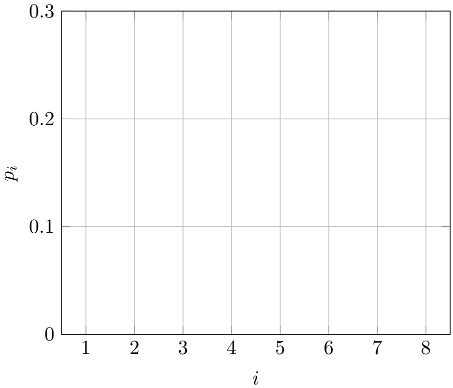
# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.

# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For One-Choice,

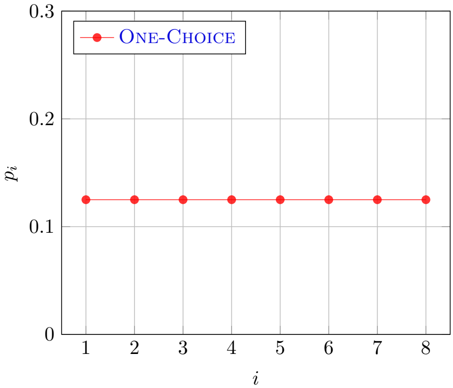
$$p_{\textsc{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$



# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For One-Choice,

$$p_{\text{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$



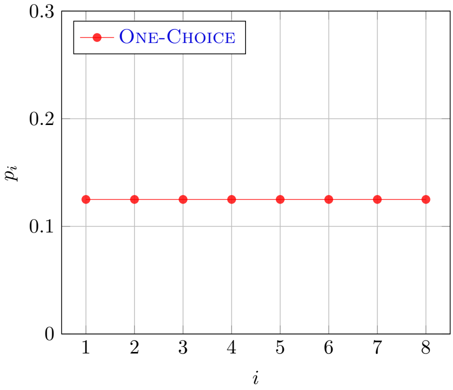
# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For One-Choice,

$$p_{\text{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$

- For Two-Choice,

$$p_{\text{Two-Choice}} = \left(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\right)$$



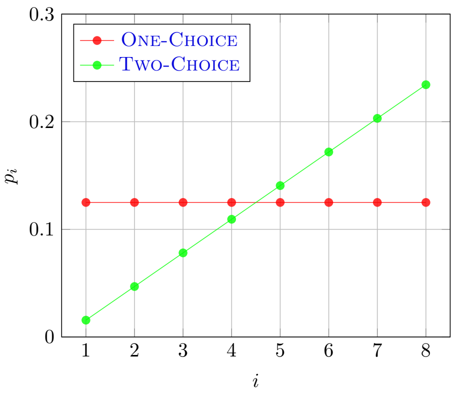
# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For ONE-CHOICE,

$$p_{\text{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$

- For TWO-CHOICE,

$$p_{\text{Two-Choice}} = \left(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\right)$$



# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For One-Choice,

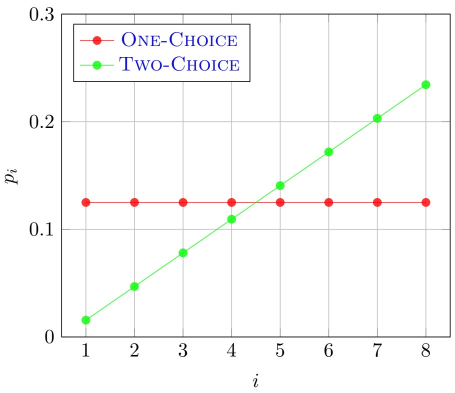
$$p_{\textsc{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$

- For Two-Choice,

$$p_{\textsc{Two-Choice}} = \left(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\right)$$

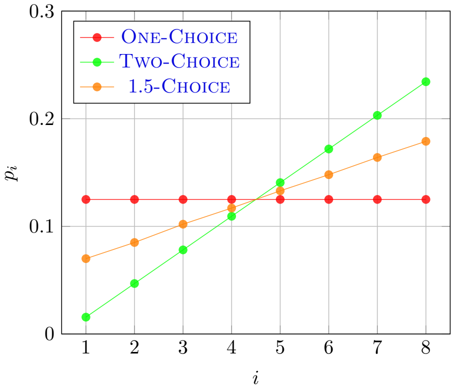
- For 1.5-Choice,

$$p_{\textsc{1.5-Choice}} = \frac{1}{2} \cdot p_{\textsc{One-Choice}} + \frac{1}{2} \cdot p_{\textsc{Two-Choice}}$$



# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For One-Choice,
$$p_{\text{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$
- For Two-Choice,
$$p_{\text{Two-Choice}} = \left(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\right)$$
- For 1.5-Choice,
$$p_{\text{1.5-Choice}} = \frac{1}{2} \cdot p_{\text{One-Choice}} + \frac{1}{2} \cdot p_{\text{Two-Choice}}$$



# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For One-Choice,

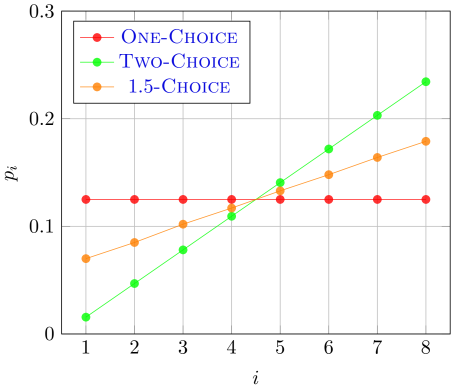
$$p_{\text{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$

- For Two-Choice,

$$p_{\text{Two-Choice}} = \left(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\right)$$

- For 1.5-Choice,

$$p_{\text{1.5-Choice}} = \frac{1}{2} \cdot p_{\text{One-Choice}} + \frac{1}{2} \cdot p_{\text{Two-Choice}}$$




- Having a time-invariant probability vector is handy for the analysis!

# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For ONE-CHOICE,

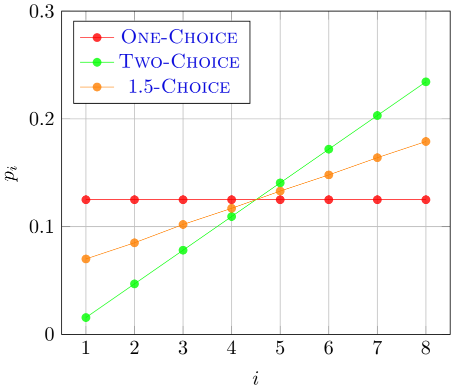
$$p_{\text{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$

- For TWO-CHOICE,

$$p_{\text{Two-Choice}} = \left(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\right)$$

- For 1.5-CHOICE,

$$p_{\text{1.5-Choice}} = \frac{1}{2} \cdot p_{\text{One-Choice}} + \frac{1}{2} \cdot p_{\text{Two-Choice}}$$




- Having a time-invariant probability vector is handy for the analysis!
- **Good:** Both TWO-CHOICE and 1.5-CHOICE have a strong bias towards light bins

# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th heaviest bin.
- For One-Choice,

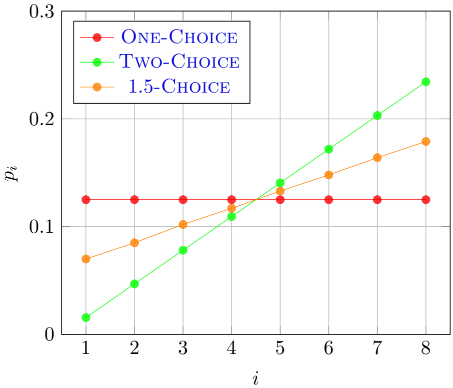
$$p_{\text{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$$

- For Two-Choice,

$$p_{\text{Two-Choice}} = \left(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\right)$$

- For 1.5-Choice,

$$p_{\text{1.5-Choice}} = \frac{1}{2} \cdot p_{\text{One-Choice}} + \frac{1}{2} \cdot p_{\text{Two-Choice}}$$



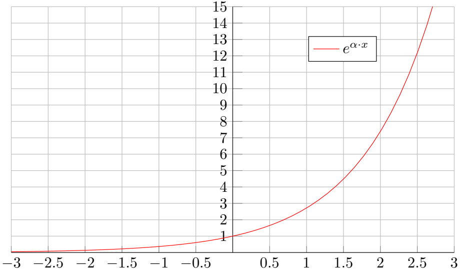

- Having a time-invariant probability vector is handy for the analysis!
- **Good:** Both Two-Choice and 1.5-Choice have a strong bias towards light bins
- **Intuition:** The more choices, the better(?)

# Tool 2: Exponential Potential

$$\Gamma^t := \underbrace{\sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{\sum_{i=1}^{n} e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}}$$

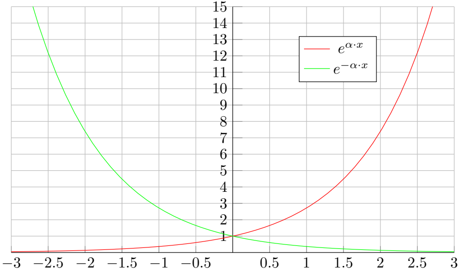
# Tool 2: Exponential Potential

$$\Gamma^t := \underbrace{\sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{\sum_{i=1}^{n} e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}}$$



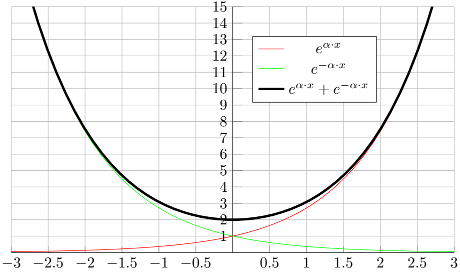
# Tool 2: Exponential Potential

$$\Gamma^t := \underbrace{\sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{\sum_{i=1}^{n} e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}}$$



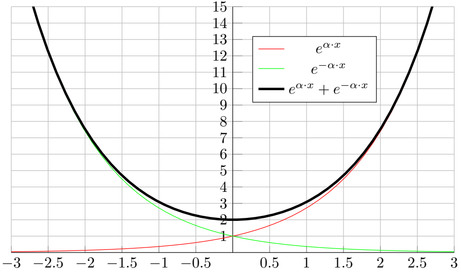
# Tool 2: Exponential Potential

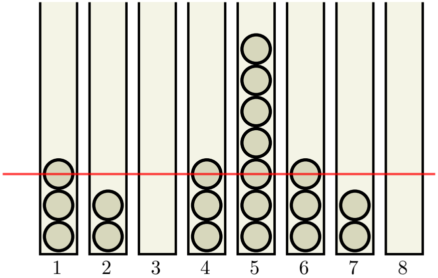
$$\Gamma^t := \underbrace{\sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{\sum_{i=1}^{n} e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



# Tool 2: Exponential Potential

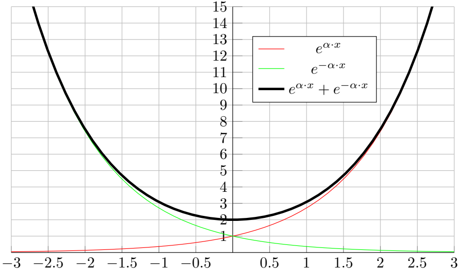
$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$

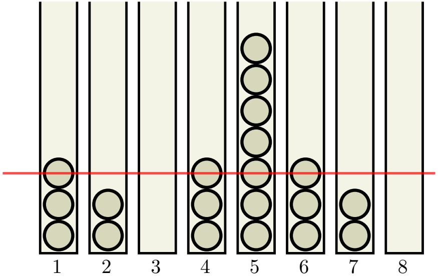


# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$

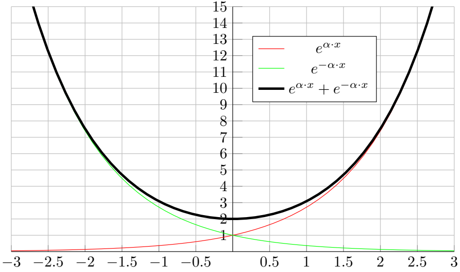


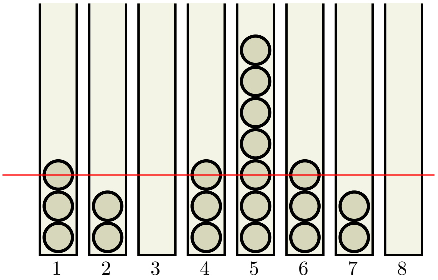




- Normalise load vector: $x^t - 2.5 =$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



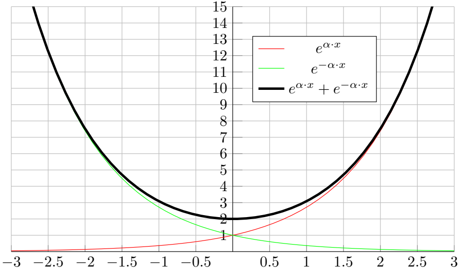



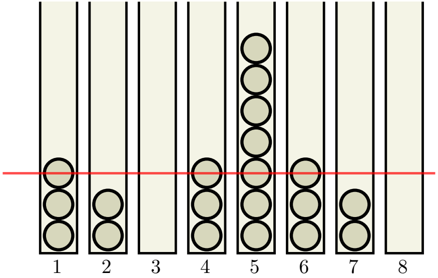

- Normalise load vector: $x^t - 2.5 =$

$(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



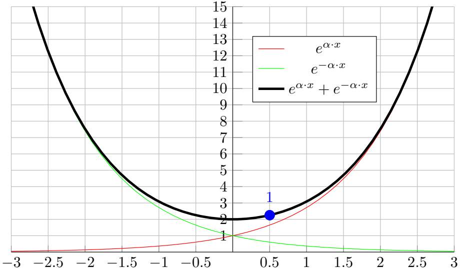



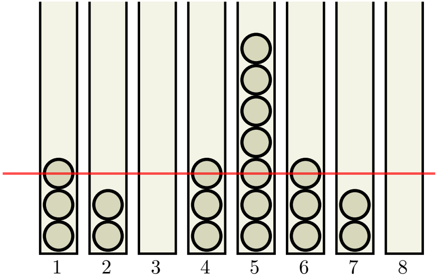

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t =$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



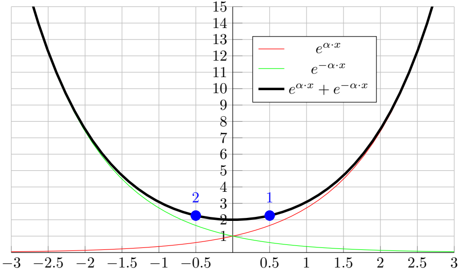



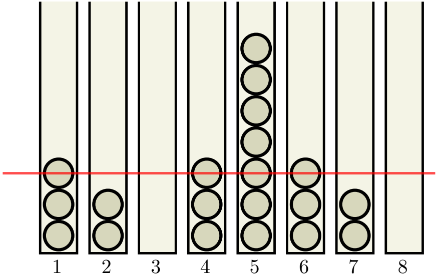

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



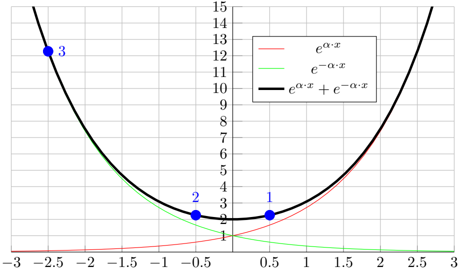



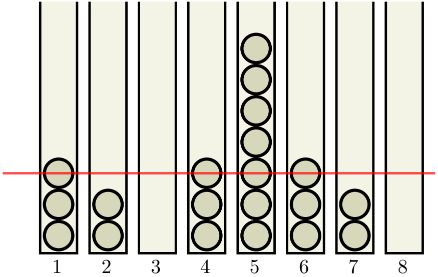

- Normalise load vector: $x^t - 2.5 = (0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25 + 2.25$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



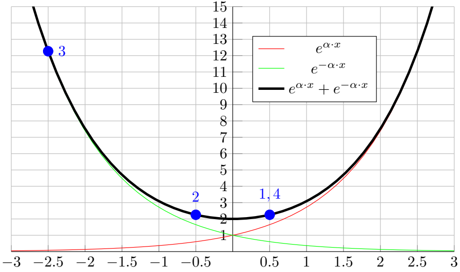



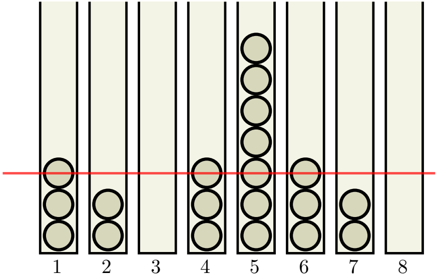

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25 + 2.25 + 12.26$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



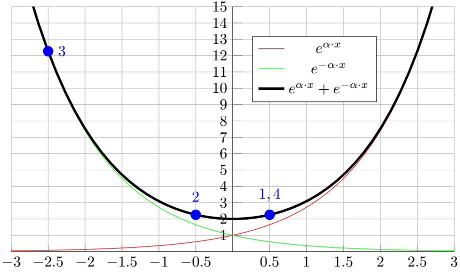



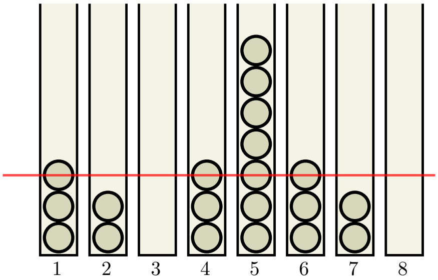

- Normalise load vector: $x^t - 2.5 = (0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25 + 2.25 + 12.26 + 2.25$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



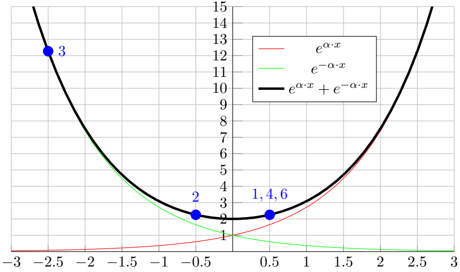



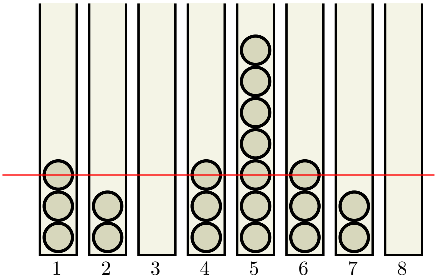

- Normalise load vector: $x^t - 2.5 = (0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25 + 2.25 + 12.26 + 2.25 + 90.03$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



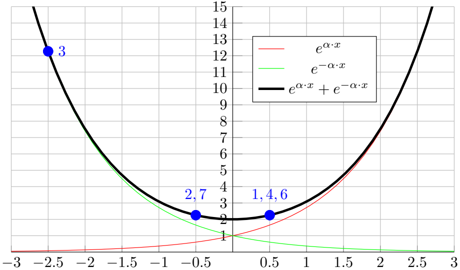



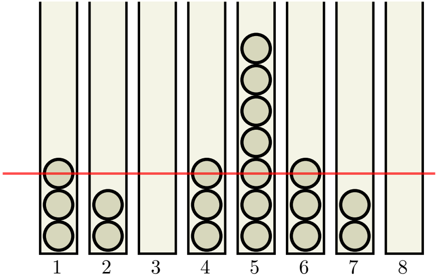

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25 + 2.25 + 12.26 + 2.25 + 90.03 + 2.25$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



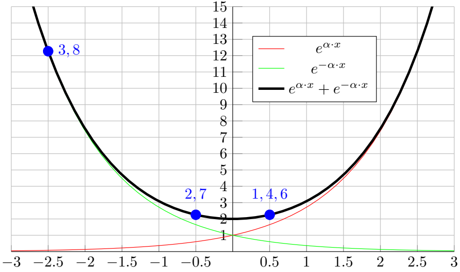



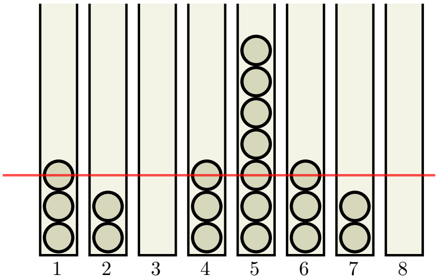

- Normalise load vector: $x^t - 2.5 = (0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25 + 2.25 + 12.26 + 2.25 + 90.03 + 2.25 + 2.25$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



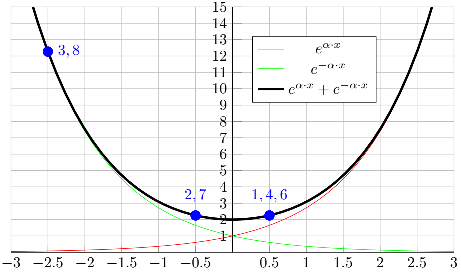



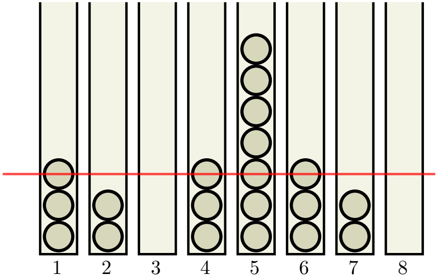

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25 + 2.25 + 12.26 + 2.25 + 90.03 + 2.25 + 2.25 + 12.26$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



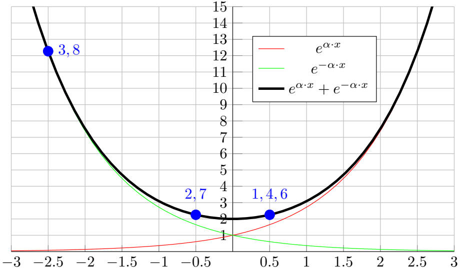

- Normalise load vector: $x^t - 2.5 = (0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

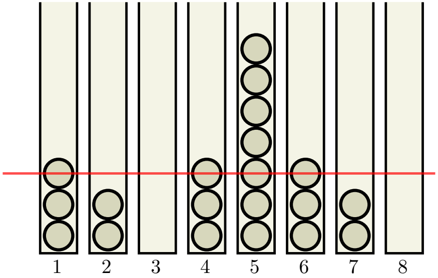
$$\Gamma^t = 2.25 + 2.25 + 12.26 + 2.25 + 90.03 + 2.25 + 2.25 + 12.26 = 125.83$$

# Tool 2: Exponential Potential

$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$

$\Gamma^t = O(\text{poly}(n)) \Rightarrow \text{Gap} = O(\log n)$




- Normalise load vector: $x^t - 2.5 = (0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25 + 2.25 + 12.26 + 2.25 + 90.03 + 2.25 + 2.25 + 12.26 = 125.83$$

# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.

# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\,\Gamma^{t+1} \mid \Gamma^t\,\right] \leq$$

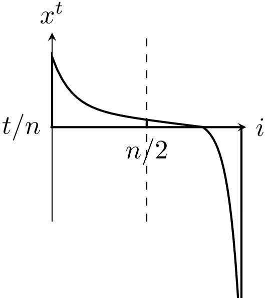
# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq$$



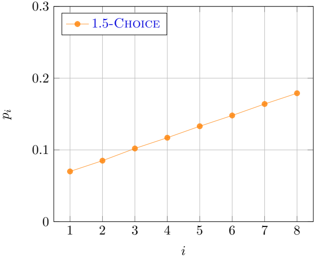
# The Change in the Exponential Potential Function

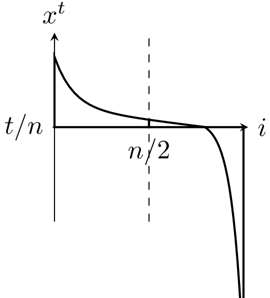
- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq$$



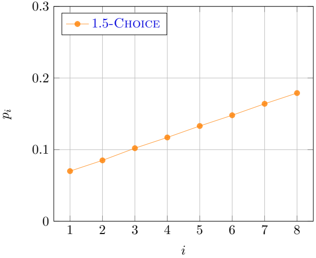
# The Change in the Exponential Potential Function

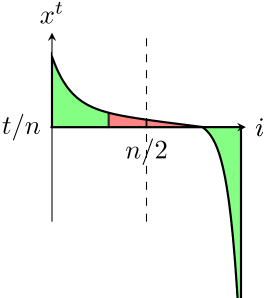
- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq$$



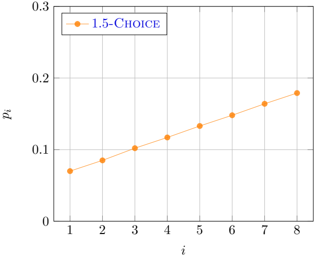
# The Change in the Exponential Potential Function

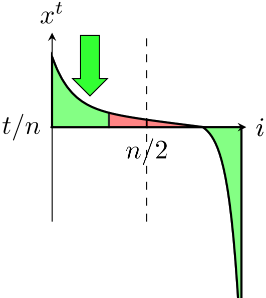
- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq$$



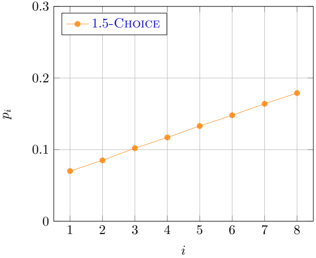
# The Change in the Exponential Potential Function

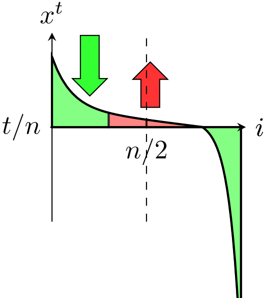
- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq$$



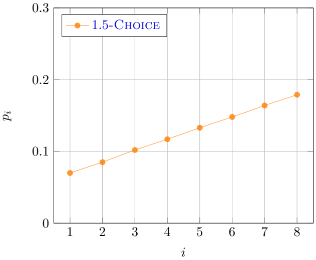
# The Change in the Exponential Potential Function

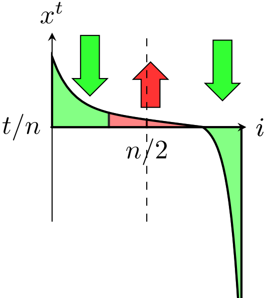
- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\,\Gamma^{t+1} \mid \Gamma^t\,\right] \leq$$



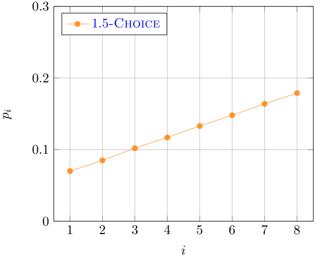
# The Change in the Exponential Potential Function

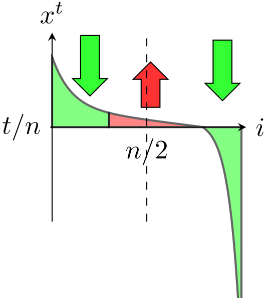
- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq \left(1 - \frac{c_1}{n}\right) \cdot \Gamma^t$$



# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq \left(1 - \frac{c_1}{n}\right) \cdot \Gamma^t + c_2.$$

# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq \left(1 - \frac{c_1}{n}\right) \cdot \Gamma^t + c_2.$$

- This implies that for any $t \geq 0$, $\mathbf{E}\left[\, \Gamma^t \,\right] \leq \frac{c_2}{c_1} \cdot n$.

# The Change in the Exponential Potential Function

- Let us analyse the change in the **exponential potential**

$$\Gamma^t := \sum_{i=1}^{n} \Big( \underbrace{e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} \Big)$$

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove a drop inequality:

$$\mathbf{E}\left[\, \Gamma^{t+1} \mid \Gamma^t \,\right] \leq \left(1 - \frac{c_1}{n}\right) \cdot \Gamma^t + c_2.$$

- This implies that for any $t \geq 0$, $\mathbf{E}\left[\, \Gamma^t \,\right] \leq \frac{c_2}{c_1} \cdot n$.
- By Markov's inequality, we get $\mathrm{Gap}(m) = O(\log n)$ with high probability.

# Mean-Thinning

# Mean-Thinning process

Mean-Thinning:

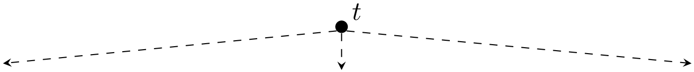
Iteration: For $t \geq 0$, sample two bins $i_1$ and $i_2$ independently u.a.r., and update:

# Mean-Thinning process

Mean-Thinning:

Iteration: For $t \geq 0$, sample two bins $i_1$ and $i_2$ independently u.a.r., and update:

$$\begin{cases} x_{i_1}^{t+1} = x_{i_1}^{t} + 1 & \text{if } x_{i_1}^{t} < \frac{t}{n} \\ \end{cases}$$

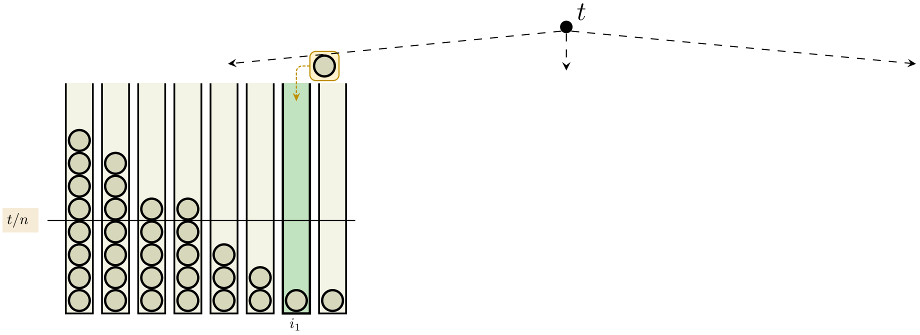


# Mean-Thinning process

**Mean-Thinning:**

Iteration: For $t \geq 0$, sample two bins $i_1$ and $i_2$ independently u.a.r., and update:

$$\begin{cases} x_{i_1}^{t+1} = x_{i_1}^t + 1 \quad \text{if } x_{i_1}^t < \frac{t}{n} \quad \checkmark \\ \end{cases}$$

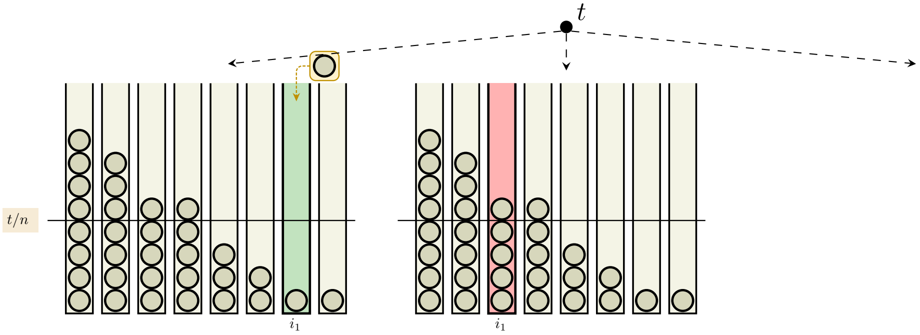


# MEAN-THINNING process

MEAN-THINNING:

Iteration: For $t \geq 0$, sample two bins $i_1$ and $i_2$ independently u.a.r., and update:

$$\begin{cases} x_{i_1}^{t+1} = x_{i_1}^{t} + 1 & \text{if } x_{i_1}^{t} < \frac{t}{n} \\ x_{i_2}^{t+1} = x_{i_2}^{t} + 1 & \text{if } x_{i_1}^{t} \geq \frac{t}{n}. \end{cases}$$

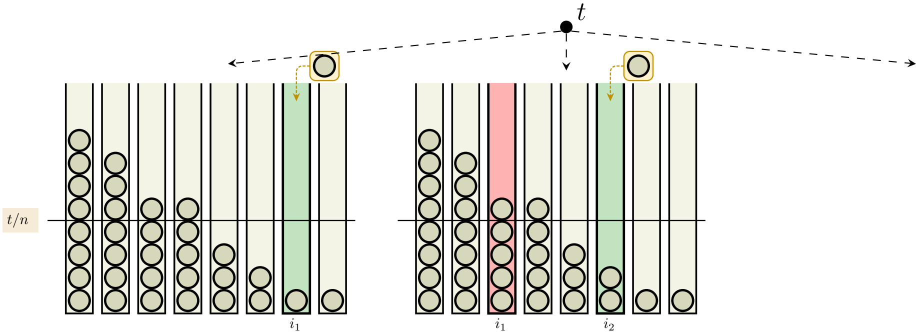


# MEAN-THINNING process

**MEAN-THINNING:**

Iteration: For $t \geq 0$, sample two bins $i_1$ and $i_2$ independently u.a.r., and update:

$$\begin{cases} x_{i_1}^{t+1} = x_{i_1}^{t} + 1 & \text{if } x_{i_1}^{t} < \frac{t}{n} \\ x_{i_2}^{t+1} = x_{i_2}^{t} + 1 & \text{if } x_{i_1}^{t} \geq \frac{t}{n}. \end{cases}$$

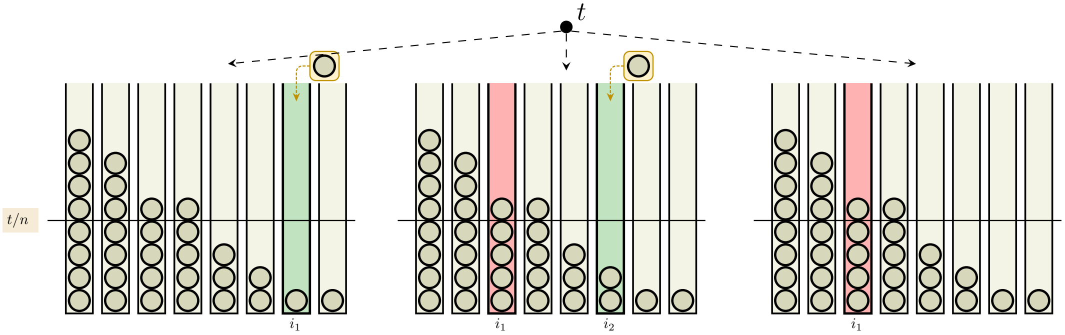


# Mean-Thinning process

**Mean-Thinning:**

Iteration: For $t \geq 0$, sample two bins $i_1$ and $i_2$ independently u.a.r., and update:

$$\begin{cases} x_{i_1}^{t+1} = x_{i_1}^t + 1 & \text{if } x_{i_1}^t < \frac{t}{n} \\ x_{i_2}^{t+1} = x_{i_2}^t + 1 & \text{if } x_{i_1}^t \geq \frac{t}{n}. \end{cases}$$

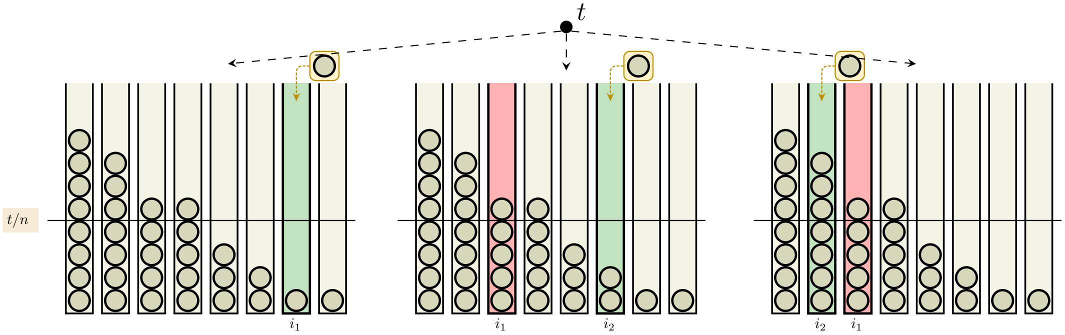


# Mean-Thinning process

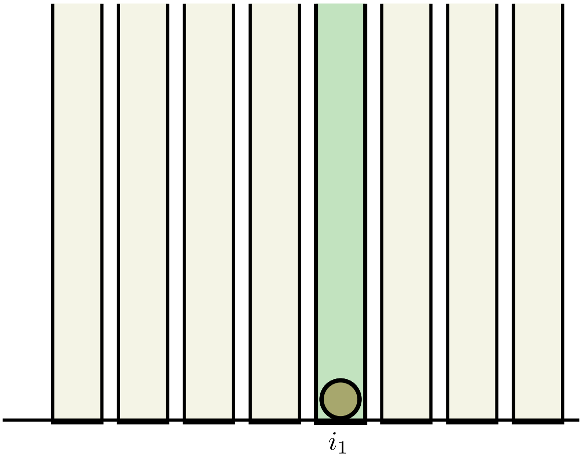
**Mean-Thinning:**

Iteration: For $t \geq 0$, sample two bins $i_1$ and $i_2$ independently u.a.r., and update:

$$\begin{cases} x_{i_1}^{t+1} = x_{i_1}^{t} + 1 & \text{if } x_{i_1}^{t} < \frac{t}{n} \\ x_{i_2}^{t+1} = x_{i_2}^{t} + 1 & \text{if } x_{i_1}^{t} \geq \frac{t}{n}. \end{cases}$$



# Mean-Thinning: Visualization



# Mean-Thinning: Our Results [LSS22]

- For all $m \geq n$, Mean-Thinning achieves w.h.p. $\text{Gap}(m) = \mathcal{O}(\log n)$

# MEAN-THINNING: Our Results [LSS22]

- For all $m \geq n$, MEAN-THINNING achieves w.h.p. $\mathrm{Gap}(m) = \mathcal{O}(\log n)$
- For sufficiently large $m$, w.h.p. $\mathrm{Gap}(m) = \Omega(\log n)$

# MEAN-THINNING: Our Results [LSS22]

- For all $m \geq n$, MEAN-THINNING achieves w.h.p. $\text{Gap}(m) = \mathcal{O}(\log n)$
- For sufficiently large $m$, w.h.p. $\text{Gap}(m) = \Omega(\log n)$
- The following implementation of MEAN-THINNING uses $2 - \epsilon$ samples (on average):
  - Sample a bin $i_1$, allocate ball to $i_1$ if its load is below the average.
  - Otherwise sample a bin $i_2$ and allocate ball to $i_2$.

# MEAN-THINNING: Our Results [LSS22]

- For all $m \geq n$, MEAN-THINNING achieves w.h.p. $\text{Gap}(m) = \mathcal{O}(\log n)$
- For sufficiently large $m$, w.h.p. $\text{Gap}(m) = \Omega(\log n)$
- The following implementation of MEAN-THINNING uses $2 - \epsilon$ samples (on average):
  - Sample a bin $i_1$, allocate ball to $i_1$ if its load is below the average.
  - Otherwise sample a bin $i_2$ and allocate ball to $i_2$.

> Bin $i_1$ (or $i_2$) can directly allocate the ball after checking whether it is underloaded $\rightsquigarrow$ no extra communication or comparison needed!

# Mean-Thinning: Why the analysis is tricky

- Let $\delta^t$ be mean quantile, i.e., the fraction of bins which are overloaded (i.e., $x^t \geq t/n$)

# Mean-Thinning: Why the analysis is tricky

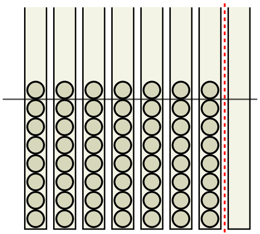
- Let $\delta^t$ be mean quantile, i.e., the fraction of bins which are overloaded (i.e., $x^t \geq t/n$)
- If $\delta^t$ is very large, say $\delta^t = 1 - 1/n$, then $p$ is very close to the One-Choice vector:

$$p_{\text{Mean-Thinning}}(x^t) = \Big(\underbrace{\frac{1}{n} - \frac{1}{n^2}, \ldots, \frac{1}{n} - \frac{1}{n^2}}_{(n-1)\text{ entries}}, \frac{2}{n} - \frac{1}{n^2}\Big).$$

# Mean-Thinning: Why the analysis is tricky

- Let $\delta^t$ be mean quantile, i.e., the fraction of bins which are overloaded (i.e., $x^t \geq t/n$)
- If $\delta^t$ is very large, say $\delta^t = 1 - 1/n$, then $p$ is very close to the One-Choice vector:

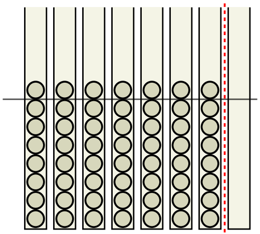
$$p_{\textsc{Mean-Thinning}}(x^t) = \Big(\underbrace{\frac{1}{n} - \frac{1}{n^2}, \ldots, \frac{1}{n} - \frac{1}{n^2}}_{(n-1)\text{ entries}}, \frac{2}{n} - \frac{1}{n^2}\Big).$$



# MEAN-THINNING: Why the analysis is tricky

- Let $\delta^t$ be mean quantile, i.e., the fraction of bins which are overloaded (i.e., $x^t \geq t/n$)
- If $\delta^t$ is very large, say $\delta^t = 1 - 1/n$, then $p$ is very close to the ONE-CHOICE vector:

$$p_{\text{MEAN-THINNING}}(x^t) = \Big(\underbrace{\frac{1}{n} - \frac{1}{n^2}, \ldots, \frac{1}{n} - \frac{1}{n^2}}_{(n-1)\text{ entries}}, \frac{2}{n} - \frac{1}{n^2}\Big).$$



**How can we prove a drop in the exponential potential?**

Combe Down Tunnel (length **1.6 kilometres**)

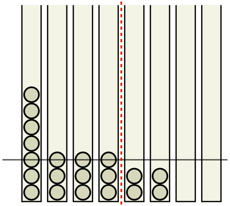
# A Closer Look at $\Gamma^t$

- Similar to the exponential potential analysis before, there is a constant $\alpha > 0$:

# A Closer Look at $\Gamma^t$

- Similar to the exponential potential analysis before, there is a constant $\alpha > 0$:
  - (**Good step**) If $\delta^t \in (\epsilon, 1-\epsilon)$ for const $\epsilon > 0$, then there are constants $c_1, c_2 > 0$

$$\mathbf{E}[\,\Gamma^{t+1} \mid \Gamma^t\,] \leq \left(1 - \frac{c_1 \alpha}{n}\right) \cdot \Gamma^t + c_2,$$



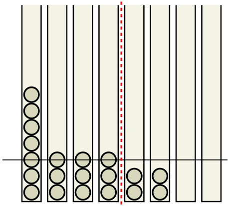
# A Closer Look at $\Gamma^t$

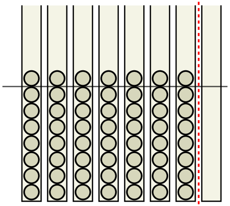
- Similar to the exponential potential analysis before, there is a constant $\alpha > 0$:
  - (**Good step**) If $\delta^t \in (\epsilon, 1-\epsilon)$ for const $\epsilon > 0$, then there are constants $c_1, c_2 > 0$

$$\mathbf{E}[\,\Gamma^{t+1} \mid \Gamma^t\,] \leq \left(1 - \frac{c_1 \alpha}{n}\right) \cdot \Gamma^t + c_2,$$

  - (**Bad step**) If $\delta^t \notin (\epsilon, 1-\epsilon)$, then there are constants $c_3, c_4 > 0$

$$\mathbf{E}[\,\Gamma^{t+1} \mid \Gamma^t\,] \leq \left(1 + \frac{c_3 \alpha^2}{n}\right) \cdot \Gamma^t + c_4.$$





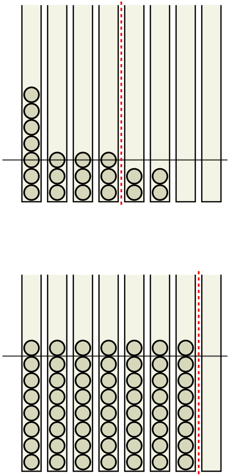
# A Closer Look at $\Gamma^t$

- Similar to the exponential potential analysis before, there is a constant $\alpha > 0$:
  - (**Good step**) If $\delta^t \in (\epsilon, 1-\epsilon)$ for const $\epsilon > 0$, then there are constants $c_1, c_2 > 0$

$$\mathbf{E}[\,\Gamma^{t+1} \mid \Gamma^t\,] \leq \left(1 - \frac{c_1\alpha}{n}\right) \cdot \Gamma^t + c_2,$$

  - (**Bad step**) If $\delta^t \notin (\epsilon, 1-\epsilon)$, then there are constants $c_3, c_4 > 0$

$$\mathbf{E}[\,\Gamma^{t+1} \mid \Gamma^t\,] \leq \left(1 + \frac{c_3\alpha^2}{n}\right) \cdot \Gamma^t + c_4.$$



**How to prove there are enough good steps?**

Skip (Slightly (More)) Technical Part

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

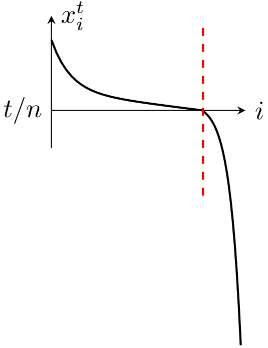
$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$

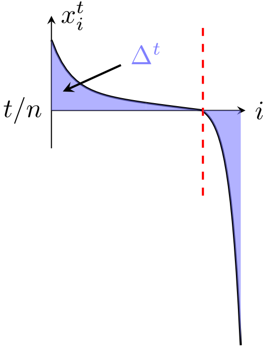


# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$

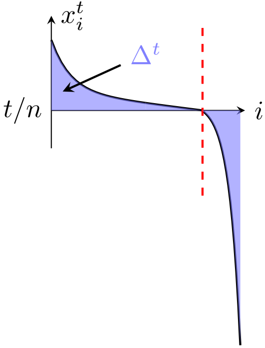


# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$



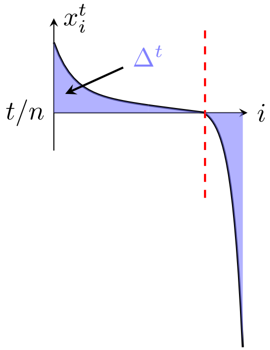

- If $\Delta^t \leq c \cdot n$, then:

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$




- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$

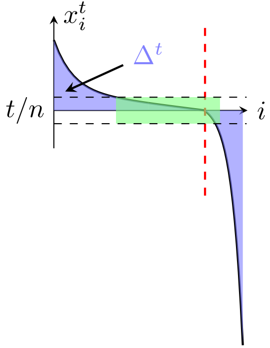
# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$

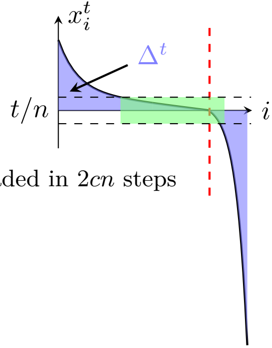


# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$



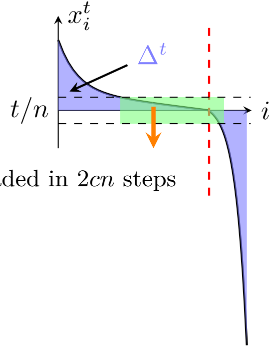

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$



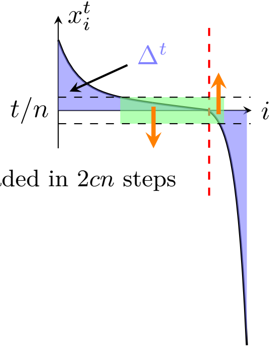

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$



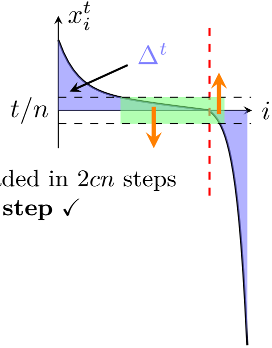

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$



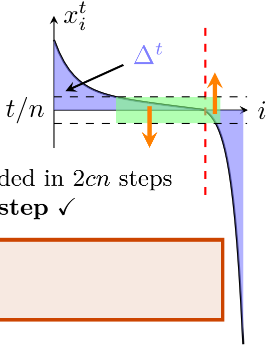

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps
  - After $2cn$ steps, enough overloaded and underloaded bins $\rightsquigarrow$ **good step** ✓

# Mean Quantile Stabilization

**How to prove there are enough good steps?**

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$




- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps
  - After $2cn$ steps, enough overloaded and underloaded bins $\rightsquigarrow$ **good step** ✓

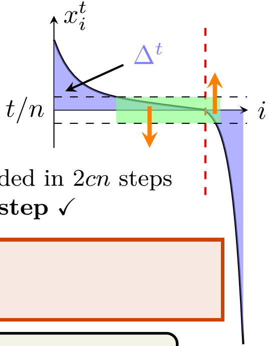
**How to prove that $\Delta^t$ is small?**

# Mean Quantile Stabilization

## How to prove there are enough good steps?

- We need a fraction of at least $\epsilon$ of overloaded and underloaded bins at those steps
- Define an absolute value potential function,

$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$

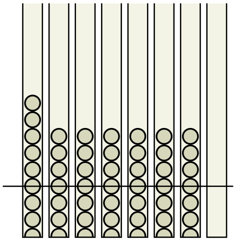



- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps
  - After $2cn$ steps, enough overloaded and underloaded bins $\rightsquigarrow$ **good step** ✓

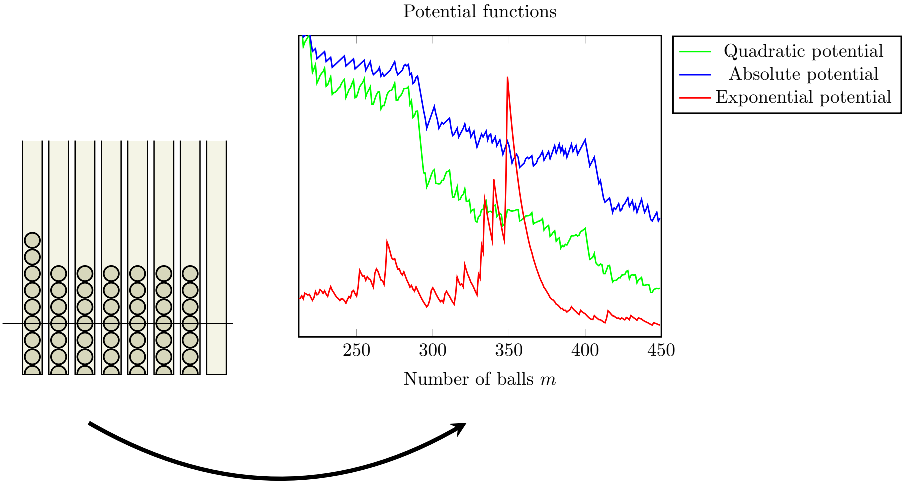
## How to prove that $\Delta^t$ is small?

Change in the quadratic potential $\Upsilon^t = \sum_{i=1}^n \left(x_i^t - \frac{t}{n}\right)^2$ is equal to $-\Delta^t + \Theta(n)$.

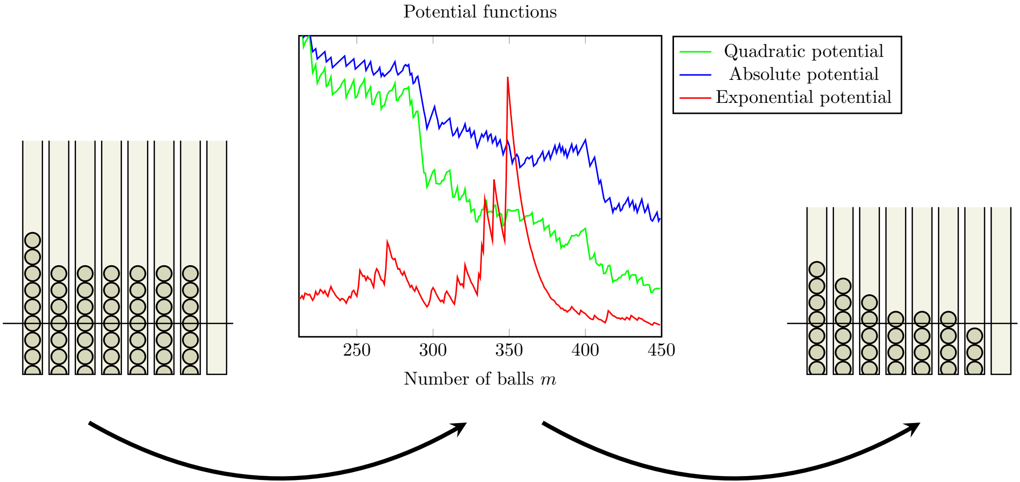
# Recovery from a bad configuration



# Recovery from a bad configuration



# Recovery from a bad configuration



**What happens if we use Thinning with a different threshold?**

## What happens if we use Thinning with a different threshold?

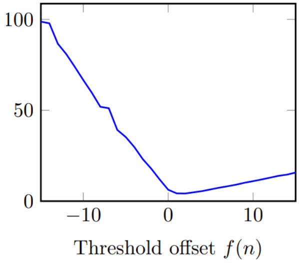
- We proved MEAN-THINNING has a gap of $\Theta(\log n)$ [LSS22]

# What happens if we use Thinning with a different threshold?

- We proved MEAN-THINNING has a gap of $\Theta(\log n)$ [LSS22]
- Simple coupling $\Rightarrow$ THINNING with threshold $\frac{t}{n} + f(n)$ has gap $O(\log n + f(n))$

# What happens if we use Thinning with a different threshold?

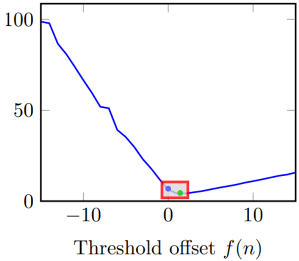
- We proved Mean-Thinning has a gap of $\Theta(\log n)$ [LSS22]
- Simple coupling $\Rightarrow$ Thinning with threshold $\frac{t}{n} + f(n)$ has gap $O(\log n + f(n))$




Gap($m$) for $n = 10^3$ and $m = 10^7$ (15 rep.)

# What happens if we use Thinning with a different threshold?

- We proved Mean-Thinning has a gap of $\Theta(\log n)$ [LSS22]
- Simple coupling $\Rightarrow$ Thinning with threshold $\frac{t}{n} + f(n)$ has gap $O(\log n + f(n))$



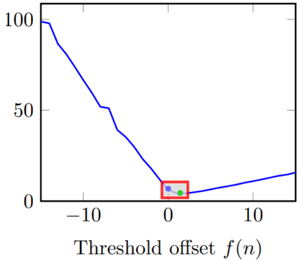

Gap($m$) for $n = 10^3$ and $m = 10^7$ (15 rep.)

**Conjecture:** For thresholds like $f(n) = \sqrt{\log n}$:

- Gap is $O(\frac{\log n}{\log \log n})$
- The average number of samples is $1 + o(1)$

# What happens if we use Thinning with a different threshold?

- We proved Mean-Thinning has a gap of $\Theta(\log n)$ [LSS22]
- Simple coupling $\Rightarrow$ Thinning with threshold $\frac{t}{n} + f(n)$ has gap $O(\log n + f(n))$




Gap($m$) for $n = 10^3$ and $m = 10^7$ (15 rep.)

**Conjecture:** For thresholds like $f(n) = \sqrt{\log n}$:

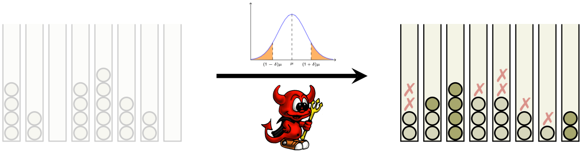
- Gap is $O(\frac{\log n}{\log\log n})$
- The average number of samples is $1 + o(1)$

Allowing the threshold to vary over time:

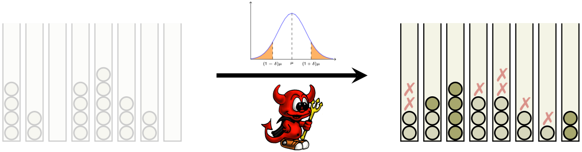
- [FGL24]: gap of $(\log n)^{1/2+o(1)}$ for specific round $m$ (and fraction of $1 - e^{-1/2\sqrt{\log\log\log n}}$ rounds)
- [LS22]: for $m = \Theta(n\sqrt{\log n})$, gap is $\Omega(\sqrt{\log n})$ w.h.p.
- [LS22]: gap is $\Omega(\frac{\log n}{\log\log n})$ for at least a $1/(\log n)$ fraction of all rounds w.h.p.

# Noisy Comparisons

# Two Choice with Noise: Noisy Load Estimates



# Two Choice with Noise: Noisy Load Estimates

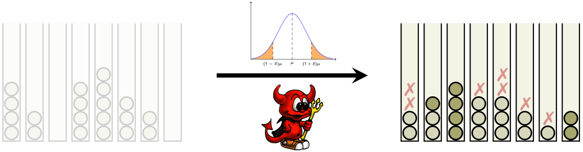


Two-Choice with Noisy Load Estimates [LS23]

Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t$ and $\widetilde{x}_j^t$
3. Allocate ball to the bin with smaller load estimate

# Two Choice with Noise: Noisy Load Estimates



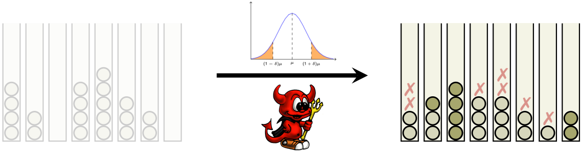
Two-Choice with Noisy Load Estimates [LS23]

Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t$ and $\widetilde{x}_j^t$
3. Allocate ball to the bin with smaller load estimate

Load Estimates could be...

# Two Choice with Noise: Noisy Load Estimates



Two-Choice with Noisy Load Estimates [LS23]

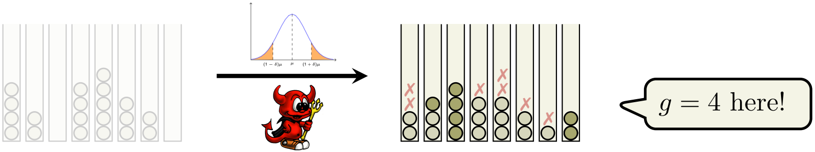
Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t$ and $\widetilde{x}_j^t$
3. Allocate ball to the bin with smaller load estimate

Load Estimates could be...

- adversarial perturbations $\rightsquigarrow \widetilde{x}_i^t \in [x_i^t - g, x_i^t + g]$

# Two Choice with Noise: Noisy Load Estimates




Two-Choice with Noisy Load Estimates [LS23]

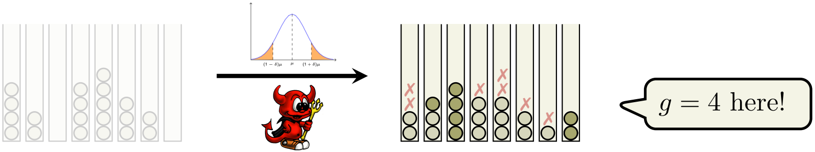
Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t$ and $\widetilde{x}_j^t$
3. Allocate ball to the bin with smaller load estimate

Load Estimates could be...

- adversarial perturbations $\rightsquigarrow \widetilde{x}_i^t \in [x_i^t - g, x_i^t + g]$

# Two Choice with Noise: Noisy Load Estimates



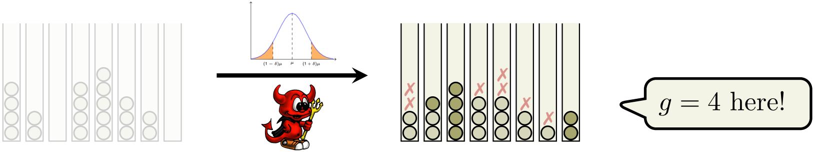

**Two-Choice with Noisy Load Estimates [LS23]**
Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t$ and $\widetilde{x}_j^t$
3. Allocate ball to the bin with smaller load estimate

Load Estimates could be...

- adversarial perturbations $\rightsquigarrow \widetilde{x}_i^t \in [x_i^t - g, x_i^t + g]$
- random perturbations of exact loads $\rightsquigarrow \widetilde{x}_i^t = x_i^t + \mathcal{N}(0, \sigma^2)$

# Two Choice with Noise: Noisy Load Estimates




Two-Choice with Noisy Load Estimates [LS23]

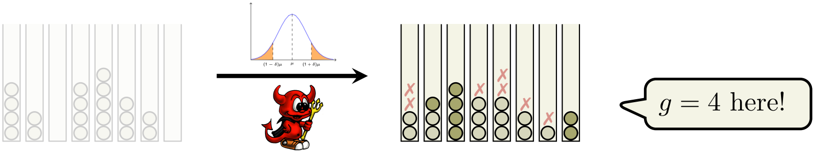
Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t$ and $\widetilde{x}_j^t$
3. Allocate ball to the bin with smaller load estimate

Load Estimates could be...

- adversarial perturbations $\rightsquigarrow \widetilde{x}_i^t \in [x_i^t - g, x_i^t + g]$
- random perturbations of exact loads $\rightsquigarrow \widetilde{x}_i^t = x_i^t + \mathcal{N}(0, \sigma^2)$
- $T$-time-delayed versions of exact loads $\rightsquigarrow \widetilde{x}_i^t \in [x_i^{t-T}, x_i^t]$

# Two Choice with Noise: Noisy Load Estimates




**Two-Choice with Noisy Load Estimates [LS23]**

Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t$ and $\widetilde{x}_j^t$
3. Allocate ball to the bin with smaller load estimate

Load Estimates could be...

- adversarial perturbations $\rightsquigarrow \widetilde{x}_i^t \in [x_i^t - g, x_i^t + g]$
- random perturbations of exact loads $\rightsquigarrow \widetilde{x}_i^t = x_i^t + \mathcal{N}(0, \sigma^2)$
- $T$-time-delayed versions of exact loads $\rightsquigarrow \widetilde{x}_i^t \in [x_i^{t-T}, x_i^t]$

# Comparison-Based Model

Let us assume that the adversary can **directly influence comparisons**:

# Comparison-Based Model

Let us assume that the adversary can **directly influence comparisons**:

$g$-BOUNDED PROCESS
Parameter: Integer $g \geq 1$
Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. If $|x_i^t - x_j^t| > g$, allocate ball to the lesser loaded bin,
3. Otherwise, allocate ball to the higher loaded bin.

# Comparison-Based Model

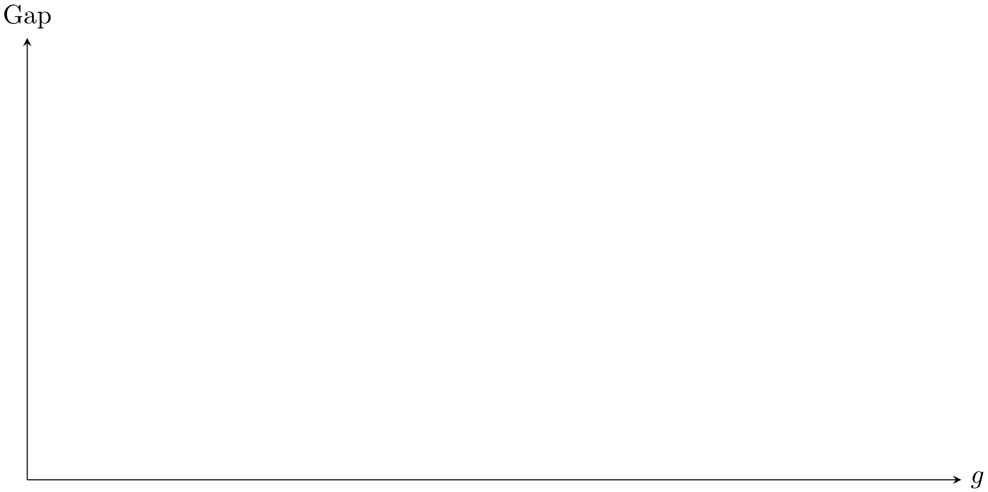
Let us assume that the adversary can **directly influence comparisons**:

$g$-BOUNDED PROCESS
Parameter: Integer $g \geq 1$
Iteration: For each $t \geq 0$:

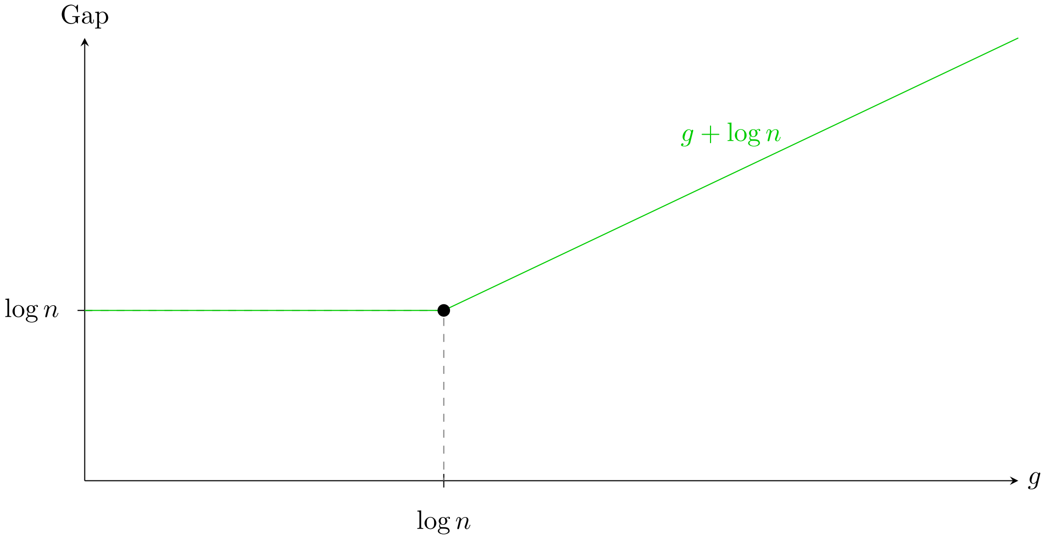
1. Sample two bins $i, j$ independently and uniformly at random
2. If $|x_i^t - x_j^t| > g$, allocate ball to the lesser loaded bin,
3. Otherwise, allocate ball to the higher loaded bin.

Adversary is greedily fooling TWO-CHOICE as often as possible!

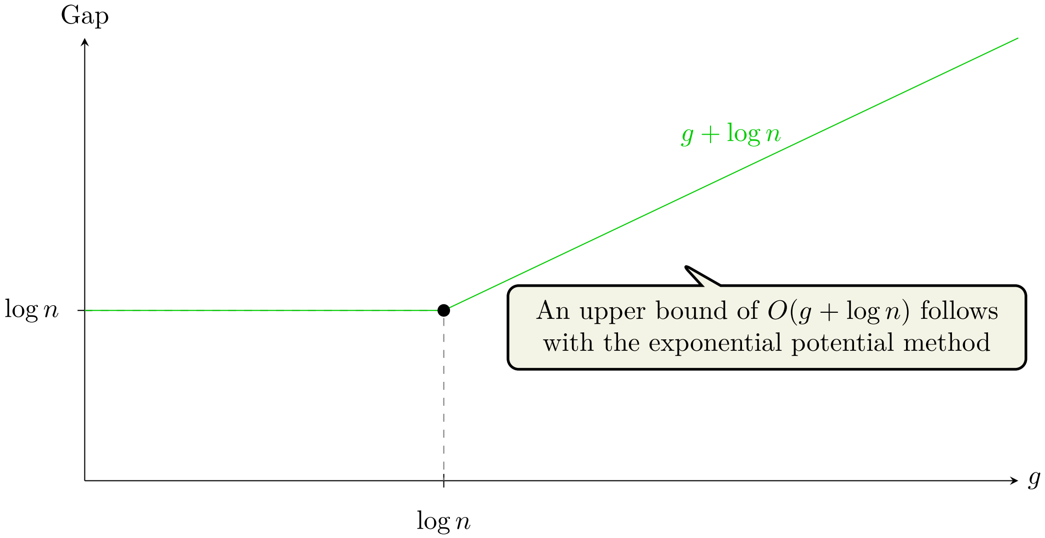
# Asymptotic Results for $g \le \log n$ [LS23]



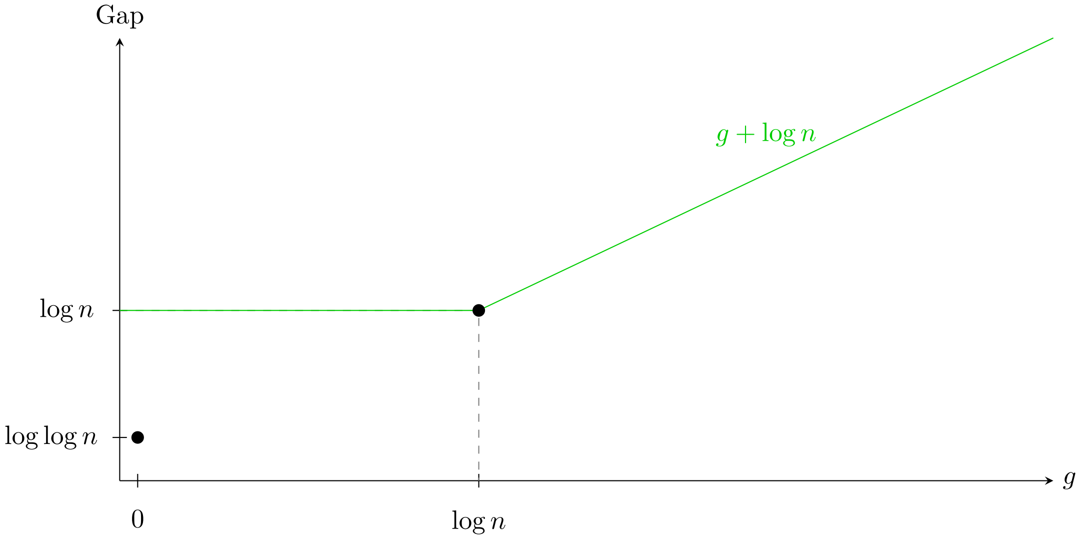
# Asymptotic Results for $g \leq \log n$ [LS23]



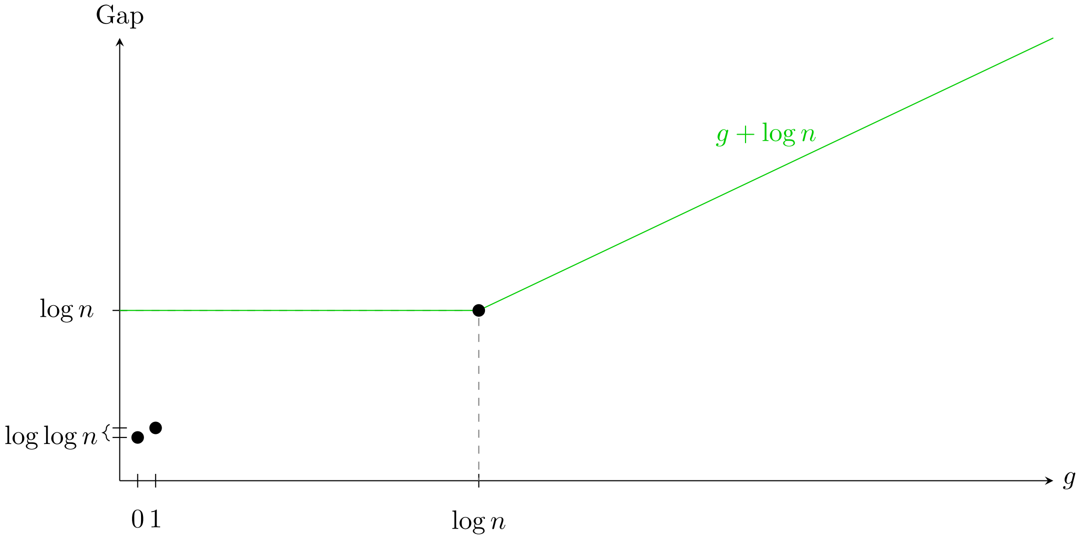
# Asymptotic Results for $g \leq \log n$ [LS23]



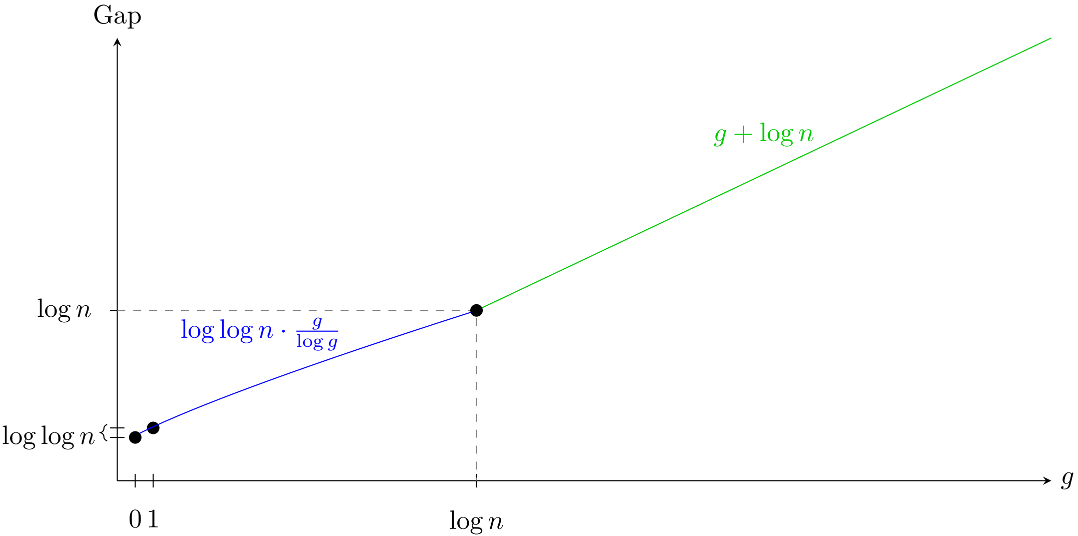
# Asymptotic Results for $g \leq \log n$ [LS23]



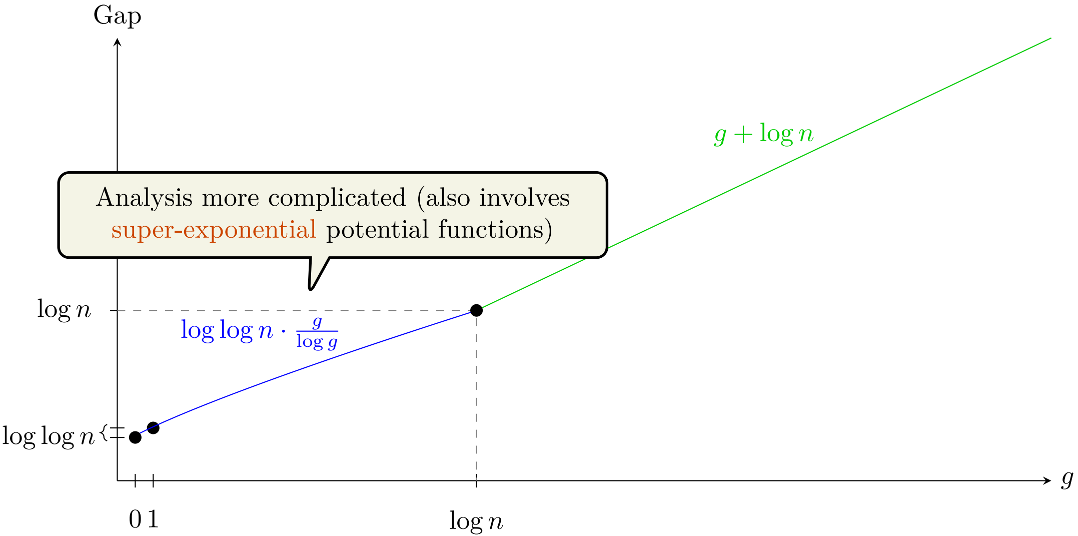
# Asymptotic Results for $g \leq \log n$ [LS23]



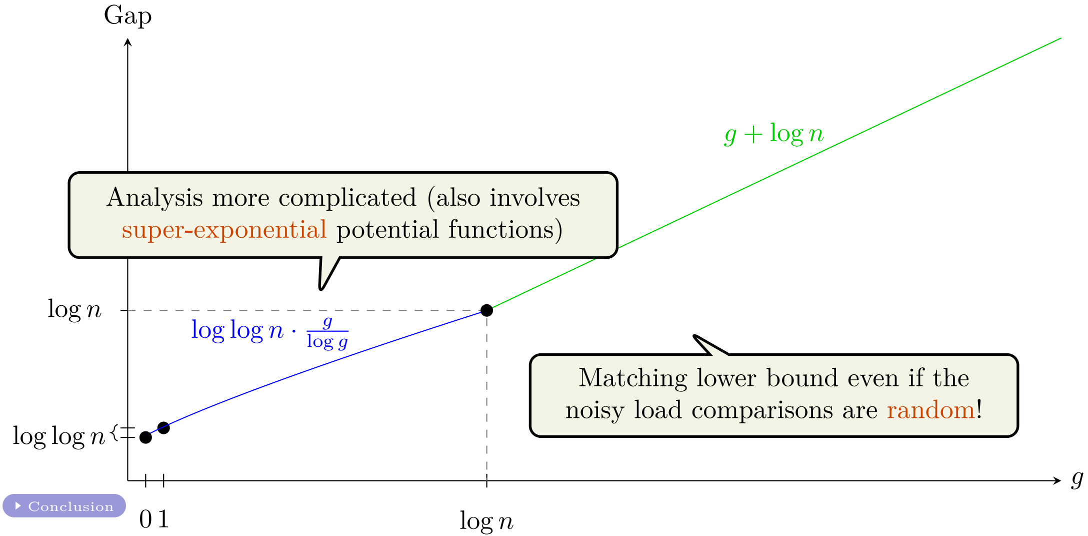
# Asymptotic Results for $g \le \log n$ [LS23]



# Asymptotic Results for $g \leq \log n$ [LS23]



# Asymptotic Results for $g \leq \log n$ [LS23]




Conclusion

# Delay Models (The Problem of Choices)

**$T$-Delay**

Parameter: Integer $T \geq 1$

Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t \in [x_i^{t-T}, x_i^t]$ and $\widetilde{x}_i^t \in [x_j^{t-T}, x_j^t]$.
3. Allocate ball to the bin with smaller load estimate.

# Delay Models (The Problem of Choices)

$T$-Delay

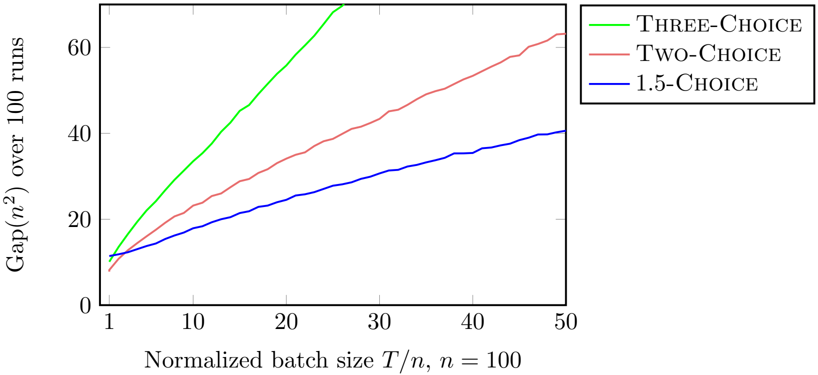
Parameter: Integer $T \geq 1$

Iteration: For each $t \geq 0$:

1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t \in [x_i^{t-T}, x_i^t]$ and $\widetilde{x}_i^t \in [x_j^{t-T}, x_j^t]$.
3. Allocate ball to the bin with smaller load estimate.

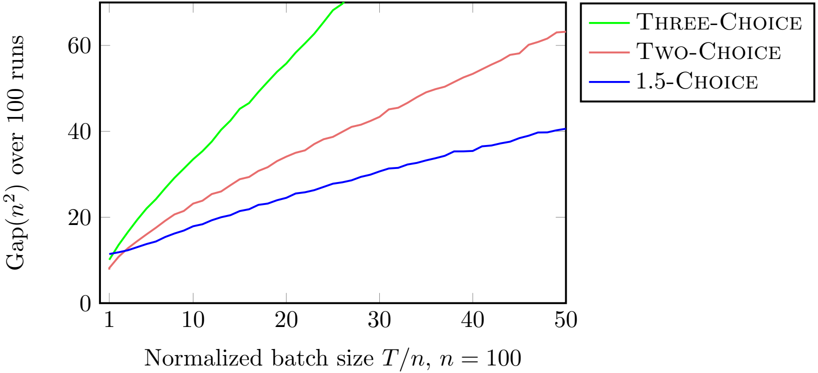
Batching model: Load values are updated every $T$ steps [BCE+12]

# Batching (1/2)



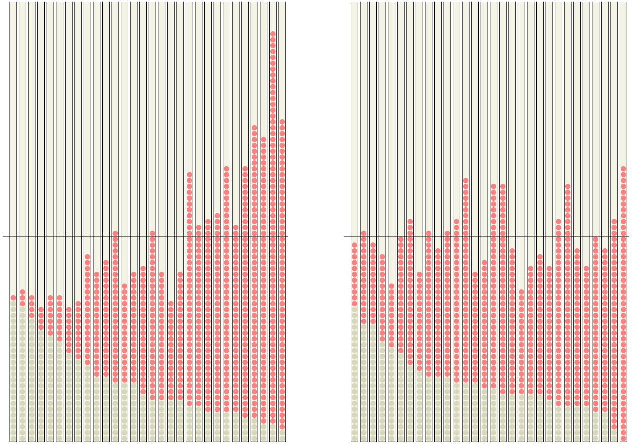
Three-Choice
Two-Choice
1.5-Choice
Gap($n^2$) over 100 runs
0
20
40
60
1
10
20
30
40
50
Normalized batch size $T/n$, $n = 100$

# Batching (1/2)




- Two-Choice (and Three-Choice) have a too strong bias towards the bins that are lightly loaded at the beginning
- Result: For $T \geq n \log n$, $(1+\beta)$-Choice with $\beta = \sqrt{(n/T) \cdot \log n}$ has $\text{Gap}(m) = O(\sqrt{(T/n) \cdot \log n})$ (optimal and quadratically better than Two-Choice)

# Batching (2/2)

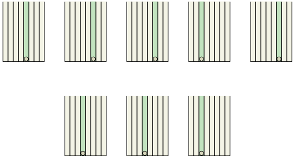


Load distribution after two additional batches of Two-Choice (left) and 1.5-Choice (right)

# Conclusion

Summary of Results:

- Tight bounds for several noisy versions of Two-Choice
- Proof techniques based on (super-)exponential and low-order potential functions
- (Some of the results extend to weighted balls and balanced allocations on graphs)



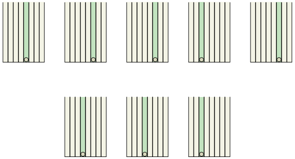
# Conclusion

Summary of Results:

- Tight bounds for several noisy versions of Two-Choice
- Proof techniques based on (super-)exponential and low-order potential functions
- (Some of the results extend to weighted balls and balanced allocations on graphs)

Open Questions:

- Are there better thresholds than Mean-Thinning?
  $\rightsquigarrow$ experiments suggest a gap of $\Theta(\frac{\log n}{\log \log n})$ is possible
- understand Balanced Allocations on sparse graphs
- other "more realistic" noise models



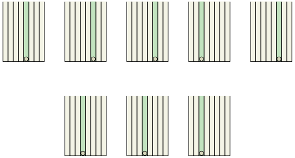
# Conclusion

Summary of Results:

- Tight bounds for several noisy versions of Two-Choice
- Proof techniques based on (super-)exponential and low-order potential functions
- (Some of the results extend to weighted balls and balanced allocations on graphs)

Open Questions:

- Are there better thresholds than Mean-Thinning?
  $\rightsquigarrow$ experiments suggest a gap of $\Theta(\frac{\log n}{\log\log n})$ is possible
- understand Balanced Allocations on sparse graphs
- other "more realistic" noise models



More visualizations:
`https://dimitrioslos.com/research/phd-thesis/index.html`
(Dimitrios Los)

# Bibliography I


- Y. Azar, A. Z. Broder, A. R. Karlin, and E. Upfal, *Balanced allocations*, SIAM J. Comput. **29** (1999), no. 1, 180–200. MR 1710347
- P. Berenbrink, A. Czumaj, M. Englert, T. Friedetzky, and L. Nagel, *Multiple-choice balanced allocation in (almost) parallel*, Proceedings of 16th International Workshop on Approximation, Randomization, and Combinatorial Optimization (RANDOM'12) (Berlin Heidelberg), Springer-Verlag, 2012, pp. 411–422.
- P. Berenbrink, A. Czumaj, A. Steger, and B. Vöcking, *Balanced allocations: the heavily loaded case*, SIAM J. Comput. **35** (2006), no. 6, 1350–1385. MR 2217150
- O. N. Feldheim, O. Gurel-Gurevich, and J. Li, *Long-term balanced allocation via thinning*, The Annals of Applied Probability **34** (2024), no. 1A, 795 – 850.
- D. Los and T. Sauerwald, *Balanced allocations with incomplete information: The power of two queries*, 13th Innovations in Theoretical Computer Science Conference (ITCS'22), vol. 215, 2022, pp. 103:1–103:23.

# Bibliography II


- ——, *Balanced allocations with the choice of noise*, J. ACM **70** (2023), no. 6, 37:1–37:84.
- D. Los, T. Sauerwald, and J. Sylvester, *Balanced allocations: Caching and packing, twinning and thinning*, Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms (SODA'22), 2022, pp. 1847–1874.
- Y. Peres, K. Talwar, and U. Wieder, *Graphical balanced allocations and the* $(1+\beta)$*-choice process*, Random Structures Algorithms **47** (2015), no. 4, 760–775. MR 3418914
- M. Raab and A. Steger, *"Balls into bins"—a simple and tight analysis*, Proceedings of 2nd International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM'98), vol. 1518, Springer, 1998, pp. 159–170. MR 1729169
- Udi Wieder, *Hashing, load balancing and multiple choice*, Found. Trends Theor. Comput. Sci. **12** (2017), no. 3-4, 275–379.