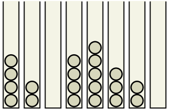
# Balanced Allocations: The Power of Choice versus Noise



Thomas Sauerwald

ADYN Summerschool on Algorithms, Dynamics, and Information Flow in Networks

1st July 2022

- Dimitrios Los, T.S., John Sylvester: Balanced Allocations: Caching and Packing, Twinning and Thinning. SODA 2022: 1847-1874
- Dimitrios Los, T.S.: Balanced Allocations in Batches: Simplified and Generalized. SPAA 2022, to appear.
- Dimitrios Los, T.S.: Balanced Allocations with the Choice of Noise. PODC 2022, to appear.

# Background

# Paradigms in Online Algorithms and Reinforcement Learning

# Balanced allocations setting

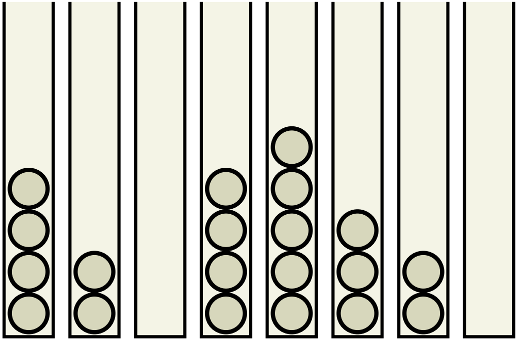
Allocate $m$ tasks (balls) into $n$ machines (bins).

# Balanced allocations setting

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 $\text{Gap}(m) = \max_{i \in [n]}(x_i^m - m/n)$.

# Balanced allocations setting

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 $\text{Gap}(m) = \max_{i\in[n]}(x_i^m - m/n)$.

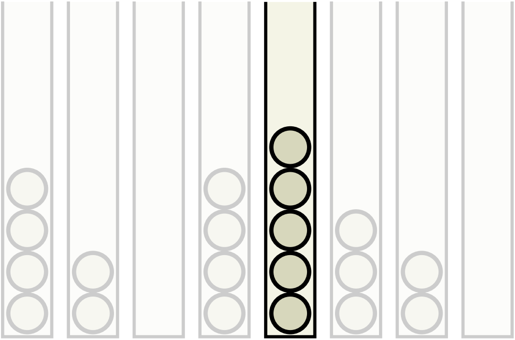


# Balanced allocations setting

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 $\text{Gap}(m) = \max_{i \in [n]}(x_i^m - m/n)$.

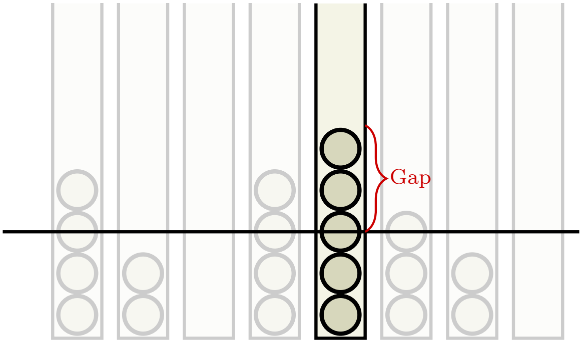


# Balanced allocations setting

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 place the 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 place the ball there.

- For $m \gg n$, w.h.p. $\operatorname{Gap}(m) = \Theta\left(\sqrt{\frac{m}{n} \cdot \log n}\right)$ (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 place the 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])

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 place the 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 place the ball in the least loaded of the two.

# One-Choice versus Two-Choice

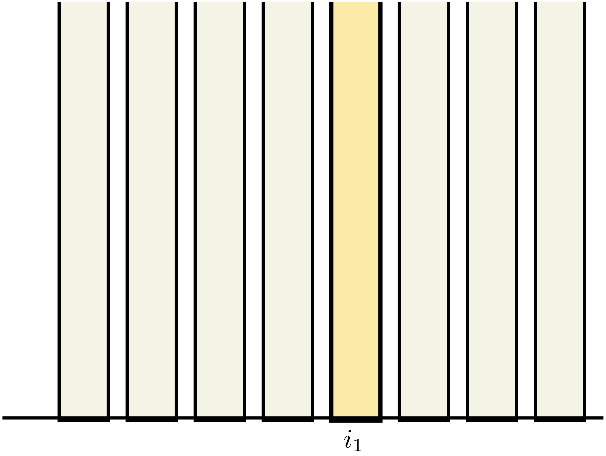
**One-Choice:**
**Iteration:** For each ball $1, 2, \ldots, m$ sample **one** bin uniformly at random (u.a.r.) and place the ball there.

- 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 place the ball in the least loaded of the two.

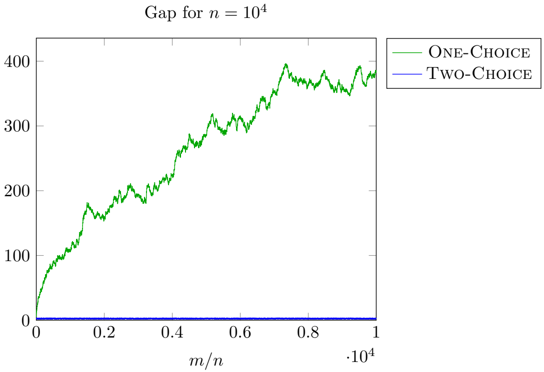
- For any $m \geq n$, w.h.p. $\text{Gap}(m) = \log_2 \log n + \Theta(1)$ [BCSV06].

# Two-Choice: Visualisation

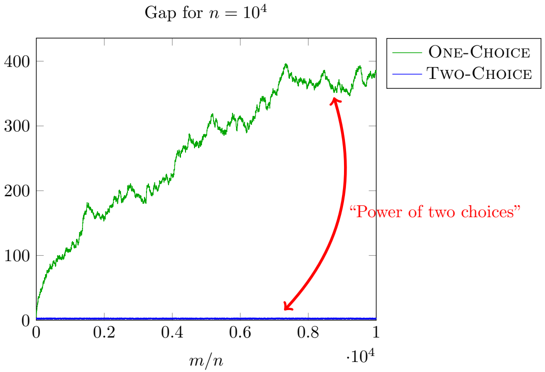


# Experiments

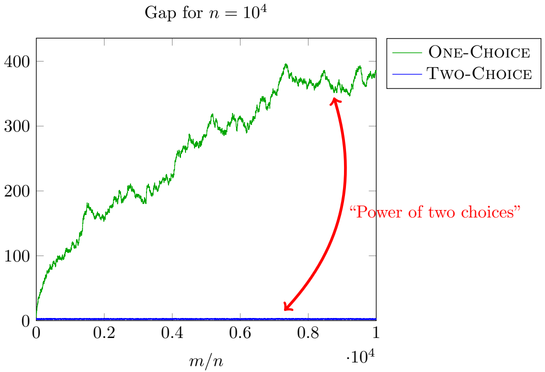
# Experiments



# Experiments



# Experiments




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

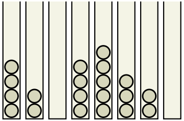
- One-Choice: gap values ranging from 328 to 520
- Two-Choice: 34 runs with gap 2; 66 runs with gap 3

# 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."*

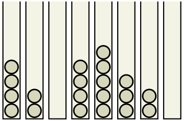
*"These include i-Google's web index, Akamai's overlay routing network, and highly reliable distributed data storage systems used by Microsoft and Dropbox, which are all based on variants of the power of two choices paradigm. There are many other software systems that use balanced allocations as an important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

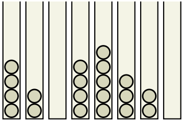
# Relaxations of the Original Model



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?

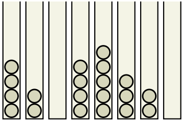
# Relaxations of the Original Model



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

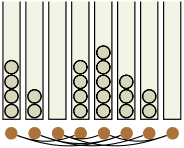
# Relaxations of the Original Model



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

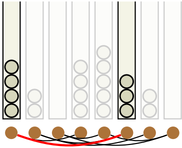


Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

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

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

# Relaxations of the Original Model

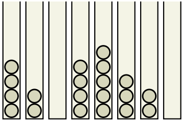


Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

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

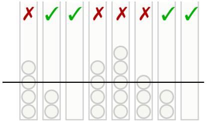


Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

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

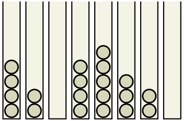


Balls arrive sequentially and sample two bins independently and uniformly at random. After receiving the correct load from the bins, the ball is placed in the least loaded bin.

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

# Relaxations of the Original Model

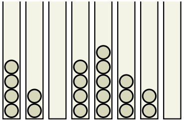


Balls arrive sequentially and sample two bins independently and uniformly at random.
After receiving the correct load from the bins, the ball is placed in the least loaded bin.

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? $\rightsquigarrow$ Threshold

# 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.

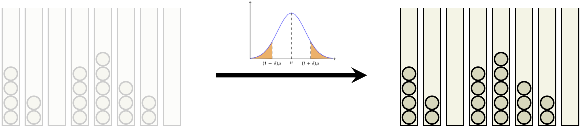
1. What if we are not (always) able to sample two bins? ⇝ 1.5-CHOICE
2. What if the two bin samples are correlated? ⇝ Balanced Allocations on Graphs

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

3b. What if the load information is outdated,

[Mitzenmacher, Richa, Sitaraman (2001)]

# 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

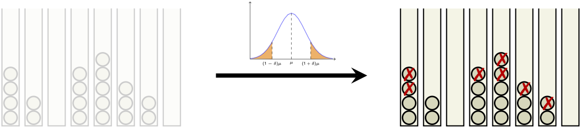
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

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

3b. What if the load information is outdated,

[Mitzenmacher, Richa, Sitaraman (2001)]

# 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

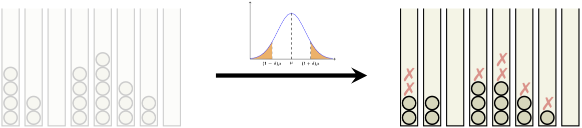
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

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

3b. What if the load information is outdated,

[Mitzenmacher, Richa, Sitaraman (2001)]

# 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

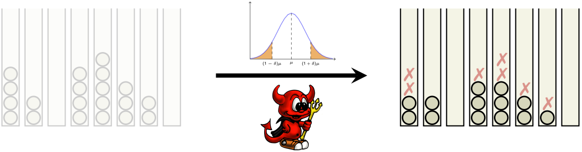
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

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

3b. What if the load information is outdated,

[Mitzenmacher, Richa, Sitaraman (2001)]

# 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.

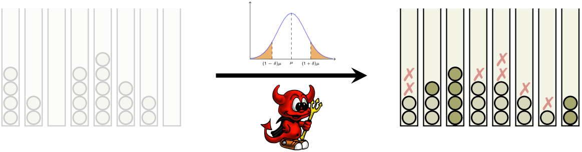
1. What if we are not (always) able to sample two bins? ⇝ 1.5-Choice
2. What if the two bin samples are correlated? ⇝ Balanced Allocations on Graphs

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

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

[Mitzenmacher, Richa, Sitaraman (2001)]

# 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

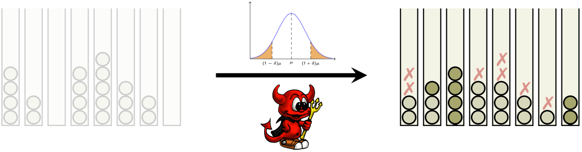
2. What if the two bin samples are correlated? $\rightsquigarrow$ Balanced Allocations on Graphs

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

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

[Mitzenmacher, Richa, Sitaraman (2001)]

# 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? $\rightsquigarrow$ THRESHOLD

3b. What if the load information is outdated, or possibly manipulated by an adversary?
$\rightsquigarrow$ Noise and Delay Models [Mitzenmacher, Richa, Sitaraman (2001)]

# 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:**
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 any other constant

1.5-Choice:

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:**
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:
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)
- 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 $m \gg n$

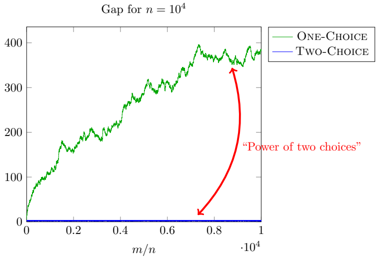
  $\vdots$ ??

- 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:**

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)
- The gap is w.h.p. $\Theta(\log n)$ [PTW15]



# Between One-Choice and Two-Choice

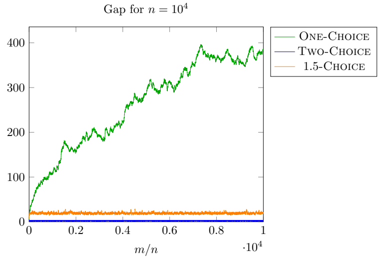
- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$

  $\vdots$ ??

- 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:**
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)
- The gap is w.h.p. $\Theta(\log n)$ [PTW15]



# Between One-Choice and Two-Choice

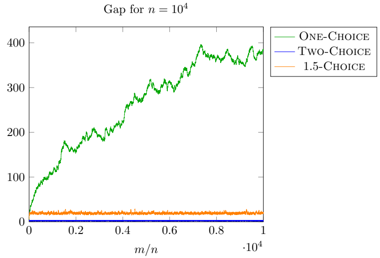
- One-Choice: large gap $\Theta(\sqrt{\frac{m}{n} \cdot \log n})$, especially when $m \gg n$

  $\vdots$ ??

- 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:**
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)
- The gap is w.h.p. $\Theta(\log n)$ [PTW15]



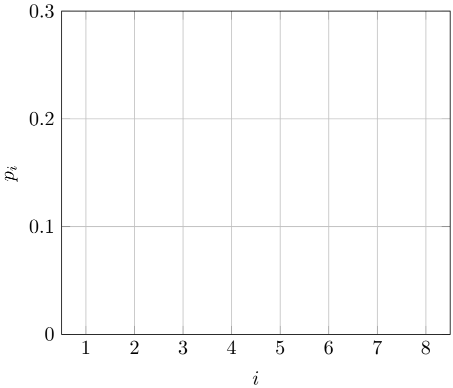
# Tool 1: Probability Vectors

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

# Tool 1: Probability Vectors

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

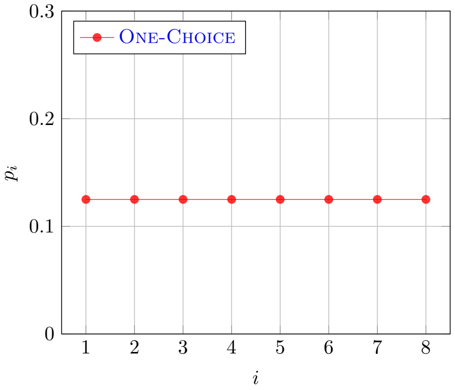
$$p_{\textsc{One-Choice}} = \Big(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\Big).$$



# Tool 1: Probability Vectors

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

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



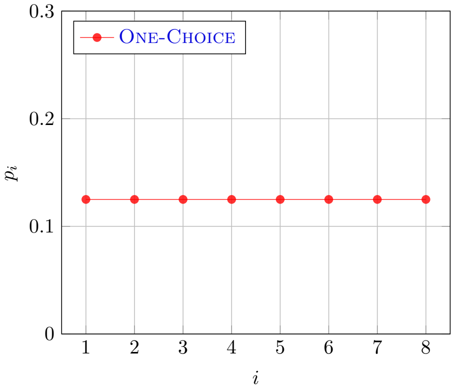
# Tool 1: Probability Vectors

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

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

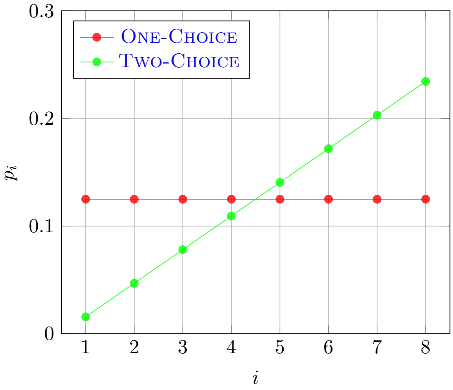
- For Two-Choice,

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



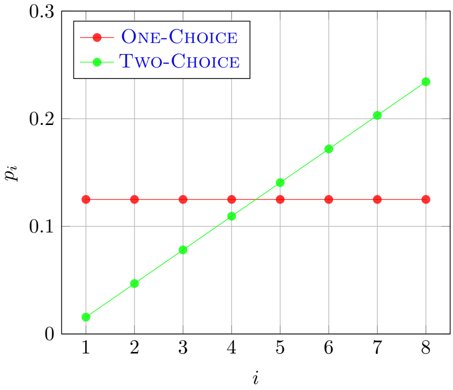
# Tool 1: Probability Vectors

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



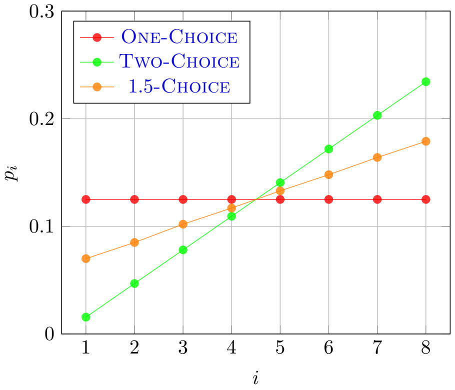
# Tool 1: Probability Vectors

- **Probability vector** $p^t$, where $p_i^t$ is the prob. of allocating to $i$-th most loaded 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 most loaded bin.
- For One-Choice,
$$p_{\textsc{One-Choice}} = \Big(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\Big).$$
- For Two-Choice,
$$p_{\textsc{Two-Choice}} = \Big(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\Big).$$
- 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 most loaded bin.
- For One-Choice,

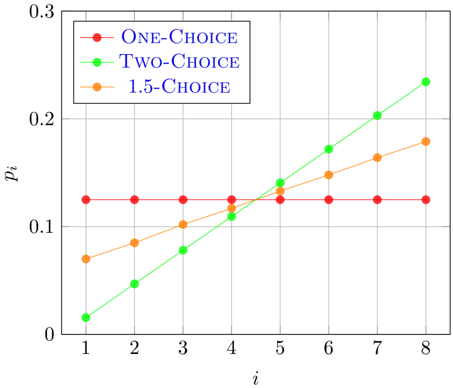
$$p_{\text{One-Choice}} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right).$$

- 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,

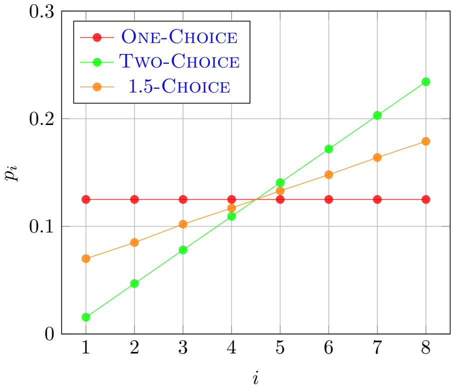
$$p_{\text{1.5-Choice}} = \frac{1}{2} \cdot p_{\text{One-Choice}} + \frac{1}{2} \cdot p_{\text{Two-Choice}}$$




- 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 most loaded bin.
- For One-Choice,
$$p_{\text{One-Choice}} = \Big(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\Big).$$
- For Two-Choice,
$$p_{\text{Two-Choice}} = \Big(\frac{1}{n^2}, \frac{3}{n^2}, \ldots, \frac{2i-1}{n^2}, \ldots, \frac{2n-2}{n^2}\Big).$$
- For 1.5-Choice,
$$p_{\text{1.5-Choice}} = \frac{1}{2} \cdot p_{\text{One-Choice}} + \frac{1}{2} \cdot p_{\text{Two-Choice}}$$



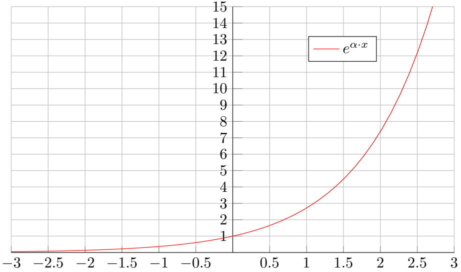

- Having a time-invariant probability vector is handy for the analysis!
- **Good:** Both Two-Choice and 1.5-Choice have a strong bias towards light bins

# Tool 2: Exponential Potential

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

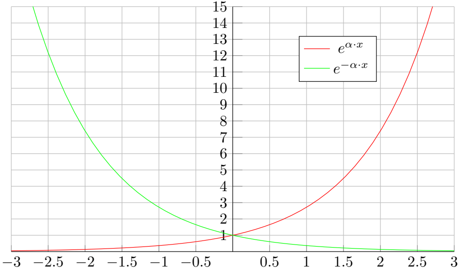
# Tool 2: Exponential Potential

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



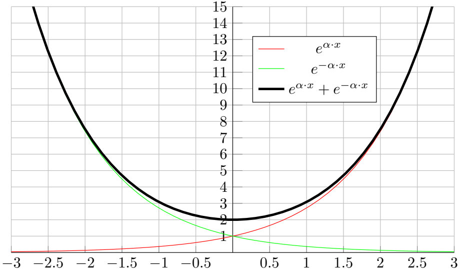
# Tool 2: Exponential Potential

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



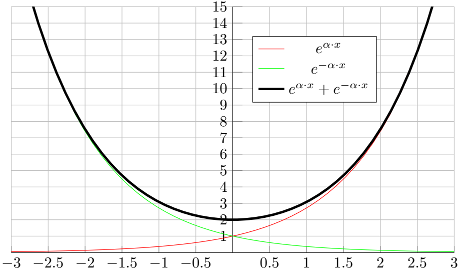
# Tool 2: Exponential Potential

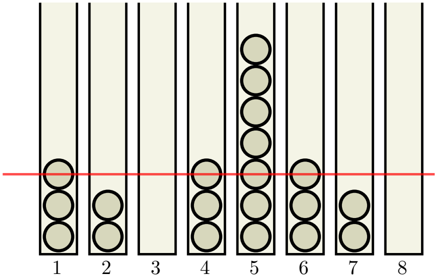
$$\Gamma^t := \underbrace{\sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)}}_{\text{Overload potential}} + \underbrace{\sum_{i=1}^{n} e^{-\alpha(x_i^t - t/n)}}_{\text{Underload potential}} = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$



# Tool 2: Exponential Potential

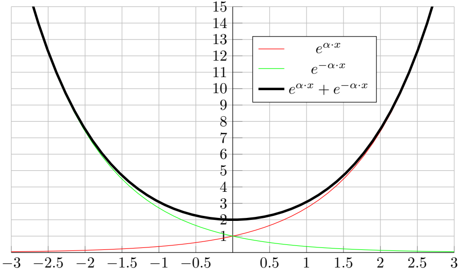
$$\Gamma^t = \sum_{i=1}^{n} e^{\alpha(x_i^t - t/n)} + e^{-\alpha(x_i^t + t/n)}$$

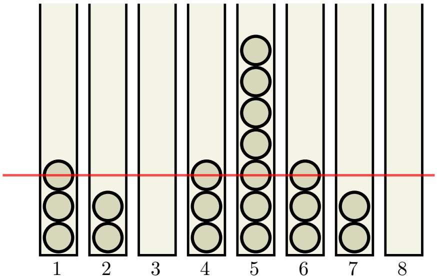


# Tool 2: Exponential Potential

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

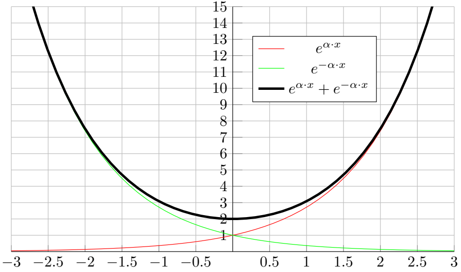


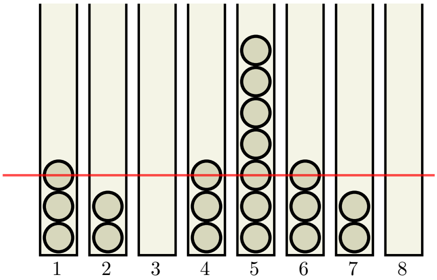




- Normalise load vector: $x^t - 2.5 =$

# Tool 2: Exponential Potential

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



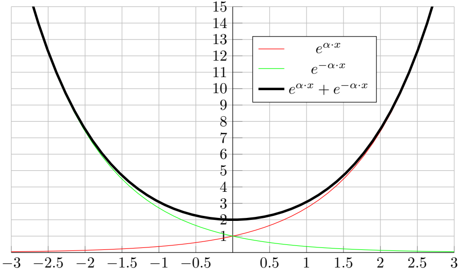



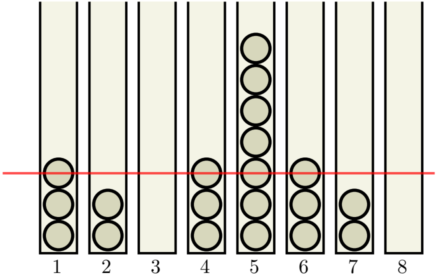

- Normalise load vector: $x^t - 2.5 =$

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

# Tool 2: Exponential Potential

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



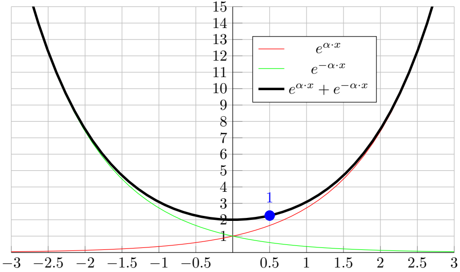



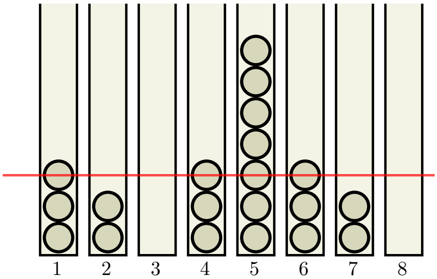

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t =$$

# Tool 2: Exponential Potential

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



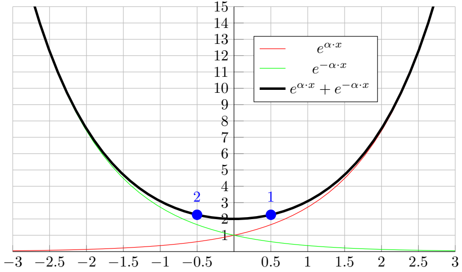



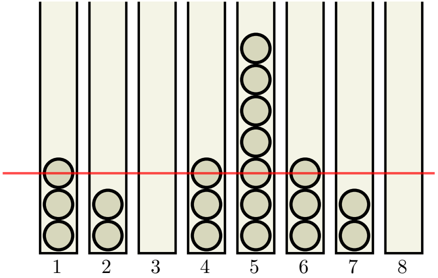

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

$$\Gamma^t = 2.25$$

# Tool 2: Exponential Potential

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



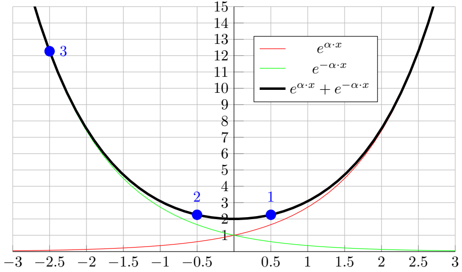



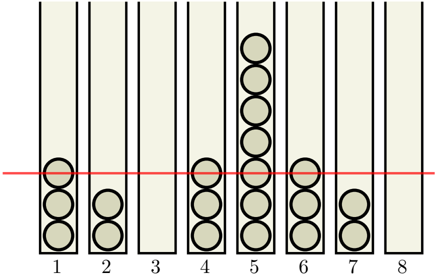

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

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

# Tool 2: Exponential Potential

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



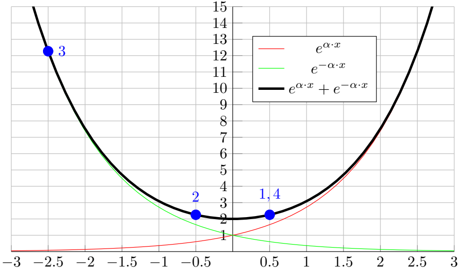



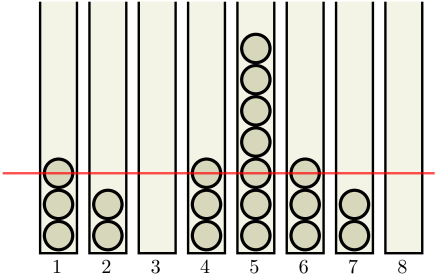

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

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

# Tool 2: Exponential Potential

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



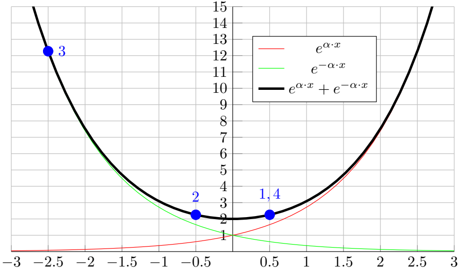



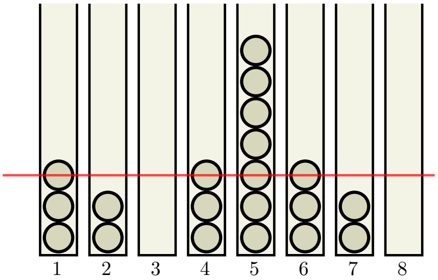

- Normalise load vector: $x^t - 2.5 = (0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

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

# Tool 2: Exponential Potential

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



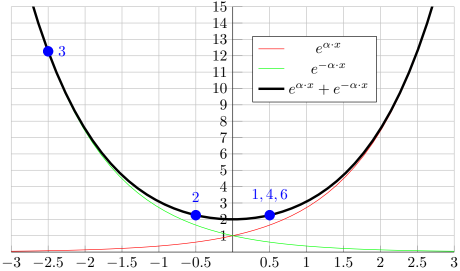



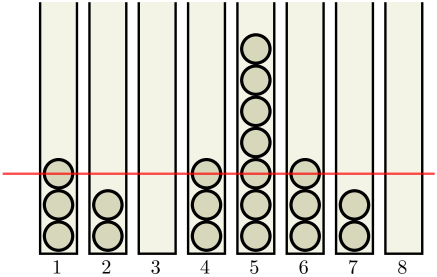

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

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

# Tool 2: Exponential Potential

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



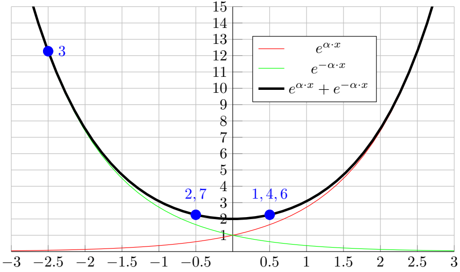



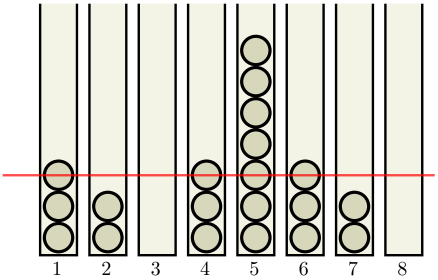

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

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

# Tool 2: Exponential Potential

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



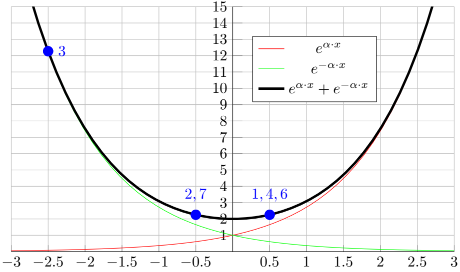



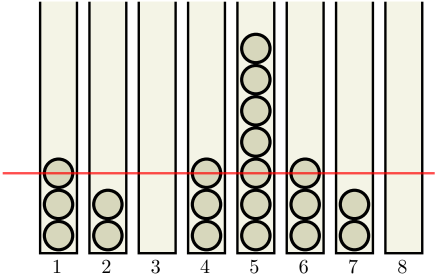

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

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

# Tool 2: Exponential Potential

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



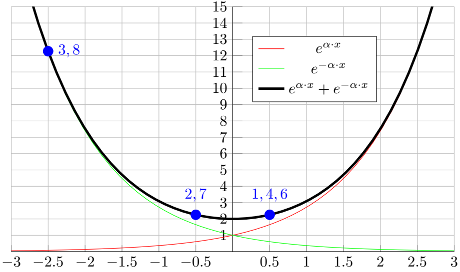



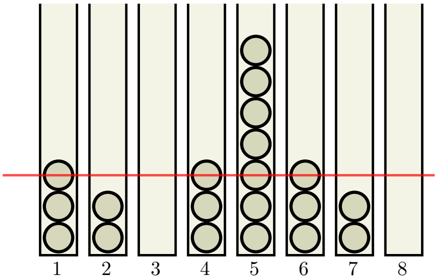

- Normalise load vector: $x^t - 2.5 =$ $(0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

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

# Tool 2: Exponential Potential

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



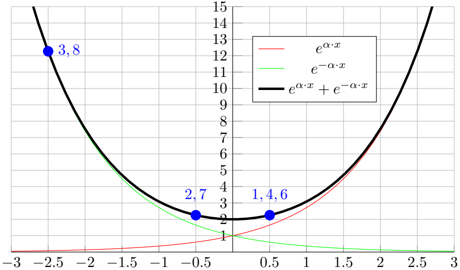

- Normalise load vector: $x^t - 2.5 = (0.5, -0.5, -2.5, 0.5, 4.5, 0.5, -0.5, -2.5)$
- Evaluate Exponential Potential Function ($\alpha = 1$):

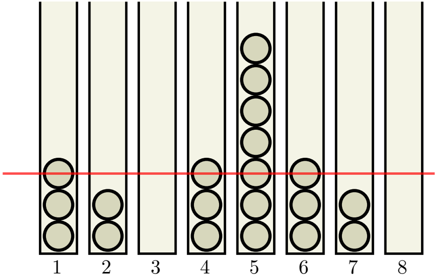
$$\Gamma^t = 2.25 + 2.25 + 12.26 + 2.25 + 90.03 + 2.25 + 2.25 + 12.26 = 125.83$$

# 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 that 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 that a drop inequality:

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

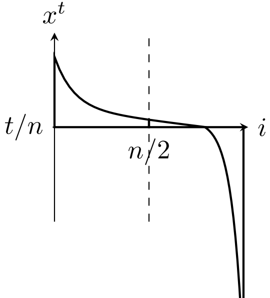
# The Change in the Exponential Potential Function

- 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 that a drop inequality:

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



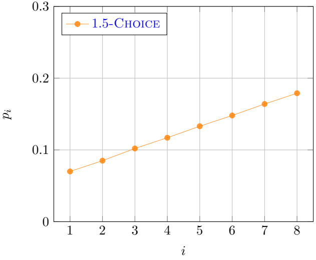
# The Change in the Exponential Potential Function

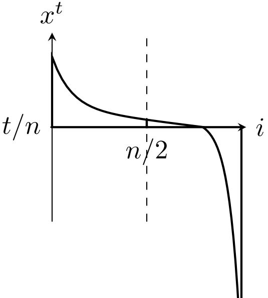
- Let us analyse the change in the **exponential potential**

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

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

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



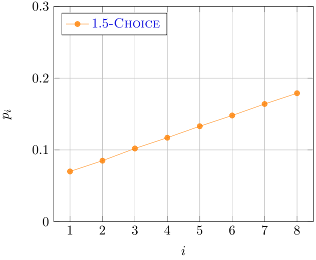
# The Change in the Exponential Potential Function

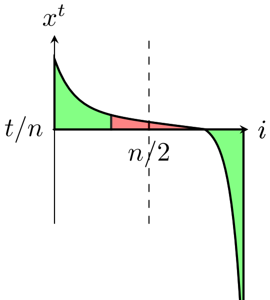
- Let us analyse the change in the **exponential potential**

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

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

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



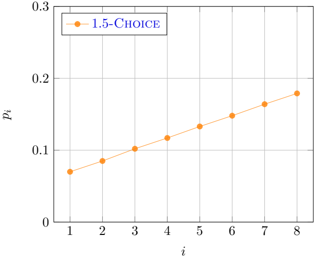
# The Change in the Exponential Potential Function

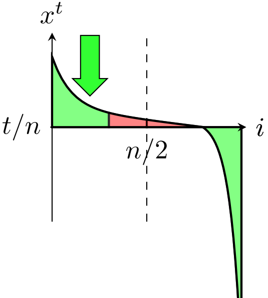
- Let us analyse the change in the **exponential potential**

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

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

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



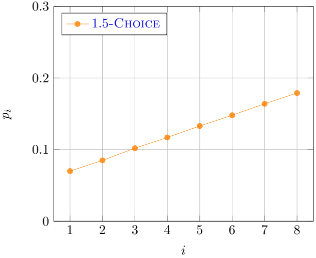
# The Change in the Exponential Potential Function

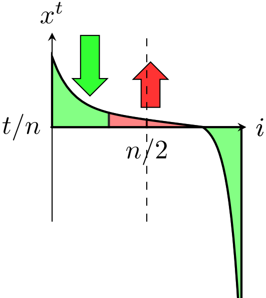
- Let us analyse the change in the **exponential potential**

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

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

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



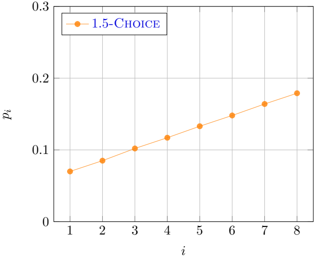
# The Change in the Exponential Potential Function

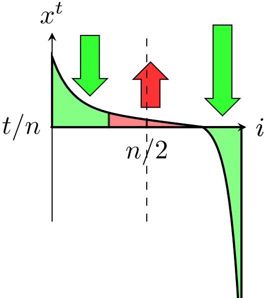
- Let us analyse the change in the **exponential potential**

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

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

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



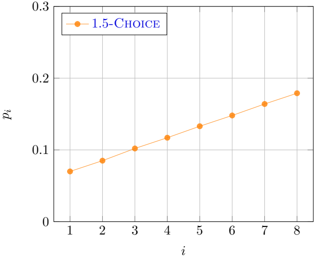
# The Change in the Exponential Potential Function

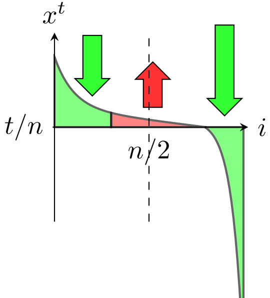
- Let us analyse the change in the **exponential potential**

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

- For 1.5-Choice, $\alpha = 1/144$.
- One can prove that 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 that 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 $t \geq 0$, $\mathbf{E}\left[\, \Gamma^t \,\right] \leq c \cdot n$ for some constant $c > 0$.

# 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 that 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 $t \geq 0$, $\mathbf{E}\left[\, \Gamma^t \,\right] \leq c \cdot n$ for some constant $c > 0$.
- By Markov's inequality, we get $\text{Gap}(m) = O(\log n)$ with high probability.

# Mean-Threshold

# Mean-Threshold process

Mean-Threshold:

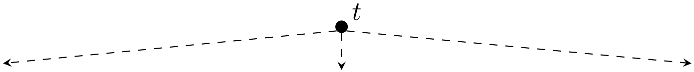
Iteration: For $t \geq 0$, sample two bins $i_1$ and $i_2$ independently u.a.r., and update:

# MEAN-THRESHOLD process

MEAN-THRESHOLD:

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

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

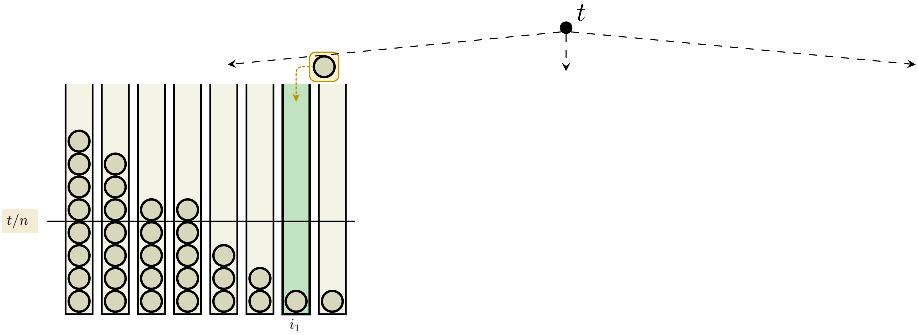


# Mean-Threshold process

**Mean-Threshold:**

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

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

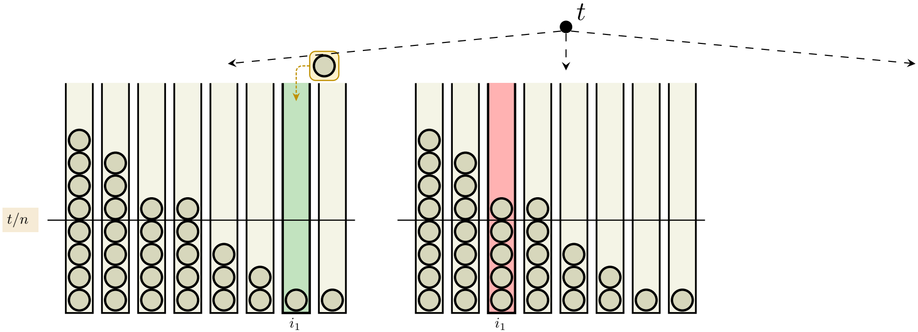


# Mean-Threshold process

**Mean-Threshold:**

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

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

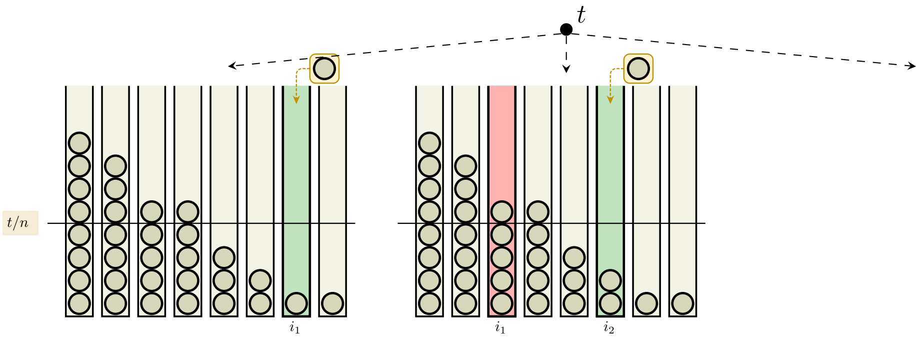


# MEAN-THRESHOLD process

**MEAN-THRESHOLD:**

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

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

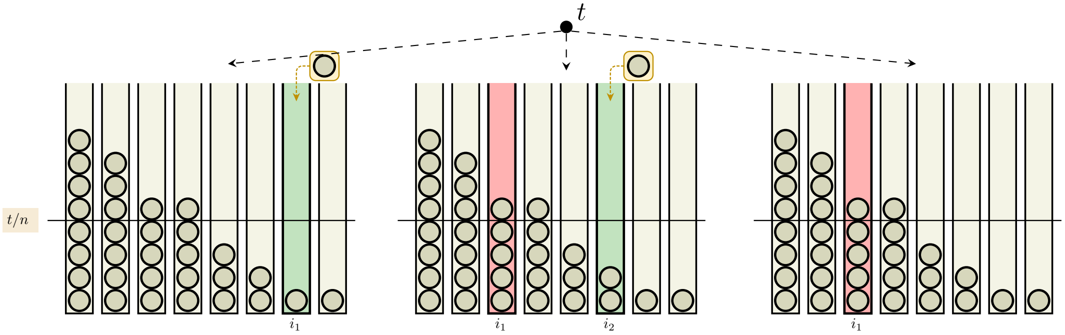


# Mean-Threshold process

**Mean-Threshold:**

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

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

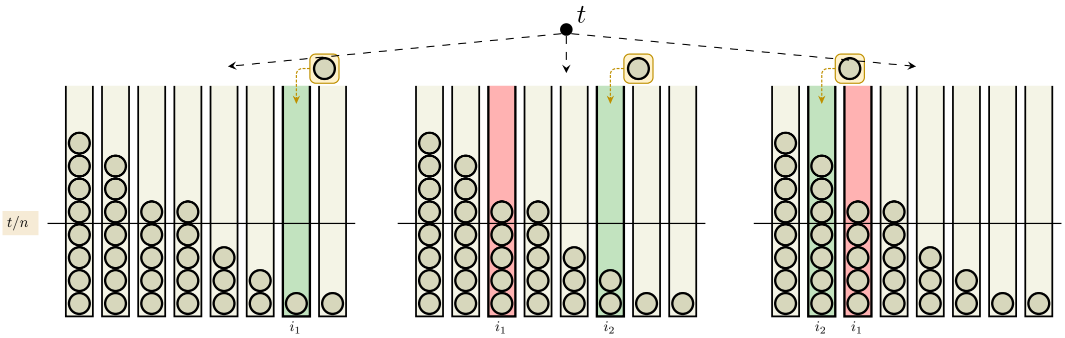


# Mean-Threshold process

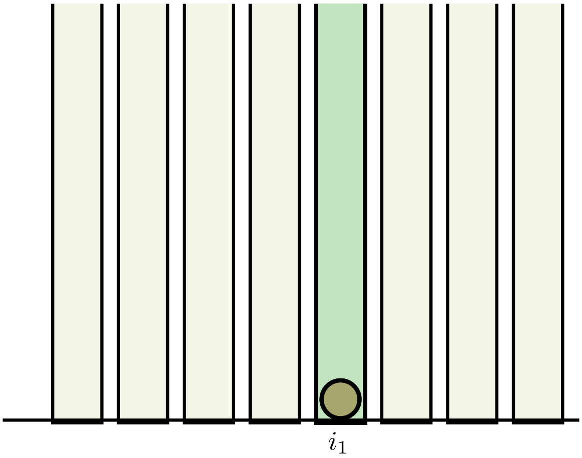
**Mean-Threshold:**

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-Threshold: Visualisation



# MEAN-THRESHOLD: Our Results

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

# Mean-Threshold: Our Results

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

# Mean-Threshold: Our Results

- For all $m \geq n$, Mean-Threshold achieves w.h.p. $\mathrm{Gap}(m) = \mathcal{O}(\log n)$
- For sufficiently large $m$, w.h.p. $\mathrm{Gap}(m) = \Omega(\log n)$
- The following implementation of Mean-Threshold 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-Threshold: Our Results

- For all $m \geq n$, Mean-Threshold 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-Threshold 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-Threshold: 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-Threshold: Why the analysis is tricky

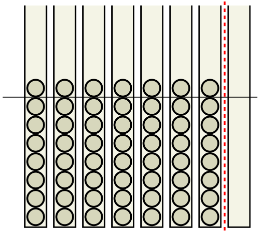
- Let $\delta^t$ be mean quantile, i.e., the fraction of bins which are overloaded (i.e., $x^t \geq t/n$)
- If $\delta^t$ is very large, say $\delta^t = 1 - 1/n$, then $p$ is very close to the One-Choice vector:

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

# Mean-Threshold: 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:

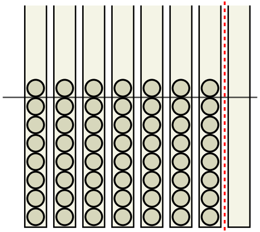
$$p_{\text{Mean-Threshold}}(x^t) = \Big(\underbrace{\frac{1}{n} - \frac{1}{n^2}, \ldots, \frac{1}{n} - \frac{1}{n^2}}_{(n-1)\text{ entries}}, \frac{2}{n} - \frac{1}{n^2}\Big).$$



# Mean-Threshold: 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:

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



What happens to the exponential potential function $\Gamma^t$?

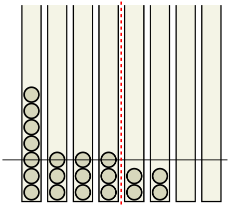
# A Closer Look at $\Gamma^t$

- Similar to the exponential potential analysis before:

# A Closer Look at $\Gamma^t$

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

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



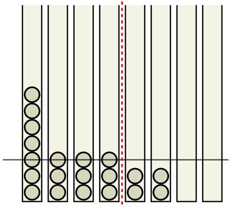
# A Closer Look at $\Gamma^t$

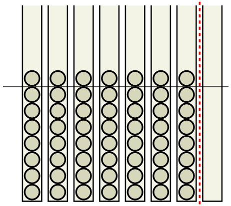
- Similar to the exponential potential analysis before:
  - (**Good step**) If $\delta^t \in (\epsilon, 1-\epsilon)$ for const $\epsilon > 0$, there are constants $c_1, c_2 > 0$

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

  - (**Bad step**) If $\delta^t \notin (\epsilon, 1-\epsilon)$, then

$$\mathbf{E}[\,\Gamma^{t+1} \mid \Gamma^t\,] \leq \left(1 + \frac{c_1 \alpha^2}{n}\right) \cdot \Gamma^t + c_2.$$





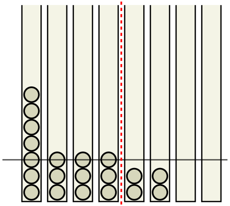
# A Closer Look at $\Gamma^t$

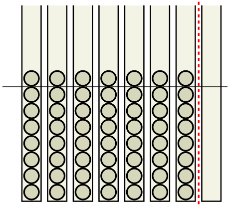
- Similar to the exponential potential analysis before:
  - (**Good step**) If $\delta^t \in (\epsilon, 1-\epsilon)$ for const $\epsilon > 0$, there are constants $c_1, c_2 > 0$

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

  - (**Bad step**) If $\delta^t \notin (\epsilon, 1-\epsilon)$, then

$$\mathbf{E}[\,\Gamma^{t+1} \mid \Gamma^t\,] \leq \left(1 + \frac{c_1 \alpha^2}{n}\right) \cdot \Gamma^t + c_2.$$





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

# Mean Quantile Stabilisation

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

# Mean Quantile Stabilisation

**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 Stabilisation

**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 a linear potential function,

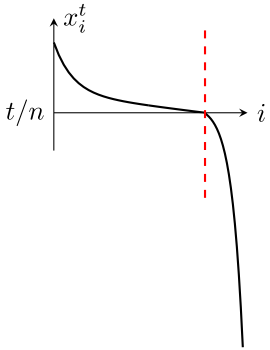
$$\Delta^t := \sum_{i=1}^{n} \left| x_i^t - \frac{t}{n} \right|.$$

# Mean Quantile Stabilisation

**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 a linear potential function,

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

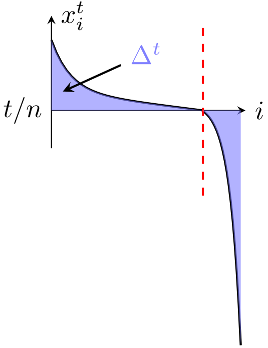


# Mean Quantile Stabilisation

**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 a linear potential function,

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



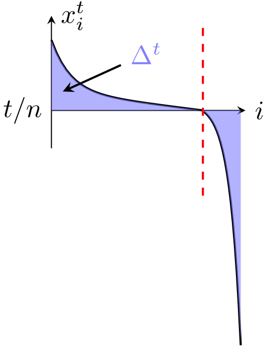
# Mean Quantile Stabilisation

**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 a linear potential function,

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

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

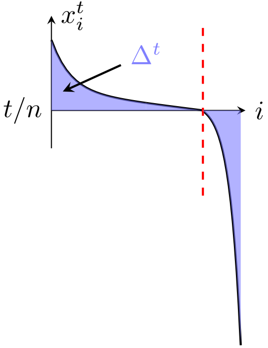


# Mean Quantile Stabilisation

**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 a linear 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 $[-2c, +2c]$

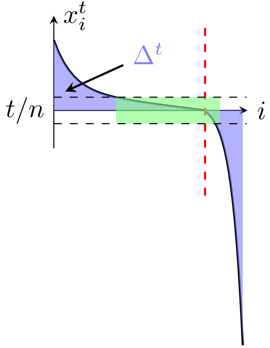
# Mean Quantile Stabilisation

**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 a linear 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 $[-2c, +2c]$

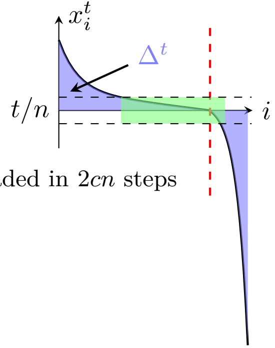


# Mean Quantile Stabilisation

**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 a linear potential function,

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



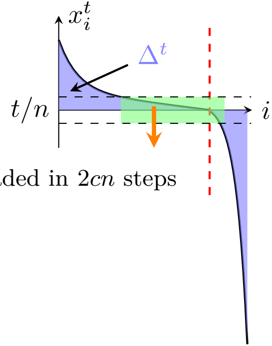

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps

# Mean Quantile Stabilisation

**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 a linear potential function,

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



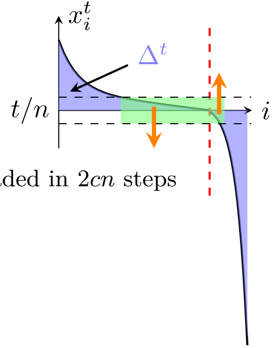

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps

# Mean Quantile Stabilisation

**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 a linear potential function,

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



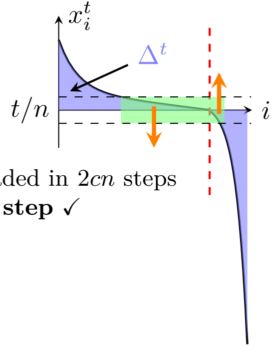

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps

# Mean Quantile Stabilisation

**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 a linear potential function,

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



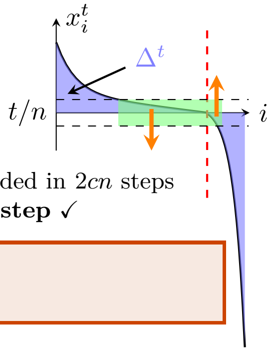

- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps
  - After $2cn$ steps, enough overloaded and underloaded bins $\rightsquigarrow$ **good step** ✓

# Mean Quantile Stabilisation

## 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 a linear 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** ✓

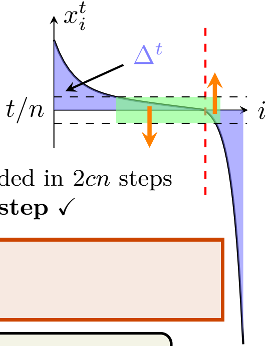
## How to prove that $\Delta^t$ is small?

# Mean Quantile Stabilisation

## 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 a linear potential function,

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

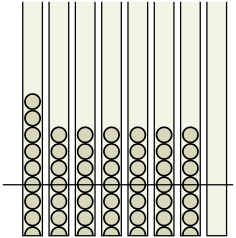



- If $\Delta^t \leq c \cdot n$, then:
  - At least $n/2$ bins have (normalised) load between $[-2c, +2c]$
  - Constant prob. for such bin to switch between overloaded/underloaded in $2cn$ steps
  - After $2cn$ steps, enough overloaded and underloaded bins $\rightsquigarrow$ **good step** ✓

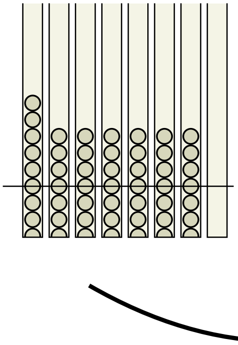
## How to prove that $\Delta^t$ is small?

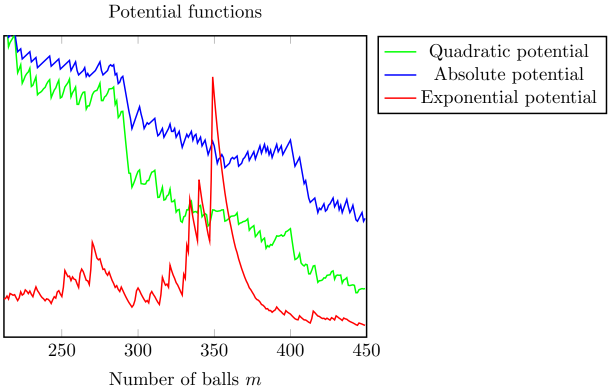
$-\Delta^t$ is essentially the gradient of the quadratic potential $\Upsilon^t = \sum_{i=1}^n \left(x_i^t - \frac{t}{n}\right)^2$.

# Recovery from a bad configuration

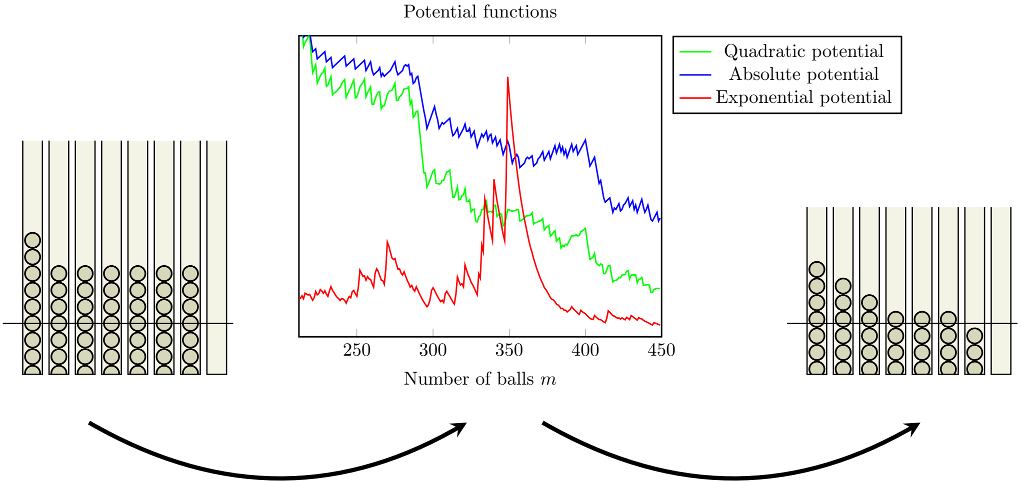


# Recovery from a bad configuration



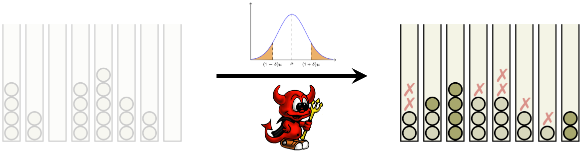


# Recovery from a bad configuration

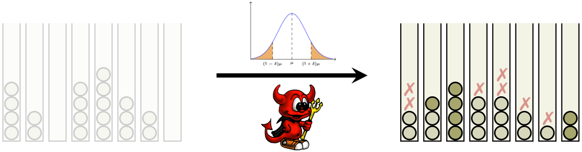


# Noisy Comparisons

# Two Choice with Noise: Model Overview



# Two Choice with Noise: Model Overview

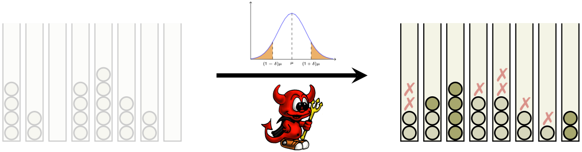



**Two-Choice** with Noise Framework

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. Place ball in the bin with smaller load estimate

# Two Choice with Noise: Model Overview



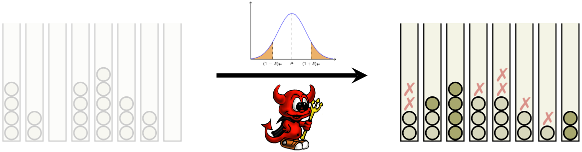
**Two-Choice with Noise Framework**

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. Place ball in the bin with smaller load estimate

Load Estimates could be...

# Two Choice with Noise: Model Overview



**Two-Choice with Noise Framework**

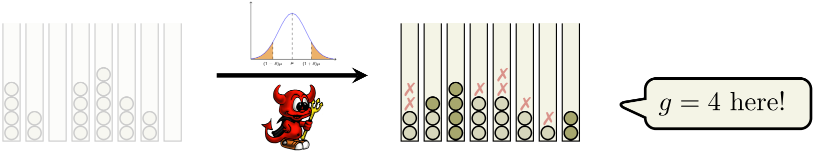
**Iteration:** For each $t \geq 0$:

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. Place ball in 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: Model Overview




**Two-Choice** with Noise Framework

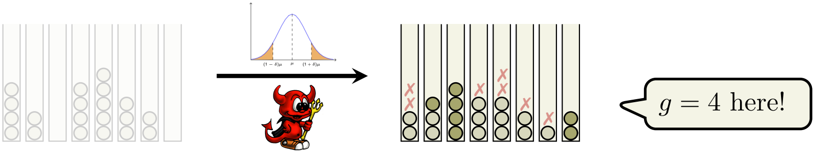
**Iteration:** For each $t \geq 0$:

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. Place ball in 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: Model Overview




**Two-Choice with Noise Framework**

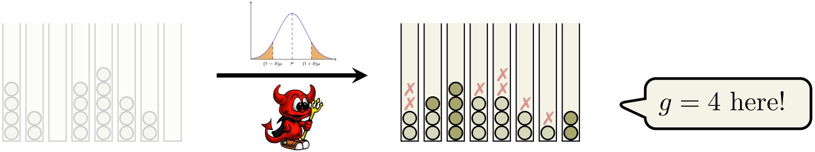
**Iteration:** For each $t \geq 0$:

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. Place ball in 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: Model Overview




**Two-Choice with Noise Framework**

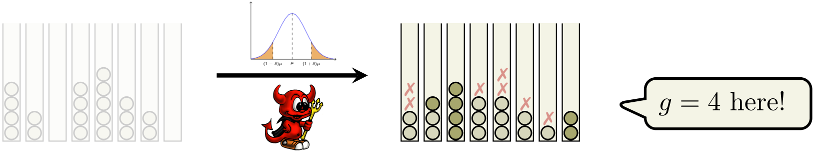
Iteration: For each $t \geq 0$:

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. Place ball in 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 the exact loads $\rightsquigarrow \widetilde{x}_i^t \in [x_i^{t-T}, x_i^t]$

# Two Choice with Noise: Model Overview




**Two-Choice with Noise Framework**

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. Place ball in 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 the 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 $|x_i^t - x_j^t| > g$, place ball in the lesser loaded bin,
3. Otherwise, place ball in the higher loaded bin.

# Comparison-Based Model

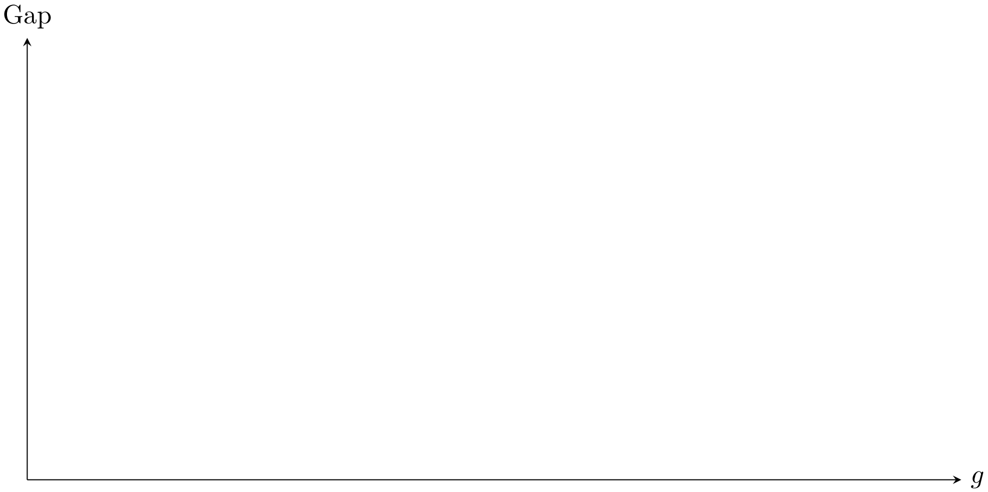
Let us assume that the adversary can **directly influence comparisons**:

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

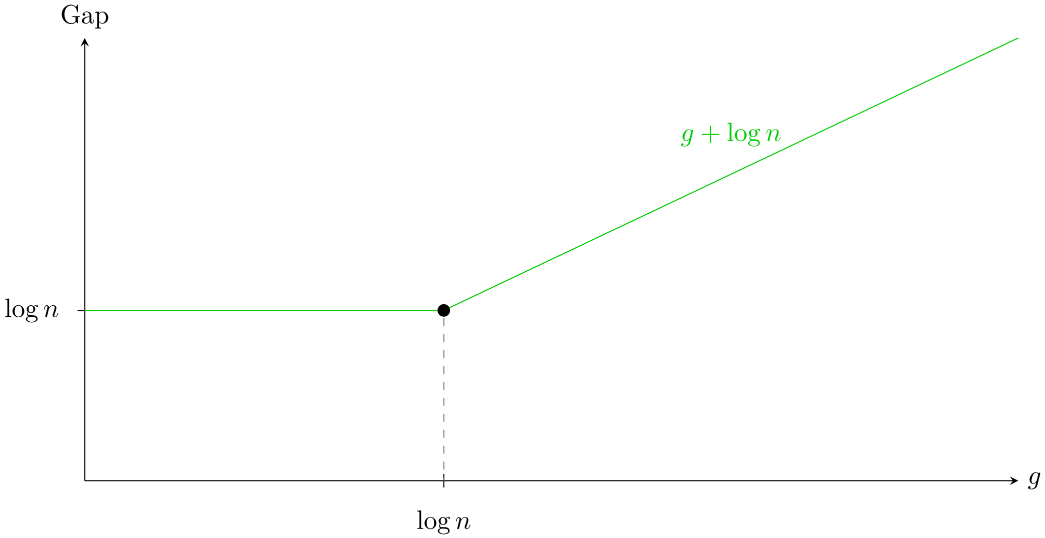
1. Sample two bins $i, j$ independently and uniformly at random
2. If $|x_i^t - x_j^t| > g$, place ball in the lesser loaded bin,
3. Otherwise, place ball in the higher loaded bin.

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

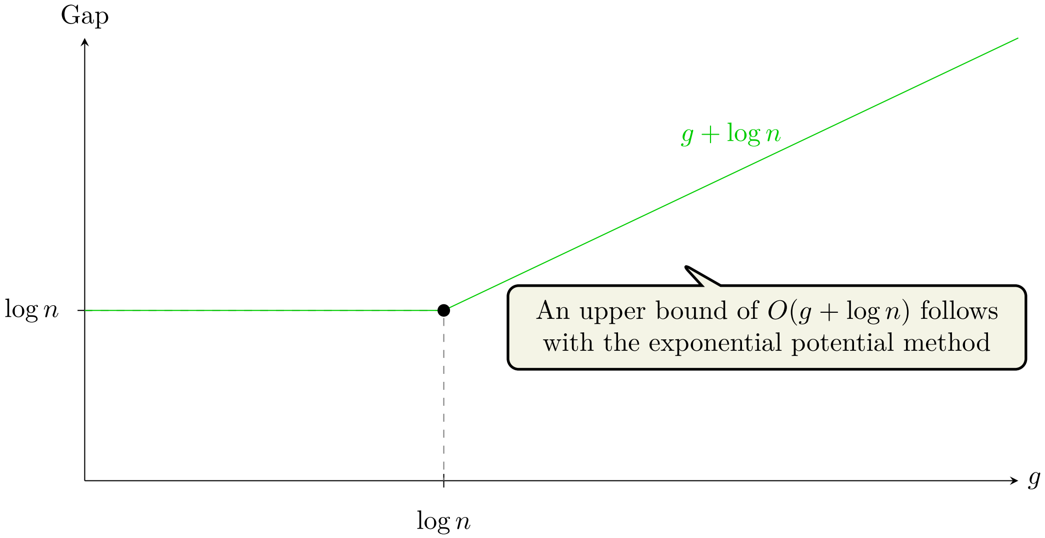
# Asymptotic Results for $g \leq \log n$



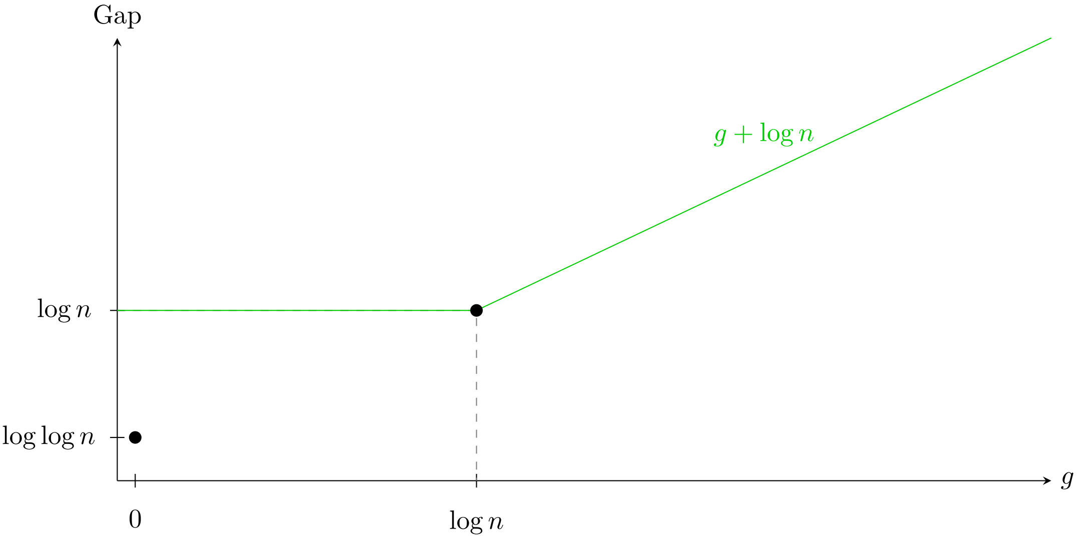
# Asymptotic Results for $g \leq \log n$



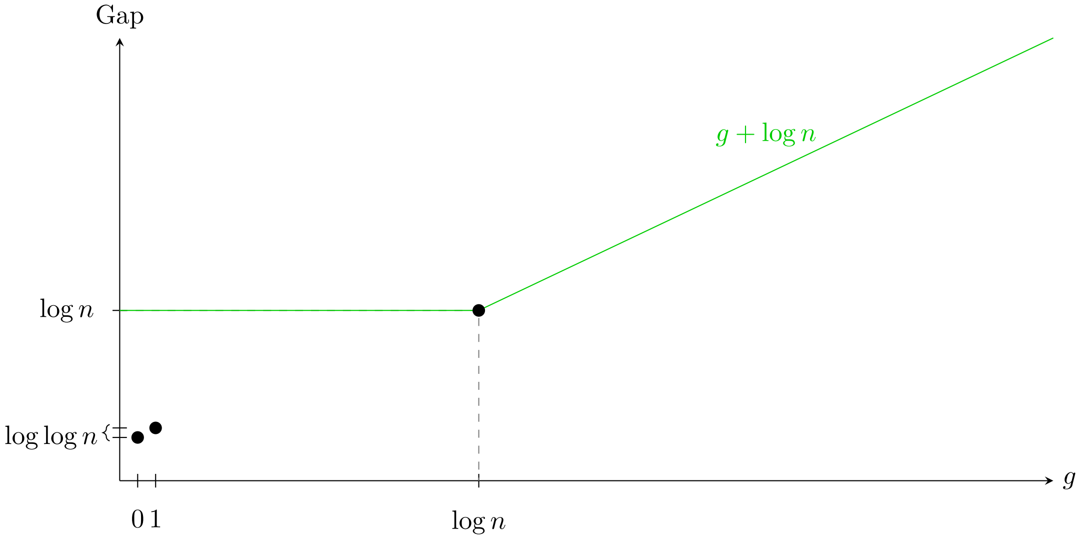
# Asymptotic Results for $g \leq \log n$



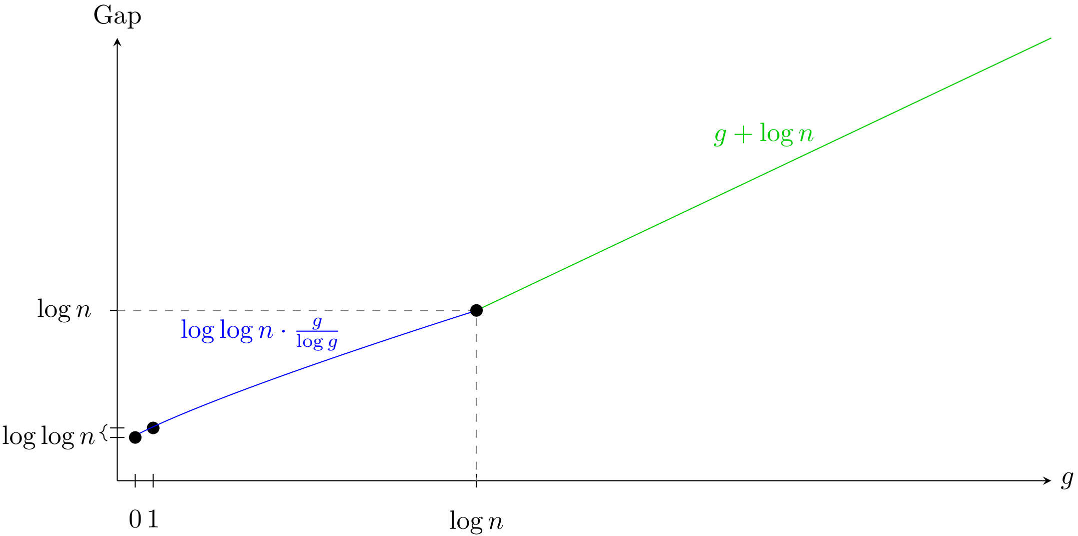
# Asymptotic Results for $g \leq \log n$



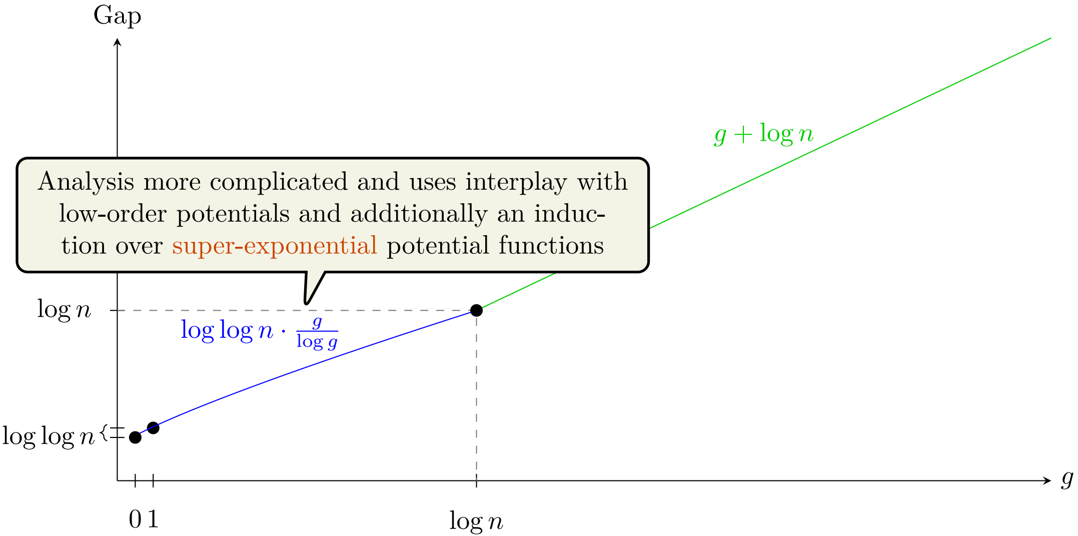
# Asymptotic Results for $g \le \log n$



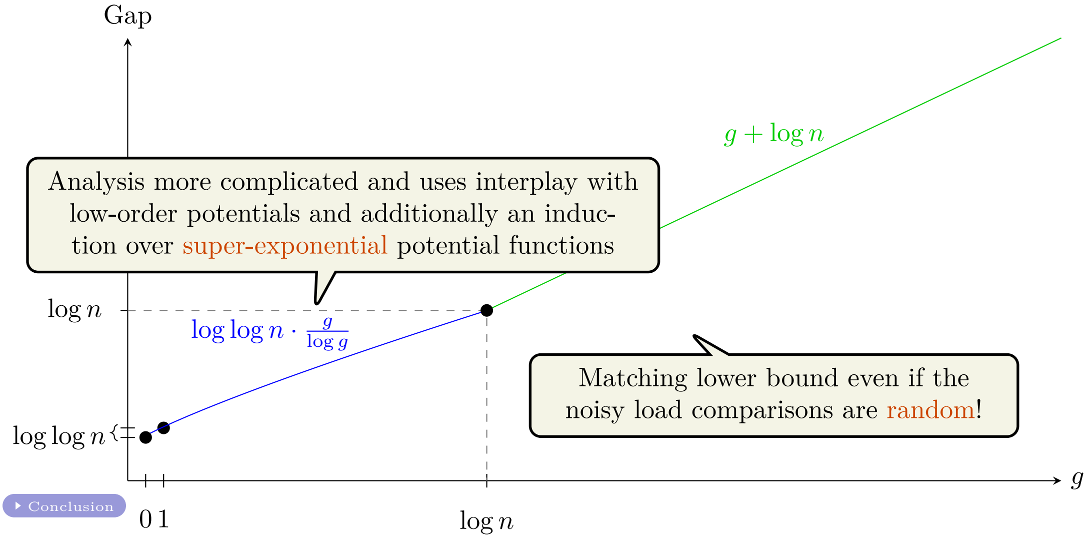
# Asymptotic Results for $g \leq \log n$



# Asymptotic Results for $g \leq \log n$



# Asymptotic Results for $g \leq \log n$




▸ 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. Place ball in the bin with smaller load estimate.

# Delay Models (The Problem of Choices)

$T$-Delay

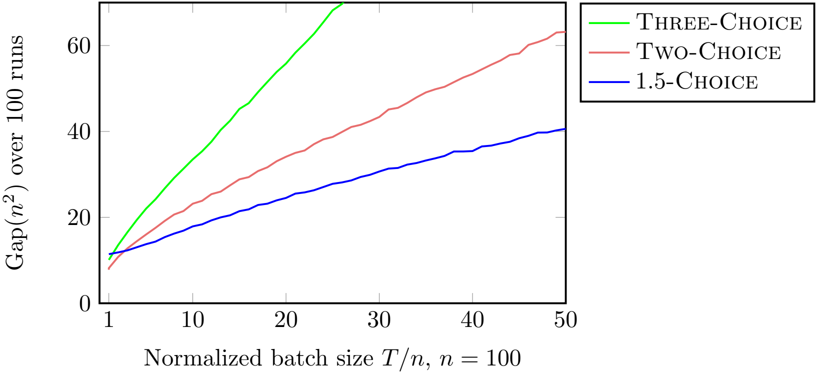
Parameter: Integer $T \geq 1$

Iteration: For each $t \geq 0$:

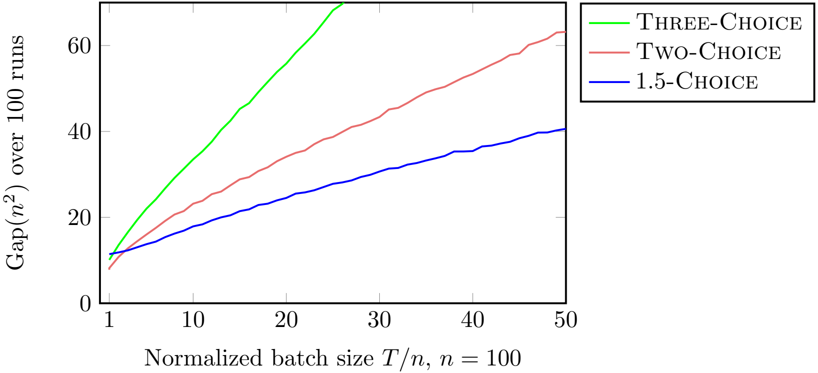
1. Sample two bins $i, j$ independently and uniformly at random
2. Receive load estimates $\widetilde{x}_i^t \in [x_i^{t-T}, x_i^t]$ and $\widetilde{x}_i^t \in [x_j^{t-T}, x_j^t]$.
3. Place ball in the bin with smaller load estimate.

Batching model: Load values are updated every $T$ steps

# Experiments with Batching



# Experiments with Batching

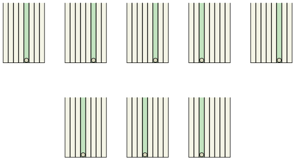



- Two-Choice (and Three-Choice) have a too strong bias towards the bins that are lightly loaded at the beginning
- With outdated information, more "unbiased" approaches like 1.5-Choice better

# 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)



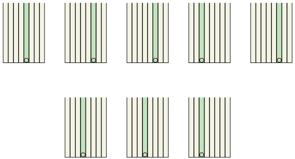
# Conclusion

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 threshold strategies than Mean-Threshold?
- Better understanding of the impact of too many choices in noisy settings (known as "Herd Phenomenon")
- more "realistic" noise models...



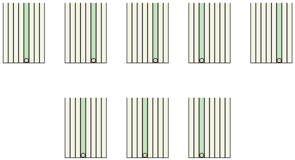
# Conclusion

## 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 threshold strategies than Mean-Threshold?
- Better understanding of the impact of too many choices in noisy settings (known as "Herd Phenomenon")
- more "realistic" noise models...



More visualisations: `tinyurl.com/lss21-visualisations`
(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, 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*, 2021, arXiv:2110.05009.
- 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.