# Online Ranking with Minimal Inversions

**Sepehr Assadi (Rutgers University), Eric Balkanski (Harvard University), Renato Paes Leme (Google)**

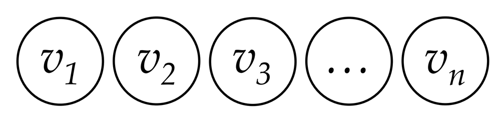
## Main Question

*What guarantees are achievable in online settings for ranking elements from pairwise comparisons?*

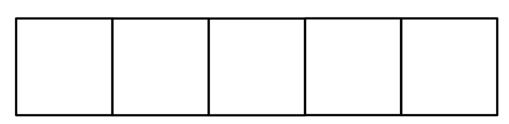
## The Secretary Ranking Problem

Generalizes the **secretary problem**:

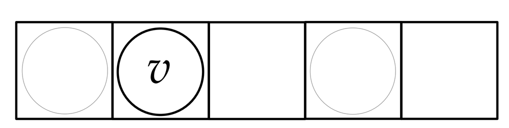
- Elements with values $v_1, \ldots, v_n$ arrive in random order




- $n$ **positions** $1, 2, 3, \ldots, n$



- Upon arrival of an element $a$:
  1. Observe its **value** $v$
  2. **Irrevocably place** $a$ **in a position** $p$ **not yet occupied**




Objective: minimize number of inversions, i.e., pairs $(a_i, a_j)$ of elements in positions s.t. $p_i < p_j$ and $v_i > v_j$.

## Trivial Bounds

Upper bound: place each element in a random available position, $O(n^2)$ inversions.

Lower bound: no information available when first element arrives, $\Omega(n)$ inversions.

## Scenarios with irrevocable online ranking decisions

- **Task assignment**, e.g., a consulting firm with teams of different skill levels and projects of different difficulty that arrive in an online fashion.
- **Reward allocation**, e.g., a university department wants to reward the best students with the best scholarships.

## Main Result

**Tight upper and lower bounds for the secretary ranking problem.**

***Theorem:***
- *There is an algorithm which obtains, w.h.p.,* $\boldsymbol{O(n^{3/2})}$ *inversions.*
- *Any algorithm obtains, w.h.p.,* $\boldsymbol{\Omega(n^{3/2})}$ *inversions.*

Main challenges
- **Algorithm**:
  - In earlier time steps, not much information
  - In later time steps, many positions are not available
- **Hardness**:
  - Requires anti-concentration inequality for correlated random variables

## The Algorithm

Upon arrival of element $a_t$ at time step $t$:

- **Estimate the true rank** of $a_t$ by rescaling its rank with respect to previous $t$ - $1$ elements and by adding some random perturbation:
  - Define $r_t = |\{a_{t'} \mid a_{t'} < a_t \text{ and } t' < t\}|$
  - Estimated rank $\tilde{\text{rk}}(a_t)$ is chosen uniformly at random in $[r_t\,(n/t), (r_t + 1)\,(n/t)]$
- **Assign** $a_t$ to the position that is the **nearest unassigned position** to estimated rank $\tilde{\text{rk}}(a_t)$:
  - Learned rank is $\pi(a_t) = \text{argmin}_{i \in R} \, |i - \tilde{\text{rk}}(a_t)|$ where $R$ is the set of available positions

Connection to linear probing in hashing:
- **Linear probing** is used to resolve collisions in hashing:
  - When a key is hashed to a non-empty cell, visit neighboring cells until empty location is found.
- Analysis of the assignment step follows similar ideas as the analysis for the linear probing hashing scheme.

## Hardness

Main ingredient: **new anti-concentration inequality** for a generic **balls in bins problem** with **correlated sampling**.

Congested ranking: two elements can be assigned the same position.

Overview of analysis:
1. **Congested** ranking is **easier** than ranking
2. The algorithm that assigns each element to its estimated rank is the **optimal algorithm**
3. Using anti-concentration bounds, this algorithm incurs $\boldsymbol{\Omega(n^{3/2})}$ **inversaions**

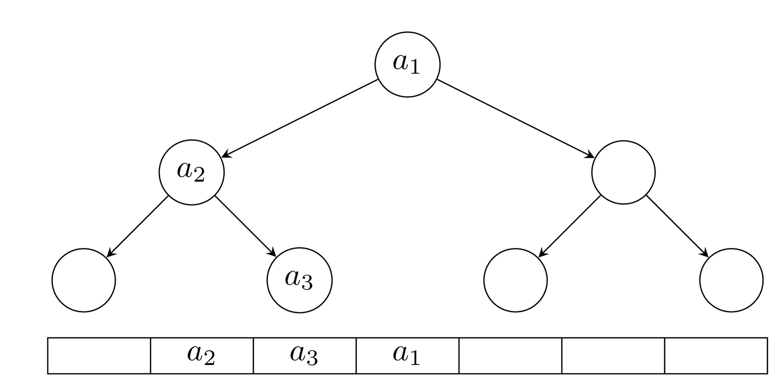
**Lemma** (anti-concentration bound)**:** Assume there are $n$ balls in a bin, $r$ red and $n$ - $r$ blue. If $t$ balls are drawn uniformly at random without replacement, for any $k$, the probability that $k$ out of these $t$ balls are red is $O(1/\sqrt{n})$.

## Extensions

Let $m$ be the number of positions and $n$ the number of elements, we consider $m \geq n$ (instead of $m = n$).

**Sparse secretary problem**:
- *How large does* $m$ *need to be to achieve* $0$ *inversions?*

Connection to **random binary trees**.




**General secretary problem**:
- *How many inversions when* $m \geq n$*?*