Improving Community-Participated Patrol for Anti-Poaching






Yufei Wu
Shanghai Jiao Tong University

Yixuan (Even) Xu
Carnegie Mellon University

Xuming Zhang
World Wide Fund For Nature China

Duo Liu
World Wide Fund For Nature China

Shibing Zhu
Heilongjiang Academy of Sciences

Fei Fang
Carnegie Mellon University

Anti-Poaching

- Security Games
  - A framework in game theory for optimizing resource allocation to protect valuable targets against adversarial threats.
- Anti-poaching
  - resources: rangers
  - targets: wildlife populations
  - adversaries: poachers

Anti-Poaching

- Security Games
  - A framework in game theory for optimizing resource allocation to protect valuable targets against adversarial threats.
- Anti-poaching
  - resources: rangers and **community members**
  - targets: wildlife populations
  - adversaries: poachers

RACPP

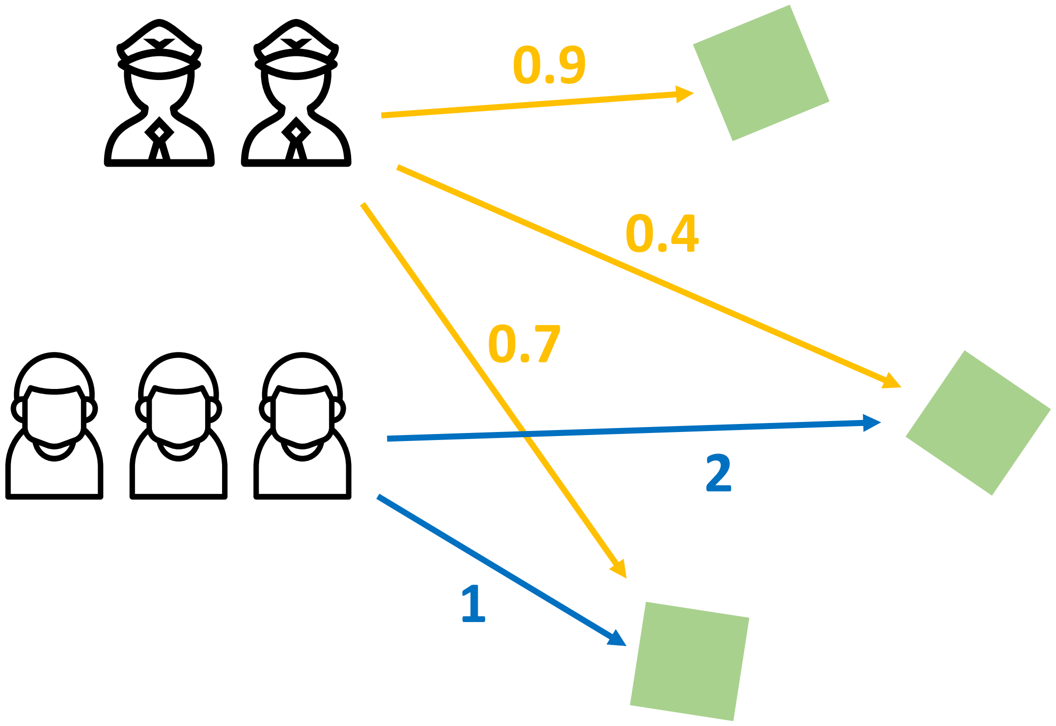
- Resources Allocation of Community Participated Patrol

- Two types of patrol resources
  - Professional rangers
    - distribute efforts among multiple targets
  - Community members (villagers)
    - patrol a single target (less flexible)

RACPP

- Resources Allocation of Community Participated Patrol

- Two types of patrol resources
  - Professional rangers
    - distribute efforts among multiple targets
  - Community members (villagers)
    - patrol a single target (less flexible)



RACPP

- Resources Allocation of Community Participated Patrol

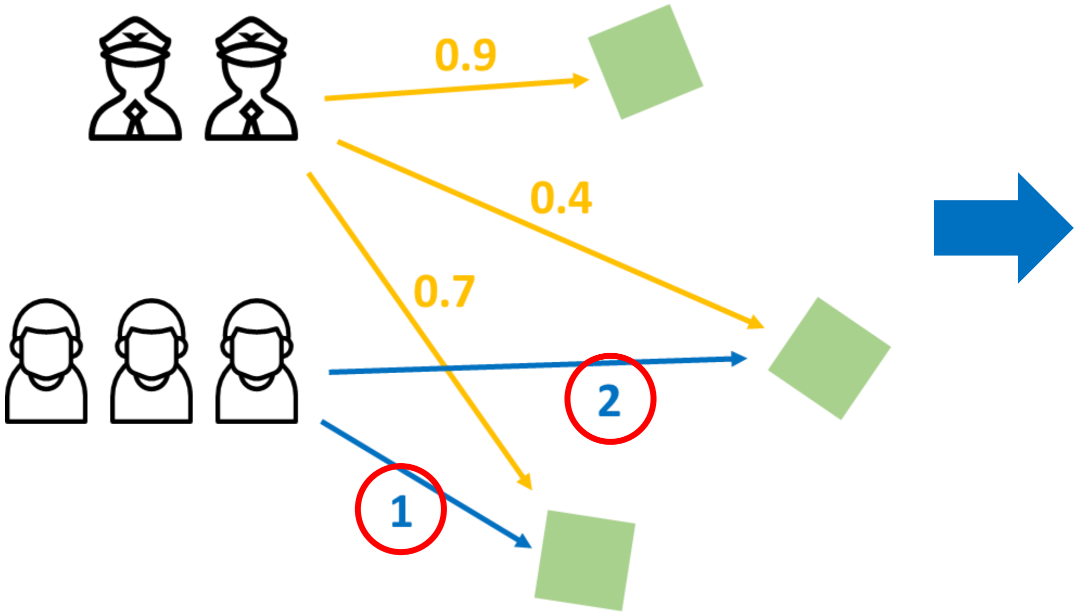
- n targets
- Rewards and penalties
  - If target i is successfully defended
    - defenders receive reward R_i^{d}
    - the attacker receives penalty P_i^{a}
  - Otherwise
    - defenders receive penalty P_i^{d}
    - the attacker receives reward R_i^{a}
- Expected utility of defenders and the attacker
- Goal
  - maximize the defenders' expected utility by adjusting defensive strategy

Mixed-Integer Linear Program Solution

- Stackelberg game
  - Defenders distribute resources
  - An attacker observes the distribution and attacks the target that maximizes his expected utility

Mixed-Integer Linear Program Solution

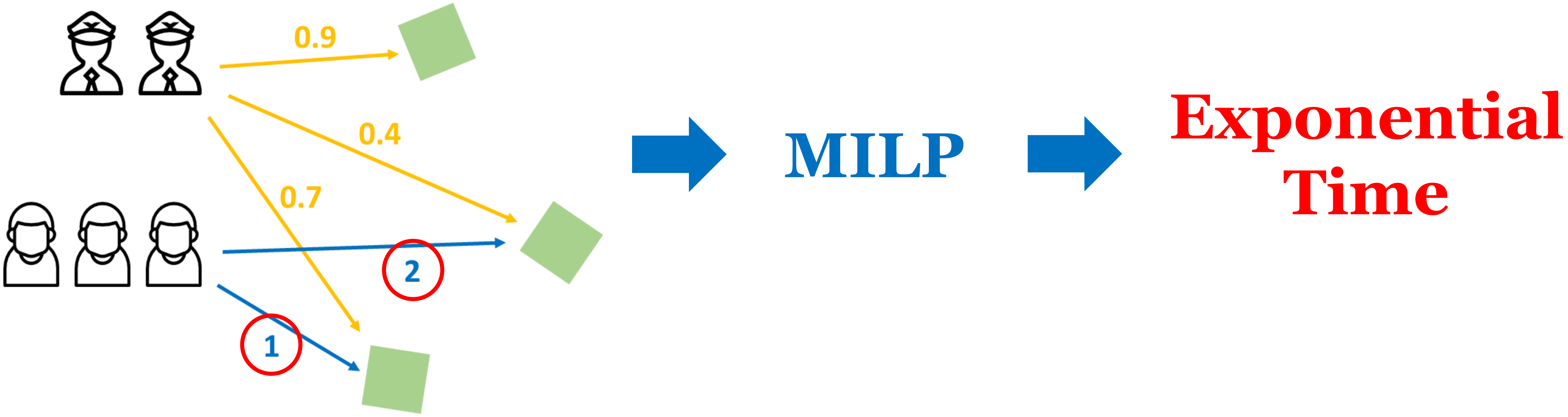
- Stackelberg game
  - Defenders distribute resources
  - An attacker observes the distribution and attacks the target that maximizes his expected utility



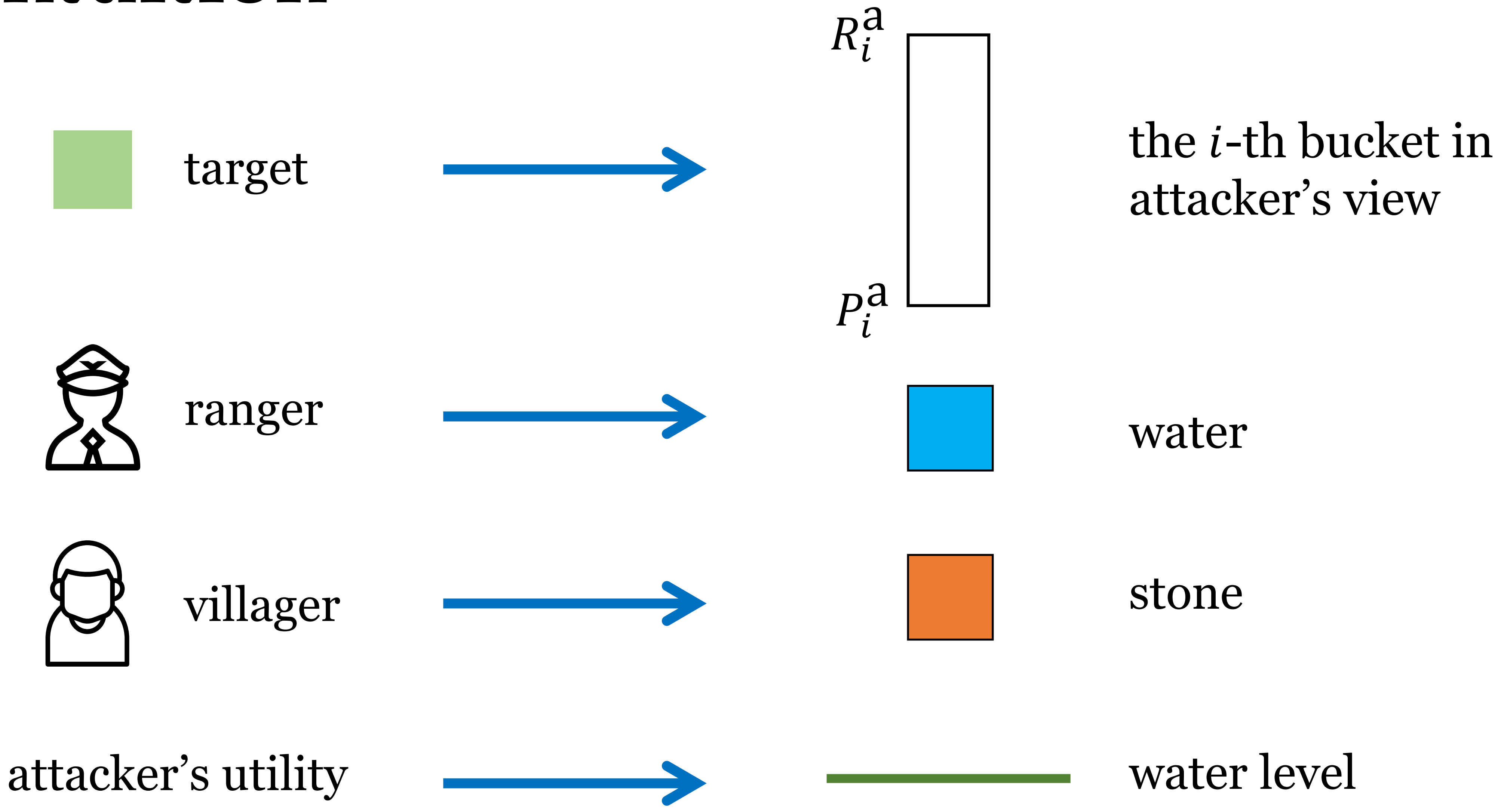
MILP

Mixed-Integer Linear Program Solution

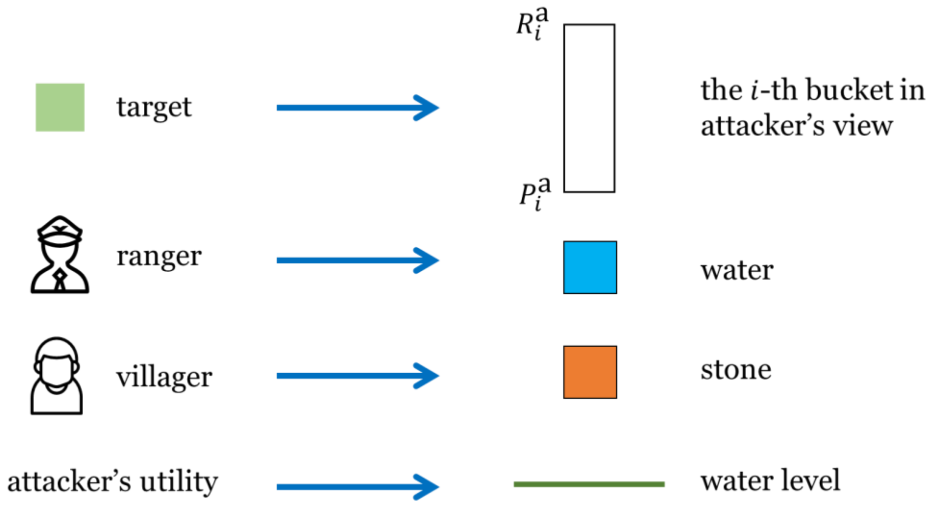
- Stackelberg game
  - Defenders distribute resources
  - An attacker observes the distribution and attacks the target that maximizes his expected utility



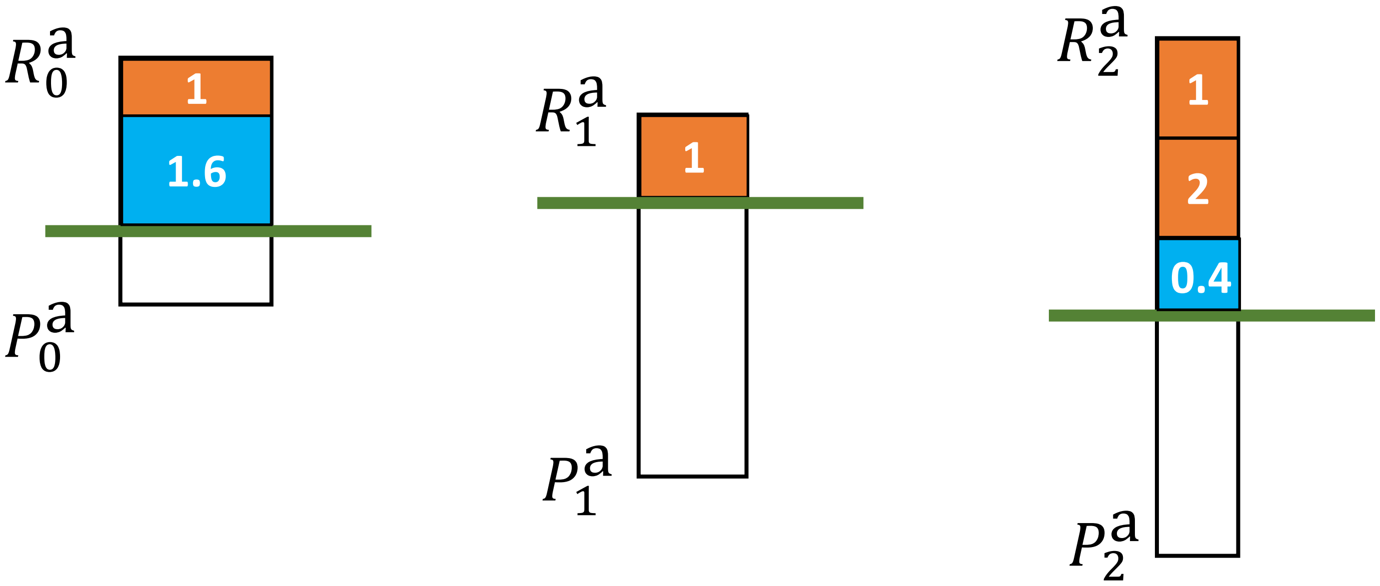
Intuition


R_i^{a}
P_i^{a}
target
the i-th bucket in attacker's view
ranger
water
villager
stone
attacker's utility
water level

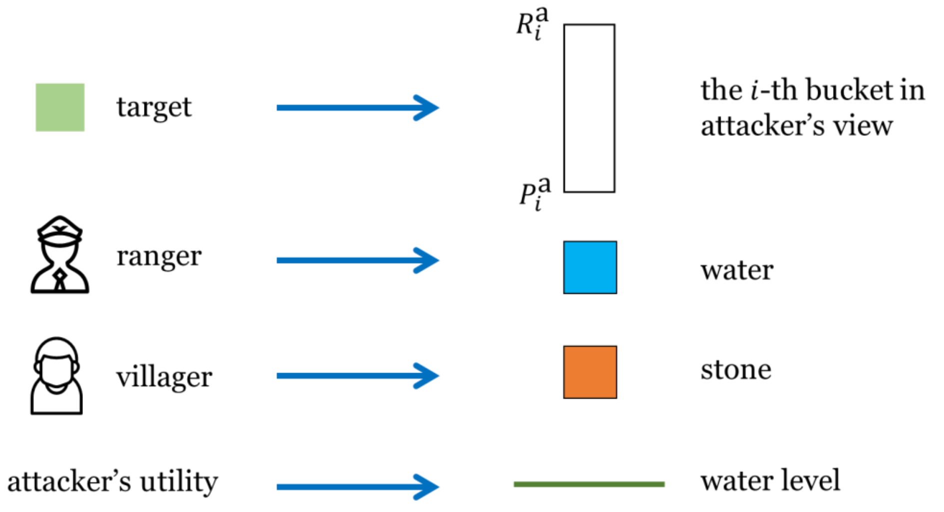
Intuition



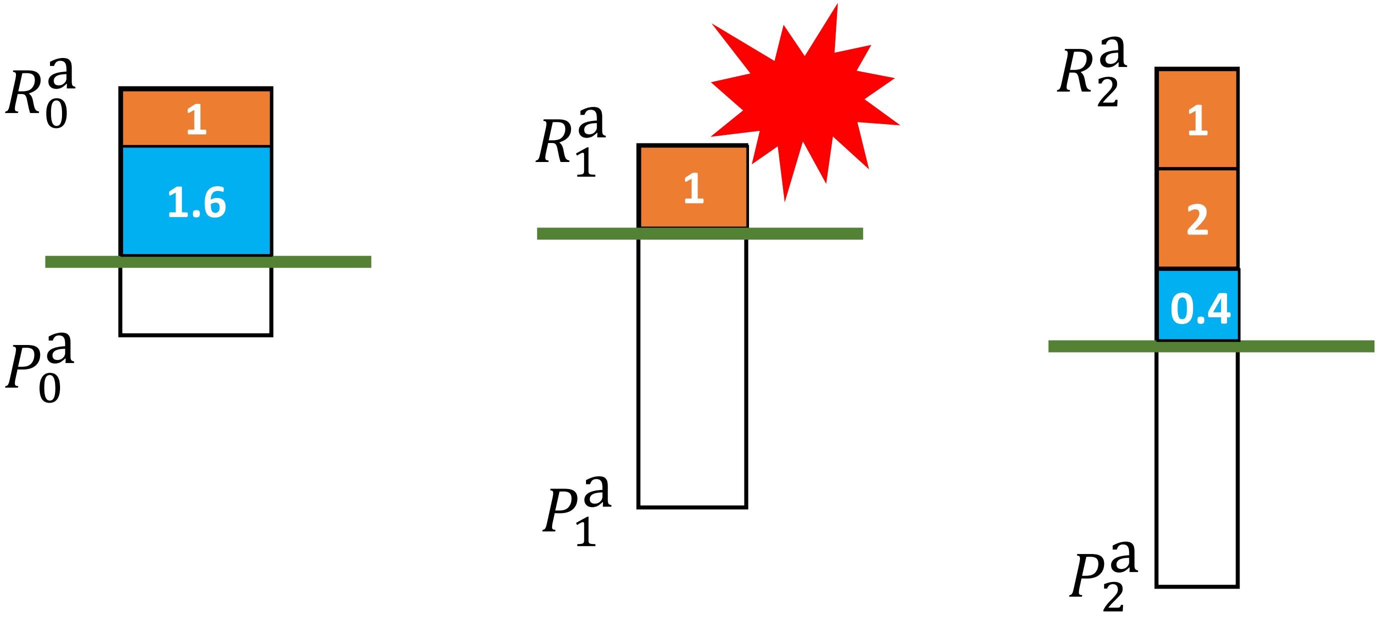
- Stone: thrown into the bucket as a whole
- Water: poured into the bucket at will
- Attacker attacks the bucket i^* with the highest water level
- Goal: Adjust resources to lower the water level of bucket i^*



Intuition

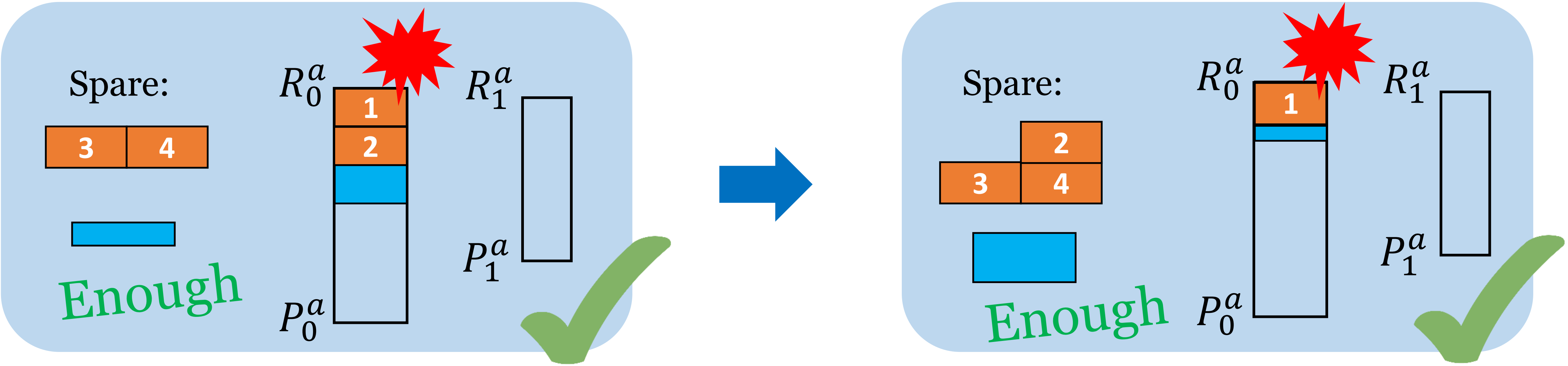


- Stone: thrown into the bucket as a whole
- Water: poured into the bucket at will
- Attacker attacks the bucket i^* with the highest water level
- Goal: Adjust resources to lower the water level of bucket i^*



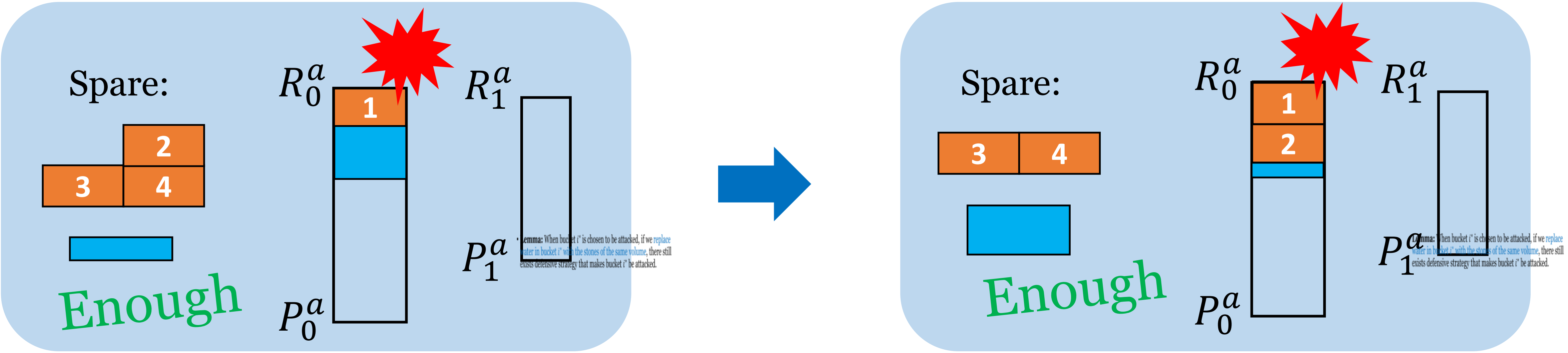
Monotonicity

- **Lemma:** When bucket i^* is chosen to be attacked, if we put fewer stones and less water in bucket i^*, there still exists defensive strategy that makes bucket i^* be attacked.



Monotonicity

- **Lemma:** When bucket i^* is chosen to be attacked, if we replace water in bucket i^* with the stones of the same volume, there still exists defensive strategy that makes bucket i^* be attacked.



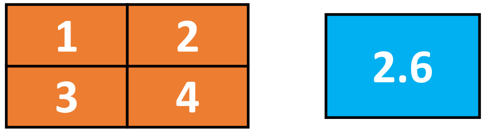
Two-Dimensional Binary Search

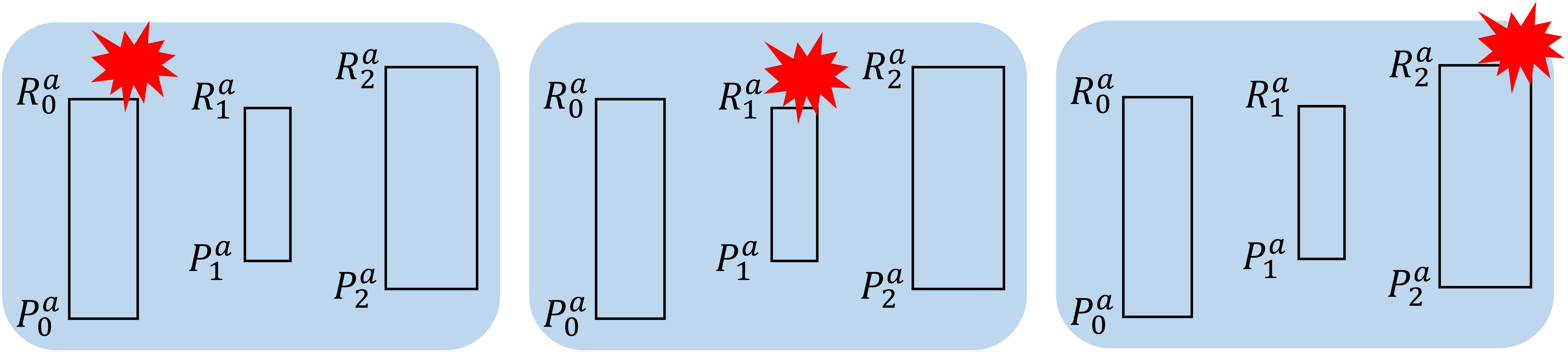
- Polynomial approximate algorithm
  - Accuracy: any desired ε
  - Complexity: $O(n^2 \log \frac{M}{\varepsilon})$
    - M is the maximum absolute reward or penalty

Two-Dimensional Binary Search

1. Iterate over all buckets as the attacked bucket i^*
   - As the one with the highest water level

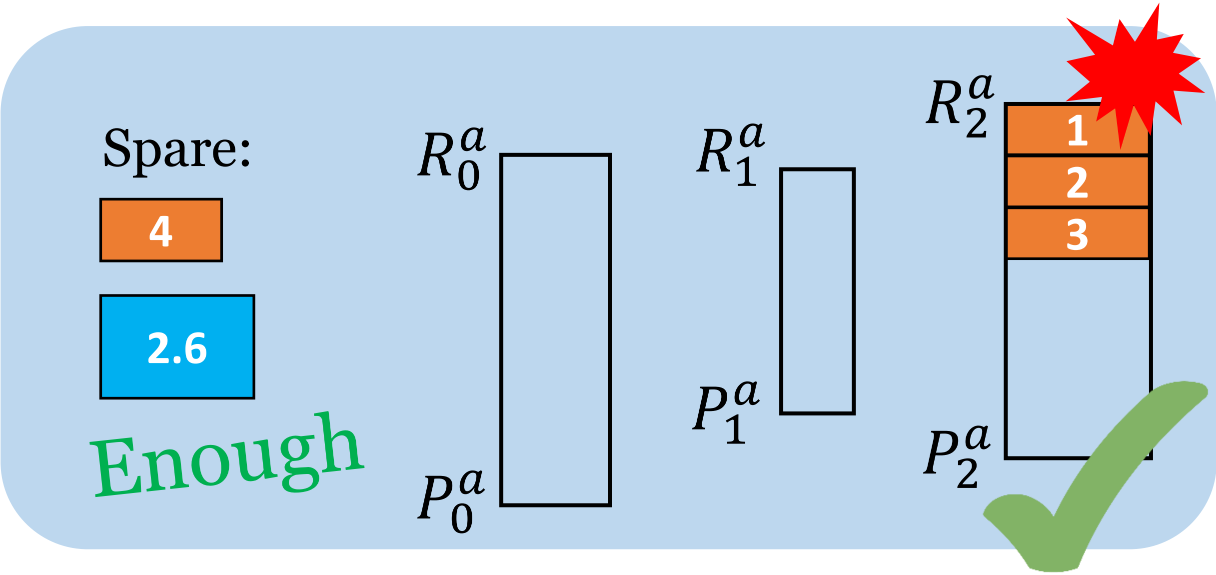
Resources:

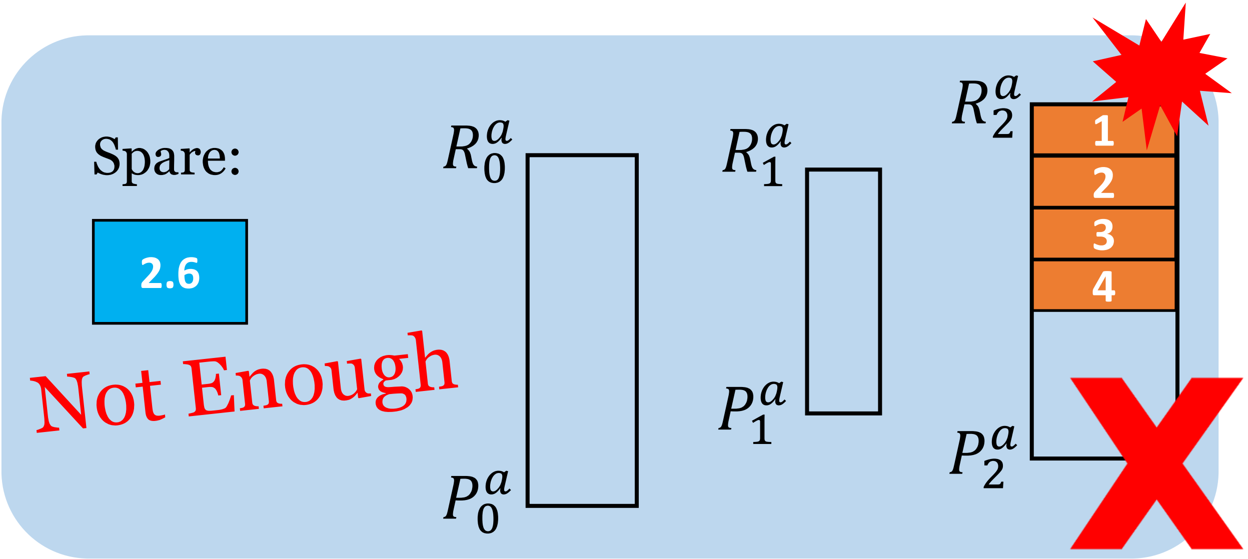




Two-Dimensional Binary Search

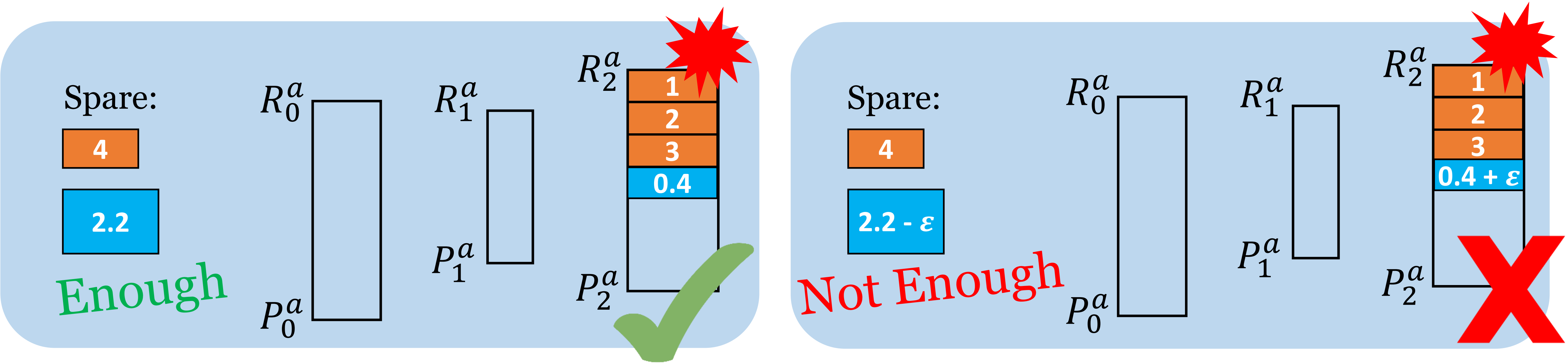
1. Iterate over all buckets as the attacked bucket i^*
2. Binary search on the max number of stones thrown into i^*
   - With enough spare resources for a defensive strategy to allow attacking bucket i^*





Two-Dimensional Binary Search

1. Iterate over all buckets as the attacked bucket i^*
2. Binary search on the max number of stones thrown into i^*
3. Binary search on the amount of water poured into i^*
   - To the desired accuracy ε
   - With enough spare resources for a defensive strategy to allow attacking bucket i^*



Two-Dimensional Binary Search

1. Iterate over all buckets as the attacked bucket i^*
   - As the one with the highest water level
2. Binary search on the max number of stones thrown into i^*
   - With enough spare resources for a defensive strategy to allow attacking bucket i^*
3. Binary search on the amount of water poured into i^*
   - To the desired accuracy ε
   - With enough spare resources for a defensive strategy to allow attacking bucket i^*

Complexity: $O(n^2 \log \frac{M}{\varepsilon})$

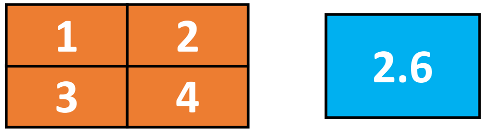
Hybrid Waterfilling

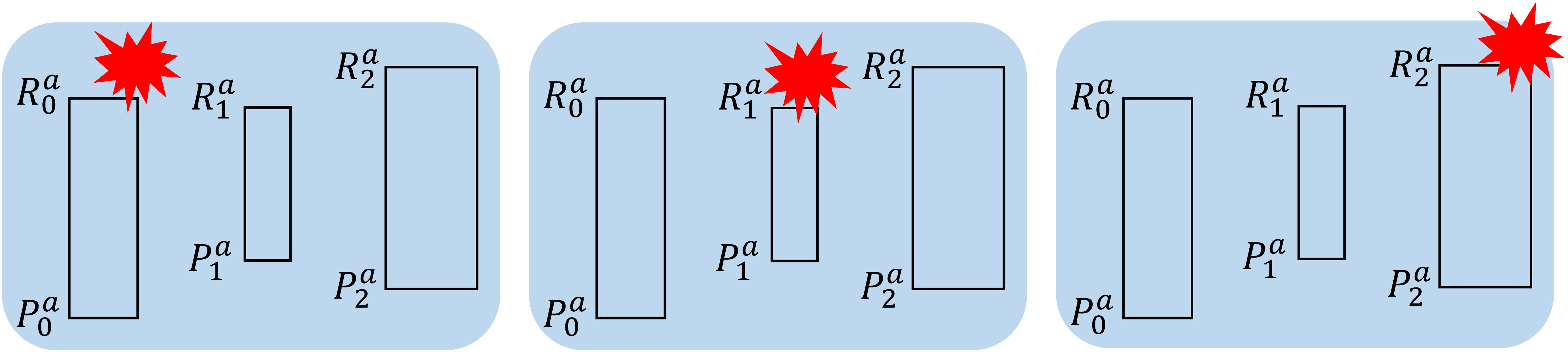
- Polynomial exact algorithm
  - Accuracy: precise
  - Complexity: $O(n^4 \log n)$

Hybrid Waterfilling

1. Iterate over all buckets as the attacked bucket i^*
    - As the one with the highest water level

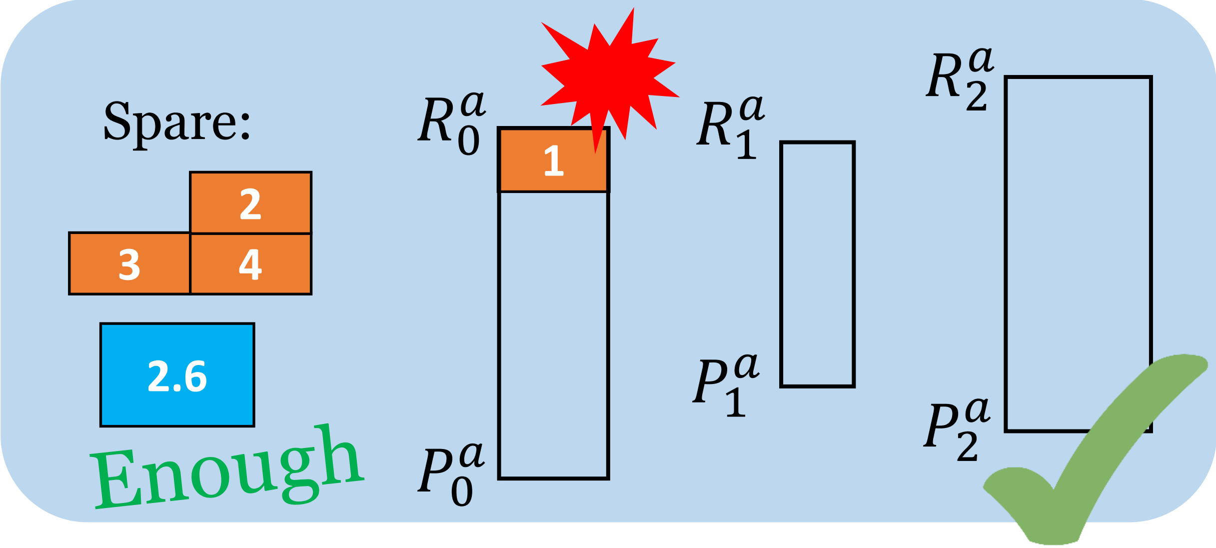
Resources:

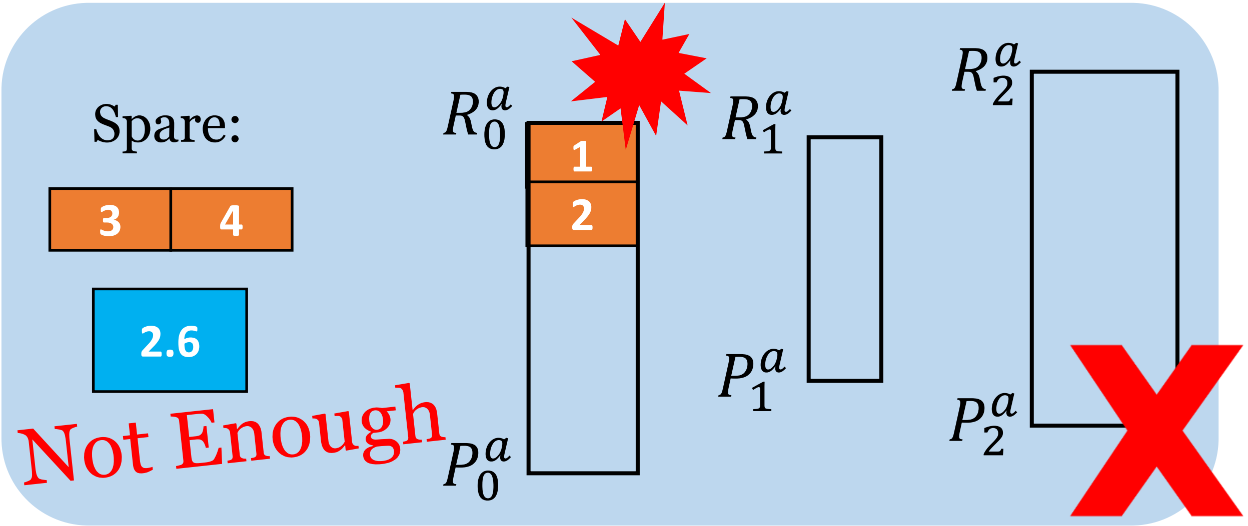




Hybrid Waterfilling

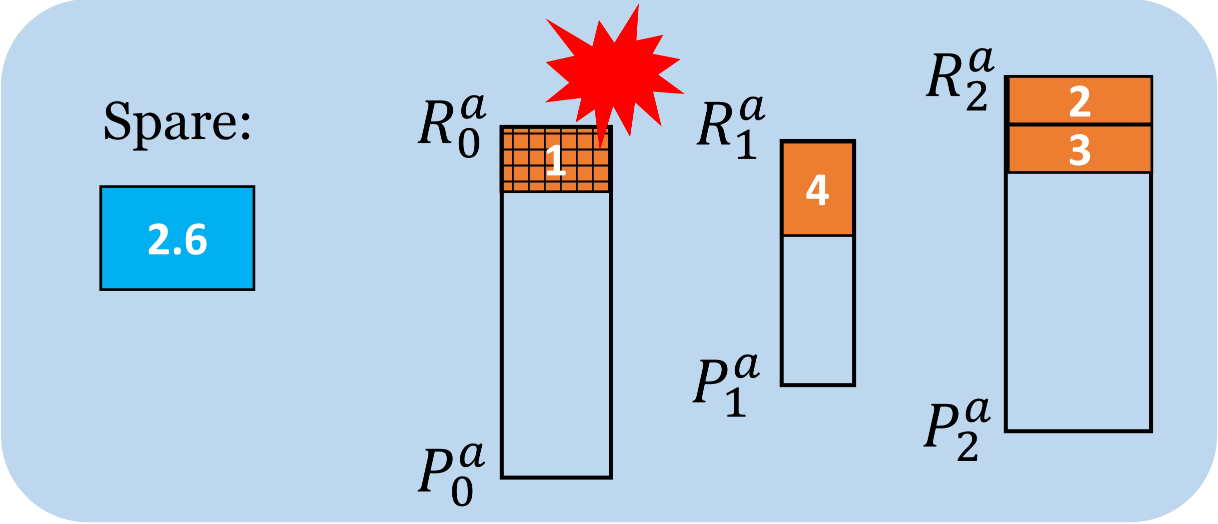
1. Iterate over all buckets as the attacked bucket i^*
2. Binary search on the max number of stones thrown into i^*
   - With enough spare resources for a defensive strategy to allow attacking bucket i^*





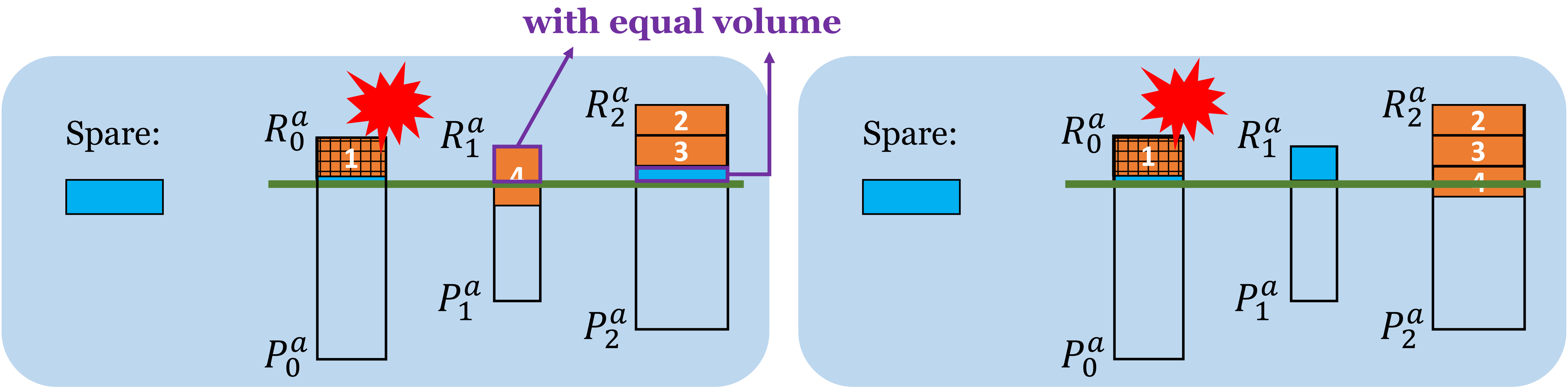
Hybrid Waterfilling

1. Iterate over all buckets as the attacked bucket i^*
2. Binary search on the max number of stones thrown into i^*
3. Greedy for spare stones
    - Throw into the bucket with the highest water level except bucket i^*



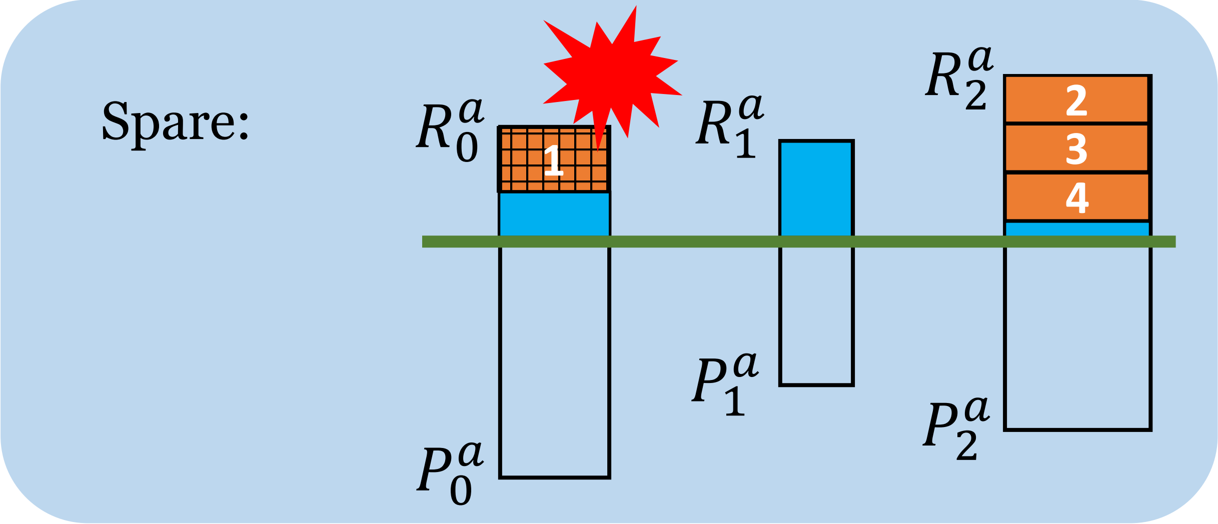
Hybrid Waterfilling

1. Iterate over all buckets as the attacked bucket i^*
2. Binary search on the max number of stones thrown into i^*
3. Greedy for spare stones
4. Waterfilling to a critical point and trigger a swap



Hybrid Waterfilling

1. Iterate over all buckets as the attacked bucket i^*
2. Binary search on the max number of stones thrown into i^*
3. Greedy for spare stones
4. Waterfilling to a critical point and trigger a swap
5. Finish Waterfilling



Hybrid Waterfilling

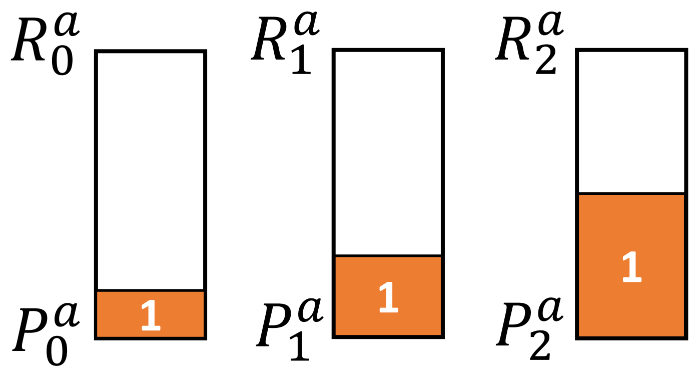
1. Iterate over all buckets as the attacked bucket i^*
   - As the one with the highest water level
2. Binary search on the max number of stones thrown into i^*
   - With enough spare resources for a defensive strategy to allow attacking bucket i^*
3. Greedy for spare stones
   - Throw into the bucket with the highest water level except bucket i^*
4. Waterfilling to a critical point and trigger a swap
   - With equal volume
5. Finish Waterfilling

Complexity: $O(n^4 \log n)$

Extensions for Practical Constraints

RACPP with Target-Specific Effectiveness

A stone has different volumes in each bucket



Adapted TDBS with unchanged complexity

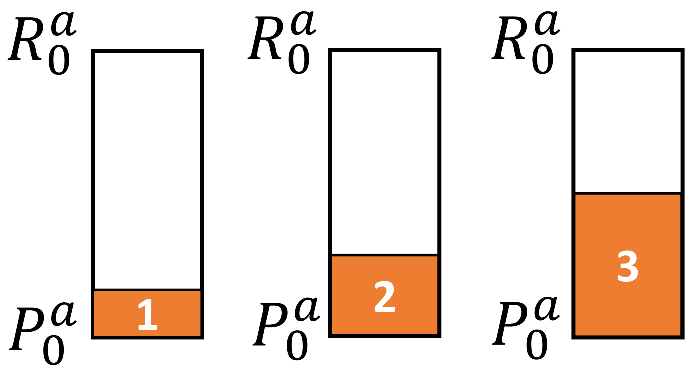
Complexity: $O(n^2 \log \frac{M}{\varepsilon})$

Extensions for Practical Constraints

RACPP with Villager-Specific Effectiveness

→

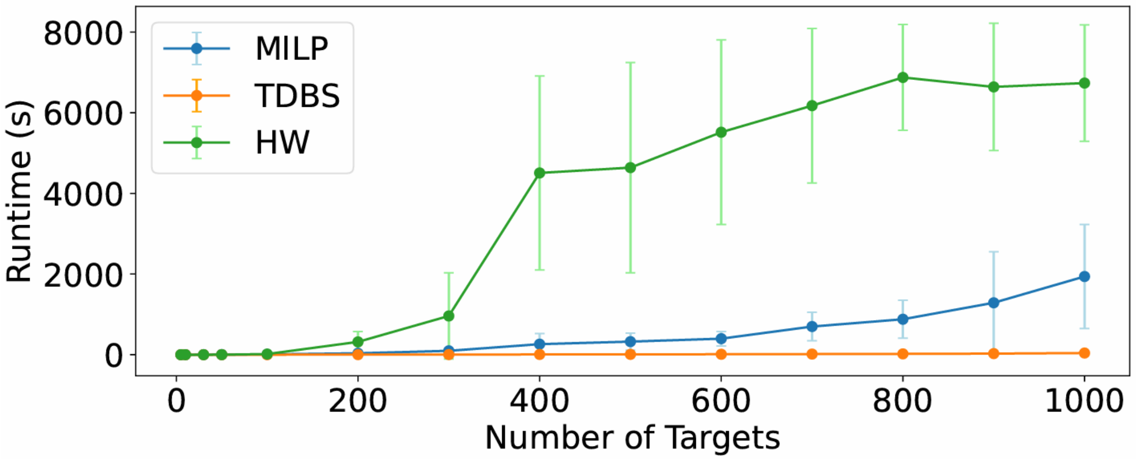
Stones have different volumes

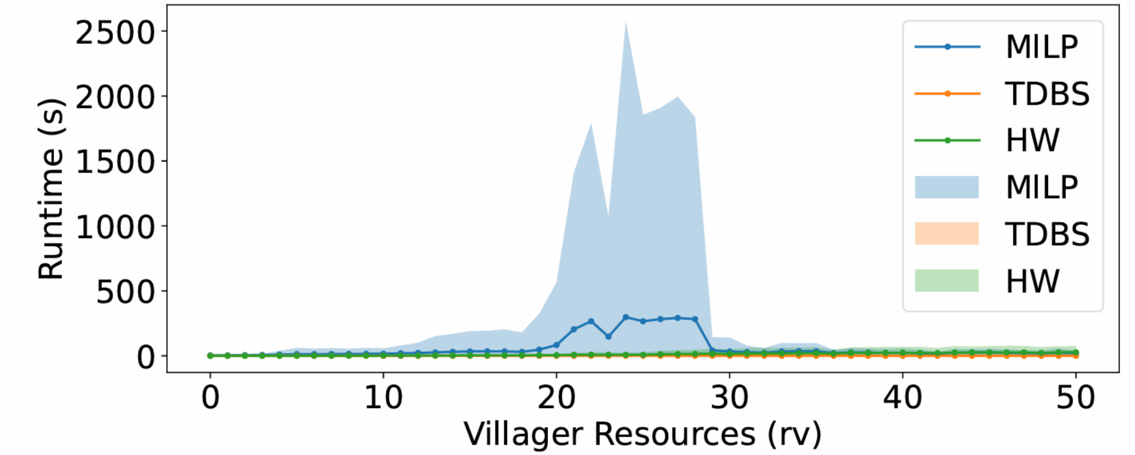


⇒

NP-Hard

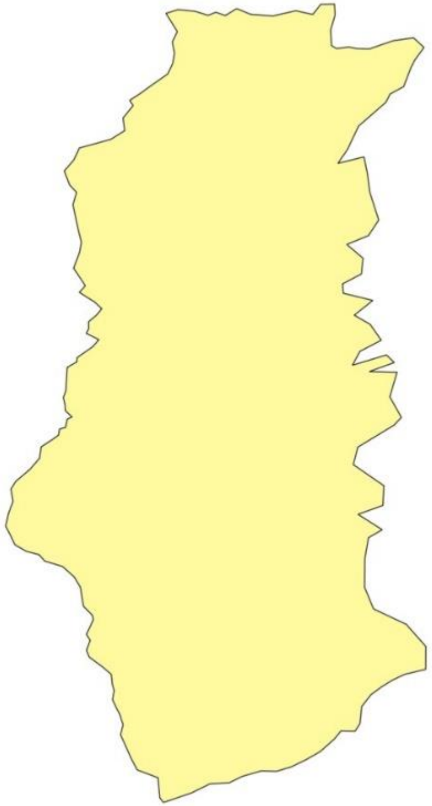
Experiments





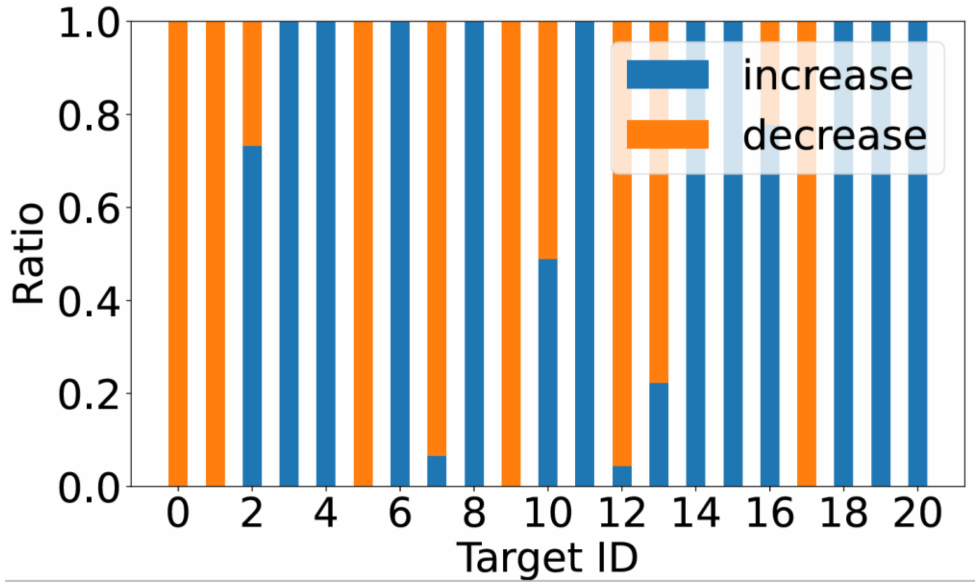
- TDBS
  - significantly faster
  - accurate enough for practical applications
- TDBS and HW
  - more stable

Case Study on Anti-poaching

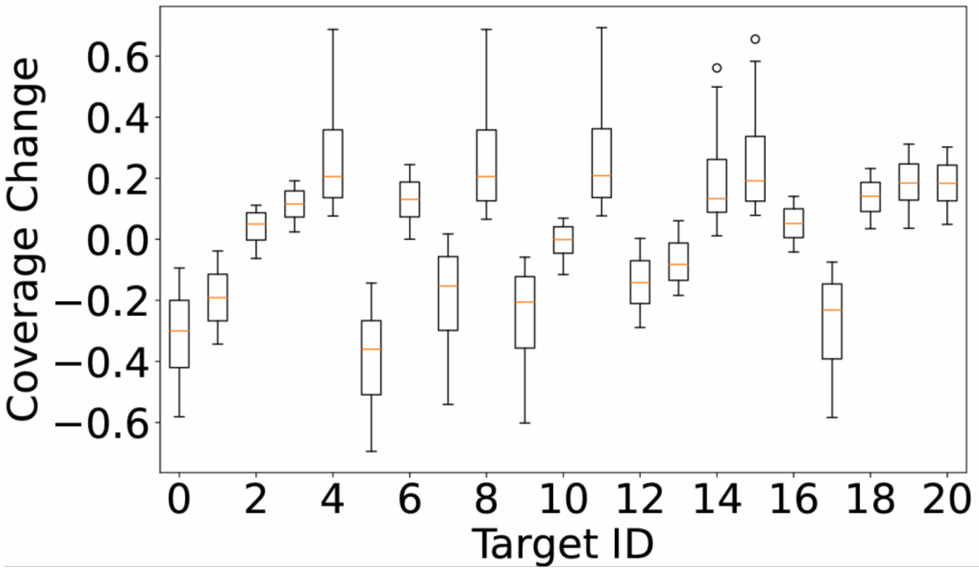


- A protected area in Northeast China
  - Home to the Manchurian tiger
  - 21 2km×2km regions
  - Defended by rangers and villagers

Case Study on Anti-poaching



Ratio of advice to increase or decrease resources on each target

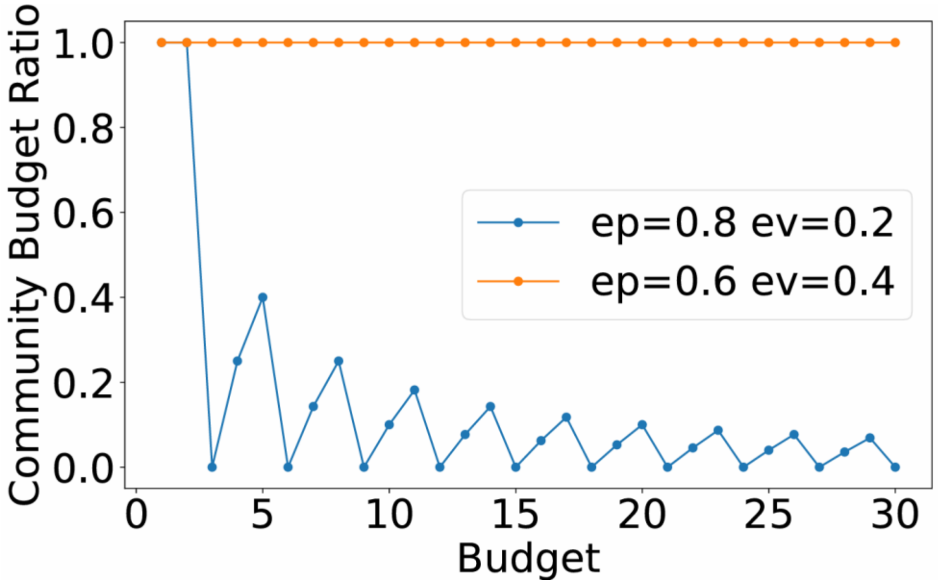


Distribution of coverage change on each target

Defenders' utility ↑ 25.9% To 152.6%

Average: 83.1%

Case Study on Anti-poaching



Cost ratio **V.S.** Effectiveness ratio

Preference for rangers or villagers

Our Contributions

- We propose a novel game-theoretic model for community-participated patrol

- We introduce two algorithms
  - A polynomial approximate algorithm: Two-Dimensional Binary Search
  - A polynomial exact algorithm: Hybrid Waterfilling

- We conduct a detailed case study
  - On a protected tiger habitat in Northeast China