Recruitment Processes In Ants Task Allocation

TEL AVIV UNIVERSITY

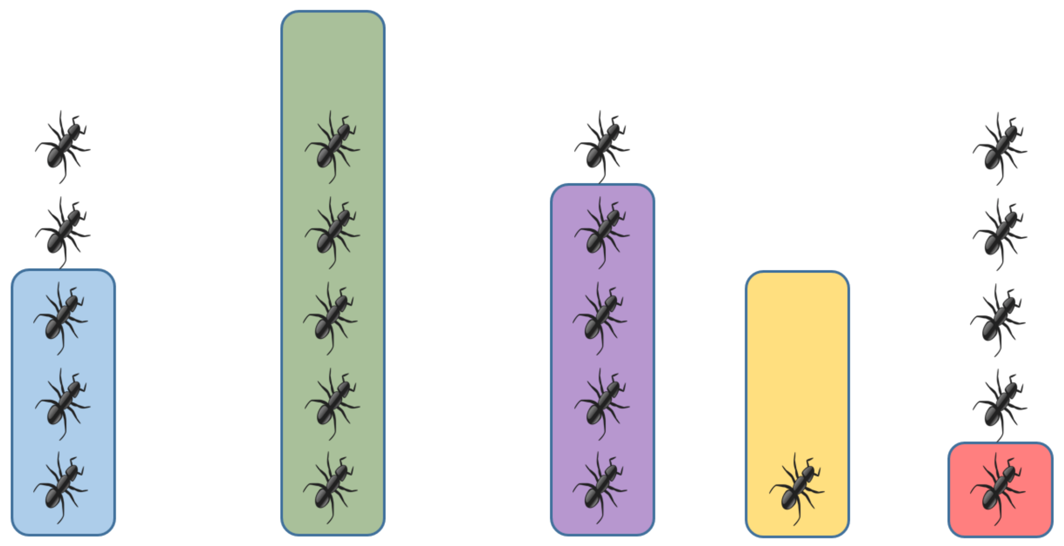
Yehuda Afek, Roman Kecher, and Moshe Sulamy

Task Allocation

- Process that adjusts # of ants engaged in different tasks
- Each ant decides which task to be active at next

Problem Definition

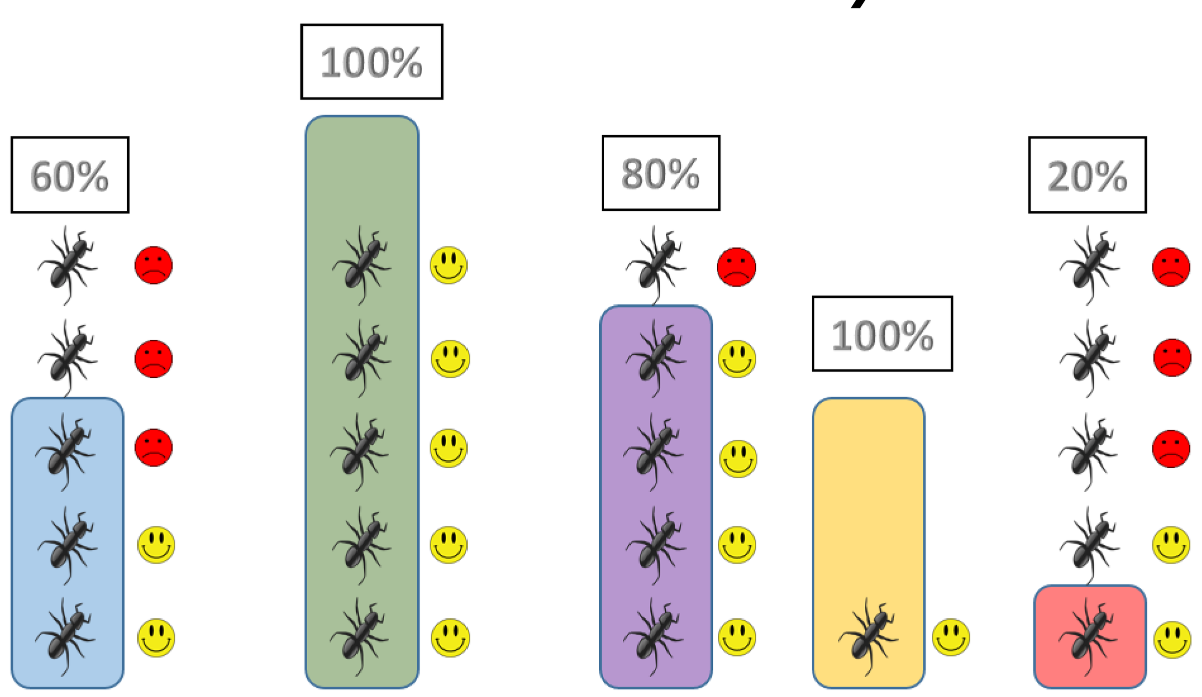
- ***n** ants, **t*** tasks
- ***X={x1, …, xt}*** - initial task assignment
- ***D={d1,…,dt}*** – demand vector (# ants required in each task)
- Goal: ***Decide*** function used by each ant to decide which task to do in next round.



Model

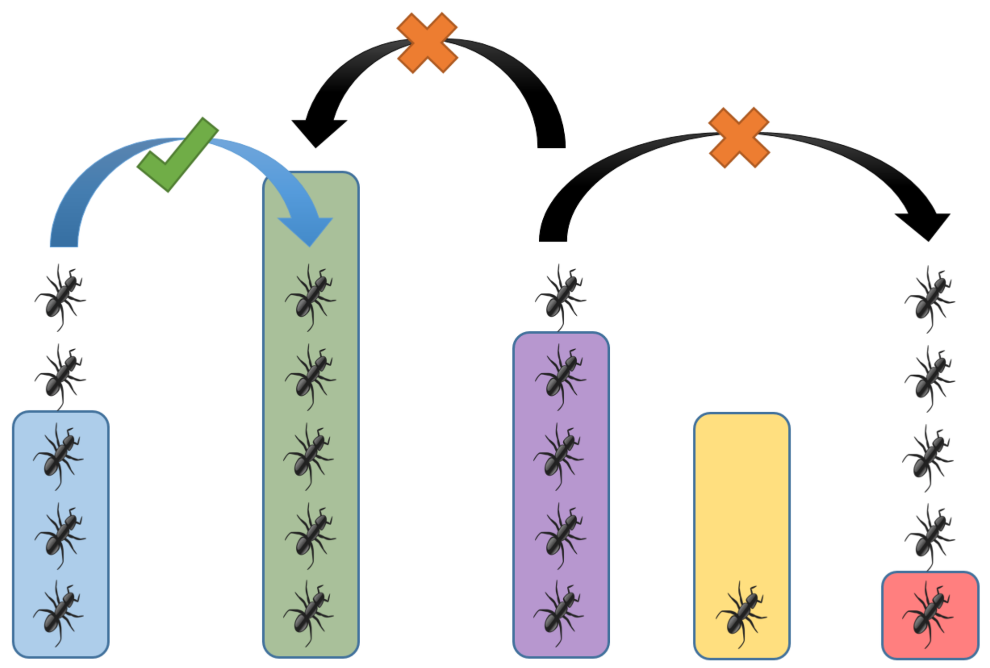
- Synchronous rounds
- Unsuccessful ants interact with another randomly-chosen ant
- Communication by 1-on-1 interaction
- Decide only according to local information, the interaction

- Ants either successful or unsuccessful:
- Success rate: if ***Xi <= Di*** then **1**, otherwise ***Di/Xi***



One-Way Task Switching

- From biological observations, ants switch tasks only in certain directions
- Thus, model allows only one-way task switching



Recruitment w/o idle ants

Decide Function

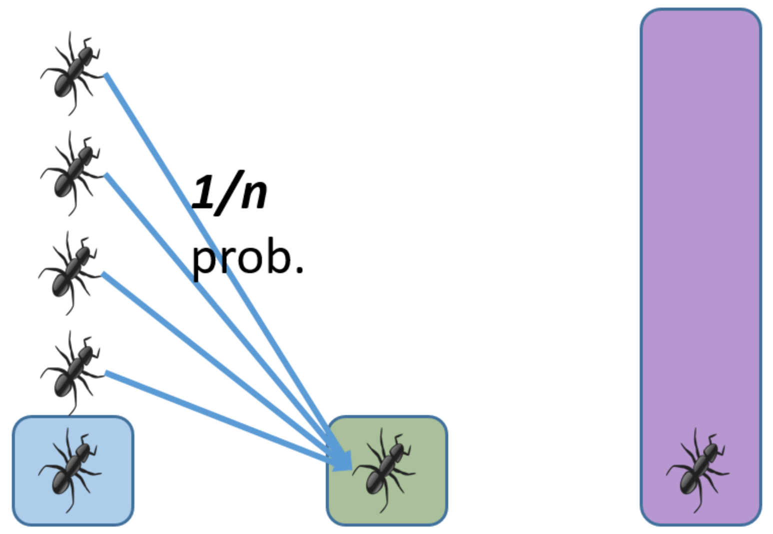
- function ***Decide*** (me, other)
- **if** (me.unsuccessful **and** other.successful **and** other.task == me.task+1) **then** me.task = me.task+1;//switch tasks
- **end if**

- Successful ants recruit unsuccessful ants to their task

Runtime Analysis

- Runtime: ***O(n ln n)*** rounds

Bottleneck Scenario



- Bottleneck prevents from switching tasks quickly to meet demand
- 1 ant recruited every ***n/x*** rounds
- ***x*** is # of ants in blue task, initially ***n-2***

$$n \cdot \sum_{x=1}^{n-2} \frac{1}{x} = O(n \cdot \ln n)$$

Recruitment *with* idle ants

- Approx. 30-50% of ants are idle
- Why?

What If …

- Idle ants are recruiters
- A fixed fraction is idle

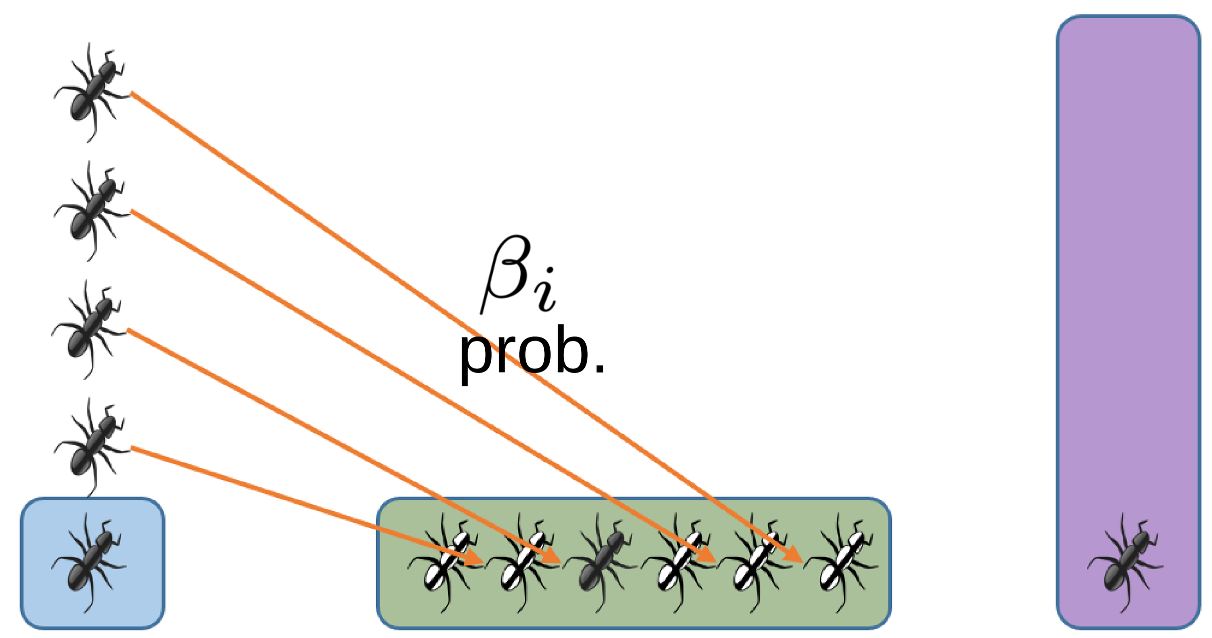
- Ants can be successful, unsuccessful, **or idle**
- Idle ants *recruit* unsuccessful ants
- Demand is met exponentially faster

Runtime Analysis

- Runtime: ***O(ln n)*** rounds with constant fraction of idle *recruiting* ants

Bottleneck Scenario

- *idle ants in white*



- Bottleneck task (green) now recruits fast, no longer an actual bottleneck
- 1 ant recruited every (*1/x**β_i)
  β_i - *fraction* of idle ants in green task

$$\frac{1}{\beta_i} \cdot \sum_{x=1}^{n} \frac{1}{x} \sim= \frac{\ln n}{\beta_i} = O(\ln n)$$

- For an average colony of *n=5000* ants and 1 interaction / second, that is **8.5 seconds** vs. **11.5 hours!**