Ants Nearby Treasure Search

- Infinite grid
- $k$ ants
  - Initially at the origin
- Food at distance $D$

- Ants have to find the food

- Optimal run-time: $\Omega(D + D^2/k)$
  
  [Feinerman, Korman, Lotker, Sereni, 2012]
Pheromones

- Ants emit pheromones [Lenzen, Radeva, 2013]
- Or not
- And sense them
- No other communication
- Biological resource
- Goal: minimize pheromone count
Ground Rules

- Every ant runs same algorithm (locally)
  - With same initial state

- Only uniform algorithms, ants have no knowledge of:
  - k, total number of ants
  - D, distance to the food
Synchronous Model

- Rounds: all ants move once per round
Synchronous Model

- **Rounds:**
  all ants move once per round

- **Assumption:** ant emission scheme
  [Emek, Langner, Uitto, Wattenhofer, 2013]

- At most one ant is emitted in each round
Asynchronous Model

- Adversary repeatedly schedules one ant

- Test&Set:
  - Sense and emit a pheromone is one atomic step

- Definition of Rounds:
  - Round ends when every ant took at least one step
  - Only for (time) complexity
Ants Models

● **FSM**: Finite State Machines
  Constant size memory

● **TM**: Turing Machines
  Unlimited memory

● Both deterministic
## Results

<table>
<thead>
<tr>
<th></th>
<th>Lower Bound</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FSM</strong> (Deterministic)</td>
<td>$\Omega(D)$ pheromones to find the food</td>
<td>$O(D)$ pheromones $O(D + D^2/k)$ time</td>
</tr>
<tr>
<td><strong>TM</strong>  (Deterministic)</td>
<td>$\Omega(k)$ pheromones for optimal run time</td>
<td>$O(k)$ pheromones $O(D + D^2/k)$ time</td>
</tr>
</tbody>
</table>

Previously known: $O(D^2)$ pheromones [Lenzen, Radeva, 2013]

Results hold for Synchronous and Asynchronous models
## Results

<table>
<thead>
<tr>
<th>Lower Bound</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FSM</strong>( (\text{Deterministic}) )</td>
<td></td>
</tr>
<tr>
<td><strong>TM</strong> ( (\text{Deterministic}) )</td>
<td></td>
</tr>
</tbody>
</table>

Previously known: \( O(D^2) \) pheromones [Lenzen, Radeva, 2013]
## Results

<table>
<thead>
<tr>
<th></th>
<th>Lower Bound</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FSM</strong> (Deterministic)</td>
<td>$\Omega(D)$ pheromones to find the food</td>
<td>$O(D)$ pheromones $O(D + D^2/k)$ time</td>
</tr>
<tr>
<td><strong>TM</strong> (Deterministic)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Previously known: $O(D^2)$ pheromones [Lenzen, Radeva, 2013]
## Results

<table>
<thead>
<tr>
<th></th>
<th>Lower Bound</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FSM</strong> (Deterministic)</td>
<td>$\Omega(D)$ pheromones to find the food</td>
<td>$O(D)$ pheromones $O(D + D^2/k)$ time</td>
</tr>
<tr>
<td><strong>TM</strong> (Deterministic)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Previously known: $O(D^2)$ pheromones [Lenzen, Radeva, 2013]

Results hold for Synchronous and Asynchronous models.
Layers

- Definition: layer $L$
- All grid cells at distance $L$ from origin
FSM Need $\Omega(D)$ Pheromones

- Assume FSM with $S$ states
  - Uses $o(D)$ pheromones
- $S+1$ consecutive pheromone-free layers exist
- Path starts and ends in same state
  - Infinite loop
FSM Algorithms

- **Problem:**
  FSM can’t count

- **Solution:**
  Use pheromones as turning points

- Similar to the idea of guides
  [Emek, Langner, Uitto, Wattenhofer, 2013]
Asynchronous FSM Algorithm

- Mark E, S, W, N
- Explore from N
  - N never longer than E, S or W
- Test&Set prevents multiple ants from exploring same layer
Synchronous FSM Algorithm

- Emission scheme breaks initial symmetry
- But what happens if two ants collide?
- Veteran ants behave differently than Newbie ants
### Results

<table>
<thead>
<tr>
<th></th>
<th>Lower Bound</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FSM</strong>&lt;br&gt;(Deterministic)</td>
<td>$\Omega(D)$ pheromones to find the food</td>
<td>$O(D)$ pheromones $O(D + D^2/k)$ time</td>
</tr>
<tr>
<td><strong>TM</strong>&lt;br&gt;(Deterministic)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Previously known: $O(D^2)$ pheromones [Lenzen, Radeva, 2013]  
Results hold for Synchronous and Asynchronous models
## Results

<table>
<thead>
<tr>
<th></th>
<th>Lower Bound</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FSM</strong>&lt;br&gt;(Deterministic)</td>
<td>$\Omega(D)$ pheromones to find the food</td>
<td>$O(D)$ pheromones $O(D + D^2/k)$ time</td>
</tr>
<tr>
<td><strong>TM</strong>&lt;br&gt;(Deterministic)</td>
<td>$\Omega(k)$ pheromones for optimal run time</td>
<td>$O(k)$ pheromones $O(D + D^2/k)$ time</td>
</tr>
</tbody>
</table>

Previously known: $O(D^2)$ pheromones [Lenzen, Radeva, 2013]

Results hold for Synchronous and Asynchronous models
Async TM Need $\Omega(k)$ Pheromones

- Assume one ant does not emit pheromones
Async TM Need $\Omega(k)$ Pheromones

- Assume one ant does not emit pheromones
- Consider same scheduling but with extra ants
  - All new ants follow that one ant
- Runtime remains the same (but more ants)
Sync TM Need \( \Omega(k) \) Pheromones

- Emit one ant
  - Until all pheromones are placed
- Emit second ant
  - Until all pheromones are placed
- Continue
  - Delay is constant

- If no new pheromones are placed, all following ants behave the same
Asynchronous TM Algorithm

- TM can **count**!
- Use pheromones to assign IDs to ants
- Static partition
  - Explore layers \( L = \text{ID} \mod \text{Total} \)
  - Occasionally update estimated Total
- Also works for the synchronous model
Future Directions

● Fault tolerance (with pheromones)

● Employ randomization
  ● To implement emission schemes
  ● To further reduce pheromone counts

● Avoid Test&Set semantics

● Direct rest of ants to the found treasure
  ● And/or back to the nest
Thanks

Questions?