Bridge Experiment with Overcrowding

Swarm Intelligence Course Project

Sébastien Cuendet & Jean-Philippe Pellet
{sebastien.cuendet, jean-philippe.pellet}@epfl.ch

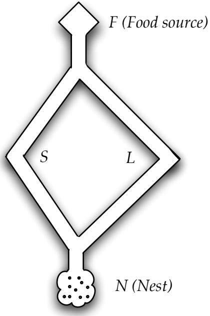
9th February, 2006

Outline

Outline

Asymmetric Bridge Experiment

- Given number of agents/ants must bring back a maximal amount of food
- Two paths: short path (length l), long path (length $r \cdot l$)
- Add obstacle avoidance
- Study effect of pheromone



Outline

Model Description

- Microscopic model: PFSM
- Fixed probabilities and dynamic probabilities
- Dynamic probs depend on system's past history

$\Rightarrow$ Not a markovian model

Modeled aspects:

- Path length
- Collisions
- Pheromone

Model Description

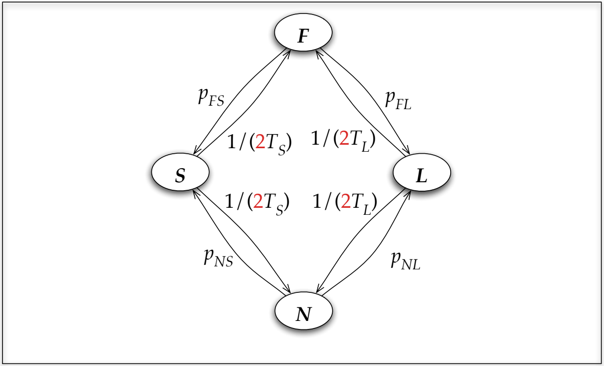
- Microscopic model: PFSM
- Fixed probabilities and dynamic probabilities
- Dynamic probs depend on system's past history

$\Rightarrow$ Not a markovian model

Modeled aspects:

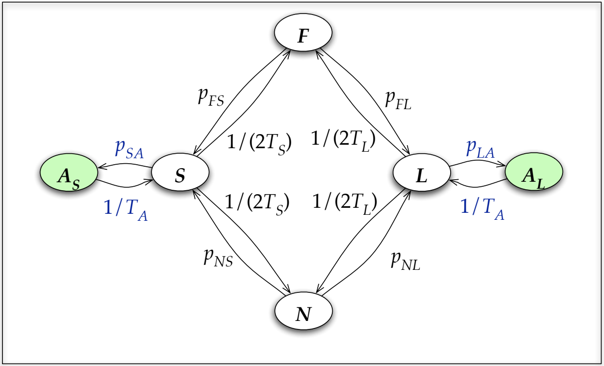
- Path length
- Collisions
- Pheromone

Designing the Model: Iteration I



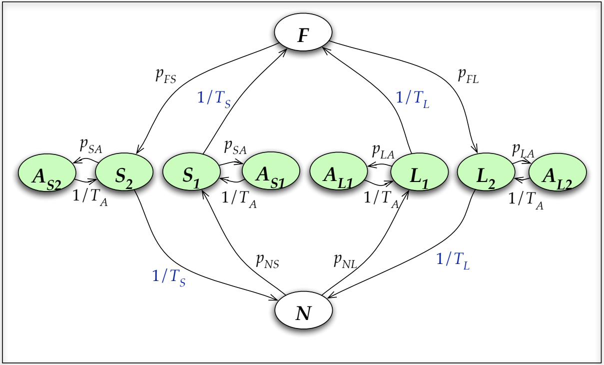
- Simplest case, no collision
- No pheromones:, $p_{NS} = p_{NL} = p_{\overline{N}}/2$, $p_{FS} = p_{FL} = 1/2$
- Path length: modeled with outgoing prob ($T_L = r \cdot T_S$)
- Implicit U-turn possibility

Designing the Model: Iteration II



- Added collision prob (dynamic):
  $p_{SA}(k) = p_r \cdot (S(k) - 1)$, $p_{LA}(k) = \frac{p_r}{r} \cdot (L(k) - 1)$
- Come back from avoidance probabilistically with prob $1/T_A$
- p_r, T_A : new model parameters. Still implicit U-turn possibility

Designing the Model: Iteration III



- Duplicated some states: now way up and way down; no U-turn

$\Rightarrow$ Probabilities must be adapted

- Collision: $p_{AS_1}(k) = p_{AS_2}(k) = p_r \cdot (S_1(k) + S_2(k) - 1)$
  $p_{AL_1}(k) = p_{AL_2}(k) = \frac{p_r}{r} \cdot (L_1(k) + L_2(k) - 1)$
- How to implement pheromones?

Modeling Pheromones I

- Idea I: count number of agents in states S_1 and S_2, use as estimate for pheromone deposited on S.

$\Rightarrow$ Problem: doesn't work (e.g. longer path is marked more). Pheromone should mark most successful path and take into account how fast ants get back using this path

- Idea II: Use two types of pheromones: one deposited on the way up and "smelled" on the way down; one deposited on the way down and smelled on the way up

$\Rightarrow$ Problem: still not OK (e.g. ants that have just entered the long path have the same influence as ants about to leave the short path)

- Idea III: use special exit states at the end of the paths; count ants going through them

$\Rightarrow$ It works!

Modeling Pheromones I

- Idea I: count number of agents in states S_1 and S_2, use as estimate for pheromone deposited on S.

$\Rightarrow$ Problem: doesn't work (e.g. longer path is marked more). Pheromone should mark most successful path and take into account how fast ants get back using this path

- Idea II: Use two types of pheromones: one deposited on the way up and "smelled" on the way down; one deposited on the way down and smelled on the way up

$\Rightarrow$ Problem: still not OK (e.g. ants that have just entered the long path have the same influence as ants about to leave the short path)

- Idea III: use special exit states at the end of the paths; count ants going through them

$\Rightarrow$ It works!

Modeling Pheromones I

- Idea I: count number of agents in states S_1 and S_2, use as estimate for pheromone deposited on S.

⇒ Problem: doesn't work (e.g. longer path is marked more). Pheromone should mark most successful path and take into account how fast ants get back using this path

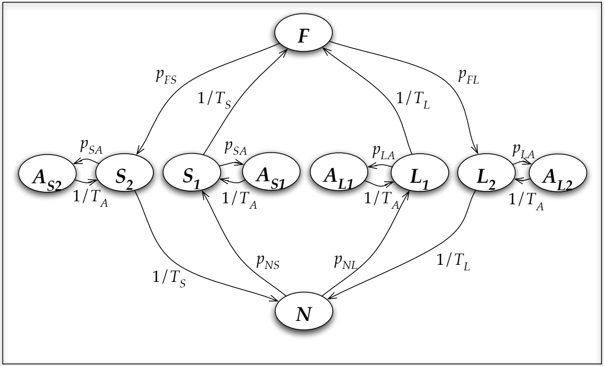
- Idea II: Use two types of pheromones: one deposited on the way up and "smelled" on the way down; one deposited on the way down and smelled on the way up

⇒ Problem: still not OK (e.g. ants that have just entered the long path have the same influence as ants about to leave the short path)

- Idea III: use special exit states at the end of the paths; count ants going through them

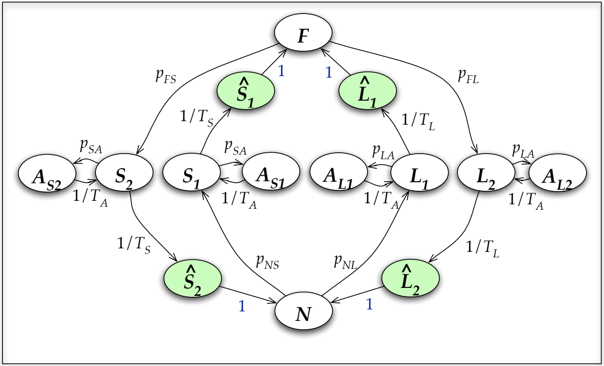
⇒ It works!

Designing the Model: Iteration III (again)



(what we had previously)

Designing the Model: Iteration IV



- Added exit states $\widehat{S}_1$, $\widehat{S}_2$, $\widehat{L}_1$, $\widehat{L}_2$
- Ants remain only one time step in exit states
- Neglect collision prob in exit states
- Pheromones implemented by varying p_{FS}, p_{FL}, p_{NS} and p_{NL}

Modeling Pheromones II

No pheromone:

- $p_{NS} = p_{NL} = p_{\overline{N}}/2$
- $p_{FS} = p_{FL} = 1/2$

With pheromone:

- "Smelled" pheromone for S at time k from the nest:

$$\Phi_{NS}(k) = \sum_{j=0}^{j_{max}} h^j \cdot \widehat{S}_2(k-j), \quad \Phi_{NL}(k) = \sum_{j=0}^{j_{max}} h^j \cdot \widehat{L}_2(k-j)$$

- Transition prob:

$$p_{NS}(k) = p_{\overline{N}} \cdot \frac{[q+\Phi_{NS}(k)]^n}{[q+\Phi_{NS}(k)]^n + [q+\Phi_{NL}(k)]^n}$$

New parameters: $p_{\overline{N}}$, h, n, q, (j_{max})

Modeling Pheromones II

No pheromone:

- $p_{NS} = p_{NL} = p_{\overline{N}}/2$
- $p_{FS} = p_{FL} = 1/2$

With pheromone:

- "Smelled" pheromone for S at time k from the nest:

$$\Phi_{NS}(k) = \sum_{j=0}^{j_{max}} h^j \cdot \widehat{S}_2(k-j), \quad \Phi_{NL}(k) = \sum_{j=0}^{j_{max}} h^j \cdot \widehat{L}_2(k-j)$$

- Transition prob:

$$p_{NS}(k) = p_{\overline{N}} \cdot \frac{[q + \Phi_{NS}(k)]^n}{[q + \Phi_{NS}(k)]^n + [q + \Phi_{NL}(k)]^n}$$

New parameters: $p_{\overline{N}}$, h, n, q, (j_{max})

Modeling Pheromones II

No pheromone:

- $p_{NS} = p_{NL} = p_{\overline{N}}/2$
- $p_{FS} = p_{FL} = 1/2$

With pheromone:

- "Smelled" pheromone for S at time k from the nest:

$$\Phi_{NS}(k) = \sum_{j=0}^{j_{max}} h^j \cdot \widehat{S}_2(k-j), \quad \Phi_{NL}(k) = \sum_{j=0}^{j_{max}} h^j \cdot \widehat{L}_2(k-j)$$

- Transition prob:

$$p_{NS}(k) = p_{\overline{N}} \cdot \frac{[q + \Phi_{NS}(k)]^n}{[q + \Phi_{NS}(k)]^n + [q + \Phi_{NL}(k)]^n}$$

New parameters: $p_{\overline{N}}$, h, n, q, (j_{max})

Outline

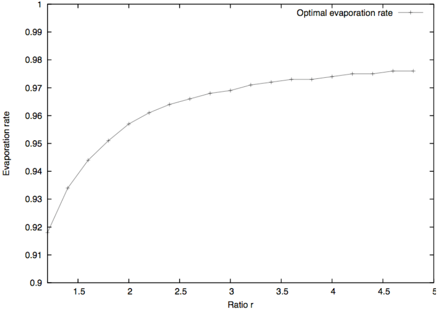
Macroscopic Model

- Equations: just read the PFSM, set of non-linear difference equations of the form

$$S_1(k+1) = S_1(k) + p_{NS} \cdot N(k) + p_{\overline{A}} \cdot A_{S_1}(k) - p_{\overline{S}} \cdot S_1(k) - p_{SA} \cdot S_1(k)$$

- Goal: determine optimal evaporation rate h given ratio r.
- Macroscopic equation are difficult to solve

$\Rightarrow$ Determine optimal h with macrosimulation



Outline

Conclusion & Outlook

- Model works according to previous case studies of the trail laying/following mechanism
- Optimal evaporation rate h is monotonic function of the ratio r, also when taking into account the overcrowding effet

Further work could include:

- Modeling of more aspects (geometry of the paths and of the robots, U-turn prob, wall avoidance, etc.)
- Realistic simulation or real-robot implementation to try to reach a zero-free parameter model

Conclusion & Outlook

- Model works according to previous case studies of the trail laying/following mechanism
- Optimal evaporation rate h is monotonic function of the ratio r, also when taking into account the overcrowding effet

Further work could include:

- Modeling of more aspects (geometry of the paths and of the robots, U-turn prob, wall avoidance, etc.)
- Realistic simulation or real-robot implementation to try to reach a zero-free parameter model

Thanks for your attention!