Ant Colony Optimization

Camazine et al. 2003
Dorigo et al. 1996
Dorigo and Gambardella 1997

Ned Dochtermann
February 15, 2006
CS 790R, University of Nevada, Reno

Ant Colony Optimization (ACO)

- Ant foraging behavior
- Ant foraging behavior as an optimization method
  - Traveling salesman problems
  - How efficient and effective versus other methods

Ant foraging behavior

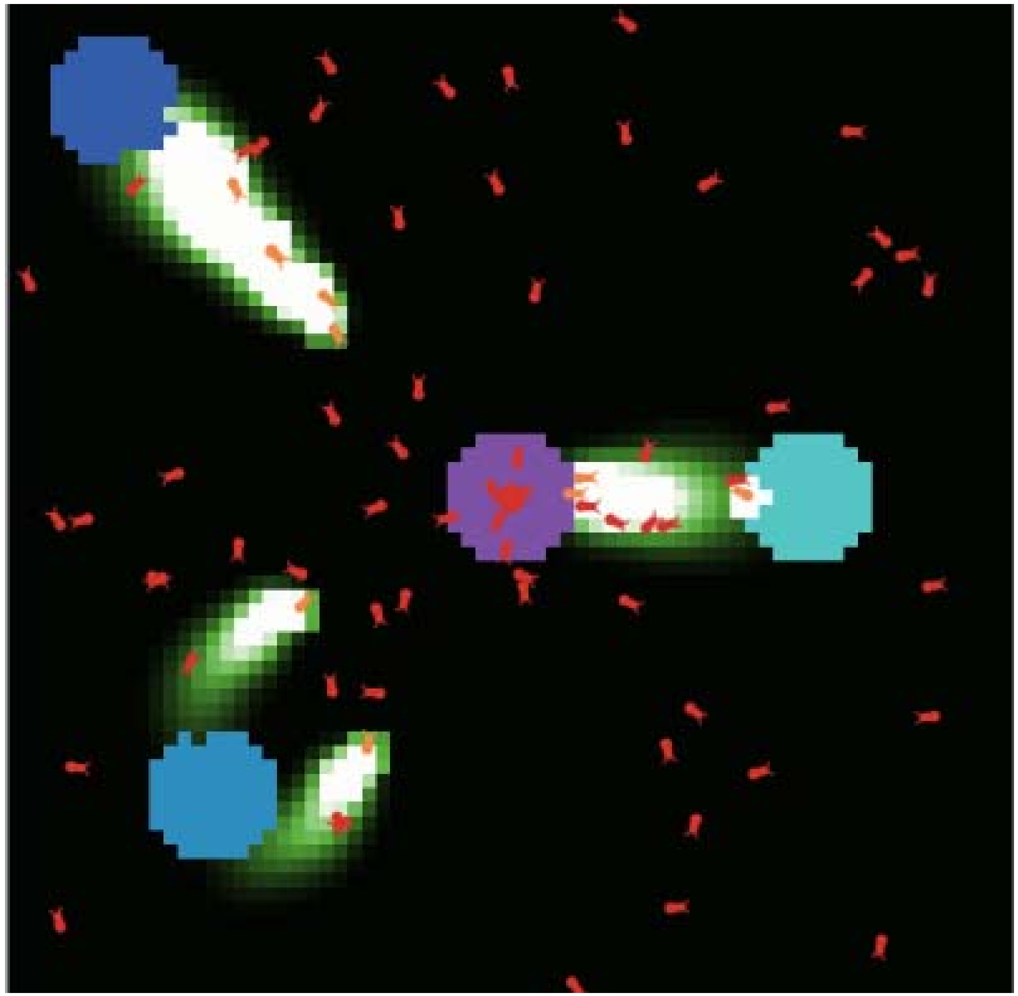
Ant foraging behavior

- Ant foraging
  - How food is found
  - How information is passed between individuals
  - Route formation/Trail-following

Ant foraging behavior

- How do ants find food?
  - Random walks
- How is information passed between foragers?
  - Odor trails (Wilson 1962)

Odor trails



Odor trails

- After finding a food source individual ants lay down chemicals
  - Only lay down pheromones when they have food
  - Modulate intensity of signal based on the number of visits to and quality of a source

Route formation / Trail formation

- Individuals follow an odor trail based on the concentration of the trail which changes according to:

$$dC_i/dt = q_i \Phi P_i - fC_i$$

where individual choice is according to:

$$P_i = [(k + C_i)^n]/[\sum(k + C_{set})^n]$$

Route formation / Trail formation

- One of the key parameters in the preceding equations is ‘q’
  - Relative values between trails → value of food source

Route formation / Trail formation

- Different value food sources
  - "better" source selected if introduced simultaneously introduced (dependent on q_1/q_2 ratio)

  - first source selected if introduced asynchronously
    - Dependent on q_1/q_2 ratio and R

Route formation / Trail formation

- The basic set of two equations can be modified and added to allow for u-turns

- These equations lead to the ability of ant foragers to find the shortest path, and the highest value source, even in the presence of obstacles

Trails and obstacles

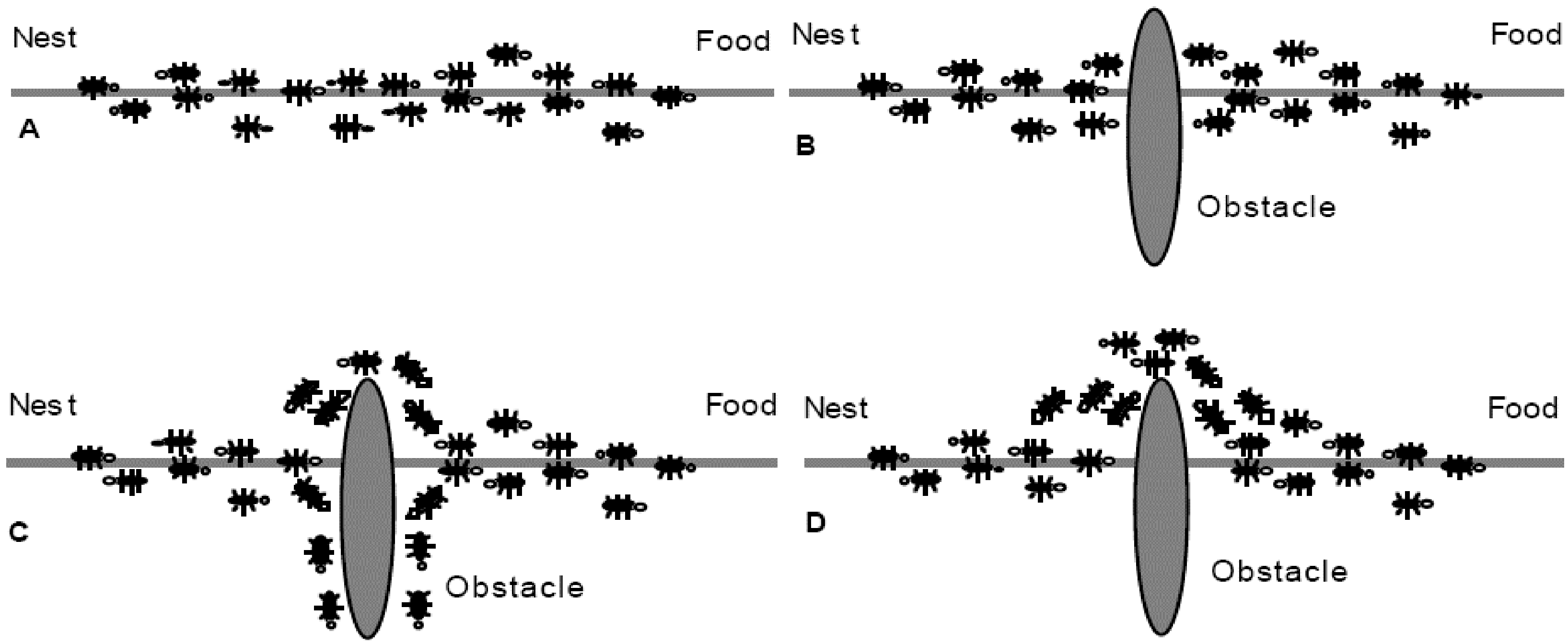


Fig.1. (A) Real ants follow a path between nest and food source. (B) An obstacle appears on the path: Ants choose whether to turn left or right with equal probability. (C) Pheromone is deposited more quickly on the shorter path. (D) All ants have chosen the shorter path.

Dorigo and Gambardella 1997

Ant Colony Optimization: The traveling salesman problem (TSP)

- Given n cities how do you find the optimal path between them?
  - Genetic algorithms
  - Maximum likelihood
  - Annealing (?)

- Why not ants?

ACO: TSP

- M. Dorigo first proposed this in his 1992 Phd dissertation
- Apply modified ant foraging rules to TSP problems (extendable to similar optimization questions)

ACO: TSP

- Maintain:
  - Trails with high pheromone levels are preferred
  - Pheromone concentration grows more quickly on short paths
  - Communication between ants via trail characteristics

Dorigo and Gambardella 1997

ACO: TSP

- Modifications:
  - Knowledge of distance between cities
  - Working memory
  - Iterative resetting of trails based on shortest route (e.g. reinforcement learning)

Dorigo and Gambardella 1997

ACO: TSP

- Main feature:
  - ants visit towns probabilistically according to a function of distance and trail/pheromone

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum\limits_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ik}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

Dorigo et al. 1996

ACO: TSP

- Updating between tours

$$\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}$$

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$$

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L_k} & \text{if k - th ant uses edge (i, j) in its tour (between time t and t + n)} \\ 0 & \text{otherwise} \end{cases}$$

Dorigo et al. 1996

ACO: TSP

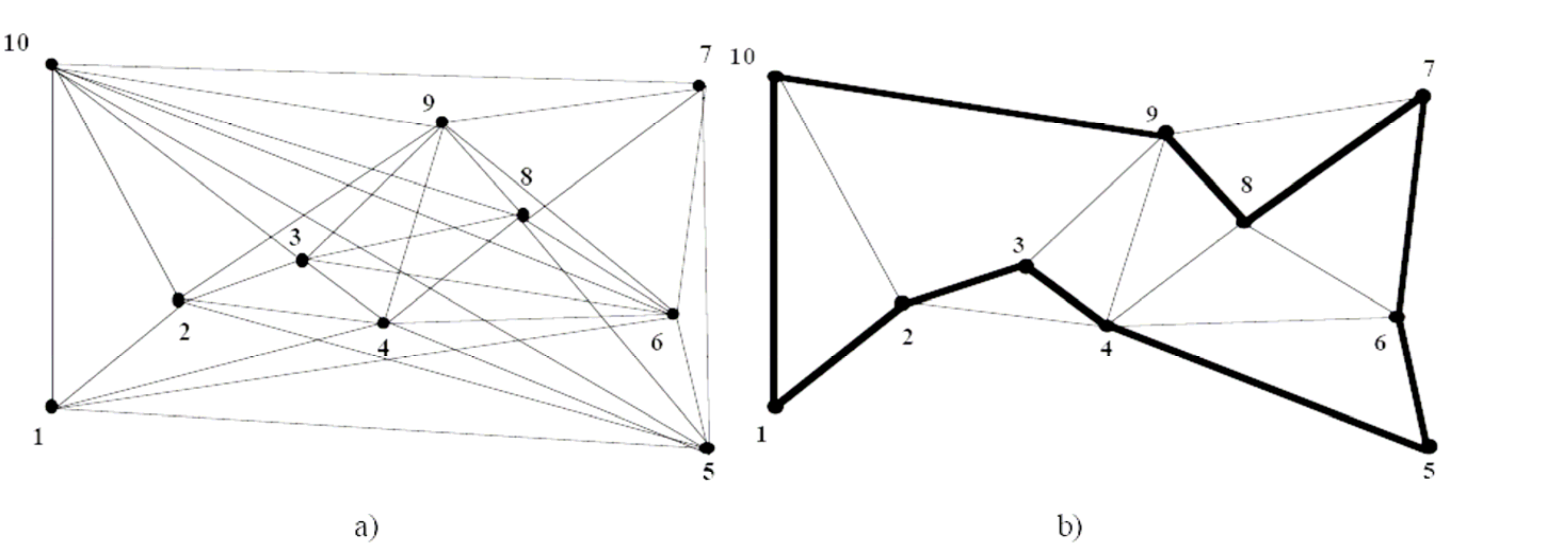


Fig. 6. Evolution of trail distribution for the CCA0 problem.
a) Trail distribution at the beginning of search.
b) Trail distribution after 100 cycles.

ACO: TSP

- Comparison to other methods

Table 1. Comparison of ACS with other nature-inspired algorithms on random instances of the symmetric TSP. Comparisons on average tour length obtained on five 50-city problems. SA = simulated annealing, EN = elastic net, SOM = self organizing map, FI = farthest insertion. Results on SA, EN, and SOM are from (Durbin and Willshaw,1987; Potvin, 1993). FI results are averaged over 15 trials starting from different initial cities. ACS was run for 1,250 iterations using m=20 ants and the results are averaged over 15 trials. The best average tour length for each problem is in boldface.

Problem name	ACS	SA	EN	SOM	FI
City set 1	**5.86**	5.88	5.98	6.06	6.03
City set 2	6.05	**6.01**	6.03	6.25	6.28
City set 3	**5.57**	5.65	5.70	5.83	5.85
City set 4	**5.70**	5.81	5.86	5.87	5.96
City set 5	**6.17**	6.33	6.49	6.70	6.71

Dorigo and Gambardella 1997

ACO: TSP

- Comparison to other methods

Table 3. Comparison of ACS with the genetic algorithm (GA), evolutionary programming (EP), simulated annealing (SA), and the annealing-genetic algorithm (AG), a combination of genetic algorithm and simulated annealing (Lin, Kao and Hsu, 1993). We report the best integer tour length, the best real tour length (in

Problem name	ACS	GA	EP	SA	AG	Optimum
Oliver30 (30-city problem)	**420** **(423.74)** **[830]**	421 (N/A) [3,200]	**420** **(423.74)** [40,000]	424 (N/A) [24,617]	**420** (N/A) [12,620]	**420** **(423.74)**
Eil50 (50-city problem)	**425** (427.96) **[1,830]**	428 (N/A) [25,000]	426 **(427.86)** [100,000]	443 (N/A) [68,512]	436 (N/A) [28,111]	**425** (N/A)
Eil75 (75-city problem)	**535** **(542.31)** **[3,480]**	545 (N/A) [80,000]	**542** (549.18) [325,000]	580 (N/A) [173,250]	561 (N/A) [95,506]	**535** (N/A)
KroA100 (100-city problem)	**21,282** **(21,285.44)** **[4,820]**	21,761 (N/A) [103,000]	N/A (N/A) [N/A]	N/A (N/A) [N/A]	N/A (N/A) [N/A]	**21,282** (N/A)

Dorigo and Gambardella 1997

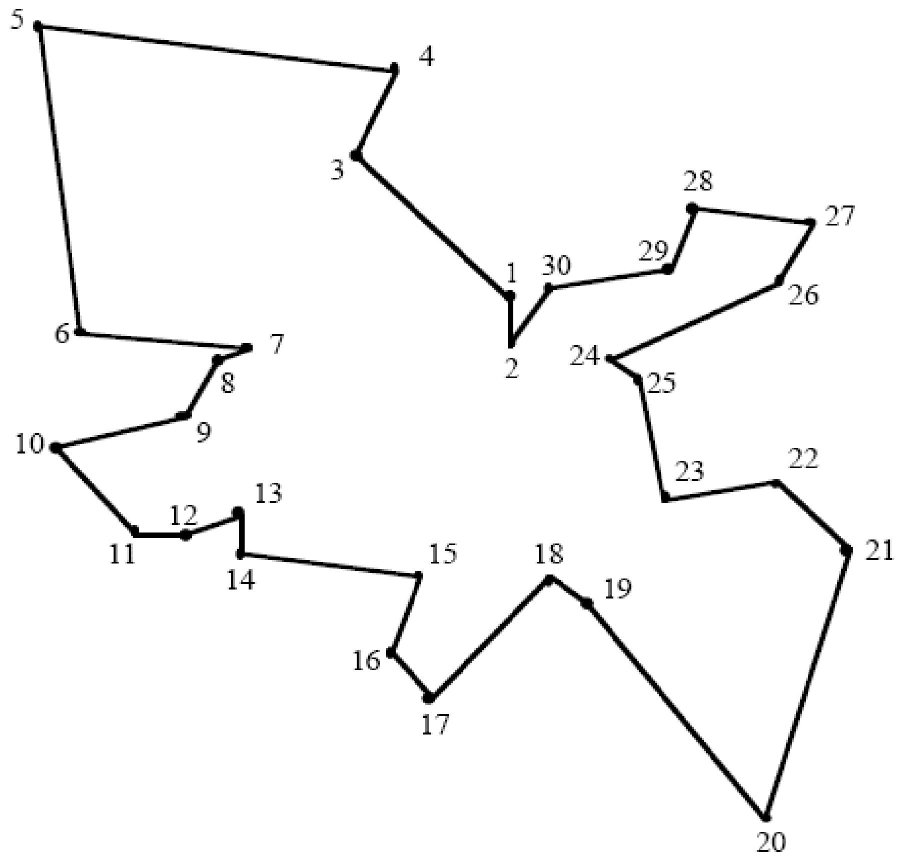


Fig. 9. The best tour obtained with 342 cycles of the *ant-cycle* algorithm for the Oliver30 problem (α=1, β=5, ρ=0.5, Q=100), real length = 423.741, integer length = 420.

Dorigo et al. 1996

ACO: TSP

- Is competitive or better than other methods in finding the optimal solution
  - faster
  - more flexible

Ant colony optimization

- Other extensions:
  - Scheduling
  - Multi-dimensional space