Ant Colony Optimization

Part 2: Simple Ant Colony Optimization

Fall 2009

Instructor: Dr. Masoud Yaghini

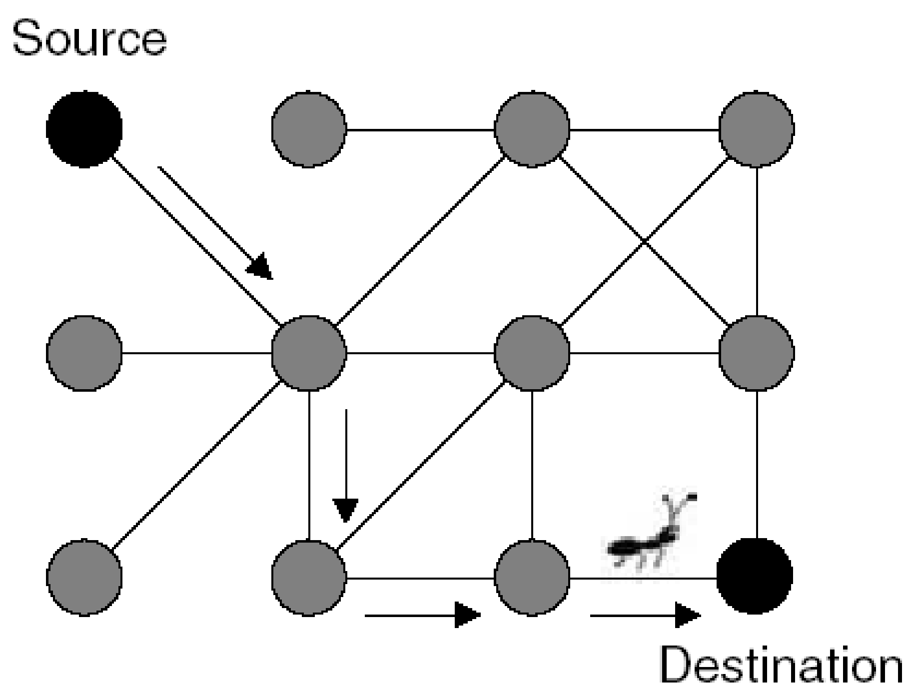
Outline

- Simple Ant Colony Optimization (S-ACO)
- Experiments with S-ACO
- References

Simple Ant Colony Optimization (S-ACO)

S-ACO

- The **simple ACO algorithm** (**S-ACO**) can be used to find a solution to **the shortest path problem** defined on the graph.



S-ACO

- A complete cycle of S-ACO:
  - **Forward ants and solution construction**
  - **Backward ants and loop elimination**
  - **Pheromone updates**
  - **Pheromone evaporation**

Forward ants and solution construction

- There are two working modes for the ants:
  - **forwards**
  - **backwards**
- Each ant builds, starting from the source node, a solution to the problem by applying a step-by-step decision policy.
- The ants memory allows them to retrace the **path it has followed** while searching for the destination node
- Pheromones are **only deposited** in backward mode.

Forward ants and solution construction

- Assume a connected graph $G = (N, A)$.
- Associated with each edge (i, j) of the graph there is a variable τ_{ij} termed **artificial pheromone trail**.
- Every **artificial ant** is capable of "marking" an edge with pheromone and "smelling" (reading) the pheromone on the trail.
- At the beginning of the search process, a constant amount of pheromone (e.g., $\tau_{ij}=1$) is assigned to all the arcs.

Forward ants and solution construction

- An ant k located at node i uses the pheromone trail $\tau_{ij}(t)$ to compute the probability of choosing j as next node:

$$p_{ij}^{k} = \begin{cases} \dfrac{\tau_{ij}{}^{\alpha}}{\sum_{j\in N_i^k} \tau_{ij}{}^{\alpha}}, & if \;\; j \in N_i^k \\ 0, & if \;\; j \notin N_i^k \end{cases}$$

- Where
  - N_i^k is the neighborhood of ant k in node i.
  - α is a parameter that controls the relative weight of pheromone trail

The neighborhood of ant *k* in node *i*

- The neighborhood of a node i contains all the nodes directly connected to node i in the graph $G = (N, A)$, except for the predecessor of node i (i.e., the last node the ant visited before moving to i).
- In this way the ants avoid returning to the same node they visited immediately before node i.
- Only in case N_i^k is empty, which corresponds to a dead end in the graph, node i's predecessor is included into N_i^k.

Forward ants and solution construction

- Ants use differences paths.
- Therefore the time step at which ants reach the destination node may differ from ant to ant.
- Ants traveling on shorter paths will reach their destinations faster.

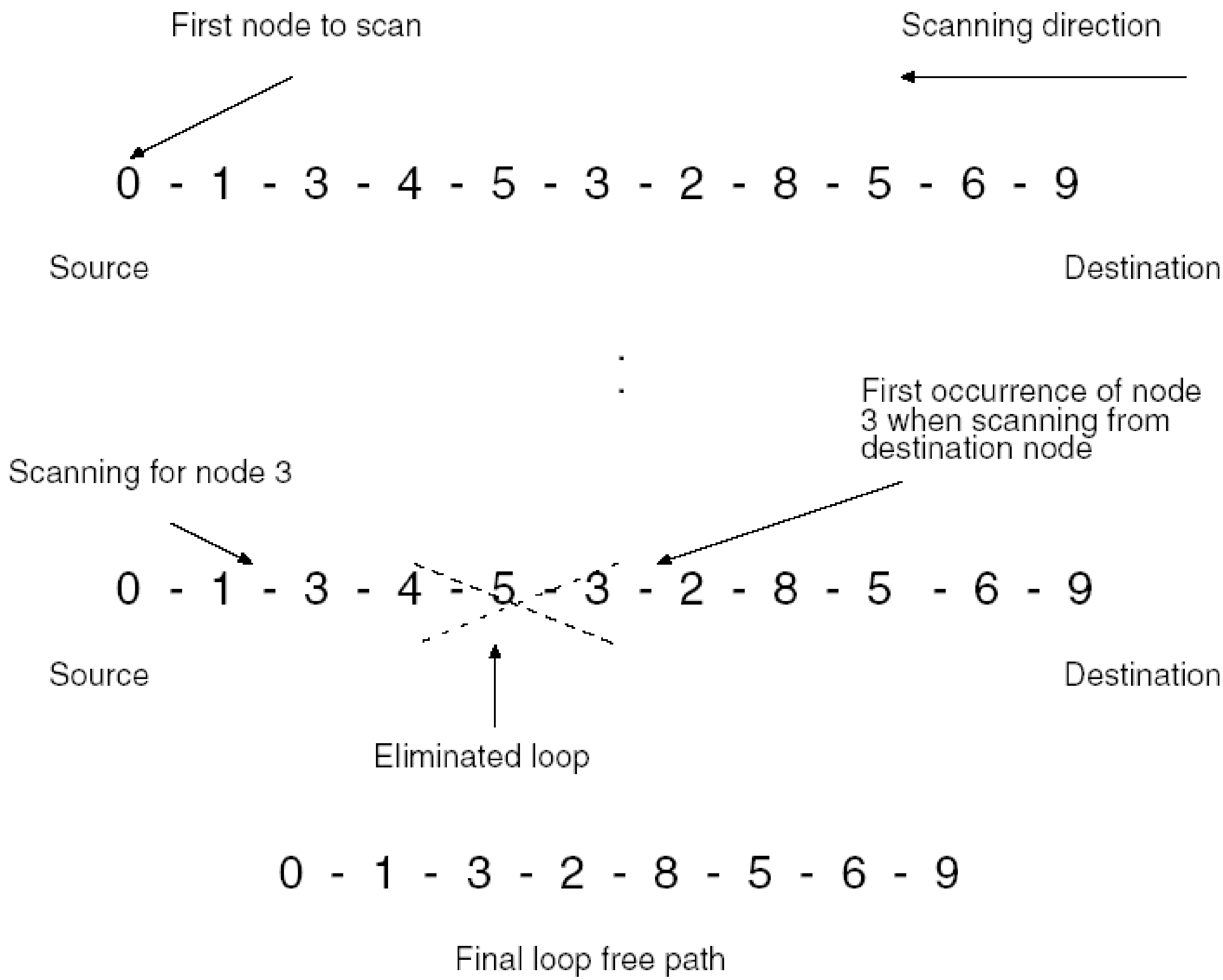
Backward ants and loop elimination

- When reaching the destination node,
  - the ant switches from the **forward mode** to the **backward mode**
- Before moving backward on their memorized path, they eliminate any loops from it has built while searching for its destination node.
- While moving backwards, the ants leave pheromones on the arcs they traversed.

Loop elimination

- Loop elimination can be done by iteratively scanning the node identifiers position by position starting from the source node
- For the node at the *i-th* position, the path is scanned starting from the destination node until the first occurrence of the node is encountered
- If we have $j > i$, the subpath from position $i + 1$ to position j corresponds to a loop and can be eliminated.

The scanning process for loop elimination



Pheromone Update

- During its return travel to the source, the *k*-th ant deposits an amount $\Delta\tau^k$ of pheromone on arcs it has visited.

$$\tau_{ij} \leftarrow \tau_{ij} + \Delta\tau^k$$

- By using this rule, the probability increases that forthcoming ants will use this arc.
- An important aspect is the choice of $\Delta\tau^k$.

Pheromone Update

Type of pheromone update:

- **The same constant value:**
  - The same constant value for all the ants.
  - Ants which have detected a shorter path can deposit pheromone earlier than ants traveling on a longer path.
- **Function of the solution quality**:
  - The ants evaluate the cost of the paths they have traversed.
  - The shorter paths will receive a greater deposit of pheromones.

Pheromone evaporation

- **Evaporation**
  - To avoid premature convergence pheromone evaporation is done
- **Convergence**
  - when the probability of selecting the arcs of particular path becomes close to 1
- An evaporation rule will be tied with the pheromones, which will reduce the chance for poor quality solutions.

Pheromone evaporation

- After each ant k has moved to the next node, the pheromones evaporate by the following equation to all the arcs:

$$\tau_{ij} \leftarrow (1-p)\tau_{ij}, \ \forall (i,j) \in A$$

  - where $p \in (0,1]$ is a parameter.

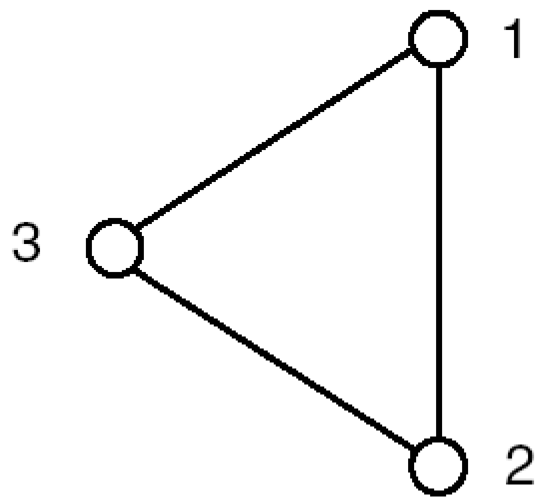
S-ACO importance aspects

- S-ACO importance aspects:
  - Number of ants
  - The value of α
  - Pheromone evaporation rate (p)
  - Type of pheromone update

Experiments with S-ACO

First Experiments with S-ACO

- The experiments were run using the double bridge
- In this model, each arc of the graph has the same length, and a longer branch is represented by a sequence of arcs.



First Experiments

1. First case:
   - Different values for the number m of ants
   - Ants depositing a constant amount of pheromone on the visited arcs ($\Delta\tau^k$ =constant)
2. Second case:
   - Different values for the number m of ants
   - Ants depositing an amount of pheromone is $\Delta\tau^k$ $=1/L^k$, where L^k is the length of ant k's path

First Experiments

- For each experiment we ran **100 trials** and each trial was stopped after each ant had moved **1000 steps** (moving from one node to the next).
- **Evaporation** was set to $p = 0$
- The parameter α was set to 2
- At the end of the trial we checked whether the pheromone trail was higher on the short or on the long path.

Results of First Experiments

- Percentage of trials in which S-ACO converged to the long path

m	1	2	4	8	16	32	64	128	256	512
without path length	50	42	26	29	24	18	3	2	1	0
with path length	18	14	8	0	0	0	0	0	0	0

- The results obtained in experiment 2 with pheromone updates based on solution quality are much better.

Influence of the parameter α

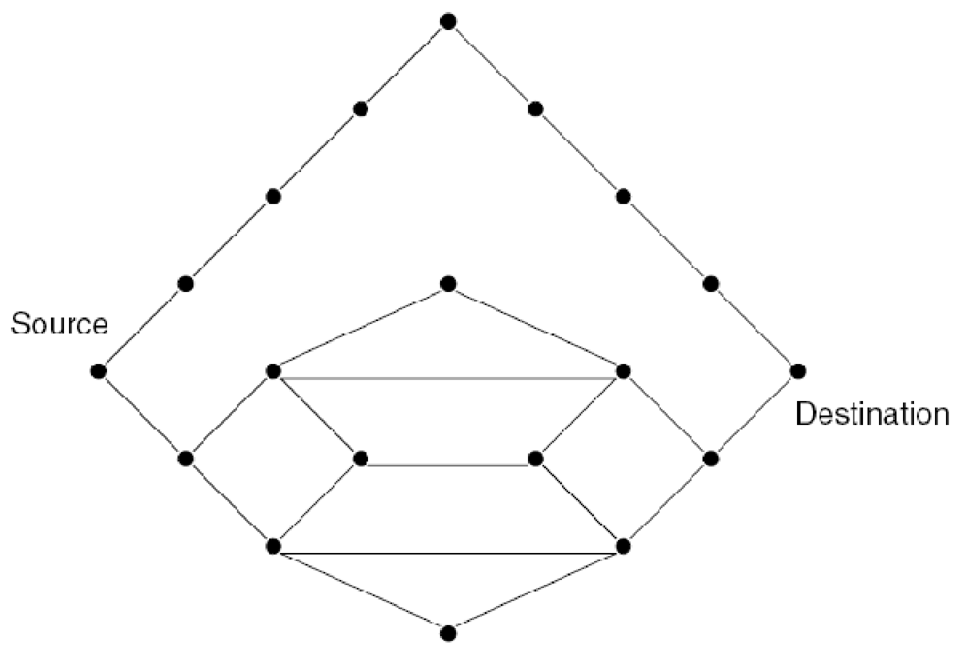
- In additional experiments, we examined the influence of the parameter α on the convergence behavior of S-ACO
- Investigating the cases where a was changed in step sizes of **0.25 from 1 to 2**.
  - In the **first case** we found that **increasing α** had a **negative effect** on the convergence behavior
  - In the **second case** the results were rather **independent** of the particular value of α.

First Experiments

- The results with S-ACO indicate that differential path length alone can be enough to let S-ACO converge to the optimal solution on small graphs
  - at the price of having to use large colony sizes, which results in long simulation times.

Second Experiments with S-ACO

- In a second set of experiments, we studied the influence that pheromone trail evaporation.
- Experiments were run using **the extended double bridge graph**



Second Experiments

- The ants deposit an amount of pheromone that is the inverse of their path length (i.e., $\Delta\tau^k = 1/L^k$)
- Before depositing pheromone, ants eliminate loops

Second Experiments

- We ran experiments with S-ACO and different settings for the evaporation rate:

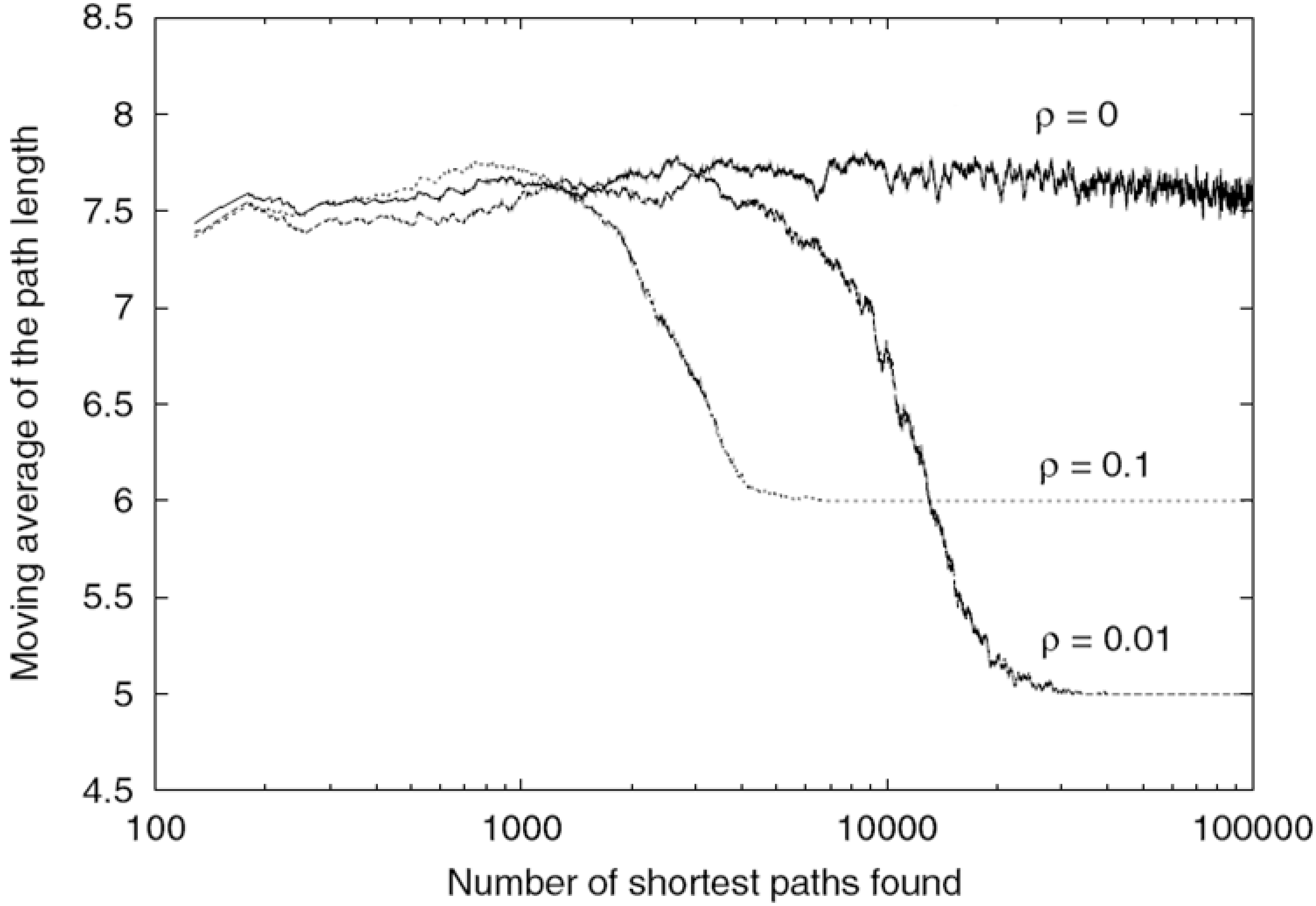
$$\rho \in \{0, 0.01, 0.1\}$$

- $\alpha = 1$ and m = 128 in all experiments.

Plot of Second Experiments

- To evaluate the behavior of the algorithm we observe the development of the path lengths found by the ants.
- We plot the moving averages of the path lengths after loop elimination (moving averages are calculated using the **4 most recent paths** found by the ants).
- In the graph of figure a point is plotted each time an ant has completed a journey from the source to the destination and back

Number of shortest paths found



Pheromone Evaporation

- If p = 0, no pheromone evaporation takes place.
- An evaporation rate of p =0.1 is rather large,
  - Because evaporation takes place at each iteration of the S-ACO algorithm
  - After ten iterations, which corresponds to the smallest number of steps that an ant needs to build the shortest path and to come back to the source, roughly 65% of the pheromone on each arc evaporates,
  - While with p = 0.01 this evaporation is reduced to around 10%.

Results: No evaporation

- If **no evaporation** is used, the algorithm does **not converge**
- It can be seen by the fact that the moving average has approximately the value 7.5, which does not correspond to the length of any path
- With these parameter settings, this result typically **does not change** if the run lasts a much higher number of iterations.

Results: With Evaporation

- With pheromone evaporation, the behavior of S-ACO is significantly different.
- After a short transitory phase, S-ACO converges to a single path
- For $\rho = 0.01$ the value of shortest path is 5
- For $\rho = 0.1$ the path of length is 6

Results: Pheromone Updates

- Without pheromone updates based on solution quality, S-ACO performance is much worse.
- The algorithm converges very often to the suboptimal solution of length 8
- The larger the parameters α or ρ, the faster S-ACO converges to this suboptimal solution.

Results: Pheromone Evaporation Rate

- The pheromone evaporation rate p can be critical.
  - when evaporation was set to a value that was too high,
  - S-ACO often converged to **suboptimal paths**
- For example, in fifteen trials with p set to 0.2, S-ACO converged:
  - once to a path of length 8
  - once to a path of length 7
  - twice to a path of length 6
- Setting p to 0.01 S-ACO converged to the shortest path in all trials.

Results: Values of α

- Large values of α generally result in a worse behavior of S-ACO
- Because they excessively emphasize the initial random fluctuations.

References

References

- M. Dorigo and T. Stützle. **Ant Colony Optimization**, MIT Press, Cambridge, 2004.

The End