Ant Algorithms and their Application in Navigation and Logistics

Anwendungen der Künstlichen Intelligenz

Sebastian Iwanowski
FH Wedel

Kap. 5:
Ameisenalgorithmen und Anwendungen

5.1 (für alle): Einführung und Anwendung in der Navigation

Ant Algorithms and their Applications

1. Motivation: Natural ant systems and their advantages for navigation
2. Details: How ant systems solve the navigation problem

für alle

3. How ant systems solve the TSP problem
4. Generalisations for the TSP problem for logistic applications
5. Further applications in graduation theses at FH Wedel

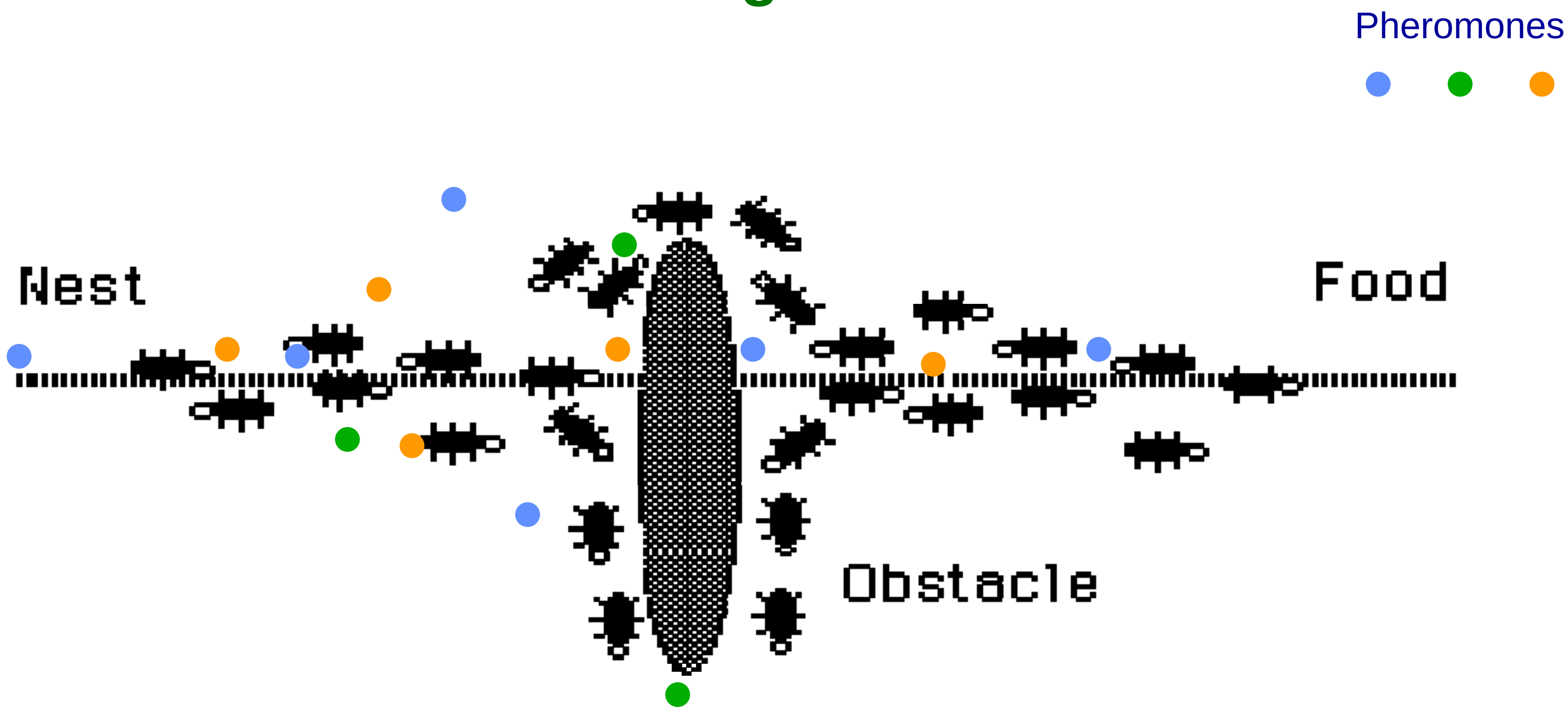
Vertiefung

6. Conclusion: Lessons learnt

für alle

Nature sets the standard

Ants searching for food



Nature sets the standard

Principal concept (nature and simulation)

- Each ant sets pheromones continuously walking on its path.

- At junctions, the probability that an ant decides for a certain direction is proportional to the pheromone concentration towards this direction.

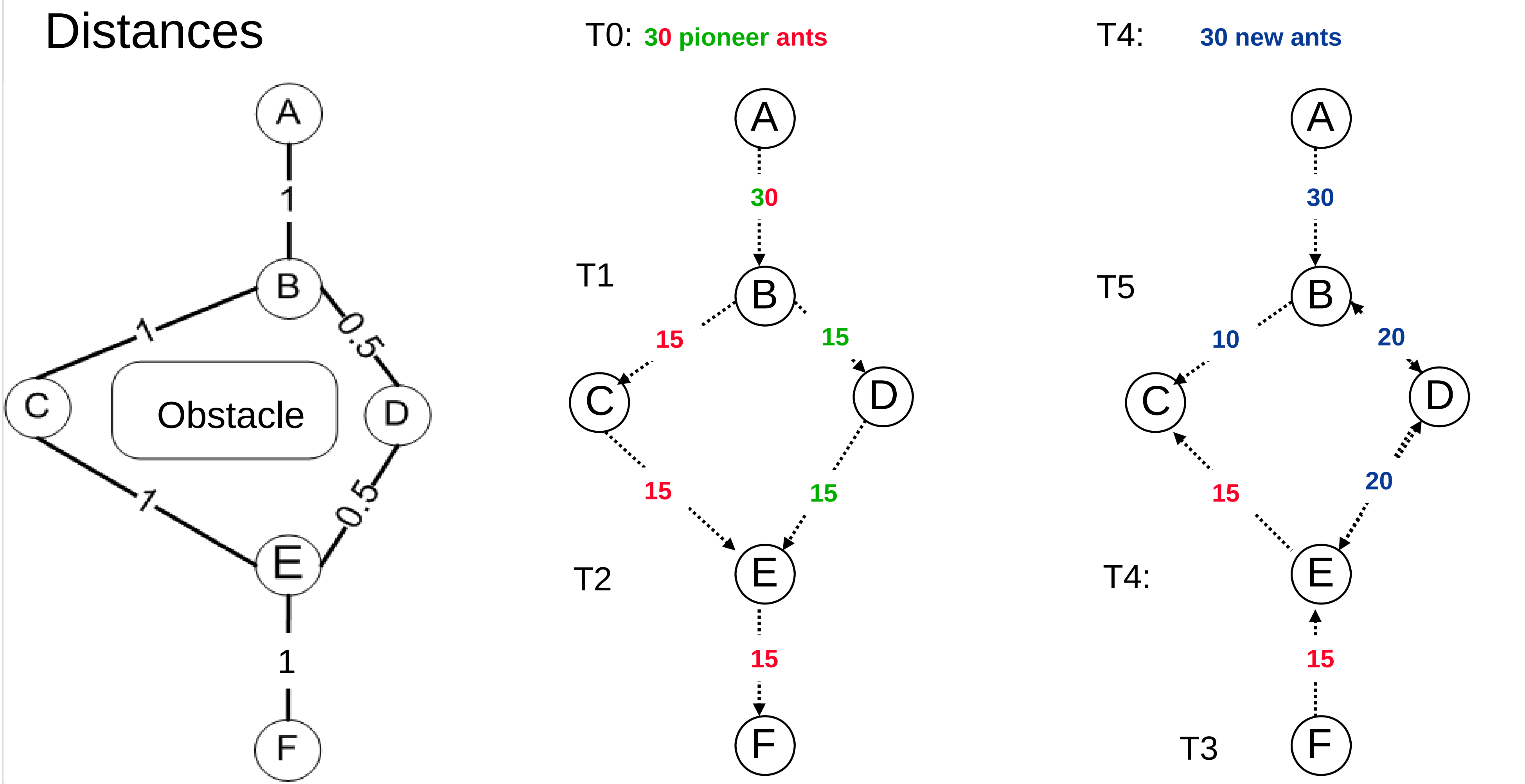
- It makes a difference if an ant is on the search for food or on its return path:

  a) Each ant returns the same path back as it came there (as soon as it found food). ***Simulation***

  b) For either direction different pheromone types are used. ***Nature***

Nature sets the standard

Advantages of probabilistic decision making: Example (alt. a)



Swarm intelligence vs. traditional algorithmics

What do ant systems solve ?

Given the location of some food supplies and the nest:
Find the shortest path between the nest and each food supply!

- Ants do not always find the optimal solution.
- Ants work in a non-deterministic way.
- Simulation of ant systems needs a lot of storage and does not guarantee any time limit.

How does traditional algorithmics solve this problem?

- with deterministic algorithms providing an optimal solution in time quadratic in the number of network nodes (Dijkstra, A*)

Why should we use ant systems ?

Swarm intelligence vs. traditional algorithmics

What do ants better than traditional algorithmics ?

Ants solve the following variant of the path finding problem:

- The parameters of the underlying graph are only known in a local scope.
- Edge costs may change any time (without the possibility of prediction).

This problem is the typical problem of road navigation.

Traditional algorithmics does not deal with such unspecified problems!

This is the chance for the applications of ant systems !

Ant Algorithms and their Applications

1. Motivation: Natural ant systems and their advantages for navigation
2. Details: How ant systems solve the navigation problem

für alle

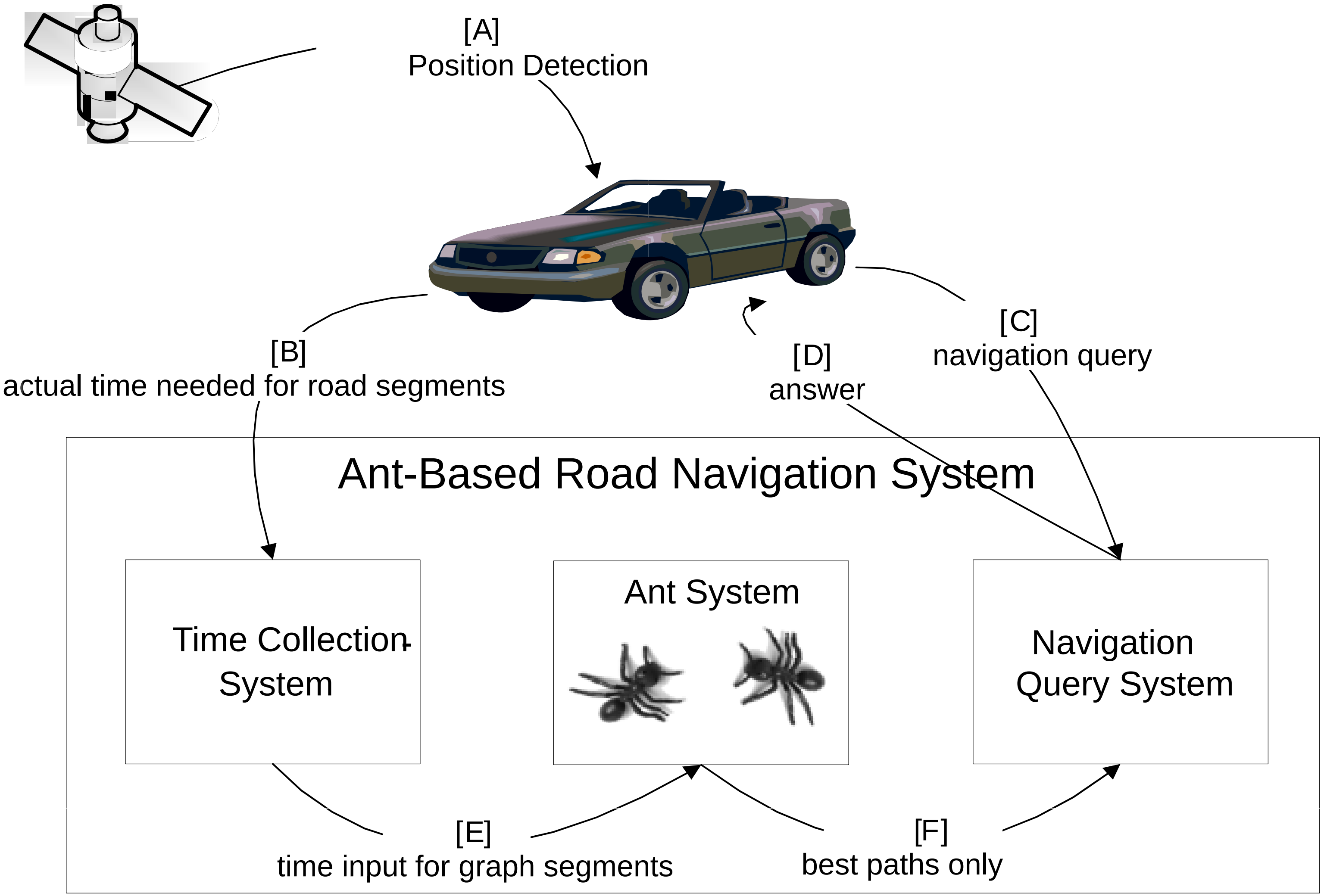
3. How ant systems solve the TSP problem
4. Generalisations for the TSP problem for logistic applications
5. Further applications in graduation theses at FH Wedel

Vertiefung

6. Conclusion: Lessons learnt

für alle

Street navigation: General Software Architecture



proposed in master thesis of Thomas Walther, 2006 (in German)

Ant Colony Optimization (ACO)

Developments by several research groups

AntNet (sophisticated, works in practice)

Dorigo M., G. Di Caro & L. M. Gambardella (1999). Ant Algorithms for Discrete Optimization. *Artificial Life,* 5(2):137-172.

http://iridia.ulb.ac.be/~mdorigo/ACO/ACO.html

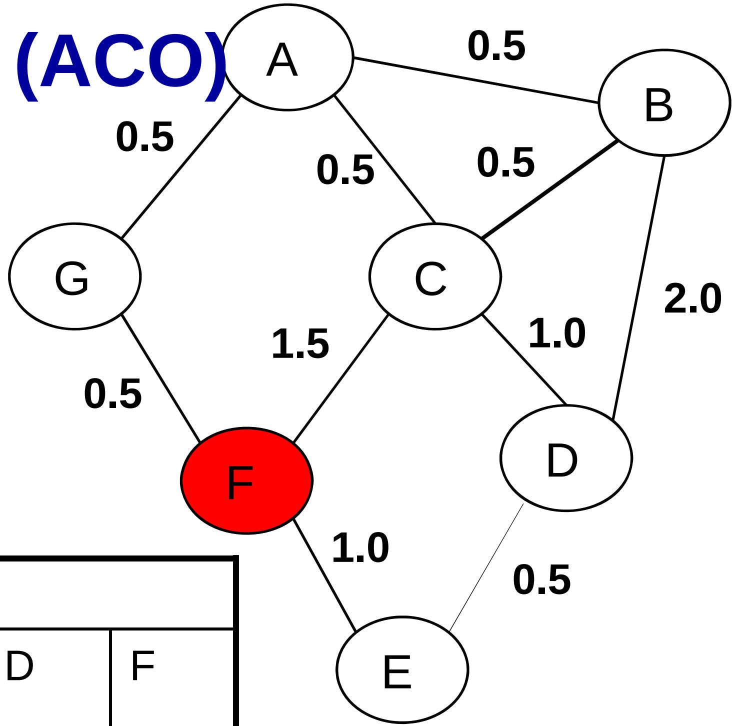
Gianni Di Caro, An Introduction to Swarm Intelligence and Metaheuristics for Combinatorial Optimization: lecture slides http://www.idsia.ch/~gianni/my_lectures.html

Ant Based Control (ABC) (easy to explain, works only for small systems)

Kroon R., *Dynamic vehicle routing using Ant Based Control*, Master‘s thesis, Delft University of Technology, 2002.

R. Schoonderwoerd, O. Holland, and J. Bruten. Ant-like agents for load balancing in telecommunications networks. In *Proceedings of the First International Conference on Autonomous Agents*, pages 209–216. ACM Press, 1997.

Ant Colony Optimization (ACO)



Base for path decision:

Pheromones in routing table for each node:

Examples for nodes F and C:

table F			
Dest \ Next	C	G	E
A	0.3	0.65	0.05
B	0.5	0.35	0.15
C	0.9	0.05	0.05
D	0.9	0.05	0.05
E	0.05	0.05	0.9
G	0.6	0.35	0.05

table C				
Dest \ Next	A	B	D	F
A	0.7	0.1	0.1	0.1
B	0.05	0.85	0.05	0.05
D	0.05	0.05	0.85	0.05
E	0.25	0.05	0.05	0.65
F	0.15	0.05	0.05	0.75
G	0.6	0.05	0.05	0.3

This need not necessarily correspond to current traffic situation !

Ant Colony Optimization (ACO)

Algorithmic processing

Alternating phases:
Construction of a route and update of pheromone values

Continuously, ants are generated from each source to each destination

Tasks of an ant running from ist source to its destination (forward ant phase):

- At each intersection, choose next edge probabilistically (according to current table entries)
- Collect and store the encountered information (edge lengths, etc.)
- Start the individual pheromone update phase for this ant when destination is reached

Tasks of the pheromone update for a single ant (backward ant phase):

- Trace back the path the corresponding ant used
- Update node infos according to the real-time information the forward ant collected

The ABC strategy for pheromone update

$$\Delta P_{s,d} = \frac{c_1}{t_{s,d}} + c_2$$

Evaporation coefficient:

This number is used to confirm the path, the ant has really used, and – simultaneously – to deminish the paths, the ant has NOT used.

Evaporation of pheromones for edges not used

$$P_{d,i} = \frac{P_{d,i}}{1 + \Delta P_{s,d}} \quad \forall\, i \neq f$$

Confirmation of pheromones for edges used

$$P_{d,f} = \frac{P_{d,f} + \Delta P_{s,d}}{1 + \Delta P_{s,d}}$$

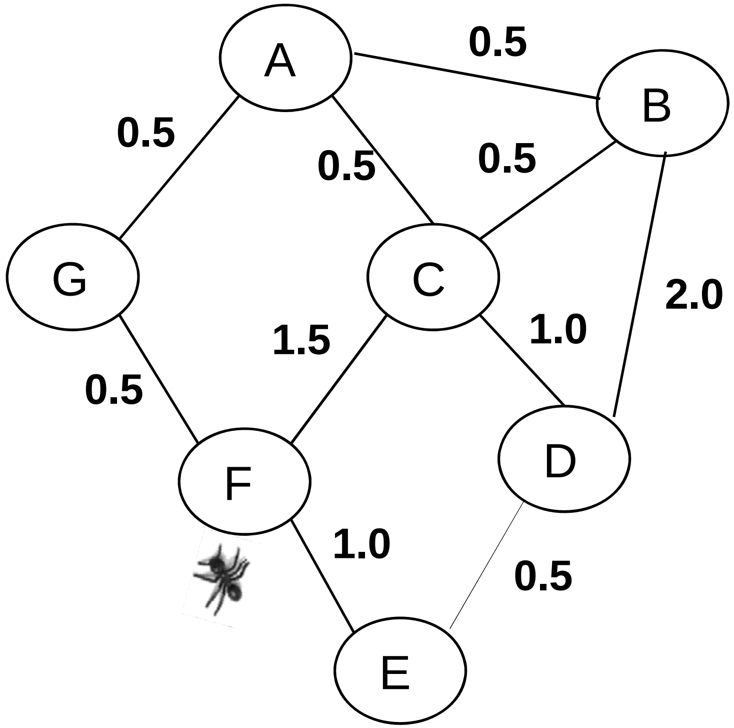
s ... source of ant

d … destination of ant

F … node which was next for ant in order to reach destination

Simple example for ABC pheromone update

Constructing the route (forward ant phase)



memory

s = F d = B

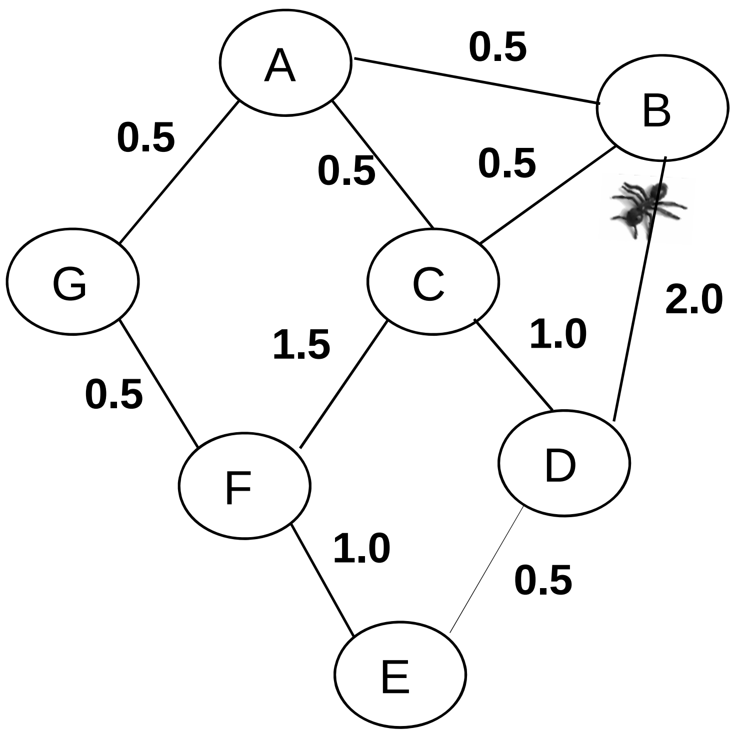
$t_{F,C}$=1.5 $t_{C,B}$=0.5

Table for F (extract)			
Next / Dest	C	G	E
B	0.5	0.35	0.15

Table for C (extract)				
Next / Dest	A	B	D	F
B	0.05	0.85	0.05	0.05

Simple example for ABC pheromone update

Updating the pheromones (backward ant phase):



memory

s = F d = B

$t_{F,C}$=1.5 $t_{C,B}$=0.5

Old Table for C (extract)				
Dest \ Next	A	B	D	F
B	0.05	0.85	0.05	0.05

New Table for C (extract)				
Dest \ Next	A	B	D	F
B	0.01	0.97	0.01	0.01

$t_{C,B} = 0.5$

$$\Delta P = \frac{2}{0.5} + 1 = 5$$

$$P_{new,A} = \frac{0.05}{1+5} = 0.01$$

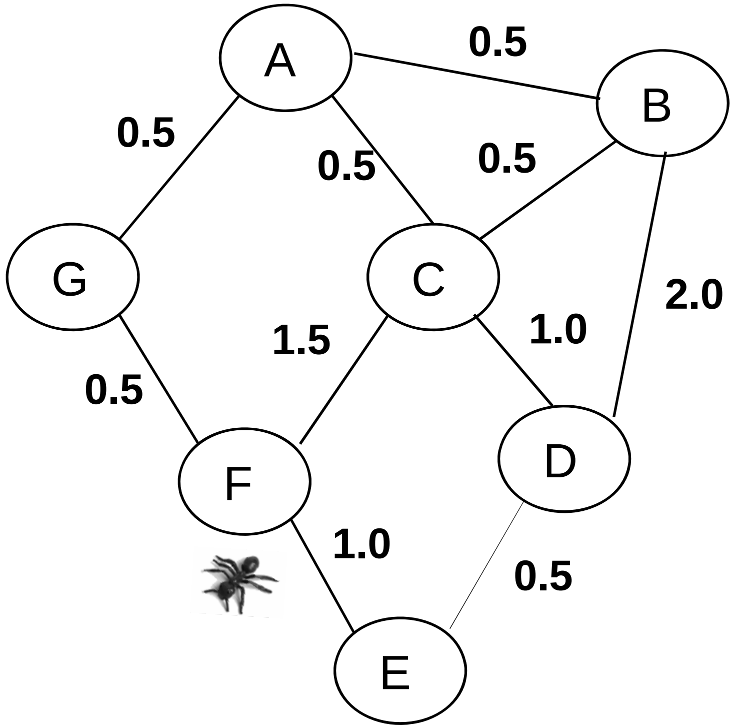
$$P_{new,B} = \frac{0.85+5}{1+5} = 0.97$$

$$P_{new,D} = \frac{0.05}{1+5} = 0.01$$

$$P_{new,F} = \frac{0.05}{1+5} = 0.01$$

Simple example for ABC pheromone update

Updating the pheromones (backward ant phase):



memory

$s = F \quad d = B$

$t_{F,C}=1.5 \qquad t_{C,B}=0.5$

$t_{F,B} = 2$

Old Table for F (extract)

Dest \ Next	C	G	E
B	0.5	0.35	0.15

New Table for F (extract)

Dest \ Next	C	G	E
B	0.83	0.12	0.05

$$\Delta P = \frac{2}{2}+1$$

$$P_{new,C} = \frac{0.5+2}{1+2} = 0{,}83$$

$$P_{new,G} = \frac{0.35}{1+2} = 0{,}12$$

$$P_{new,E} = \frac{0.15}{1+2} = 0{,}05$$

System design issues: Distributing the ant system

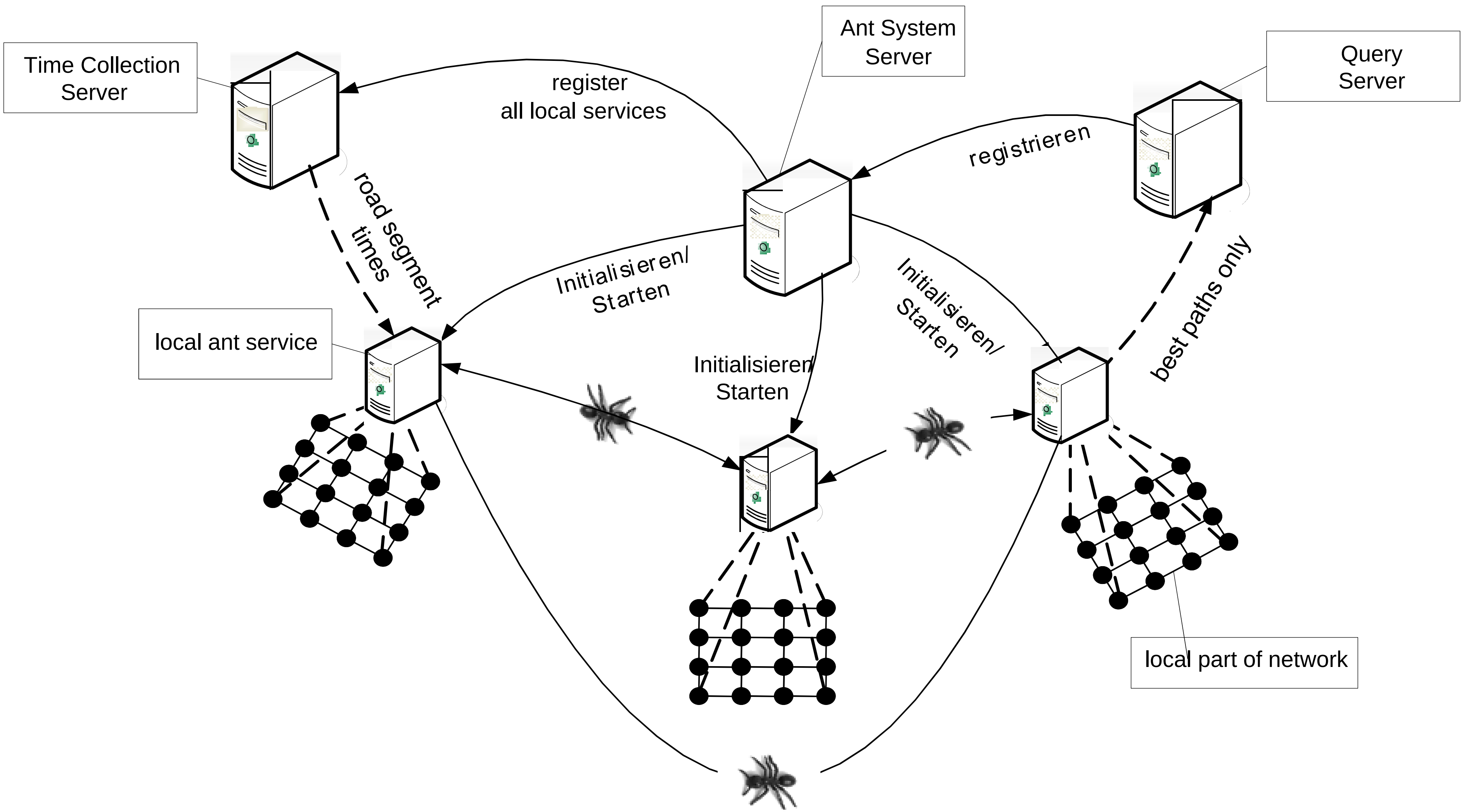
Big networks are a problem:

- Amount of ants grows quadratic in network size
- More ants produce a higher computing effort

Solution:

- Distribute computing load to several peer services
- This is enabled by indirect and local communication

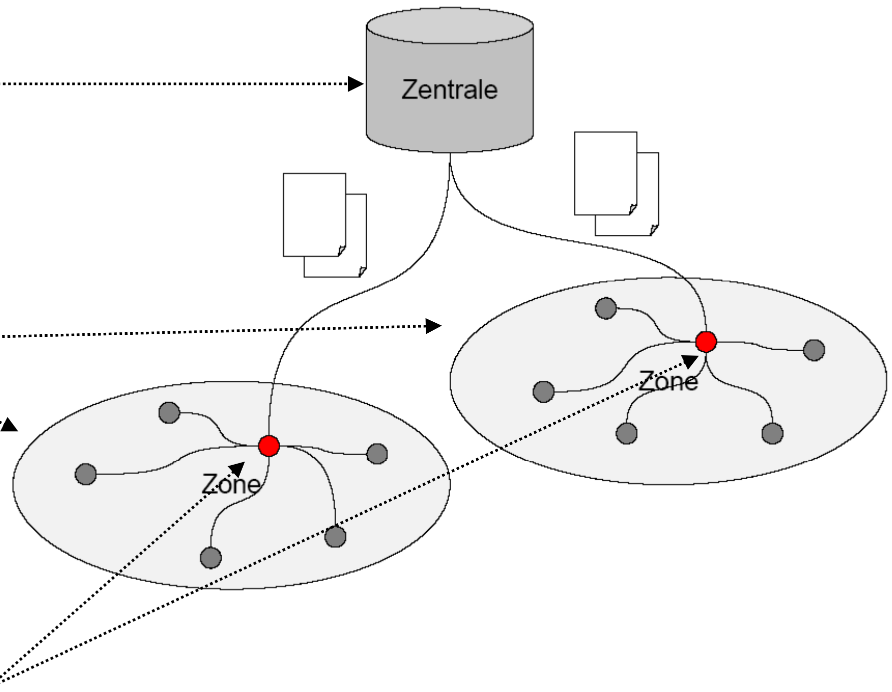
System design issues: Distributing the ant system



Mobile use of pheromones

There the ant algorithm is performed.

Pheromones of respective zone are distributed to all vehicles residing in respective zone.



Only very few vehicles ask directly at central server.

discussion und details in master thesis of Michael Suthe, 2007 (in German)

Advantages to traditional navigation methods

Compressed storage of data:

- Pheromones store compressed information collected from many segments.
- This enables feasible mobile distribution.

Concurrent computation of data:

- Ant system computes pheromones off-board.
- This enables greater computing capacities.
- **Eager computing before actual query → quicker reply**.

Middleware connecting data collection and data use:

- Dynamic data is collected from all of the country and distibuted to each user.
- Central server is mediator between data collectors and data users.
- This grants the general advantage of each middleware:
  Less communication effort between providers and users.

Ant Algorithms and their Applications

1. Motivation: Natural ant systems and their advantages for navigation
2. Details: How ant systems solve the navigation problem

für alle

3. How ant systems solve the TSP problem
4. Generalisations for the TSP problem for logistic applications
5. Further applications in graduation theses at FH Wedel

Vertiefung

➡ 6. Conclusion: Lessons learnt

für alle

Ant Algorithms and their Applications

When should ant algorithms be applied ?

- For navigation only in the dynamic case: When rapid and unexpected changes matter

- For logistics even in the static case: When the computation of the mathematically best solution is not feasible or not necessary