

Künstliche Intelligenz

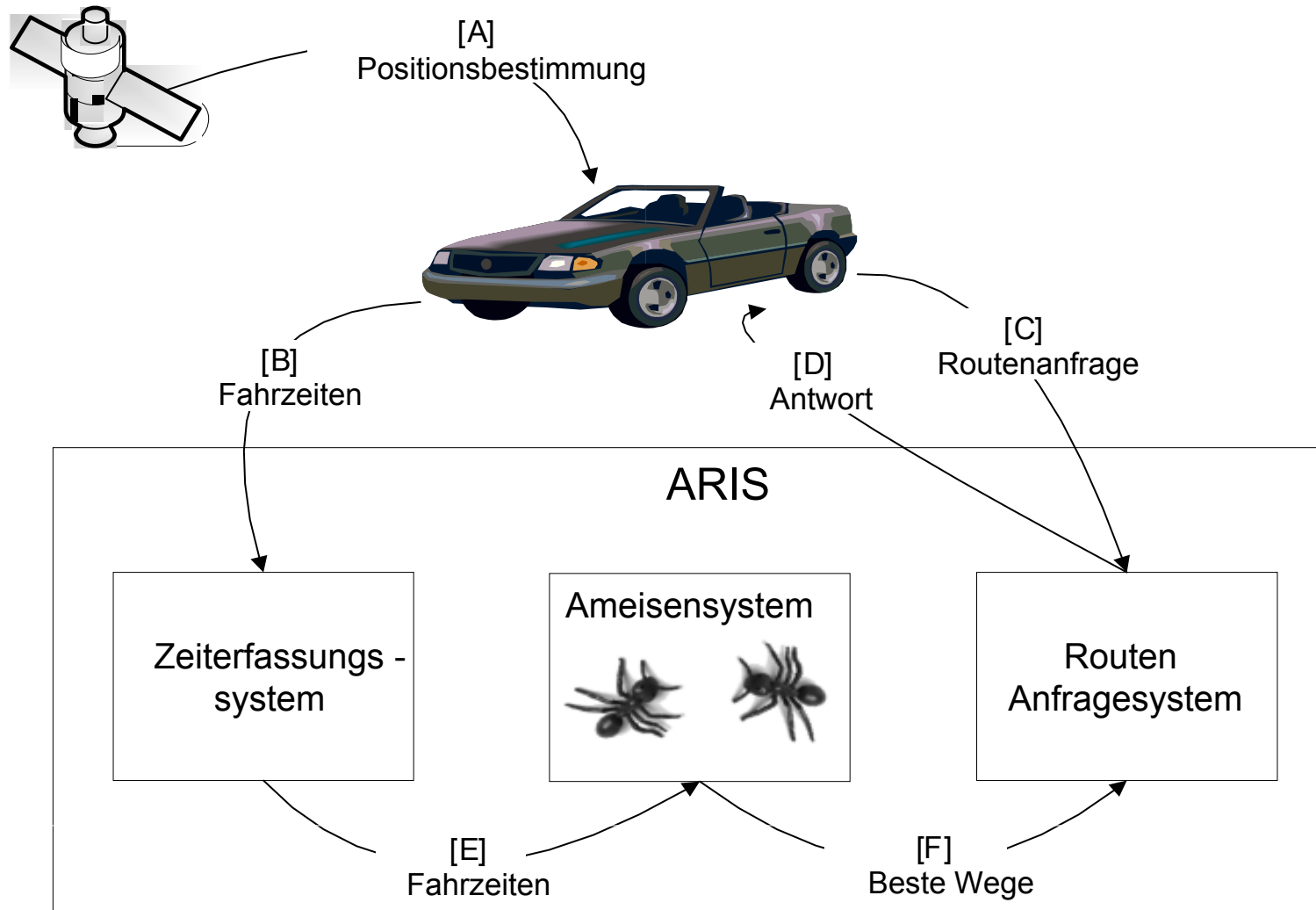
Sebastian Iwanowski
FH Wedel

Kap. 7:
Ameisenalgorithmen

7.2: Anwendung auf Dynamische Verkehrsnavigation

*Einige der hier vorgestellten Folien stammen aus früheren Präsentationen
von **Thomas Walther**, M.Sc. der FH Wedel*

Systemdesign: Systemkomponenten



ARIS = Ameisenbasiertes RoutenInformationssystem

Systemdesign: Systemkomponenten

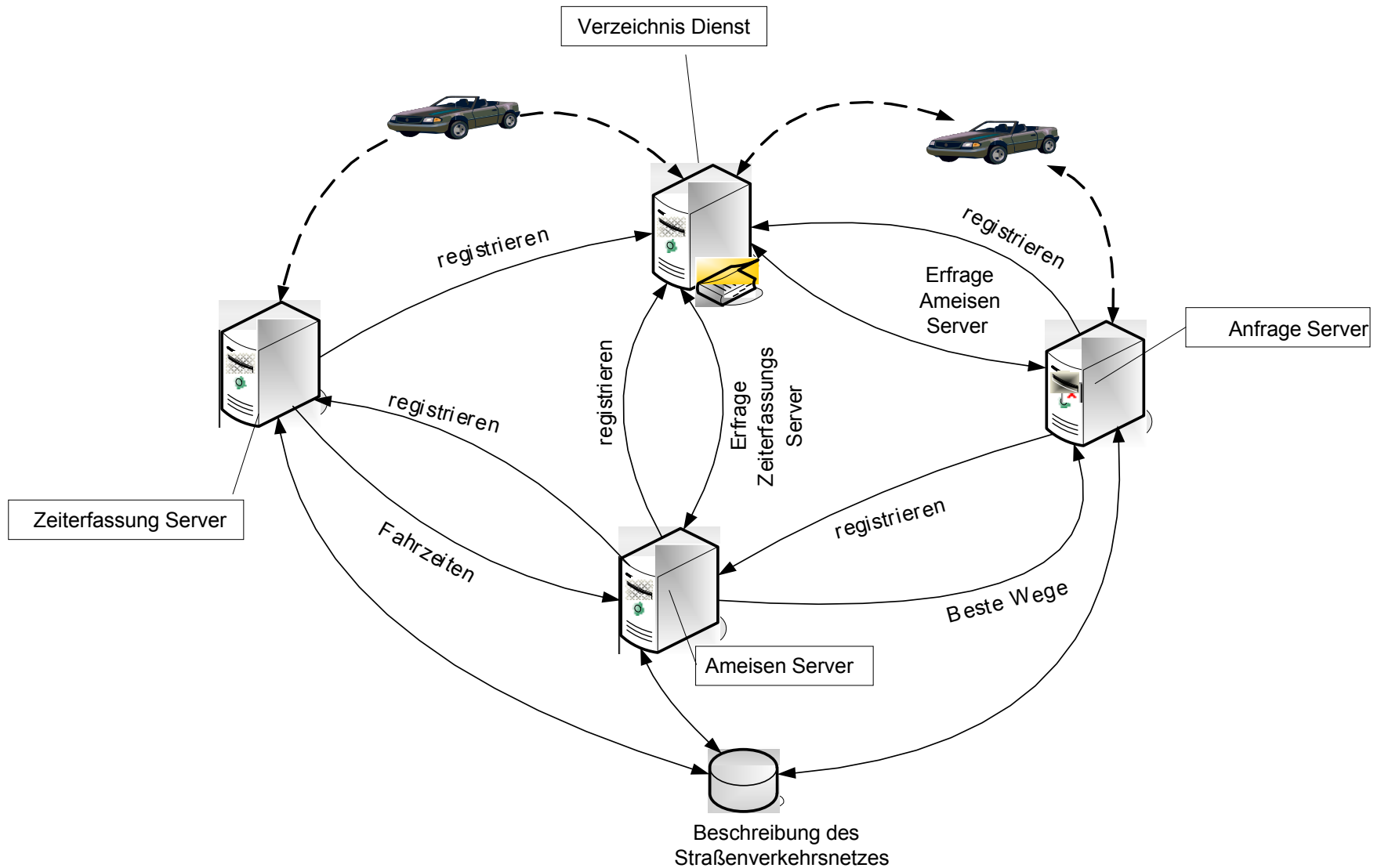
Zeiterfassungssystem

- Abbildung der Verkehrslage für die einzelnen Wegstrecken
- Verschiedene Datenquellen: Induktionsschleifen, Kameras, Fahrzeuge, ...

Fahrzeuge als Datenquelle

- Komplexe Erfassung notwendig
- Problematik geringer Informationsdichten
- Abgelegt werden die gemittelten Fahrzeiten (über ein Zeitfenster)

Systemdesign: Verteilung der Systemkomponenten



Systemdesign: Verteilung des Ameisensystems

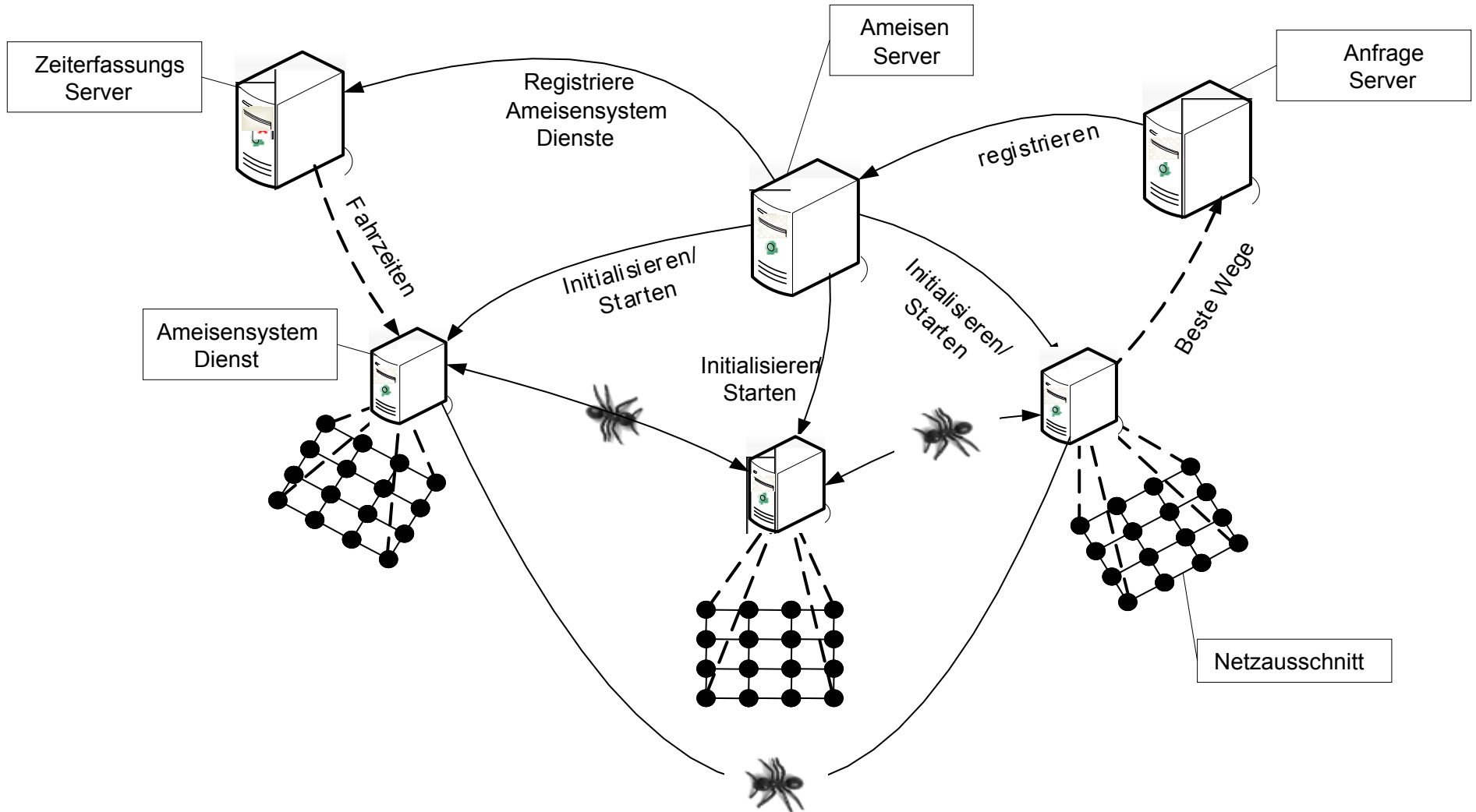
Große Netze sind problematisch:

- Je größer das Netz desto mehr Ameisen benötigt
- Je mehr Ameisen desto größer ist die Rechenlast

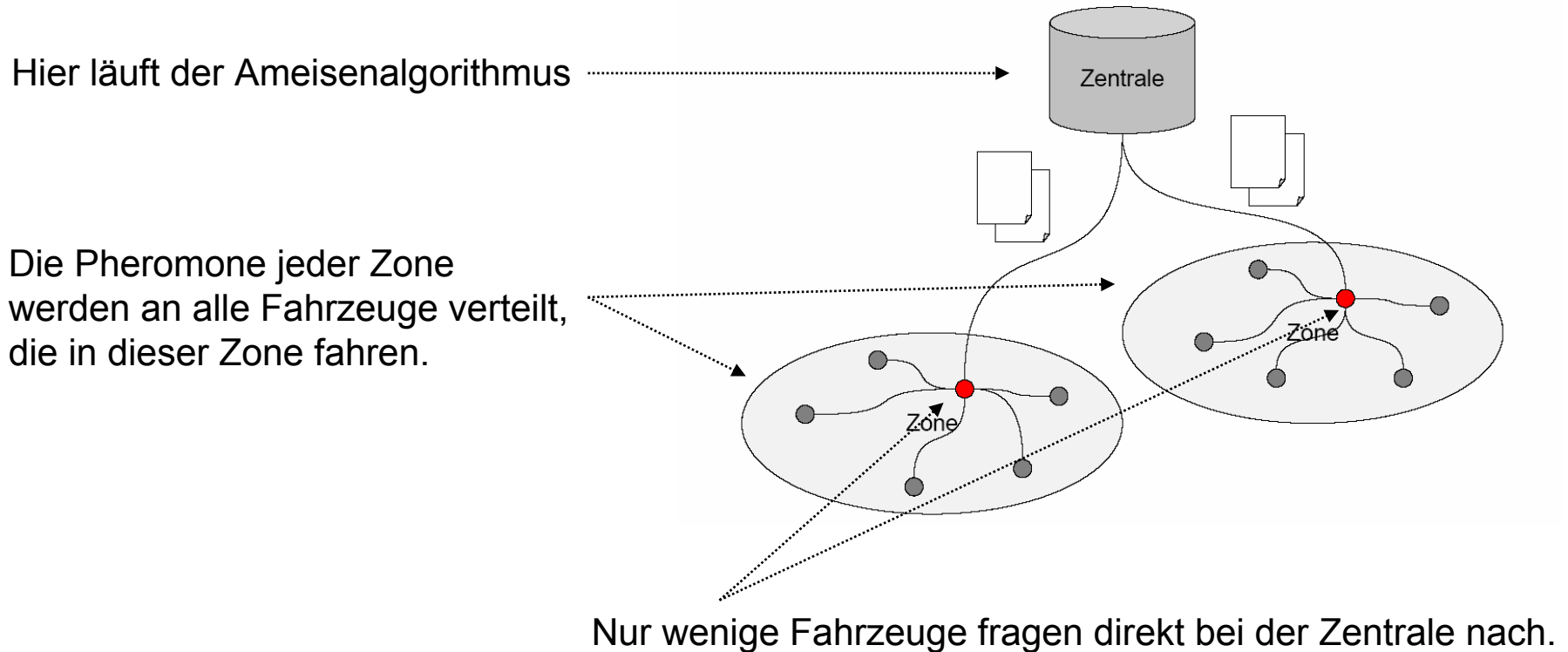
Lösung:

- Verteilung der Rechenlast auf gleichartige Dienste
- Begünstigt durch indirekte und lokale Kommunikation

Systemdesign: Verteilung des Ameisensystems



Mobile Nutzung der Pheromontabellen



Diskussion und Details in Masterarbeit Michael Suthe

Nutzen gegenüber herkömmlichen Navigationsverfahren

Komprimierte Speicherung der Daten:

- Pheromone speichern die zusammengefasste Situation von vielen Einzelstrecken
- Übertragungsdatenmenge wird dadurch beherrschbar.

Nebenläufige Berechnung der Daten:

- Ameisensystem errechnet die Daten außerhalb der Fahrzeuge
- Dadurch sind größere Rechnerkapazitäten einsetzbar.
- Die Berechnung kann bereits vor der Abfrage erfolgen.

Middleware zwischen Datenerfassung und Datennutzung:

- Dynamische Daten werden vom ganzen Land zentral erfasst und weiterverteilt.
- Die Zentrale fungiert als Middleware zwischen Datensammlern und Datennutzern
- Dadurch erhält man den Vorteil jeder Middleware:
weniger Kommunikationsaufwand zwischen vielen Providern und vielen Nutzern.

Künstliche Ameisenverfahren

Entwicklungen verschiedener Forschergruppen

AntNet

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)

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.

Künstliche Ameisenverfahren

Grundlage für Wegentscheidung:

Routentabelle für jeden Knoten

Beispieltabellen für 2 Knoten:

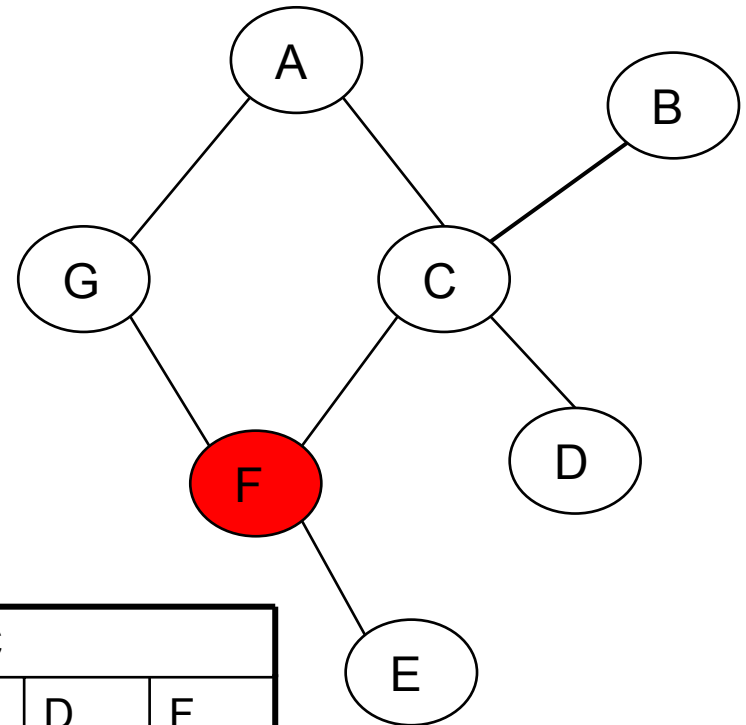


Tabelle F				
Next \ Dest	C	G	E	
A	0.3	0.65	0.05	
B	0.5	0.35	0.15	
C	0.9	0.05	0.05	
D	0.9	0.05	0.05	
E	0.05	0.05	0.9	
G	0.6	0.35	0.05	

Tabelle C					
Next \ Dest	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	

Künstliche Ameisenverfahren

Algorithmischer Ablauf

Vorwärts- und Rückwärtsameisen

Kontinuierliche Generation von Vorwärtsameisen von jeder Quelle zu jedem Ziel

Funktionsweise der Vorwärtsameisen:

- Wahrscheinlichkeitsgesteuerte Wegwahl (gemäß aktuellen Tabellenwerten)
- Sammeln und Merken von Weginformationen (Wegzeiten etc.)
- Starten einer Rückwärtsameise am Ziel

Funktionsweise der Rückwärtsameisen:

- Rückverfolgung des Wegs der Vorwärtsameise
- Aktualisierung der Knoteninformation mit Hilfe der Weginformationen der Vorwärtsameise

Details eines Ameisenverfahrens (ABC)

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

Verdampfungskoeffizient:

Mit diesem lässt die Rückwärtsameise die Pheromone der anderen Wege „verdampfen“ und verstärkt gleichzeitig den eigenen Weg.

Verdampfung der Pheromone

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

Erhöhung der Pheromonkonzentration

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

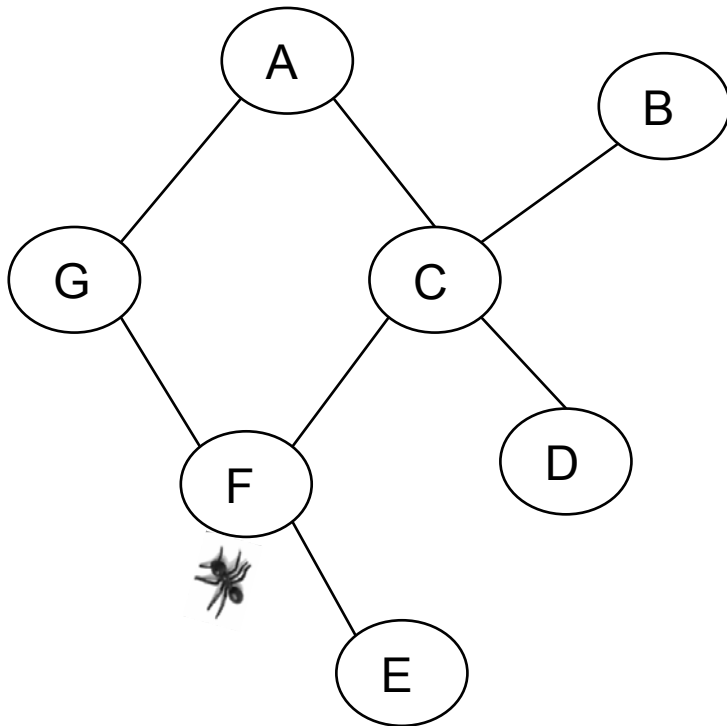
s ... Startknoten der Vorwärtsameise

d ... Zielknoten der Vorwärtsameise

f ... Knoten über den die Rückwärtsameise den aktuellen Knoten erreicht hat

Details eines Ameisenverfahrens (ABC)

Verhalten der Vorwärtsameise



Gedächtnis

$s = F$ $d = B$

$\{k, t_k\} = \{F, 0.5\} \quad \{C, 1.5\}$

Tabelle F (verkürzt)

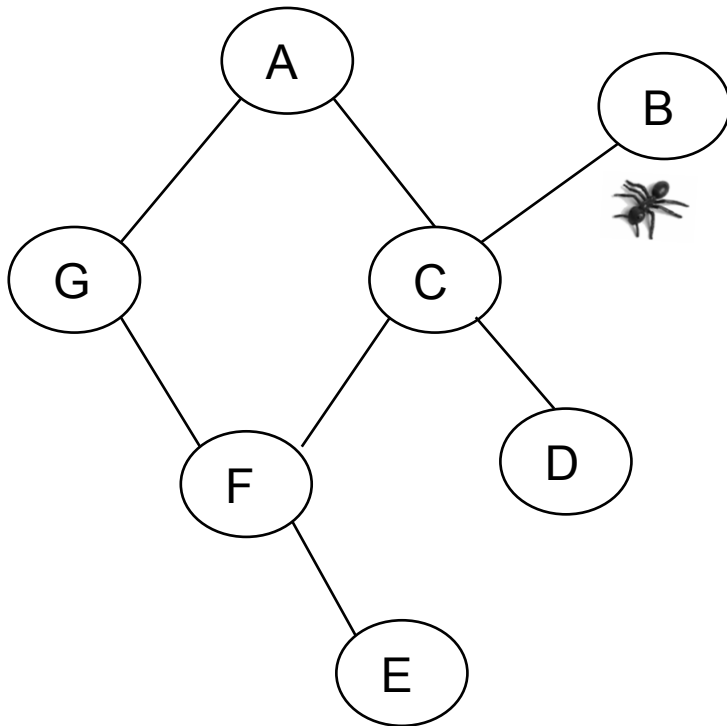
Next \ Dest	C	G	E
B	0.5	0.35	0.15

Tabelle C (verkürzt)

Next \ Dest	A	B	D	F
B	0.05	0.85	0.05	0.05

Details eines Ameisenverfahrens (ABC)

Verhalten der Rückwärtsameise



Gedächtnis

$$s = F \quad d = B$$

$$\{k, t_k\} = \{F, 0.5\} \quad \{C, 1.5\}$$

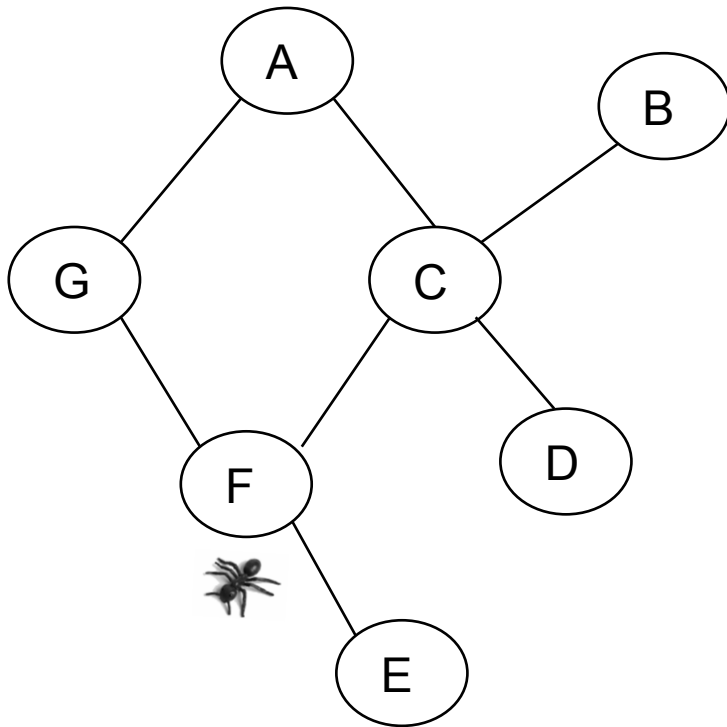
Tabelle C (verkürzt)

Next \ Dest	A	B	D	F
B	0.05	0.85	0.05	0.05

Pheromonaktualisierung verändert die Tabelle des Knotens C nicht, da als minimaler Wert für jeden Eintrag 0.05 erhalten bleiben soll.

Details eines Ameisenverfahrens (ABC)

Verhalten der Rückwärtsameise



Gedächtnis

$$s = F \quad d = B$$

$$\{k, t_k\} = \{F, 0.5\} \quad \{C, 1.5\}$$

Alte Tabelle F (verkürzt)			
Next	C	G	E
Dest			
B	0.5	0.35	0.15

Neue Tabelle F (verkürzt)			
Next	C	G	E
Dest			
B	0.83	0.12	0.05

$$t_{F,B} = 2$$

$$\Delta P_{F,B} = \frac{2}{2} + 1$$

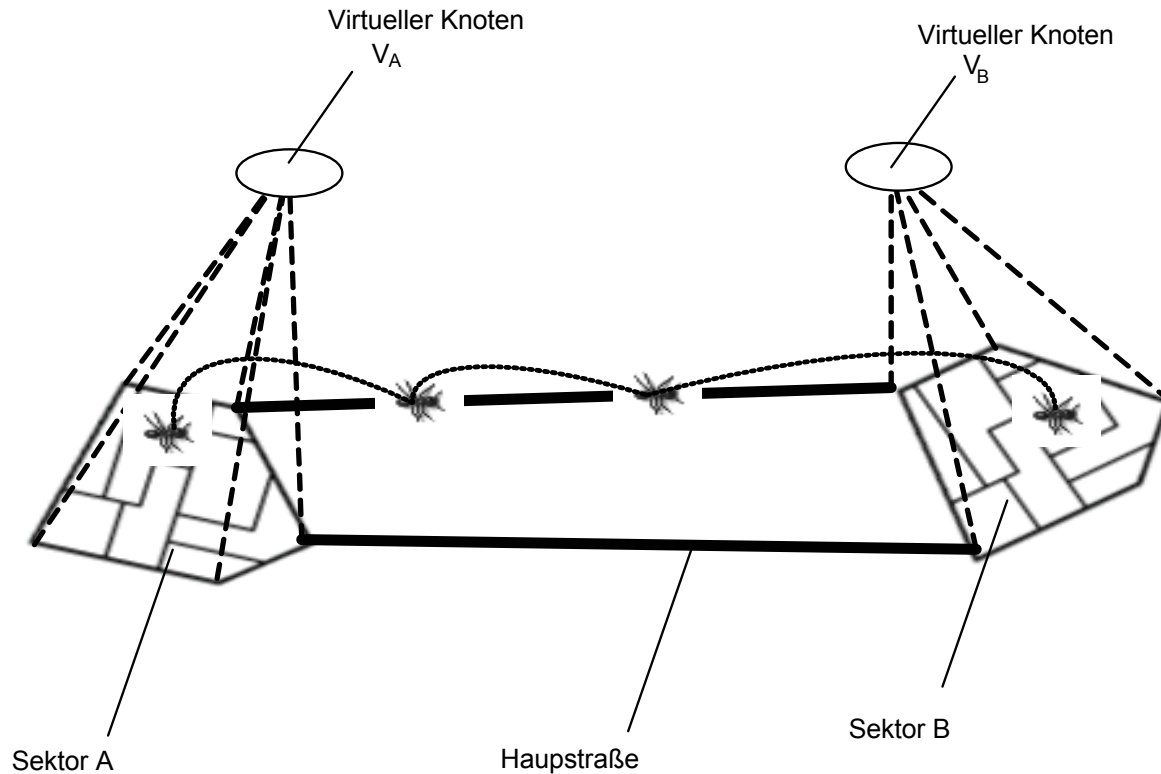
$$P_{\text{new},C} = \frac{0.5 + 2}{1 + 2} = 0,83$$

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

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

Verteiltes Ameisensystem: Hierarchische Verfahren

Gleichmäßiger top-down-Ansatz*



- Sektor-Level-Knoten
- Grenzknoten
- Top-Level-Knoten

Routentabelle
Grenzknoten

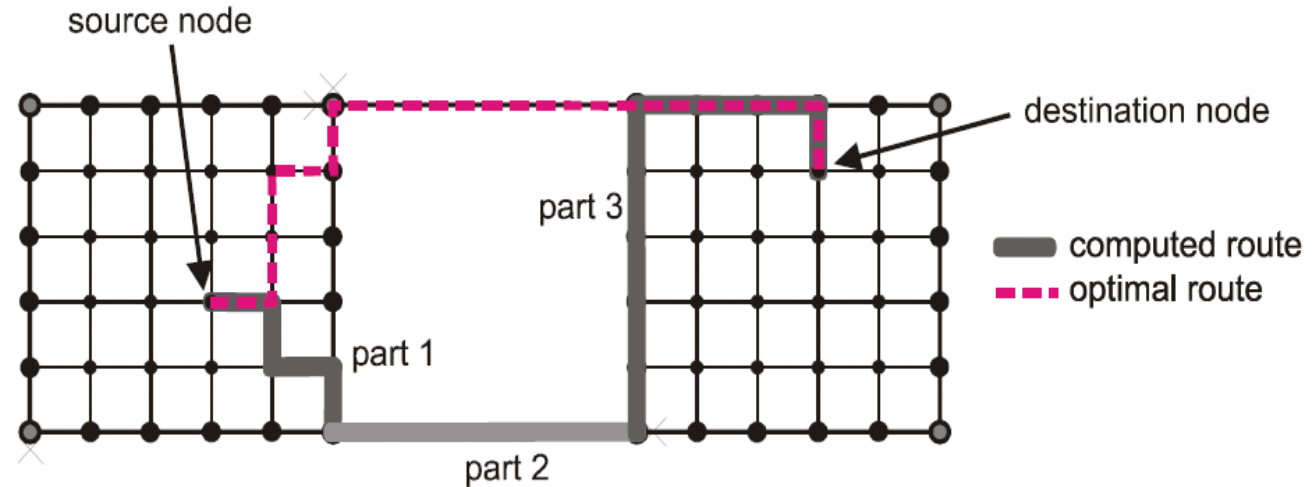
	s_1	s_2	s_n	T_1	T_k	V_1	V_m
s1	0.7	-	0.2	-	-	0.3	-
s2	0.8	-	-	-	-	0.2	-
...							
sn	0.2	-	-	0.7	0.6	0.8	-
T1	-	0.1	-	0.1	0.2	0.1	0.1
...							
Tk	-	0.2	-	0.6	0.2	0.1	0.1
V1	-	0.6	-	0.2	0.2	0.1	0.1
...							
Vm	-	0.6	-	0.2	0.2	0.1	0.1

*Henrik Dibowski: Hierarchical routing system using ant based control, TU Delft (NL) / TU Dresden, Diplomarbeit, 2003

Verteiltes Ameisensystem: Hierarchische Verfahren

Gleichmäßiger top-down-Ansatz

Problem:



Problembeschreibung bereits in Diplomarbeit Dibowski, wiederaufgegriffen in Masterarbeit Walther

Verbesserungen zur Lösung dieses Problems:

Masterarbeit Walther, S. 52-54

- 1) Erweiterung der Routentabelle: Explizite Erwähnung der entfernten Sektor-Level-Knoten
- 2) Verwendung von entfernten Zuordnungen Grenzknoten – Sektor-Level-Knoten

Verteiltes Ameisensystem: Hierarchische Verfahren

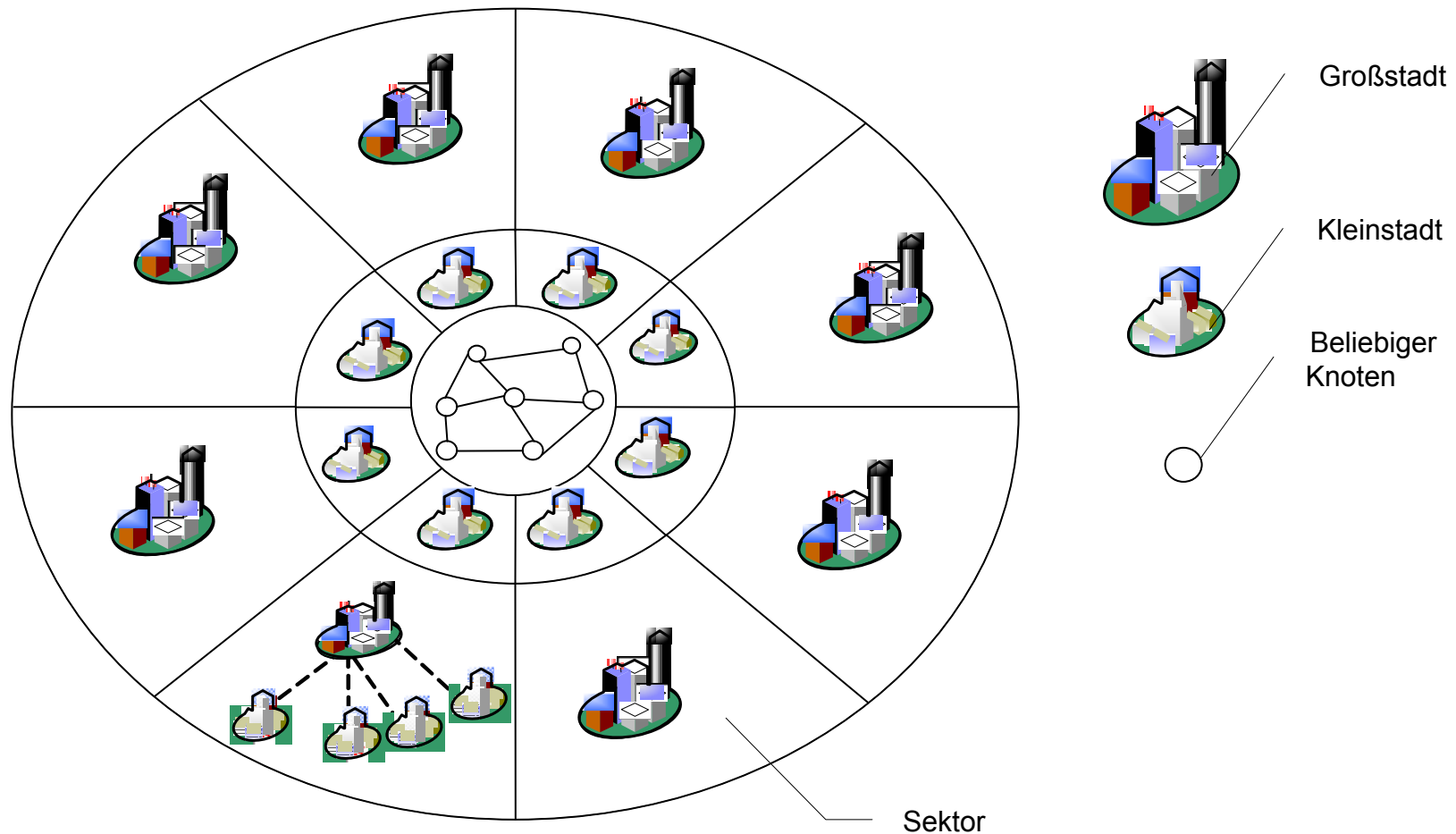
Unscharfes Routen

Konzeptionelle Idee:

- *Genaue* Wegwahl für nahe Ziele
- *Unschärfe* Wegwahl für entfernte Ziele
- Definition von Orientierungspunkten
- Knotenspezifische Sektordefinitionen
- Keine Wegwahleinschränkung

Verteiltes Ameisensystem: Hierarchische Verfahren

Unscharfes Routen



Masterarbeit Walther, S. 54-60

Verteiltes Ameisensystem: Hierarchische Verfahren

Unscharfes Routen

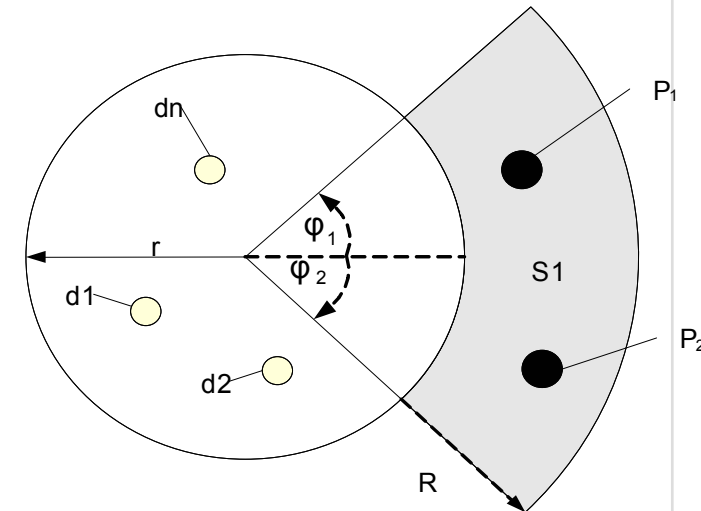
Wegwahl der Ameisen bleibt für nahe Ziele unverändert

Wegwahl für entfernte Sektoren:

- Zielknoten ist ein Orientierungspunkt
- An jedem Knoten ändert sich die Sicht
- Sektordefinitionen ändern sich

Vorteile:

- Verringerung der Anzahl der möglichen Ziele
- Der bester Weg zu einem Sektor gilt für alle Knoten des Sektors



	S_3	S_4
d1	0.3	0.7
d2	0.5	0.5
...		
dn	0.4	0.6
...		
S1, P1	0.65	0.35
S1, P2	0.4	0.6
...		

$$S1 = \{ \varphi_1, \varphi_2, r, R \}$$

Zusammenfassung der Masterarbeit von T. Walther

Ameisenalgorithmen

- Einzelne Individuen sind sehr einfach
- Kollektives Verhalten ermöglicht die Lösung komplexer Probleme

Konzeptionelle Verbesserungen der Masterarbeit

- Beschränkung der Zielknoten und Wege
- Erweiterung der hierarchischen Verfahren

Praktische Ergebnisse der Masterarbeit

- Implementierung eines Verkehrssimulators
- Implementierung des AntNet – Algorithmus
- Prototyp für ein Routeninformationssystem als verteiltes System
- Funktionsdemonstration mit zeilenbasierter Interaktion

Aufgaben für zukünftige Arbeiten

Konzeptionelle Arbeiten

- Ausgestaltung des Konzepts „Unscharfes Routen“
- Fahrzeiterfassung
- Personalisierte Verkehrsinformation:
 - Fahrzeugklassen berücksichtigen
 - Integration/Kombination verschiedener Optimierungsziele
 - Integration von historischen Daten

Praktische Arbeiten

- Verwendung eines besseren Verkehrssimulators (bessere Bedienungsfläche)
- Implementierung hierarchischer Verfahren
- Anbindung einer realen Fahrzeiterfassung