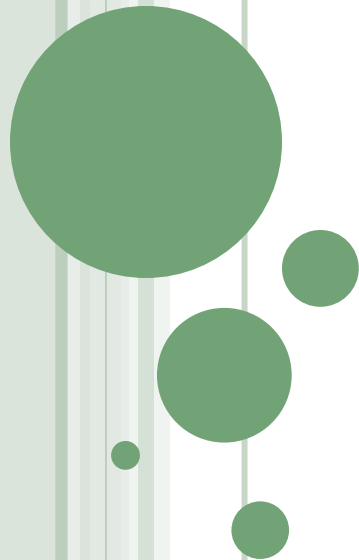


# **VERKEHR UND LOGISTIK**

## **THEMA 09**

### **FUNKTIONSWEISE VON AMEISENSYSTEMEN BEIM VEHICLE ROUTING PROBLEM**

**Timo Höltgen**



# GLIEDERUNG

- Erinnerung
- Traveling Salesman Problem
- ACO für das Traveling Salesman Problem
- ACO für das Vehicle Routing Problem
- Praxisbeispiel

# ZUR ERINNERUNG

## WAS GENAU WAR NOCHMAL ACO?

- Ant Colony Optimization (ACO) bezeichnet verschiedene Algorithmen, welche künstliche Ameisen zur Lösung eines kombinatorischen Optimierungsproblems verwenden.
- Vorbild sind echte Ameisen auf der Suche nach Futter.

## WAS GENAU WAR NOCHMAL ACO?

- Ameisen hinterlassen Pheromone.
- Andere Ameisen folgen bevorzugt diesen Spuren.
- Kurze Wege werden schneller abgelaufen und somit öfter besprüht → effiziente Wege wirken attraktiver.
- Nicht alle Ameisen folgen den gelegten Spuren → neue (bessere) Wege können gefunden werden.

## WAS GENAU WAR NOCHMAL DAS VRP?

- Ein Lieferant hat eine Flotte von Lieferfahrzeugen.
- Er muss eine gewissen Anzahl von Kunden beliefern.
- Jedes Fahrzeug hat eine eigene Route mit eigenen Kunden, welche nur von diesem Fahrzeug beliefert werden.
- Die Fahrzeuge starten von einem Depot und besuchen jeden Kunden ihrer Route. (TSP)

# TRAVELING SALESMAN PROBLEM

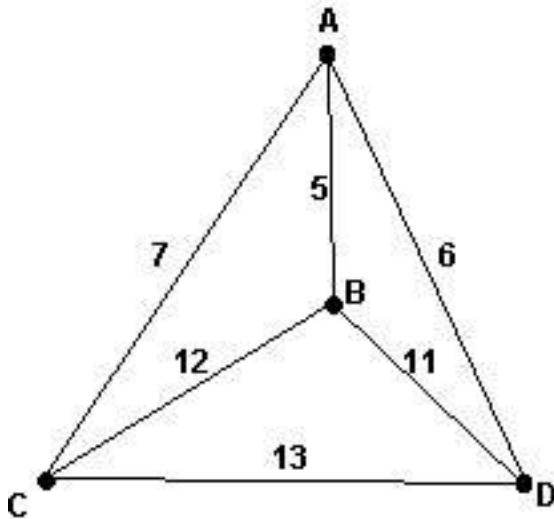
## WAS GENAU WAR NOCHMAL DAS TSP?

- Das TSP ist ein kombinatorisches Optimierungsproblem.
- Ziel ist es den kürzesten Weg einer Rundreise zu finden.
- Reisender möchte von einem Startpunkt aus gewisse Ziele besuchen und danach wieder zum Startpunkt zurückkehren.
- Reihenfolge der besuchten Orte ist beim grundlegenden TSP egal.



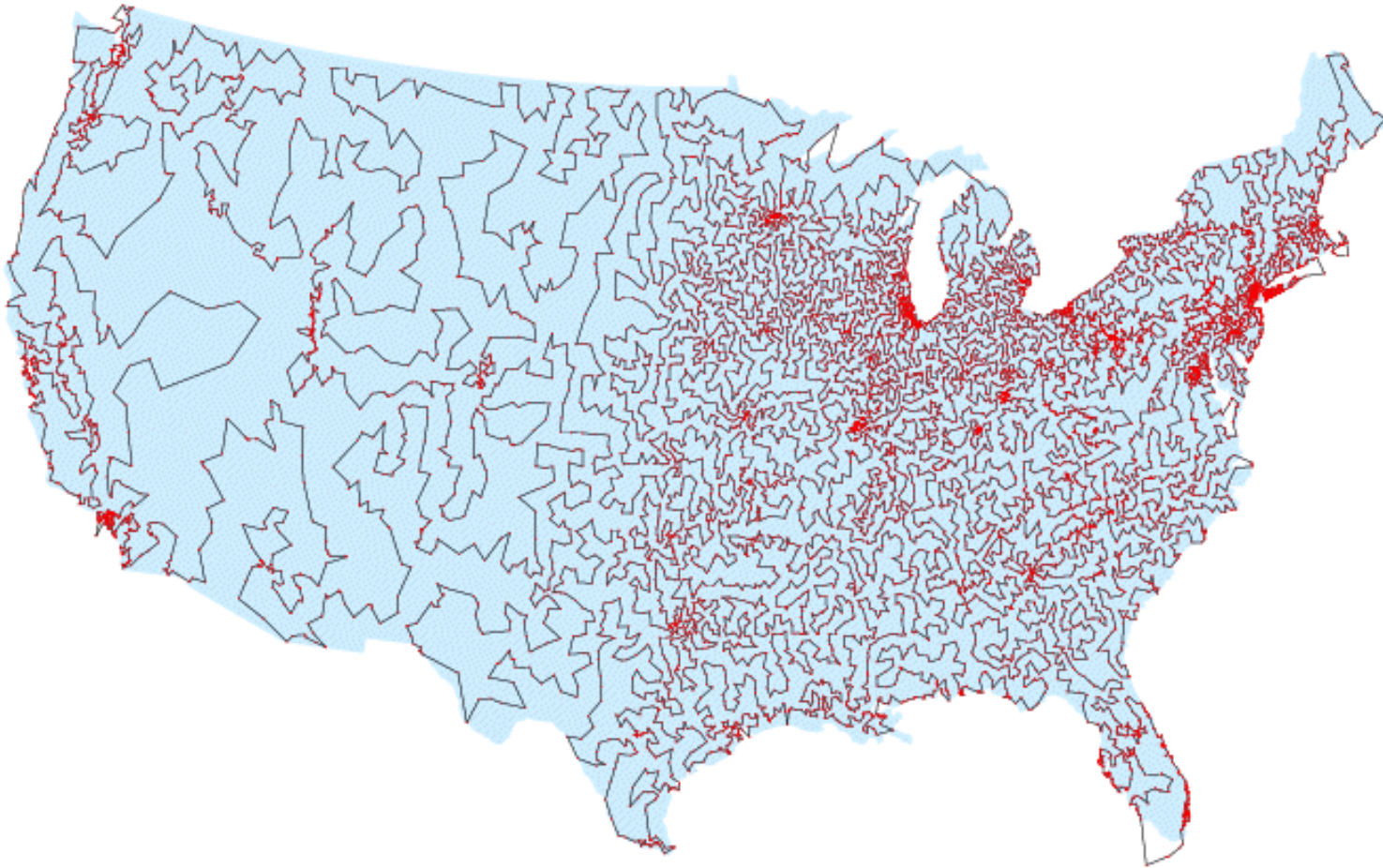
# TSP ALS GRAPH

- TSP als gewichteter Graph  $G = (N, A, d)$ .
- $N$  ist die Menge der Knoten und  $A$  die Menge der Kanten.
- Jede Kante  $(i, j) \in A$  besitzt eine Distanz  $d_{ij}$ . ( $i, j \in N$ )



- Eine Lösung: ACBDA (36)

# BEISPIEL FÜR EIN TSP: RUNDREISE DURCH DIE USA



# ACO FÜR DAS TRAVELING SALESMAN PROBLEM

# ÜBERBLICK

- Der erste ACO Algorithmus heißt Ant System (Marco Dorigo 1992) und wurde am TSP getestet.
- TSP ist NP-schwer.
- TSP ist ein leicht zu verstehendes Problem.
- TSP kommt echten Ameisen auf Futtersuche sehr nah.

# ALGORITHMUS SKELETT

1. Initialisierung der Parameter und Pheromone.
2. Solang ein bestimmtes Abbruchkriterium noch nicht erfüllt ist wiederhole:
  1. Lasse die künstlichen Ameisen eine komplette Tour konstruieren.
  2. (führe extra Prozeduren aus.)
  3. Aktualisiere die Pheromone.

## INITIALISIERUNG

- Anfängliche Pheromonstärke nur ein wenig höher als der durchschnittliche Betrag an Pheromonen, die während einer Iteration gelegt werden.
- Ist der Betrag am Anfang zu klein, werden die Ameisen von den ersten Spuren zu stark beeinflusst.
- Ist der Betrag zu groß, fallen die ersten konstruierten Touren kaum ins Gewicht.
- Anzahl der Ameisen  $m$  wird auf Anzahl der Knoten  $n$  gesetzt.

# KONSTRUKTION DER TOUR

1. Ein Startpunkt für jede Ameise wird festgelegt. Am besten eine Ameise auf jedem Knoten.
2. Jede Ameise wandert von Knoten zu Knoten, bis alle Knoten besucht wurden.
3. Rückkehr zum Startpunkt.

## KONSTRUKTION DER TOUR

- Formel zur Berechnung der Wahrscheinlichkeit, den Knoten  $j$  als nächsten zu besuchen.

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in \mathcal{N}_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \quad \text{if } j \in \mathcal{N}_i^k$$

- $\tau_{ij} \rightarrow$  Pheromonstärke.  
(Wie gut **war** die Strecke)
- $\eta_{ij} \rightarrow$  heuristischer Gehalt. Hier  $1/d_{ij}$   
(Wie gut **ist** die Strecke)
- $\alpha$  und  $\beta \rightarrow$  Parameter zur Steuerung des Einflusses.
- $\mathcal{N}_i^k \rightarrow$  Menge der brauchbaren Knoten.



# KONSTRUKTION DER TOUR

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in \mathcal{N}_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \quad \text{if } j \in \mathcal{N}_i^k$$

- $\tau_{ij} = 0,2; \eta_{ij} = 1/d_{ij}$

- $\alpha = 1; \beta = 2$

- $p_{ac}^k = 0.23$

- $p_{ab}^k = 0.45$

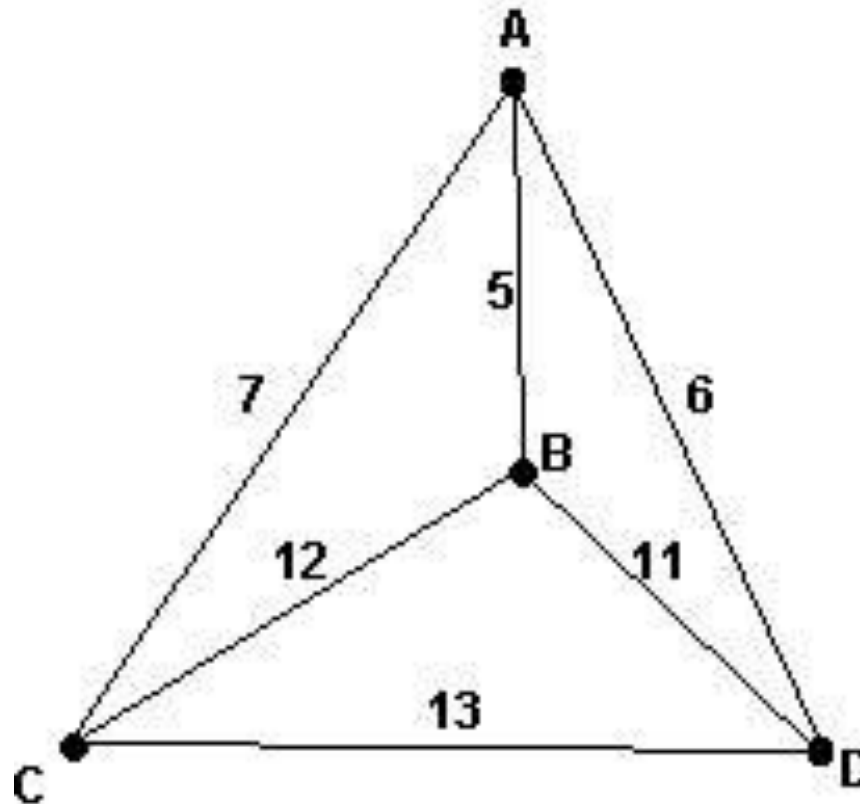
- $p_{ad}^k = 0.32$

- $\tau_{ac} = 0,6$

- $p_{ac}^k = 0.48$

- $p_{ab}^k = 0.31$

- $p_{ad}^k = 0.21$



## PHEROMON UPDATE

- Update der Pheromone beginnt nachdem alle Ameisen alle Knoten besucht haben.
- Konstruierte Touren können eventuell noch optimiert werden.
- Pheromonstärke kann in Abhängigkeit der Tour-Qualität dosiert werden.

## PHEROMON UPDATE

- Zuerst werden die Beträge der Pheromone auf allen Kanten um einen konstanten Betrag verringert. (Pheromon Evaporation)
- Schlechte Touren werden immer weniger attraktiv.
- Formel für die Evaporation
$$\tau_{ij} \leftarrow (1 - \rho) \tau_{ij}, \quad \forall (i, j) \in A$$
- $\rho \rightarrow$  Die festgelegte Rate der Evaporation.  $0 < \rho \leq 1$

## PHEROMON UPDATE

- Nach der Evaporation werden neue Spuren gelegt.
- Jede Ameise erhöht den Betrag auf den Kanten, auf denen sie gelaufen ist.

- $$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k, \quad \forall (i, j) \in A$$

- $\Delta\tau_{ij}^k \rightarrow$  Der Betrag den Ameise  $k$  auf der Kante  $(i, j)$  deponiert.
- $\Delta\tau_{ij}^k = 1/L^k$ , wenn Kante  $(i, j)$  zur Tour von  $k$  gehört, ansonsten  $\Delta\tau_{ij}^k = 0$ .
- $L^k$  ist die Länge der Tour von Ameise  $k$ .

## ANMERKUNGEN

- Die Wahrscheinlichkeit für alle Knoten auszurechnen kann bei großen TSPs zu schlechter Performance führen.
- Deshalb werden „Nearest-Neighbour-Lists“ eingesetzt.
- Sämtliche Distanzen zwischen zwei Knoten sind bekannt.
- Für jeden Knoten: Liste der nächsten Nachbar-Knoten.
- Ameise entscheidet sich nur noch zwischen diesen Knoten.

## ANMERKUNGEN

- Weitere Verbesserung des Algorithmus bringt das Einführen von Elite-Ameisen.
- Nach Ende aller Touren werden die Längen berechnet.
- Nur die „besten“ Ameisen legen Pheromone.
- Also die Ameisen, welche die besten Lösungen konstruiert haben.

# ACO FÜR DAS VEHICLE ROUTING PROBLEM

## VOM TSP ZUM VRP

- VRP vereinigt mehrere TSPs.
- Es liegt nahe sehr ähnliche Algorithmen für das VRP zu verwenden, wie für das TSP.
- Zusätzliche Komplexität: Mehrere Rundreisen (Routen).
- Zusätzliche Bedingungen: Kapazität der Fahrzeuge, Zeitfenster der Belieferung.

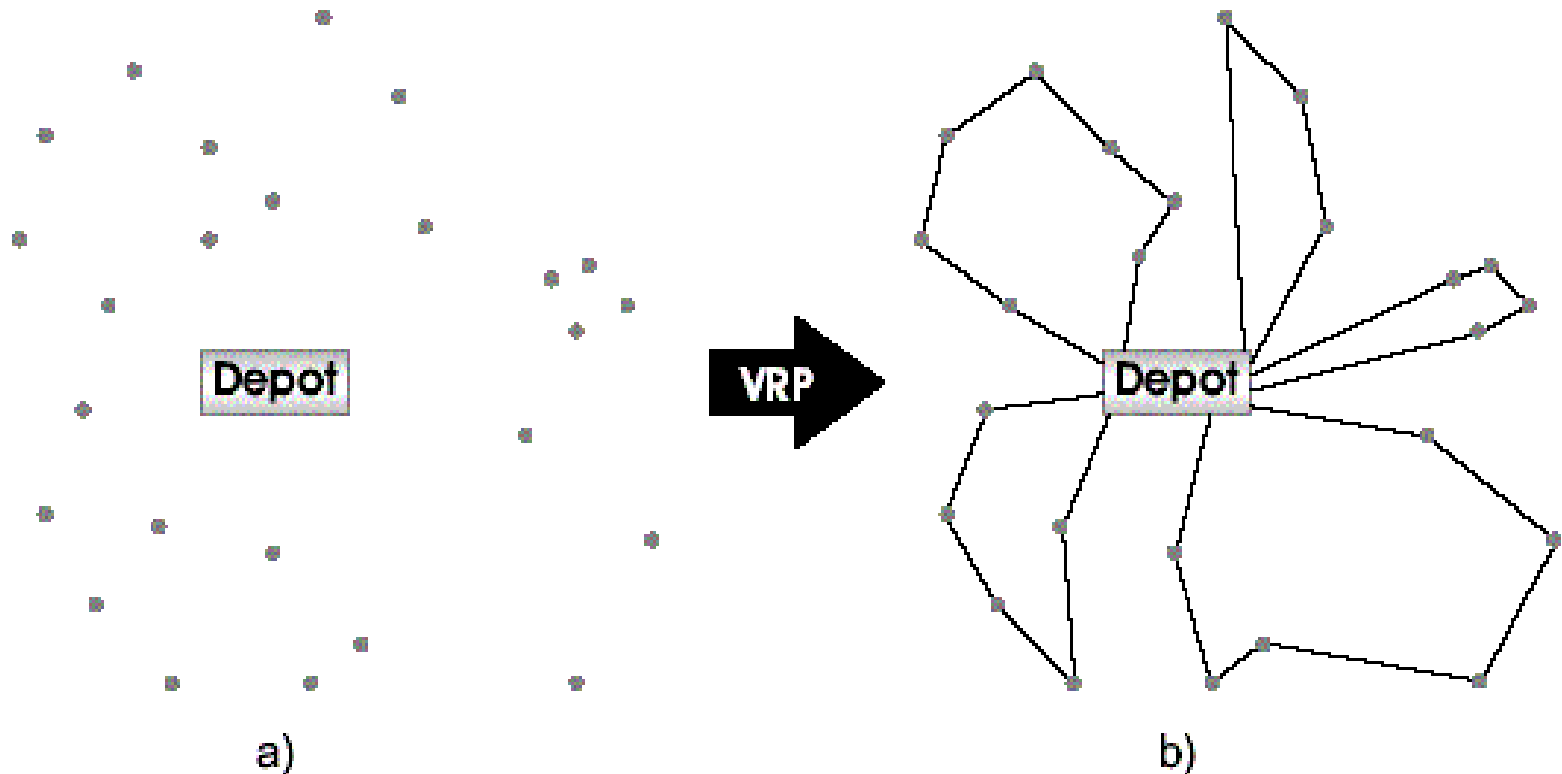


# CAPACITATED VEHICLE ROUTING PROBLEM

- CVRP ist das klassische VRP. Deshalb das erste VRP, auf welches ein ACO-Algorithmus angewendet wurde.
- Ameisen wählen einen Kunden nach dem anderen wie beim TSP.
- Überschreitung der Kapazität oder der maximal erlaubten Routenlänge des Fahrzeugs → Ameise geht zurück zum Depot.

# CAPACITATED VEHICLE ROUTING PROBLEM

- Heuristischer Gehalt ist gegeben durch folgende Formel:  $\eta_{ij} = d_{i0} + d_{0j} - d_{ij}$



# CAPACITATED VEHICLE ROUTING PROBLEM

- Es werden „Nearest-Neighbour-Lists“ und Elite-Ameisen eingesetzt, wodurch sich das Updaten der Pheromone ändert.

$$\tau_{ij}^{\text{new}} = p\tau_{ij}^{\text{old}} + \sum_{\mu=1}^{\sigma-1} \Delta\tau_{ij}^{\mu} + \sigma\Delta\tau_{ij}^*$$

- $\sigma \rightarrow$  Festgelegte Anzahl an Elite-Ameisen.
- $\mu \rightarrow$  Index der Elite-Ameisen.
- $\Delta\tau^{\mu} \rightarrow$  Betrag an Pheromon.  $(\sigma - \mu) / L^{\mu}$ .
- $\Delta\tau^* \rightarrow$  Betrag an Pheromon, welcher nur auf die Beste Route gelegt wird.  $1 / L^*$ .

## VRP WITH TIME WINDOWS

- Typisches VRP in der Realität.
- Größter Unterschied zum TSP: Zeitfenster.
- Die Kunden können nicht zu jeder Zeit beliefert werden.
- Berücksichtigung von zwei Problemen:
  - Minimierung der Routenanzahl (Fahrzeuge)
  - Minimierung der kompletten Reisezeit.
- Minimierung der Routenanzahl hat Vorrang.

## VRP WITH TIME WINDOWS

- Effizientester ACO Algorithmus für das VRPTW → MACS-VRPTW. (Multi-Ant-Colony-System)
- Zentrale Idee: Zwei parallele Ameisen Kolonien.
- Eine zur Minimierung der Routenanzahl (ACS-VEI) und eine zur Minimierung der Reisezeit (ACS-TIME).
- Beide Kolonien sind unabhängig → sie benutzen eigene Pheromone.
- Die Beste Lösung wird mit Pheromonen aus beiden Kolonien belegt.

# VRP WITH TIME WINDOWS

## ○ MACS-VRPTW:

1. Eine brauchbare Startlösung wird mit Hilfe einer geeigneten Heuristik erstellt.
2. ACS-VEI wird mit einem Fahrzeug weniger gestartet und versucht eine bessere Lösung zu finden.
3. ACS-TIME wird auch gestartet und versucht die Zeit zum Beliefern aller Kunden zu verringern.
4. Wenn durch ACS-VEI eine bessere Lösung gefunden wird, wird die Anzahl an Fahrzeugen um Eins verringert und wieder bei Schritt 2 begonnen.

## VRP WITH TIME WINDOWS

- Brauchbare Knoten für die Ameisen:
- Alle noch nicht besuchten Knoten, bei denen die Ankunftszeit und die Verladezeit nicht das Zeitfenster verletzen.
- Bei der Berechnung der Wahrscheinlichkeit einen Knoten zu besuchen fließt die Zeit in den heuristischen Gehalt ein.
- Je weniger Zeit zur Belieferung eines Kunden bleibt, umso „näher“ wird er für die Ameisen erscheinen.

# PRAXISBEISPIEL



## EINE VRPTW APPLIKATION

- Eine der größten Supermarktketten der Schweiz.
- Güter auf Paletten zu über 600 Filialen liefern.
- Lieferungen innerhalb von bestimmten Zeitfenstern.
- Es gibt verschiedene Fahrzeugtypen. Nicht jedes Fahrzeug kann jeden Laden beliefern.
- Die Fahrzeuganzahl ist nicht begrenzt.

## EINE VRPTW APPLIKATION

- Jede Tour muss innerhalb eines Tages vollendet sein.
- Daten zur Erstellung des Graphen wurden über Jahre gesammelt.
- Es wurden die Distanz und die mögliche Fahrgeschwindigkeit für die Gewichtung der Kanten berücksichtigt.

## LÖSUNG

- Benutzt wurde ANTRROUTE, eine leichte Modifikation des MACS-VRPTW Algorithmus.
- Ameisen entscheiden sich zufällig am Anfang ihrer Tour für einen Fahrzeugtyp.
- „Wartekosten“ wurden eingeführt, damit ein Fahrzeug nicht zu früh bei einem Laden ankommt.

# LÖSUNG

- Zwei Konfigurationen:
  - AR-RegTW berücksichtigt regionale Planungsvorgaben und hat Zeitfenster von jeweils einer Stunde pro Laden.
  - AR-Free lockert diese Restriktionen.
- 52.000 Paletten sollten über 20 Tage zu 6.800 Standorten geliefert werden.
- ANTRROUTE wurde jeden Tag mit den aktuellen Lieferungen gestartet und hatte im Durchschnitt schon nach 5 min. gute Lösungen anzubieten.
- Die menschlichen Planer des Unternehmens brauchten pro Tag ca. 3 Stunden, um auf eine geeignete Lösung zu kommen.

# LÖSUNG

- ANTRROUTE im Vergleich mit den menschlichen Tourenplanern des Unternehmens.

	Planer des Unternehmens	AR-RegTW	AR-Free	AR-RegTW vs. Planer	AR-Free vs. Planer
<b>Totale Anzahl der Touren</b>	2.056	1.807	1.614	12,11%	21,50%
<b>Totale Strecke in km</b>	147.271	143.983	126.258	2,23%	14,27%
<b>Durchschnittliche Fahrzeugauslastung</b>	76,91%	87,35%	97,81%	10,44%	20,9%

**VIELEN DANK FÜR IHRE  
AUFMERKSAMKEIT**

**GIBT ES NOCH FRAGEN?**