



Technische
Universität
Braunschweig

Institut für Wirtschaftsinformatik
Lehrstuhl Decision Support
Prof. Dr. Dirk Christian Mattfeld



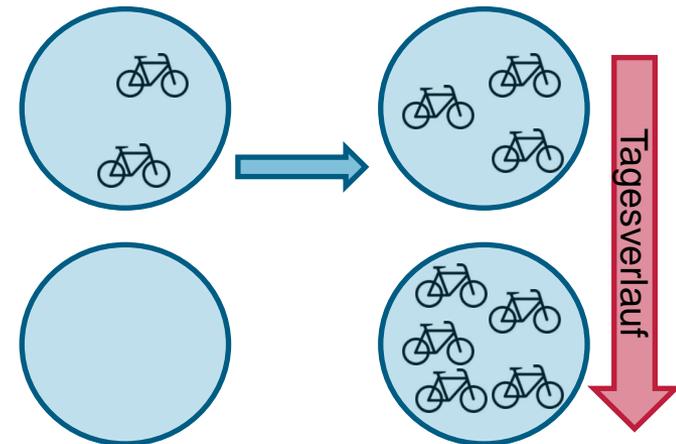
Ein Dekompositionsansatz für die Repositionierung in Bike-Sharing-Systemen

DoWoNo 2014, Clausthal

Viola Ricker

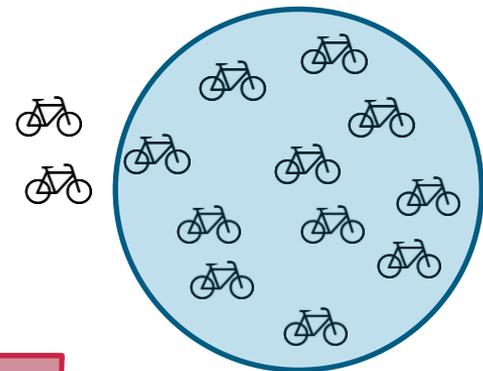
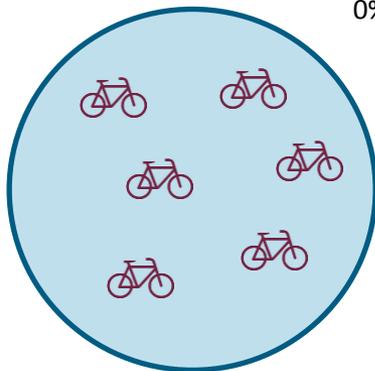
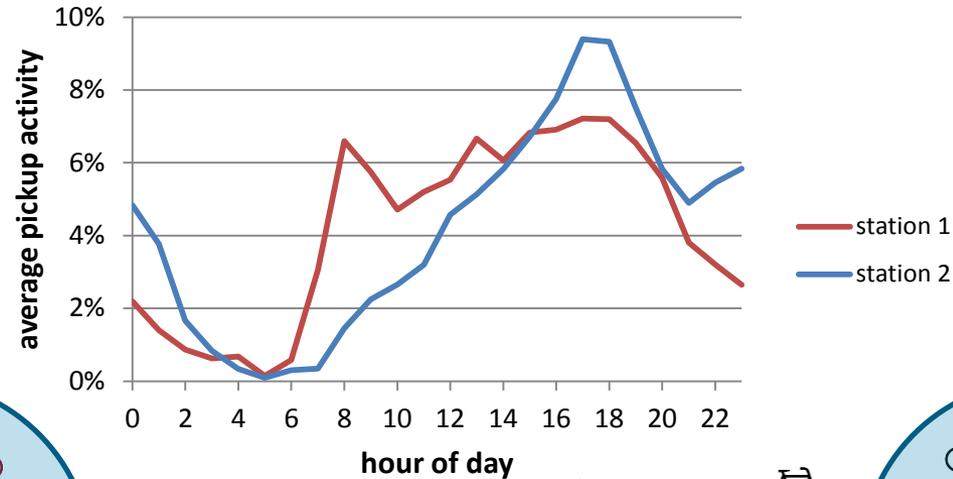
Herausforderungen von Bike-Sharing-Systemen (BSS)

- Mobilitätsdienstleistung BSS: Bereitstellung von Fahrrädern als nachhaltiges, individuelles, öffentliches Verkehrsmittel für innerstädtische Kurzstrecken
 - Schneller und einfacher Zugang durch Informationssystem-Unterstützung:
 - Einwegnutzung
 - Kurze Entleihzeiten
 - Managementziel: Sicherstellung der Servicequalität durch Bereitstellung von Fahrrädern und Fahrradboxen
 - Problem: **Ungleichgewicht in der Fahrradverteilung** durch räumliche und zeitliche Nachfrageschwankungen
- ➔ **Wie kann die Servicequalität in stationsbasierten BSS sichergestellt werden?**



Operative Kosten: Repositionierung

pickups at two different stations



Problemcharakteristika

- Konkurrierende Ziele:
 1. Kundenzufriedenheit maximieren:
 - a) Verfügbarkeit von Fahrrädern
 - b) Verfügbarkeit von Fahrradboxen
 2. Transportzeit für Repositionierungen minimieren

→ Maße für beide Ziele in Zeiteinheiten ausdrücken:

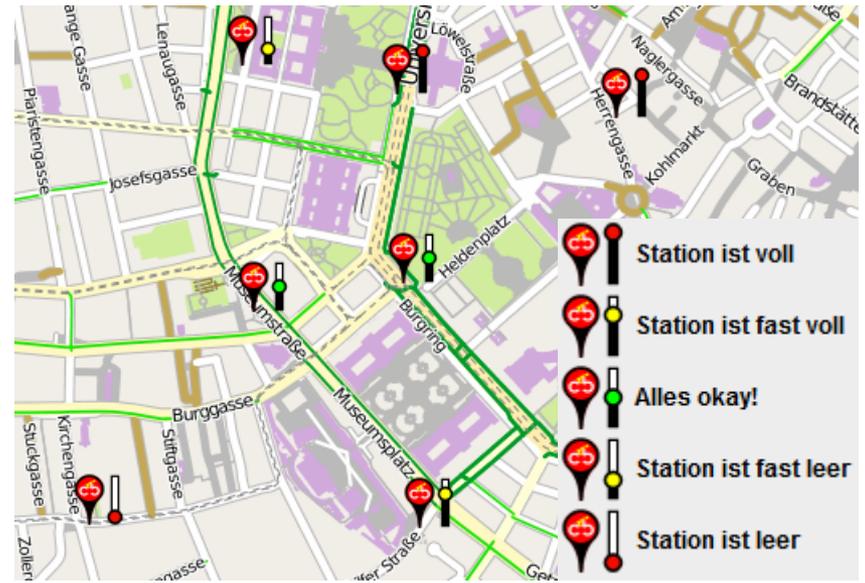
- Transportzeit:

$$\sum_{\text{Transporter}} \text{Repositionierungsfahrten}$$

- Umwegzeit:

$$\sum_{\text{Stationen}} \text{Kundenwege}$$

*Ziel: min Transportzeit + w * Umwegzeit*



Dynamisches und stochastisches Problem: Inventory routing problem with pickups and deliveries

zu treffende Entscheidungen:

- Anzahl Räder
- Anzahl Boxen



inventory problem

- Startort
- Zielort



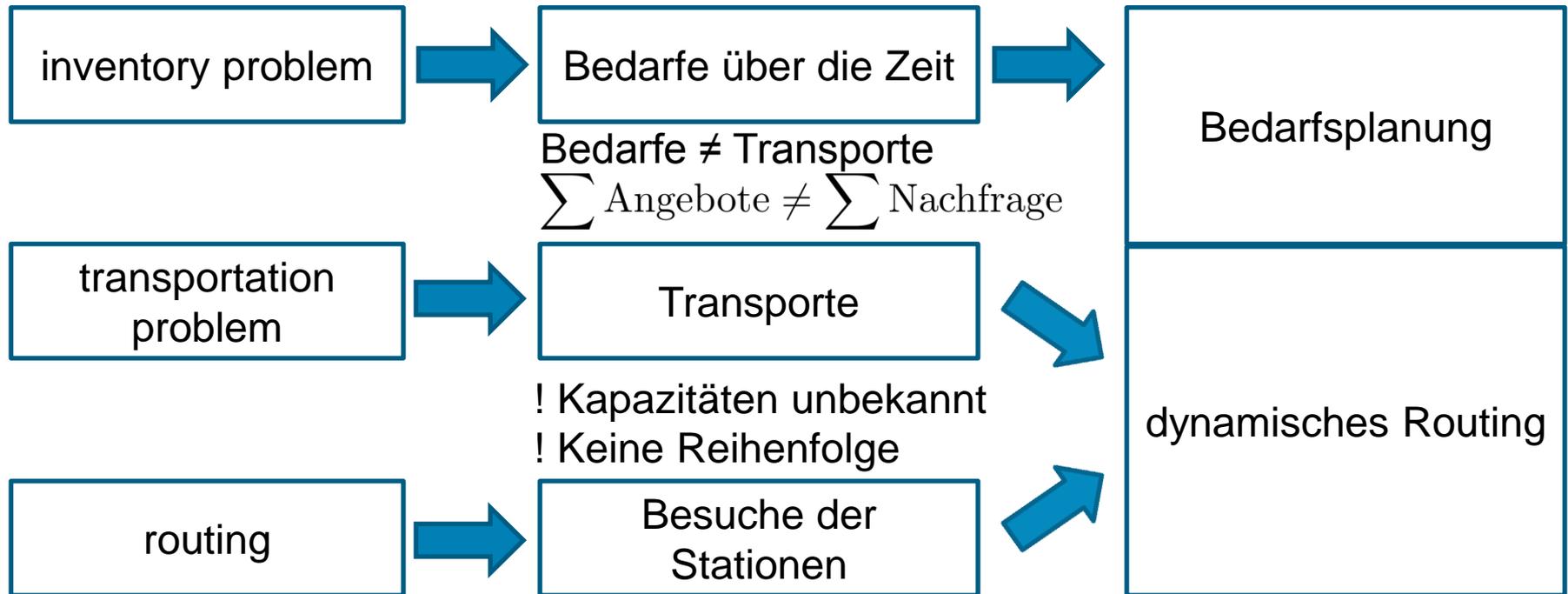
transportation problem

- Reihenfolge
- Transporter

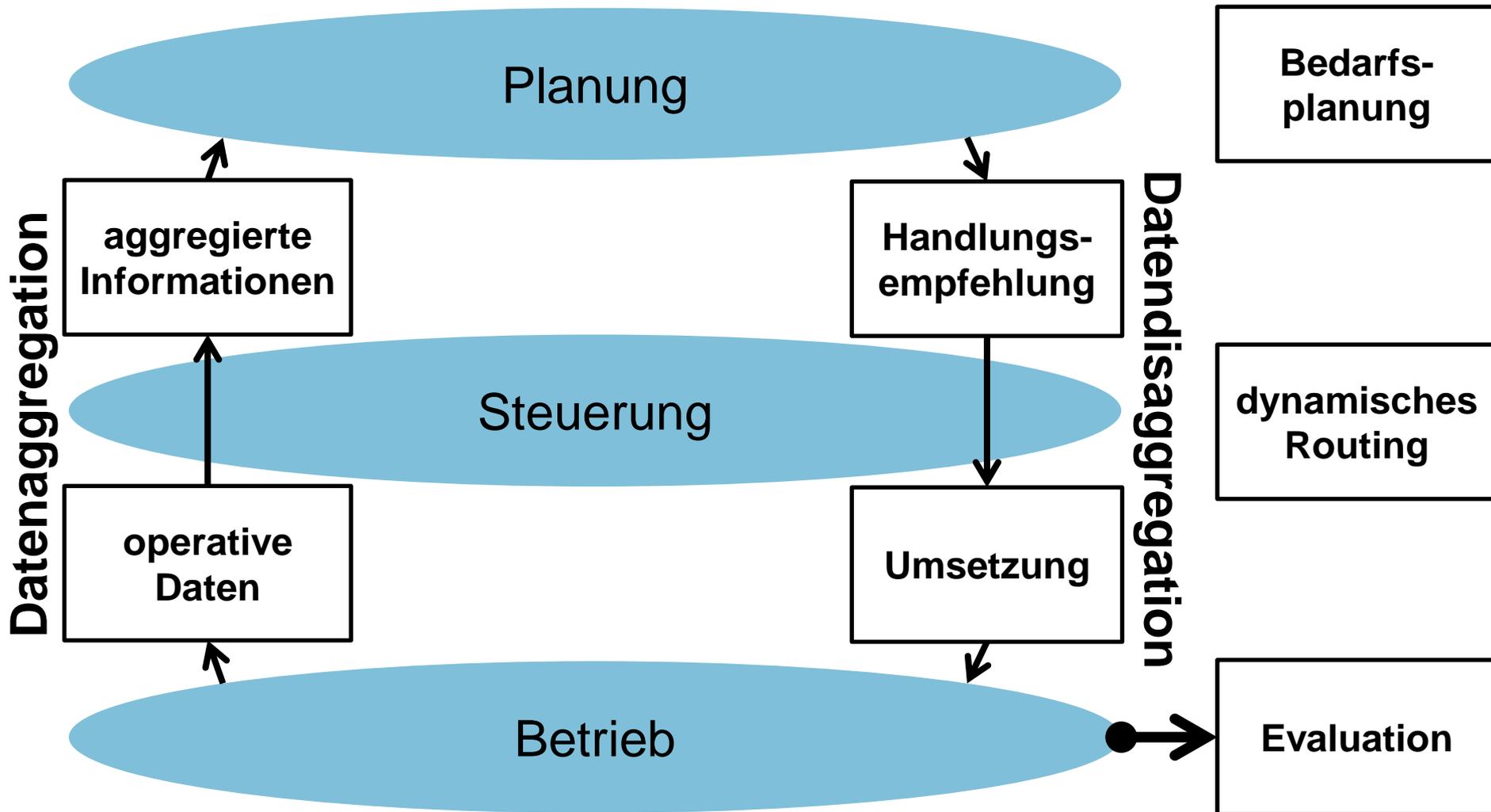


routing with pickup and delivery

Dekomposition



Schematisches Lenkungsebenenmodell



Betriebsebene → Simulation

- Daten von Citybike in Wien
- System mittlerer Größe:
58 Stationen, ca. 700 Räder, ca. 1750 Fahrten pro Tag im Sommer
- Google-Fahrtzeiten
- Exponentialverteilung für die Nachfrage: Parameter λ pro Station und Stunde
- Zielstationswahl nach historischen Daten diskret verteilt

- Umwegzeiten als Maß für Kundenzufriedenheit
 - fehlgeschlagene Entleiherung: Laufweg zur nächsten Station
 - fehlgeschlagene Rückgabe: Radweg zur nächsten Station
- Depot mit einer Anzahl an Transportern
- Simulationshorizont: 5 Tage + Ausschwingzeit



Simulationsergebnis

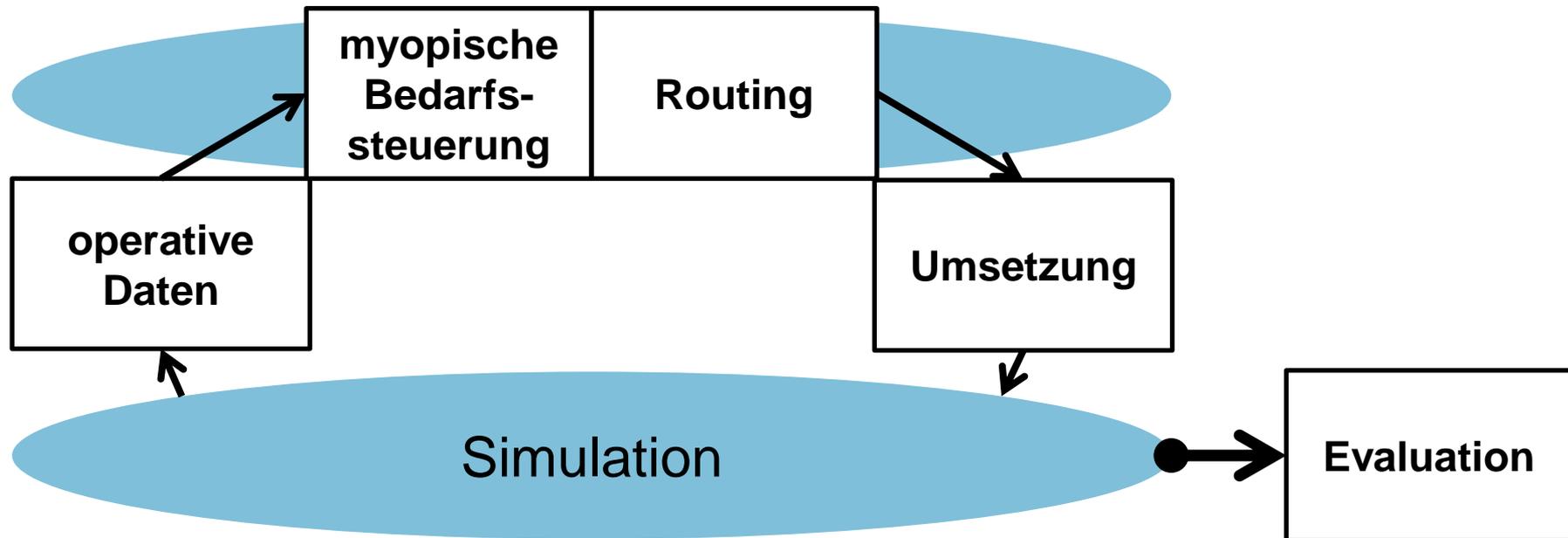
- Erhebliche Umwegzeiten wenn nicht eingegriffen wird:

detour time walk	detour time bike	failed rentals per day	failed returns per day
45,9 h/day	40,1 h/day	451,8	572,9

- Knapp 70% der Kunden hat im Mittel eine nicht erfolgreiche Entleiherung oder Rückgabe
 - Pro Misserfolg entsteht ein Umweg von ca. 5 min zu Fuß oder 3,5 min mit dem Rad
 - Insgesamt ca. 86 Std Kundenumweg pro Tag!
- ➔ Betreiberseitiges Repositionieren notwendig!

Steuerung

- Operative Daten: Beobachtete Stationsbestände
- Myopische Bedarfssteuerung: Entscheidung über Bedarfe
- Routing:
 - Greedy-Starttour aus den ermittelten Bedarfen mit Kapazitätsbeachtung
 - Splitting der Touren unter Berücksichtigung der Transportzeit
 - verschiedene Verbesserungsheuristiken



Beispiel einer myopischen Bedarfssteuerung

- Regelbasierte Bedarfssteuerung
- Auslösen von Repositionierungen durch Unterschreiten/Überschreiten von Pufferbeständen an den Stationen (bspw. 20% und 80%)
- Auffüllen/Abtransport von ca. 50% der Stationskapazität
 - Beispiel:

Stationskapazität	10 Boxen
Untergrenze	2 Fahrräder
Obergrenze	8 Fahrräder
Transportmenge	5 Fahrräder
- Ausgleich des Bedarfsvektors nötig, falls Angebot und Nachfrage nicht gleich sind
- Wahl der jeweils relativ am meisten oder wenigsten gefüllten Station zum Ausgleich

Simulationsergebnisse mit Steuerung

buffer	transportation time [h/day]	service time [h/day]	detour time walk [h/day]	detour time bike [h/day]	failed rentals	failed returns	number of tours	relocated bikes
0%	12,8	4,9	5,2	7,5	60,4	135,1	13,8	145,7
10%	17,0	7,0	3,7	4,4	41,2	81,9	17,1	208,7
20%	20,1	9,0	3,1	5,2	35,0	94,9	19,7	258,0

- Test mit jeweils 100 Simulationsdurchläufen
- Deutlicher Rückgang der Umwegzeiten
(ca. 9 – 13 Std. im Vergleich zu 86 Std. ohne Eingriff ins System)
- Größere Puffer → mehr Transporte
(knapp 18 Std. Arbeitszeit pro Tag ohne Puffer, gut 29 Std. bei 20% Puffer)
- Frühes kurzichtiges Reagieren kann zu entgegengesetzten Fehlmengen führen

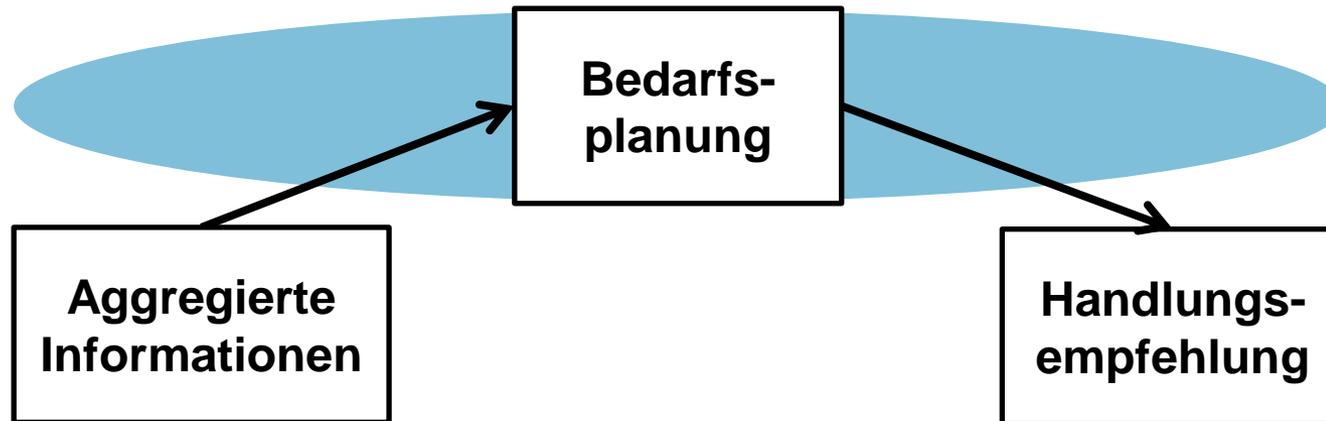
Nachteile der Myopischen Optimierung

Keine Vorausschau bezüglich

- Schwankungen oder Umkehrung in der Kundennachfrage:
 - Keine Berücksichtigung von Selbstregulierung im System
 - Reaktion nur auf Zustand
- Kapazitätsrestriktionen:
 - Transporte nicht ohne Zeitverzögerung möglich
 - Transporter möglicherweise nicht verfügbar

→ Planung vorteilhaft

Planung



- Bedarfsplanung: Antizipierende dynamische und stochastische Optimierung
 - Große Zustands- und Aktionsräume
→ Optimierung auf aggregierter Ebene
- Aggregierte Stationsbestände und Informationen über Umwegzeiten
- Handlungsempfehlung: Grobe Aussage über die zu repositionierenden Mengen pro Station
- S_t grober Stationszustand (vordefinierte Füllgrade)
- d_t grobe Entscheidung
- ω_t stochastischer Einfluss

Optimierungsmodell: Approximative Dynamische Programmierung (ADP)

- Markov-Prozess:

$$S_{t-1}^d \xrightarrow{\omega_{t-1}} S_t \xrightarrow{d_t} S_t^d \xrightarrow{\omega_t} S_{t+1} \xrightarrow{d_{t+1}} S_{t+1}^d$$

- Bellman Gleichung:

$$V_t(S_t) = \min_{d \in \mathcal{D}} (C_t(S_t, d_t) + \mathbb{E} \{ \gamma V_{t+1}(S_{t+1}) | S_t \})$$

- Bewertung über post-decision-Zustand:

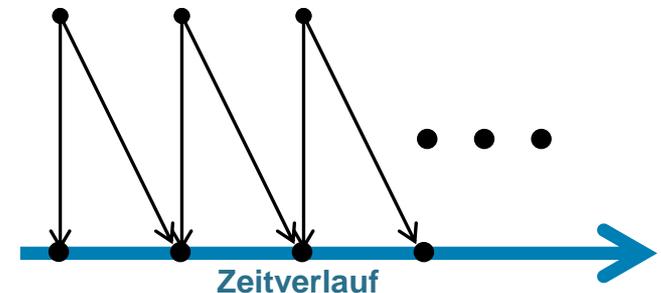
$$V_{t-1}^d(S_{t-1}^d) = \mathbb{E} \left\{ \min_{d \in \mathcal{D}} (C_t(S_t, d_t) + \gamma V_t^d(S_t^d) | S_{t-1}^d) \right\}$$

- Entscheidungsauswahl:

$$\hat{v}_t^n = \min_{d \in \mathcal{D}} (c_{trans}(d_t) + c_{pen,t-1} + \gamma \bar{V}_t(S_t^d))$$

- Value update:

$$\bar{V}_{t-1}^{dn}(S_{t-1}^{dn}) = (1 - \alpha_{n-1}) \bar{V}_{t-1}^{n-1}(S_{t-1}^{dn}) + \alpha_{n-1} \hat{v}_t^n$$



Abschätzung der Transportkosten

- Paralleles Lernen einer guten Abschätzung der Transportzeitenanteile für die einzelnen Stationen:
- Startwerte:

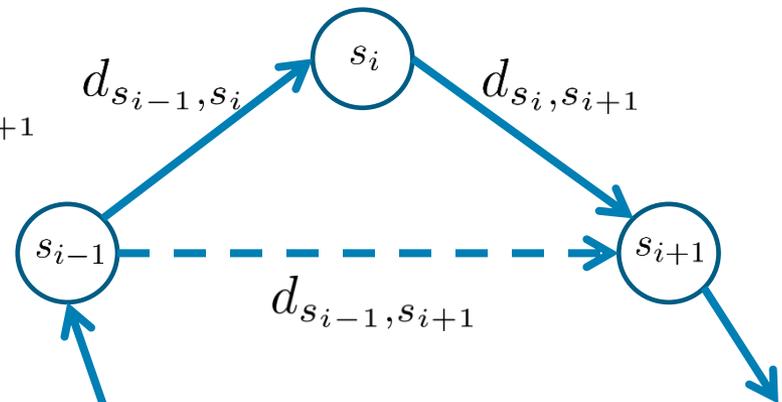
$$\bar{c}_{trans}^0(s) = d(s_0, s), \forall s \in \text{Stationsmenge}$$

- Update:

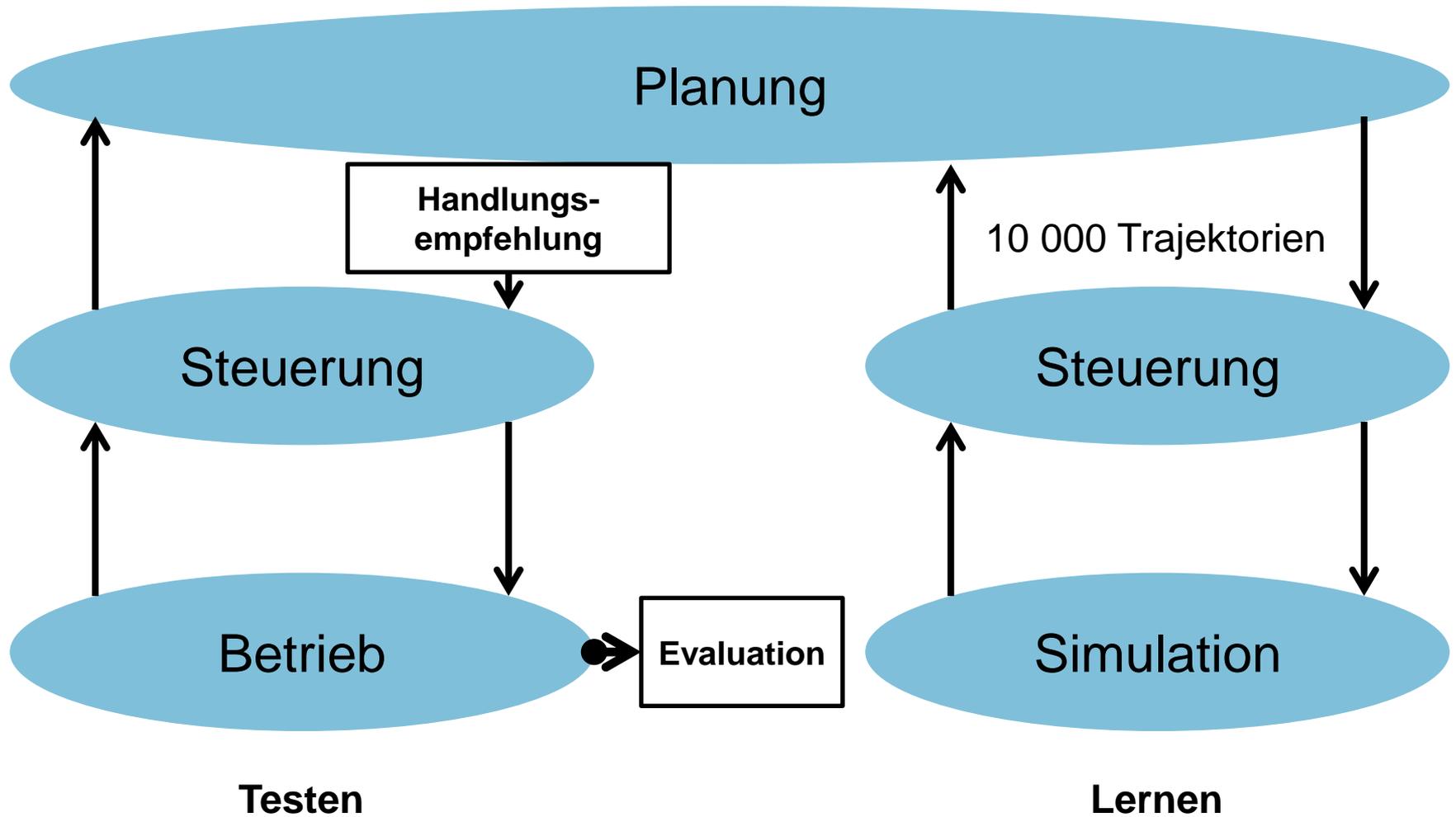
$$\bar{c}_{trans}^n(s) = (1 - \alpha_{n-1}) \cdot \bar{c}_{trans}^{n-1}(s) + \alpha_{n-1} \cdot \hat{c}_{trans}^n$$

$\hat{c}_{trans}^n(s)$ Transportzeitzuwachs für die jeweilige Station im Transportplan

$$\hat{c}_{trans}^n(s_i) = d_{s_{i-1}, s_i} + d_{s_i, s_{i+1}} - d_{s_{i-1}, s_{i+1}}$$



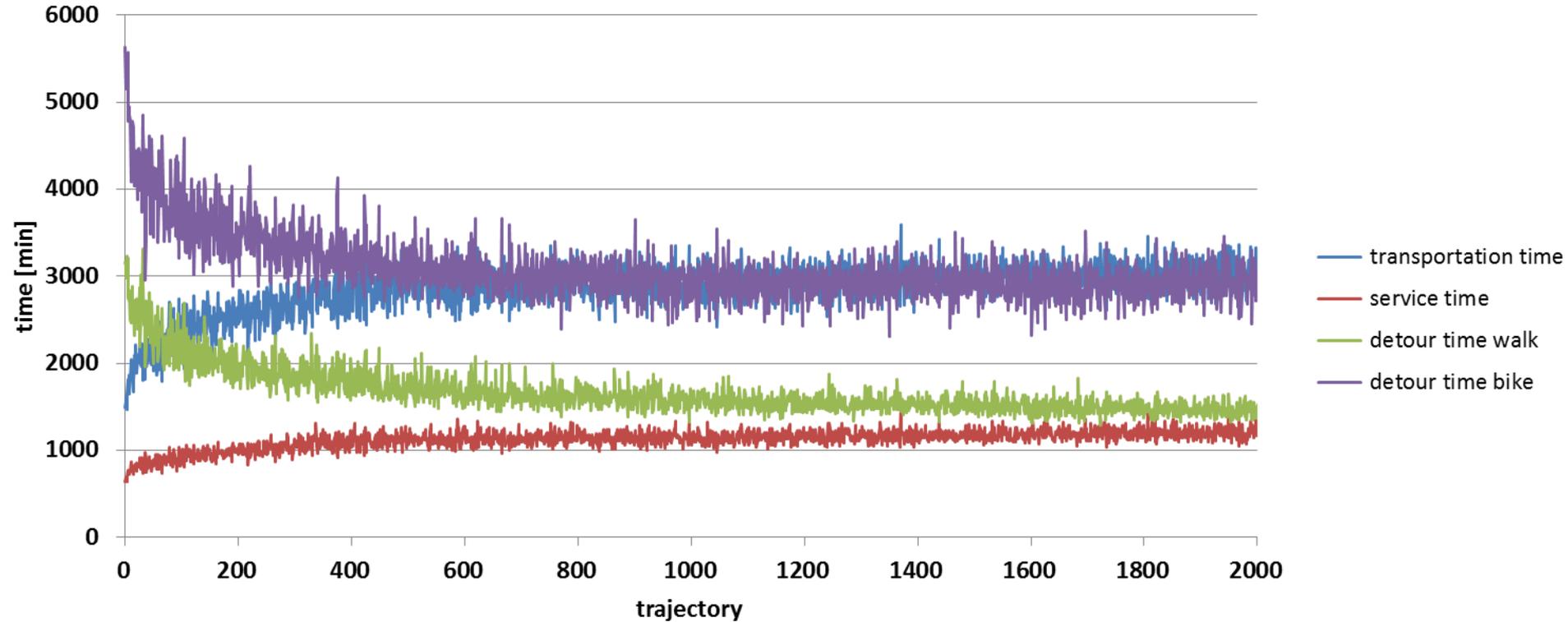
Ablaufdiagramm



Experimente

- Simulation des Bike-Sharing-Systems über 5 Tage
- 5 Zustände gestaffelt von „leer“ über „ausgeglichen“ zu „voll“
- 3 Entscheidungsmöglichkeiten „wegtransportieren“, „nichts tun“ und „hintransportieren“
- Lernen einer Aktion an jeder Station für jede Stunde eines Tages und jeden Zustand
- 10000 Trajektorien
- Vergleich mit myopischem Ansatz
 1. Gegensteuernde Repositionierung, wenn Schwellwerte unter- oder überschritten sind
 2. ADP-Handlungsempfehlung nach 10000 Trajektorien

Vergleich der Ergebnisse



	total time	transportation time	service time	detour time walk	detour time bike	failed rentals	failed returns	number of tours	relocated bikes
balance heuristic 0%	30,4	12,8	4,9	5,2	7,5	60,4	135,1	13,8	145,7
ADP	26,7	10,0	4,1	4,2	8,4	59,5	176,4	12,1	124,2

Zusammenfassung und Ausblick

- Repositionierung:
Wichtige Maßnahme zur Sicherstellung der Servicequalität
- Myopische Optimierung bedingt geeignet
- ADP erlaubt Einbeziehung zukünftiger Ereignisse in die Entscheidung

Nächste Schritte:

- Verbesserung des ADP
 - Untersuchung der Lernperformance bei verschiedenen Updateparametern α
 - Variation des Parameters w zur Gewichtung der verschiedenen Zielfunktionsanteile
 - Integration von Heuristikwissen
- Evaluierung
 - Testsettings über die Daten von CityBike hinaus
 - Für welche Art von Inputdaten eignet sich ADP?

Danke für die Aufmerksamkeit!