



Ameisenkolonien und evolutionäre Algorithmen zur Lösung logistischer Probleme

Prof. Dr. Thomas Bousonville

Hochschule für Technik und Wirtschaft des Saarlandes

- Saarland University of Applied Sciences -

Saarbrücken

Systemwissenschaftliches
Kolloquium

28.10.2009



Agenda

- Komplexität
- Naturinspirierte Verfahren zur Lösung statischer Probleme
 - Genetische / Evolutionäre Algorithmen
 - Ameisenkolonien
 - Tabu Suche
 - Vergleich der Verfahren
 - Anwendungen
- Umgang mit Dynamik in der Planung logistischer Systeme
 - Rollierende Planung
 - Hierarchische Planungsmodelle
 - Selbststeuerung in der Logistik



Welche logistischen Entscheidungen sind komplex?

- Operative Entscheidungen in der Logistik
 - Bestellmengen und -zeitpunkte
 - Bestimmung der Reihenfolge der Kunden bei der Auslieferung/Abholung von Waren
 - Einplanung von Produktionsaufträgen

- Strategische Entscheidungen im der Logistik
 - Standort eines neuen Distributionszentrums
 - Fremdvergabe von logistischen Dienstleistungen (Lager, Fuhrpark)
 - Aufgabenverteilung zwischen zentralen und dezentralen Logistikeinheiten

Auch operative Probleme können komplex sein



Komplexität

- Maßgröße für die Komplexität berechenbarer Problemstellungen:

Aufwand (z.B. Zeitaufwand)
zur "Lösung" des Problems in Abhängigkeit
der Problemgröße (Input)

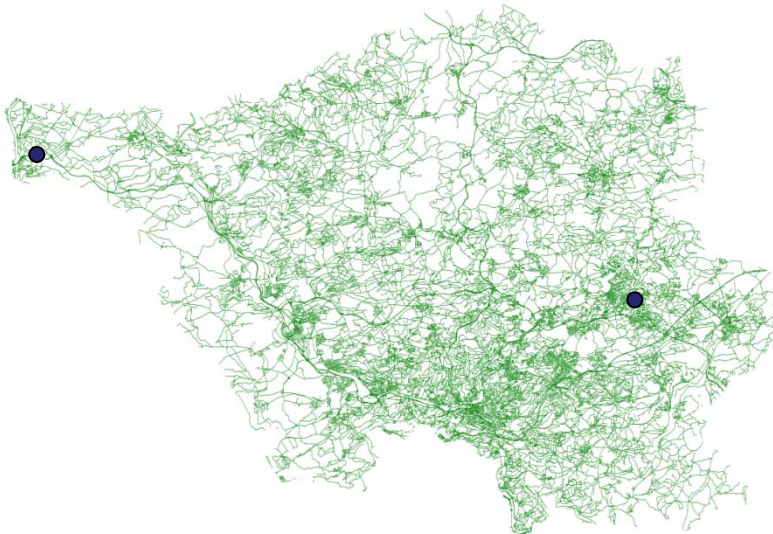
Komplexitätsbegriff der theoretischen Informatik

- „Lösen“ bedeutet hier die Auswahl der Handlungsoption, mit der man dem gewünschten Ziel am nächsten kommt
- „Lösen“ durch Anwenden einer Verfahrensvorschrift (Algorithmus)
- Komplexität eines Algorithmus: $O(\cdot)$ -Notation
 - Beispiele: $O(1)$, $O(n)$, $O(n \log n)$, ...
- Abgrenzung zum Komplexitätsbegriff in der Systemtheorie
 - Verhalten nichtlinearer, dynamischer Systeme



Beispiele (1): Kürzeste Wege

- Finde kürzesten Weg eines Fahrzeugs von A nach B
- Aufwand $O(n^2)$ [mit Dijkstra]
 - Falls Straßennetz mit 100 Knoten, Berechnung 0,001 sec
 - => Straßennetz mit 53.000 Knoten => ~ 280.000 mal länger



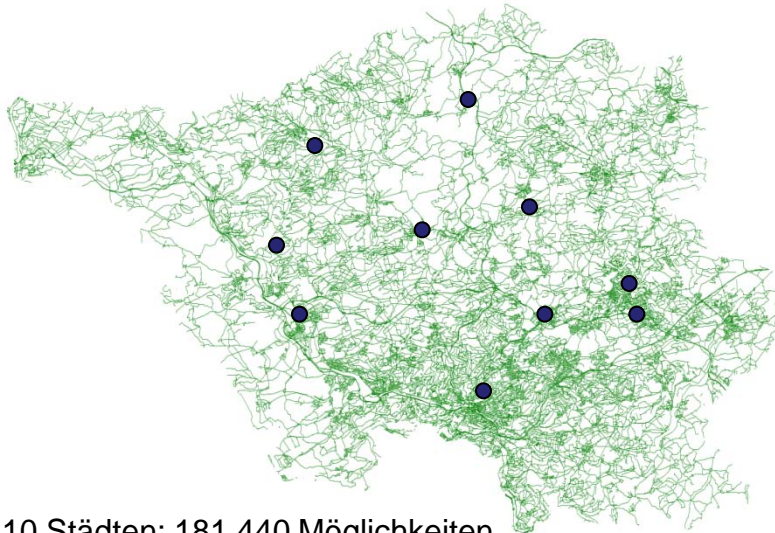
Gilt als praktisch lösbares Problem ...

... da der Aufwand zum Finden der besten Lösung polynomial mit der Problemgröße steigt.



Beispiele (2): Rundreise

- Finde kürzesten Weg um eine Menge von Kunden mit Zeitungen zu beliefern, dort Stückgut einzusammeln etc. (Probl. d. Handlungsreisenden)
- Größe Lösungsraum: $(n-1)!/2$
 - Nicht polynomial
 - Kein polynomialer Algorithmus bekannt



bei 10 Städten: 181.440 Möglichkeiten
 bei 15 Städten: 43.589.145.600 Mögk.

Praktisch nicht (begrenzt) lösbares Problem

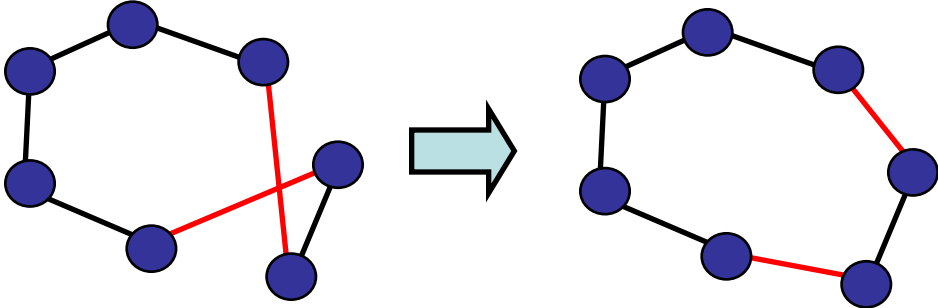
... da der Aufwand zum Finden der besten Lösung mehr als polynomial mit der Problemgröße steigt.

“Größtes” exakt gelöstes Problem lag 2001 bei

- 15.112 Knoten durch
- 110 parallele Computer mit
- 22,6 Jahren Rechenzeit (500 Mhz)

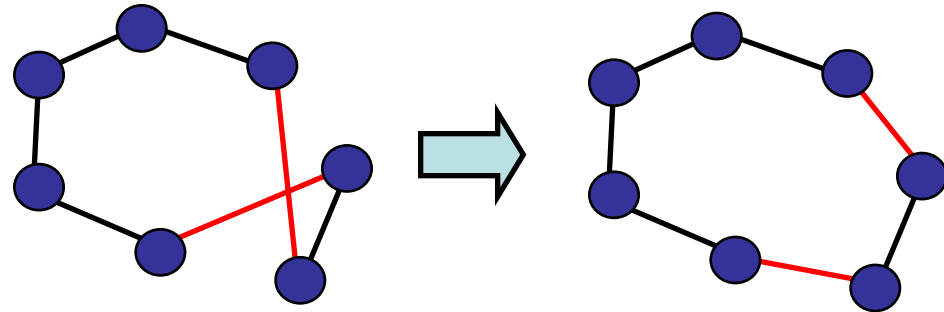


Heuristische Lösungsmöglichkeiten

- Konstruktionsheuristik
 - Erzeugt eine Ausgangslösung
 - Beispiel: Nächster Nachbar
 - ❖ Wähle einen Startknoten
 - ❖ Aus den noch verbliebenen Knoten wähle den nächstgelegenen als Folgeknoten
- Verbesserungsverfahren
 - Startet mit einer vorliegenden Lösung
 - Modifikation eines Teils der Lösung
 - Beispiel: 2-opt
 - 
 - Iteration bis keine Verbesserung mehr (lokale Optimierung)



Heuristische Lösungsmöglichkeiten



- Lokale Suche
 - Operator (Move), z.B. 2-opt, definiert einen lokalen Suchraum (= „Nachbarschaft“)
 - Allg. k-opt
- Problem:
 - Steckenbleiben im lokalen Optimum

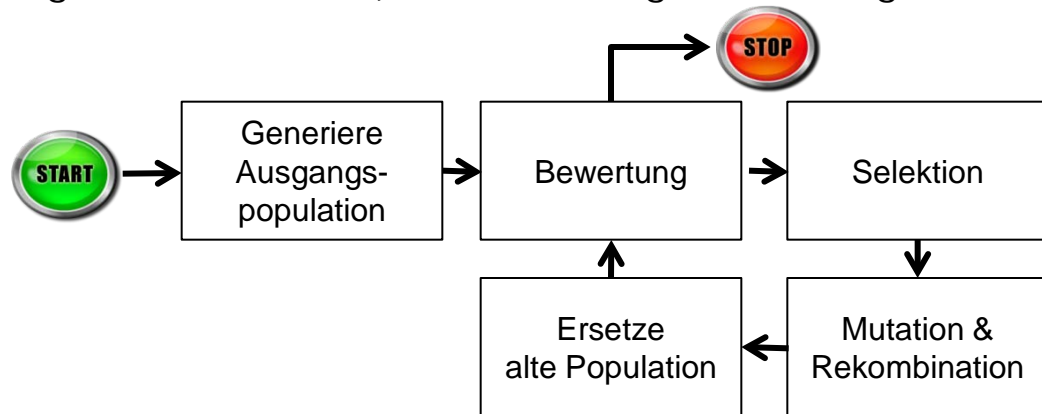


Die Evolution als Problemlöser?

- Übertragung der (technischen) Problemstellung in die Natur
 - Hier: Man sucht eine Lösung die optimal zum Problem „passt“ (es löst)
- Biologische Metapher: Evolutionsprozess als Anpassungsprozess
- Anpassung von Individuen (= „Lösungen“) an die Umwelt (= „Problem“)
 - Mutation (+Rekombination) zur der Generierung neuer Eigenschaften
 - Selektion zur Auswahl der besten Lösungen
 - Schrittweise Anpassung an die Umwelt (= Finden von guten Lösungen für “das Problem”)

Prinzip eines evolutionären Algorithmus

Bei binärer Problemdarstellung:
Bez. als „Genetischer Algorithmus“



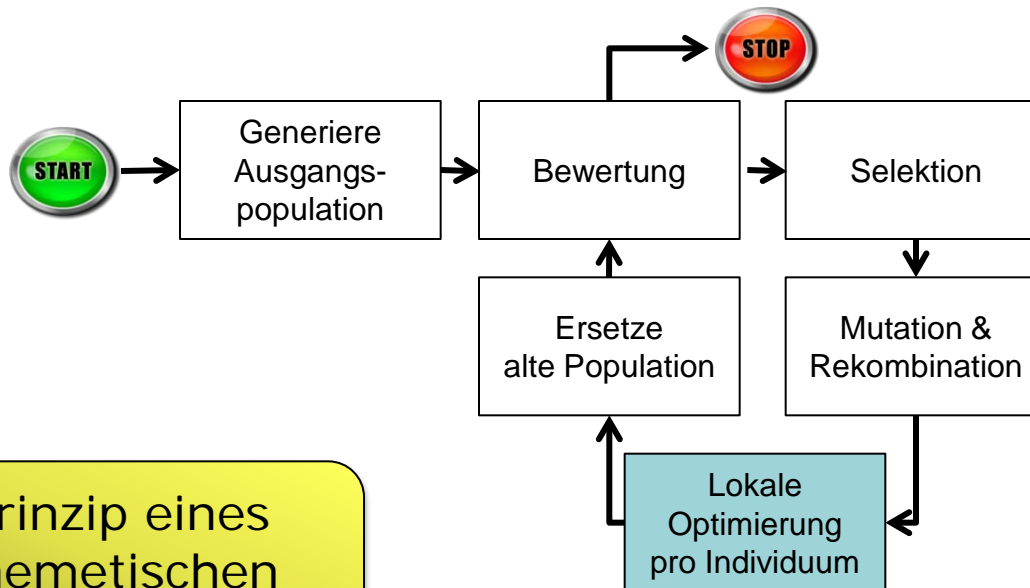


Evolutionary algorithm: Pseudo code

```
procedure Standard_GA
begin
    t=0
    P(t) := InitPop( )
    while not (Stop(t,P(t)))
        t := t+1
        Evaluate P(t-1)
        Select P', P'' ∈ P(t-1) for reproduction
        Mutation (i)    i ∈ P'
        Crossover (i,j)    i,j ∈ P''
        P(t) := NewGeneration (P(t-1), P', P'')
    end
end
```



Kombination mit lokaler Suche (Hybridisierung)



Prinzip eines memetischen Algorithmus

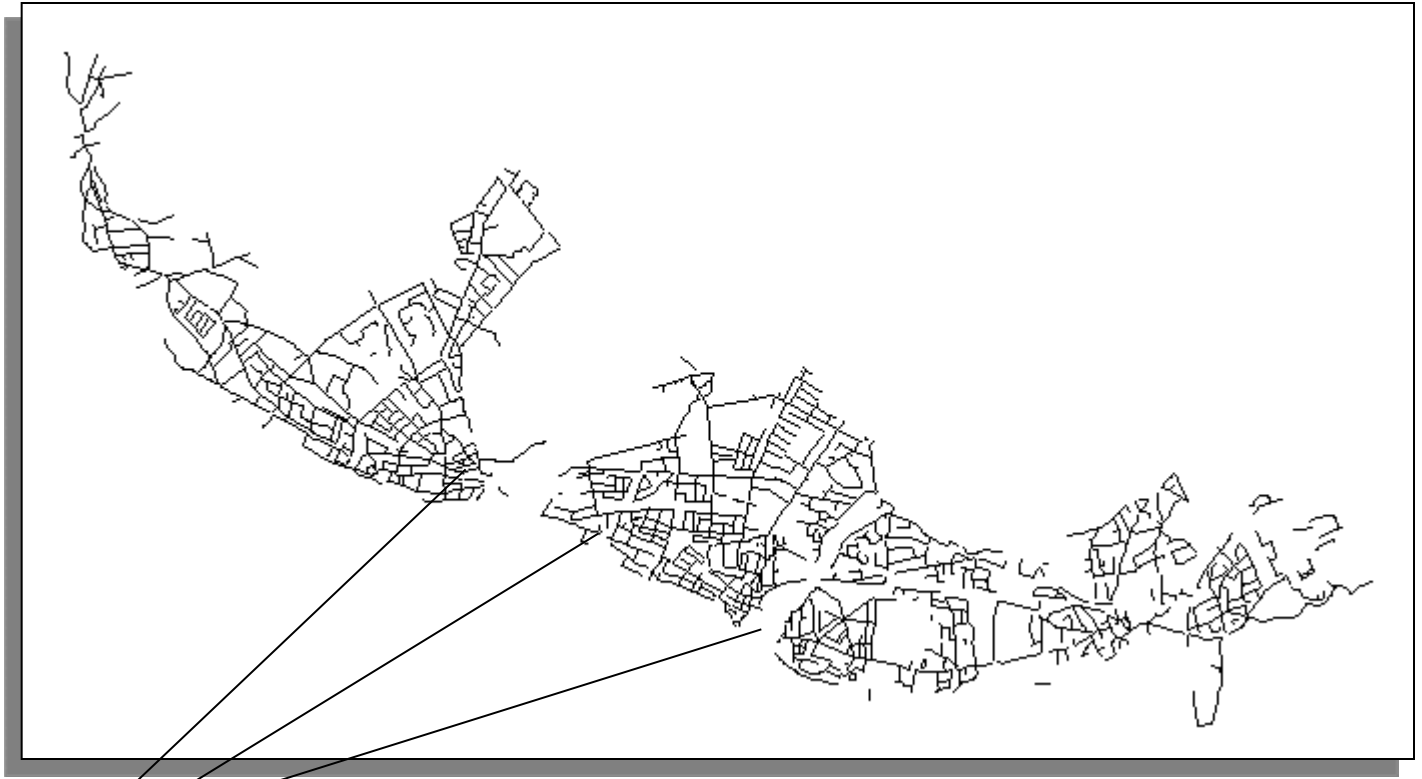


Routing in garbage collection





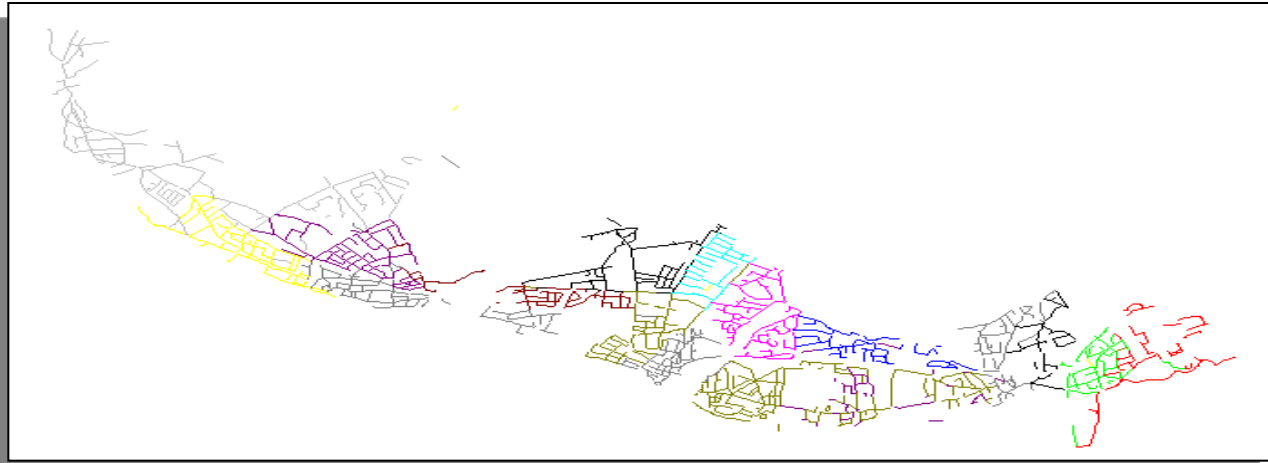
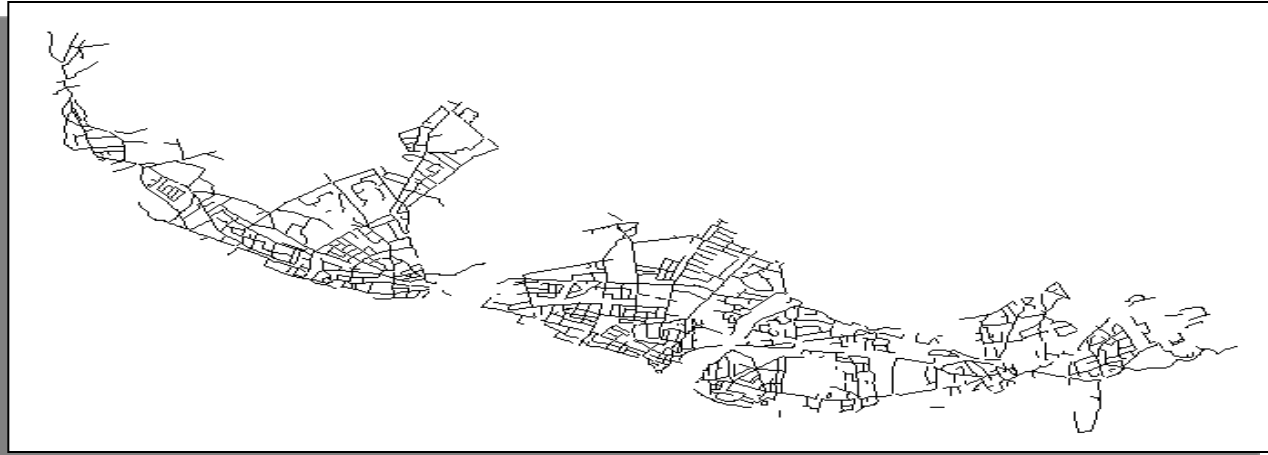
Distribution of garbage



Disconnected graph



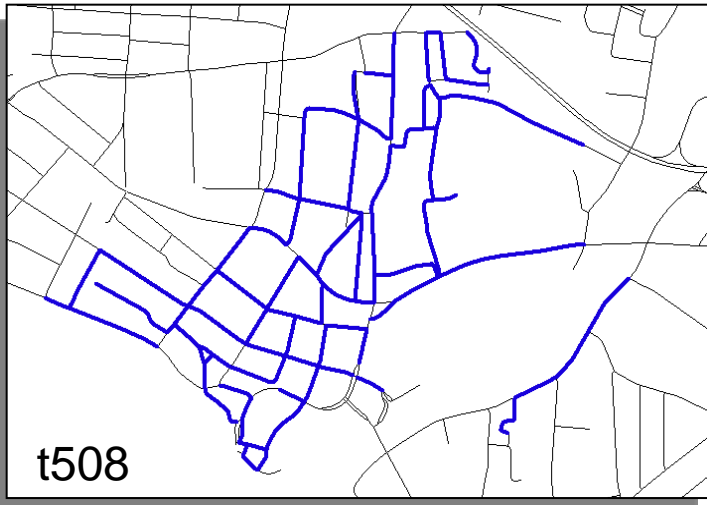
Partition in districts





Uncapacitated Routing

- Problem:
 - Determine the order in which the street segments are serviced
 - ... minimizing the total length
- Taking into account:
 - One way roads
 - Turn restrictions
 - Waiting times at crossings





Repräsentation des Problems

$$ind1 = ((s_1, d_1), \dots, (s_{s_n}, d_{s_n}))$$

- Unique mapping:
 - *Every edge has to have a logical direction: (s_i, d_i)*
- Reconstruction of a tour from the genotype
 - *Connect s_i and s_{i+1} with their shortest path*
- Shortest Paths:
 - *Turn penalties are included*

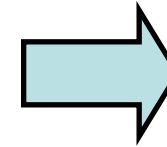


Operatoren

- Crossover
 - OX (Davis 1985)

$$p_1 = (1,2,3,4,5,6,7,8,9)$$

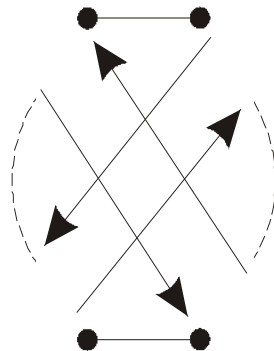
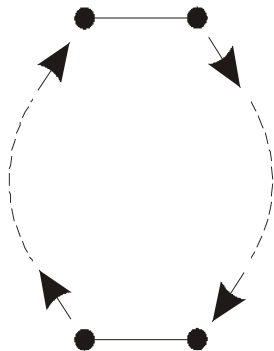
$$p_2 = (4,5,2,1,8,7,6,9,3)$$



$$o_1 = (*,*,*,4,5,6,7,*,*)$$

$$o_2 = (2,1,8,4,5,6,7,9,3)$$

- Mutation

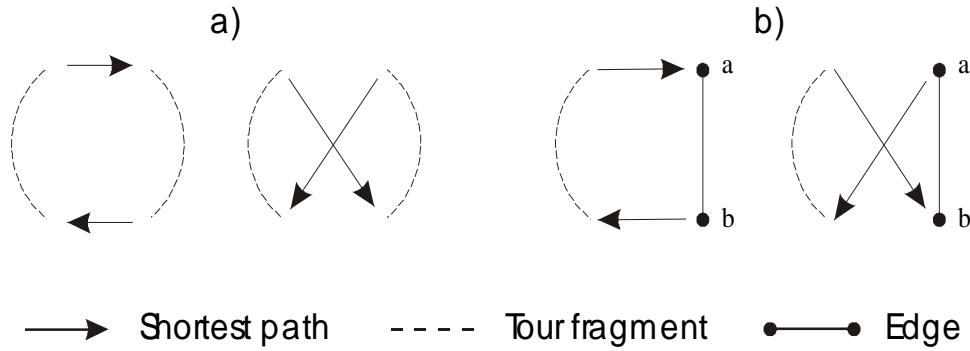


- Shortest path
- - - Tour fragment
- Edge

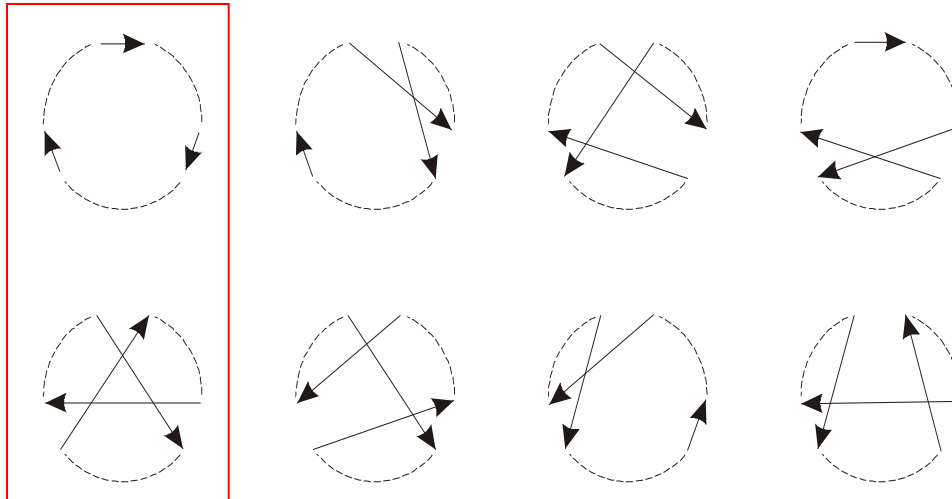


Lokale Suche

2-Opt



3-Opt





```

procedure MRPPTP_EALS
begin
   $t := 0$ 
  init population  $P_t$  of size  $n$ 
  for each individual  $i \in P_t$  do
    3-OptS ( $i$ )
  end
  while not converged do
    evaluate all  $i \in P_t$ 
    select  $m$  parents for mutation
    select  $c/2$  parent pairs for crossover
    copy the resulting offsprings to  $P_{t+1}$ 
    copy the  $n-m-c$  best individuals offsprings  $P_t$  of to  $P_{t+1}$ 
     $t := t+1$ 
    for each individual  $i \in P_t$  do
      DirOr-Opt ( $i$ )      in odd generations
      3-OptS ( $i$ )        in even generations
    end
  end
end
end

```

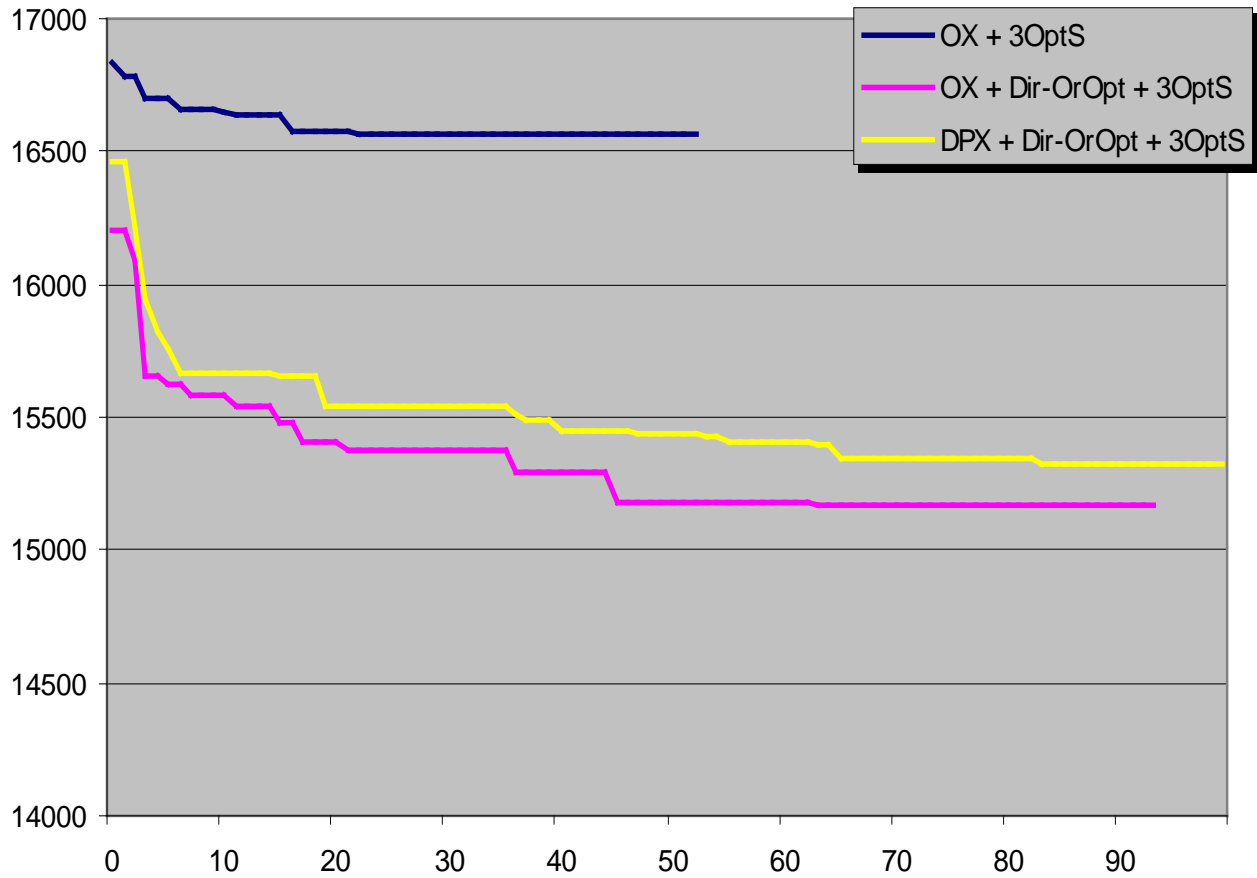
Stochastic Universal Sampling

Constant population size

Variable Neighborhood search



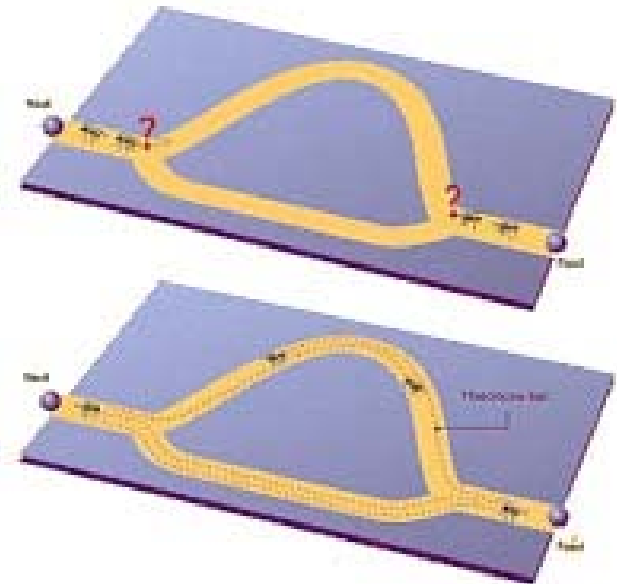
Konfiguration





Weitere naturinspirierte Vorgehensweisen

- Ameisenkolonien
 - Ameisen verstreuen Duftstoffe (Pheromone) als Orientierung für Artgenossen
 - Auf kurzen Wegen hohe Dichte an Pheromonen
 - Zieht weitere Ameisen an
- Doppelbrückenexperiment
 - 1) Gleich lange Wege
 - 2) Ein Weg doppelt so lang
 - 3) Nachträglich Hinzufügen eines kurzen Weges
- Beobachtungen
 - => Konvergenz auf effiziente Wege zwischen Bau und Nahrungsquelle
 - Entstehen eines stabilen Musters auf Makroebene ("Kollektive Intelligenz")





Ant systems

```
procedure AS
```

```
Begin
```

```
    Init_pheromones()
```

```
    while (stop_crit_not_satisfied)
```

```
        while (resources_available)
```

```
            propagate_new_ant()
```

```
        end
```

```
        pheromone_update()
```

```
        daemon_activities()
```

```
    end
```

```
end
```



Anwendung auf Routingprobleme

- *propagate_new_ant()*

$$p_{ij}^k = \frac{(\tau_{ij})^\alpha (1/d_{ij})^\beta}{\sum_{m \in N_i^k} (\tau_{im})^\alpha (1/d_{im})^\beta}$$

Lösungskonstruktion
(stochastisch & heuristisch)

- *pheromone_update()*

$$\Delta \tau_{ij}^k = 1/C^k$$

Verstärkung guter
Lösungsteile

$$\tau_{ij} \leftarrow (1 - \rho) \tau_{ij}$$

Evaporation (Vermeidung
frühzeitiger Konvergenz)

- *daemon_activities()*
 - Lokale Optimierung



Weitere naturinspirierte Vorgehensweisen

- Tabu Suche
 - Vermeide Schritte, die zu einer schon bekannten Situation führen
 - Mögliche Umsetzung: "Tabu-Liste" mit zuletzt ausgeführten Schritten / Maßnahmen
- Simulated Annealing (Simulierte Abkühlung)

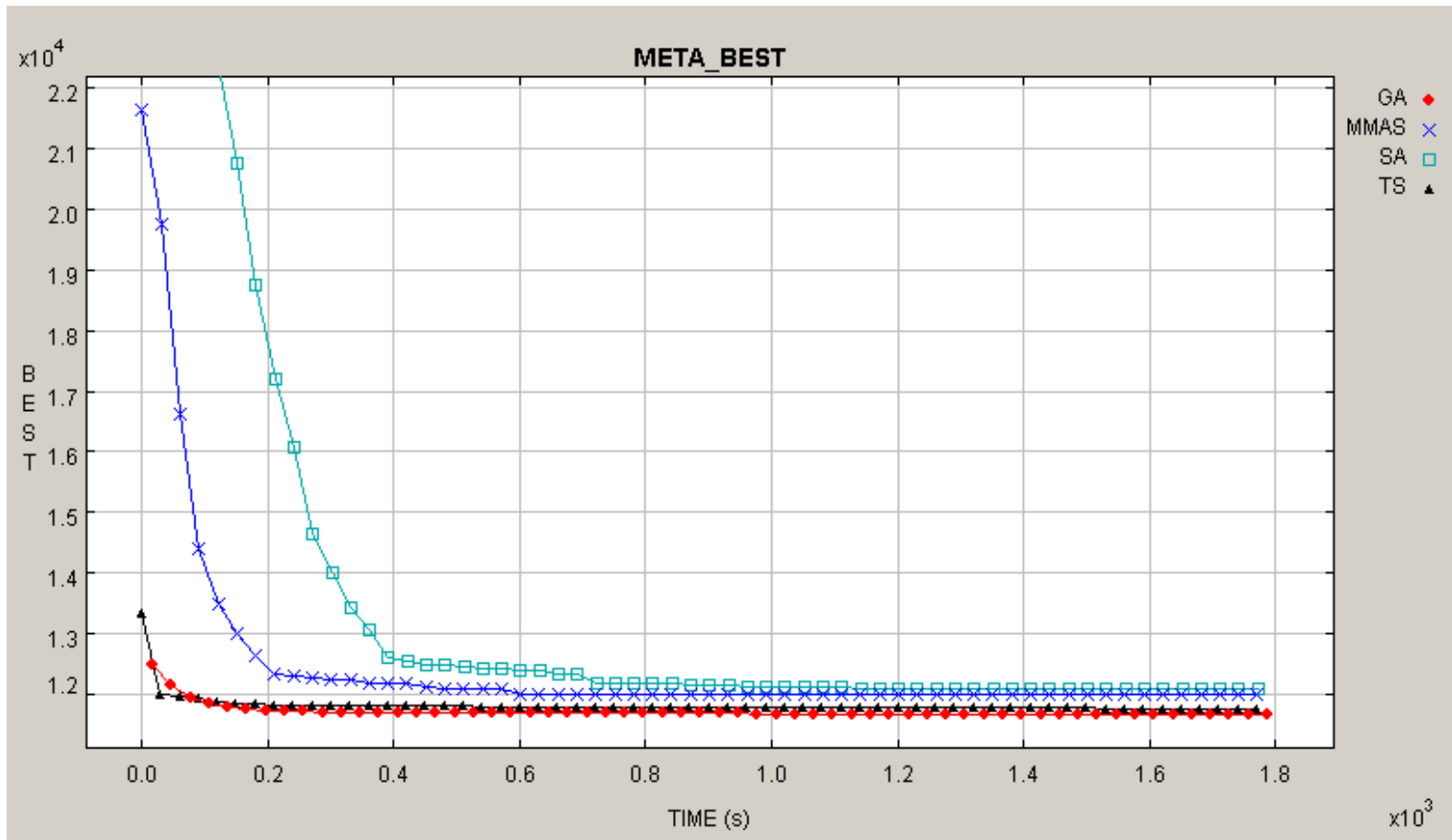


Tabu Search

```
procedure TS
begin
  initialize TabuMemory
  while ( stop_criterion_not_satisfied )
     $s' = \mathbf{BestAllowedNeighbor}(s, N(s), \mathbf{TabuMemory})$ 
    TabuMemory.add ( move( $s, s'$ ) )
     $s = s'$ 
    TabuMemory.add ( $s$ )
    if ( escape_triggered_by_TabuMemory )
      perform_diversifying_move
  end
end
```



Vergleich verschiedener Metaheuristiken

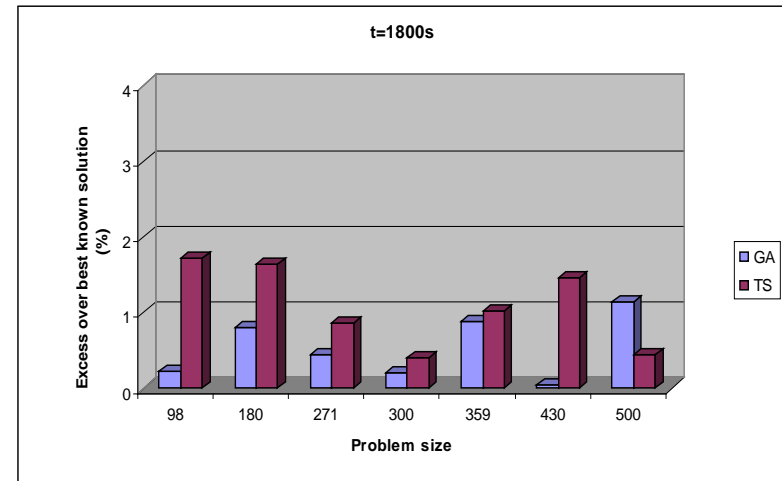
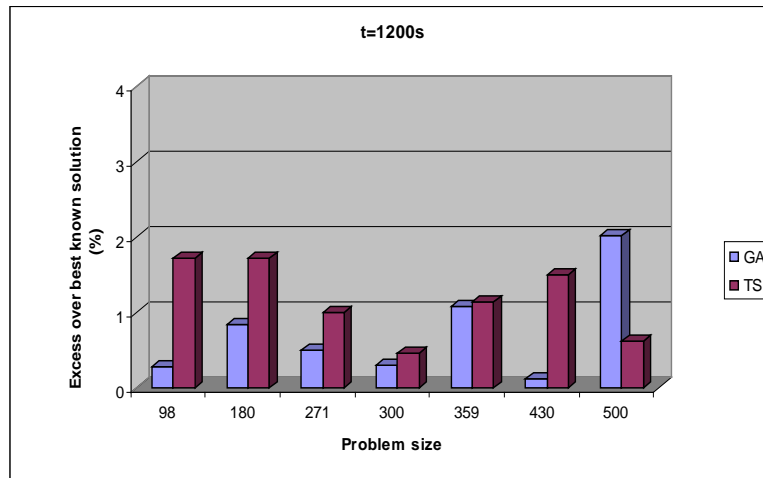
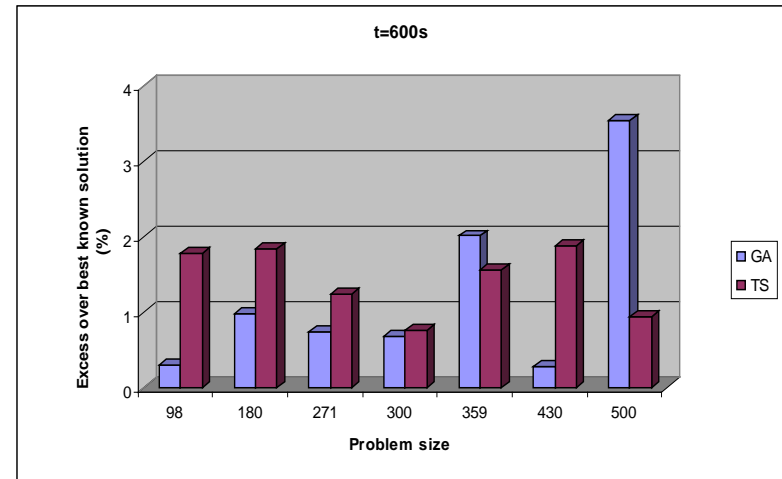


for a given instance (180 edges)



Vergleich GA und TS

Instance	Size	t = 600s		t = 1200s		t = 1800s	
		GA	TS	GA	TS	GA	TS
P102241	98	0.29	1.77	0.28	1.72	0.23	1.72
P820410	180	0.97	1.83	0.83	1.71	0.81	1.65
P143374	271	0.74	1.24	0.49	0.99	0.45	0.86
P122614	300	0.67	0.75	0.29	0.46	0.21	0.41
P1637710	359	2.01	1.55	1.07	1.13	0.89	1.03
P204211	430	0.27	1.87	0.11	1.50	0.04	1.46
P1840110	500	3.54	0.94	2.02	0.61	1.14	0.45





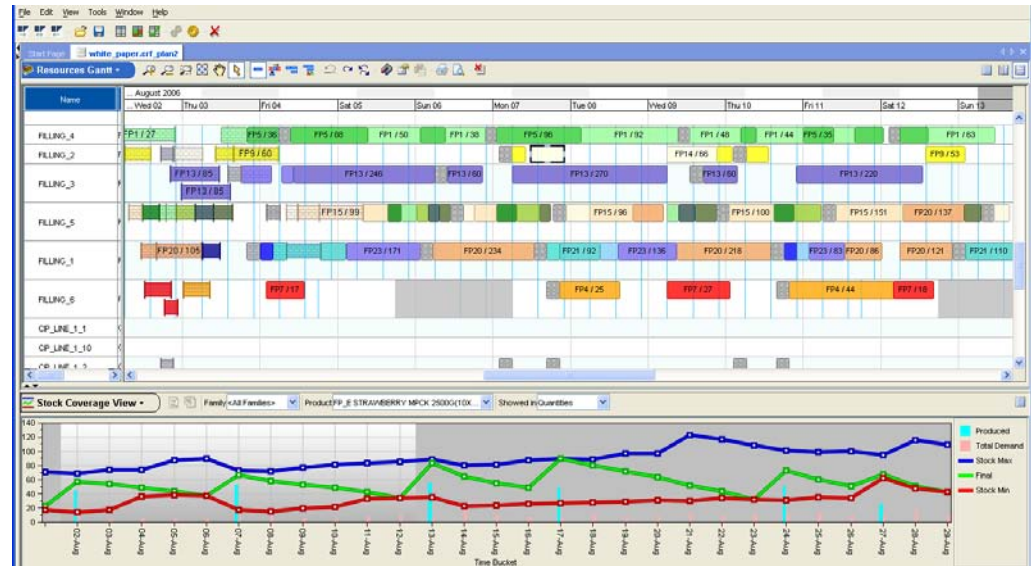
Klassifizierung der Verfahren als Metaheuristiken

- Gemeinsame Charakteristika
 - Keine problemspezifische Vorgehensweise
 - ❖ Generische Anwendbarkeit
 - Kombinierbarkeit mit lokaler Suche
 - Alle Verfahren sind aus einer Analogie mit der Natur entstanden
 - Stochastische Algorithmen
 - ❖ Keine Garantie, die optimale (beste) Lösung (Handlungsoption) zu finden
 - ❖ Keine Garantie über die Nähe zur besten Lösung



- Evolutionäre Algorithmen:

- Standardsoftware:
- Erfolgreicher Einsatz in Supply Chain Management Software (z.B. von SAP)
 - ❖ Lösung von Produktionsplanungsproblemen



- Viele Individuallösungen

- Ant Colony Systeme

- AntOptima (www.antoptima.com)
- Optimierung der Distributionslogistik bei Air Liquide U.S. (Harper, Davis 2008)



Optimale Pläne in einer dynamischen Welt

- Dargestellte Anwendungen
 - Geschlossene Modelle, insbesondere zeitlich
 - ❖ Zeitaspekt in Form von Multiperiodenmodellen
 - Festlegung der „Lösung“ im Detail (keine Regeln/Dynamik)
 - Problem:
 - ❖ Keine „Reaktionsaktionsfähigkeit“ der Lösung auf unvorhergesehene Ereignisse
 - Konsequenz
 - ❖ Praktische Lösungsqualität ist oft weit von der theoretischen „Optimalität“ entfernt



Rollierende Planung

Planungszeitpunkt:



Ende Dez.

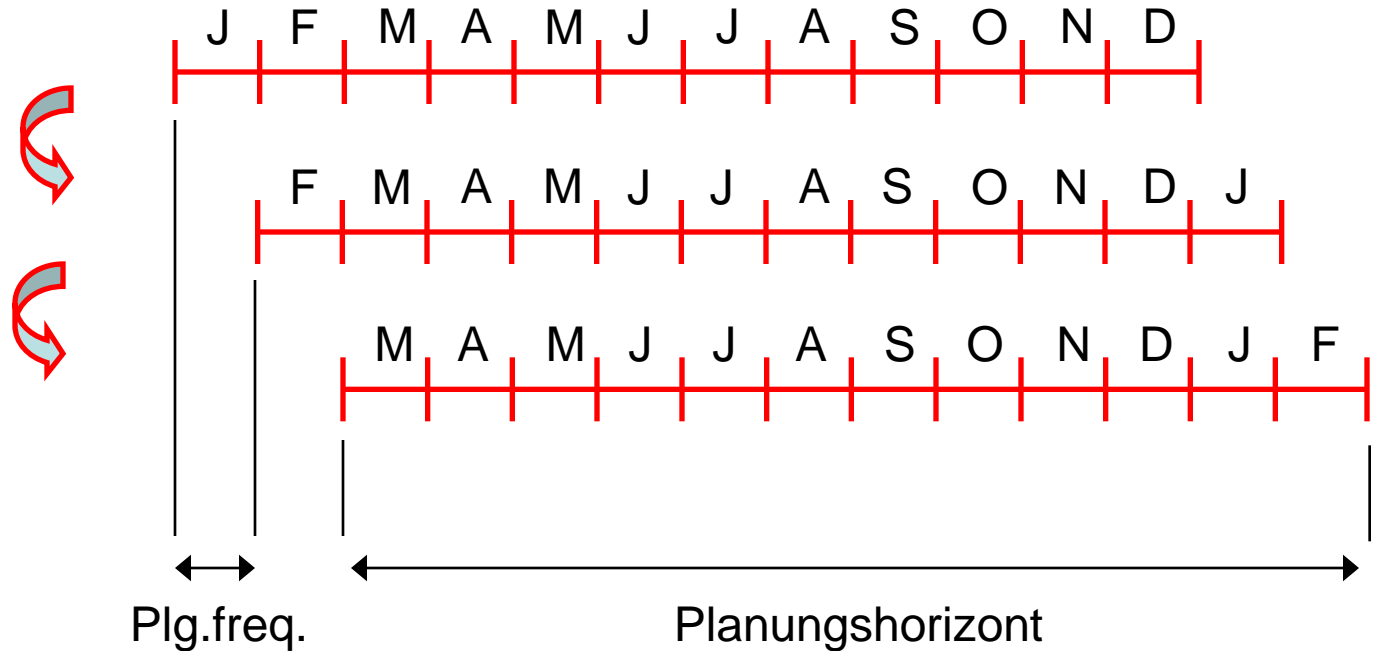
Ende Jan.

Ende Feb.

Beispiel:

Planungshorizont = 1 Jahr

Planungsfrequenz = monatlich



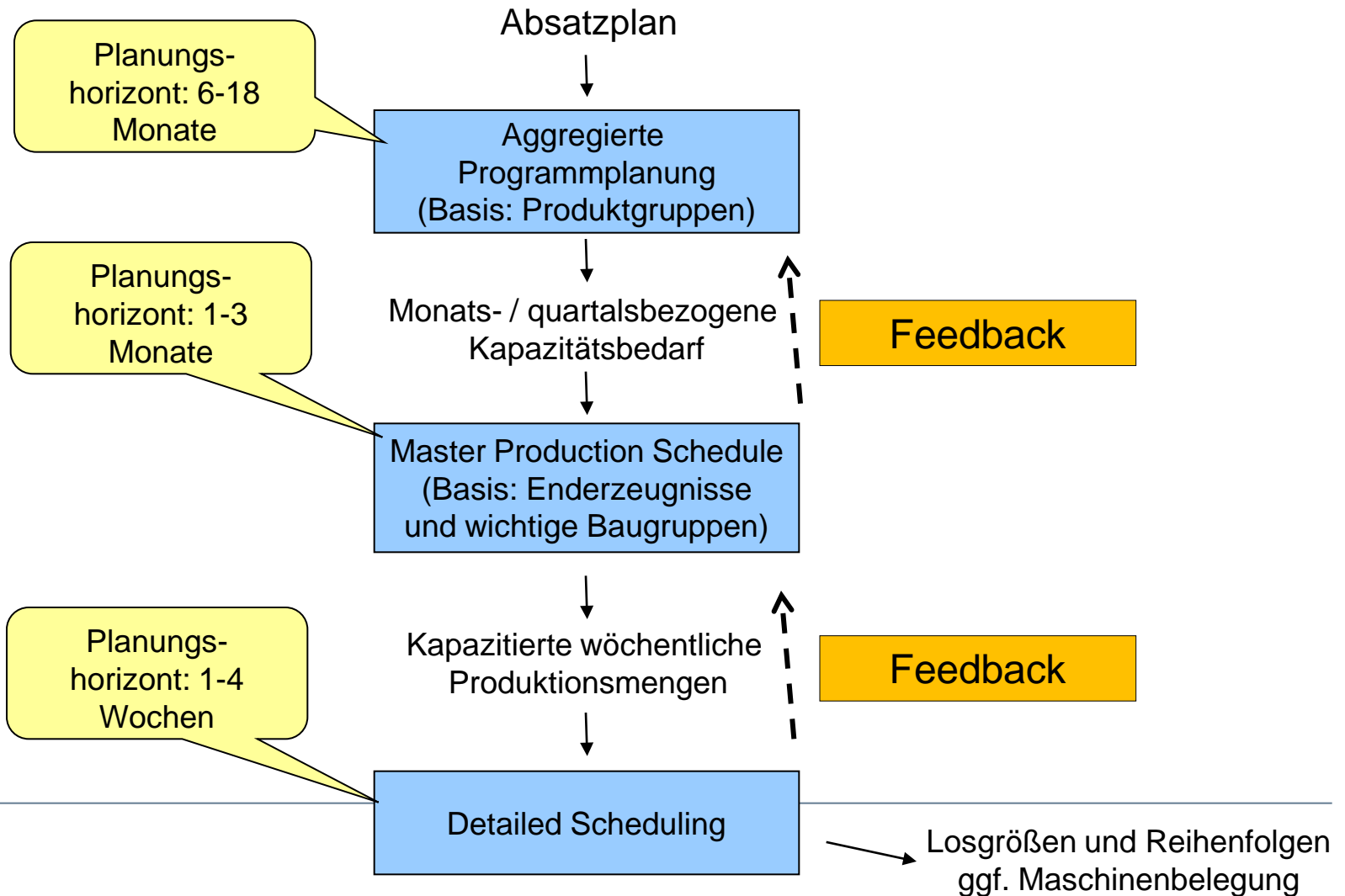


Planungsgrößen auf verschiedenen Zeit- und Aggregationsebenen

- Hierarchische Planungsmodelle (Schneeweiß 2002)
 - Dekomposition Gesamtproblem in verschiedene Ebenen mit unterschiedlichen Zeithorizonten
 - Mastermodell-Submodelle
 - ❖ Mastermodell gibt Entscheidungsspielraum für die Submodelle vor
 - ❖ Rückkopplung falls keine Lösung / Probleme im Submodell
 - Beispiel 1: Fuhrparkdisposition
 - Beispiel 2: Hierarchische Produktionsplanung (Hax and Meal 1975)



PPS-Sukzessivplanungskonzept (MRP II)





Alternativansätze zur zentralen Planung

- Beherrschung von Komplexität durch Zerlegung in selbststeuernde Regelkreise (Einheiten)
 - Festlegung von Regeln statt Determination von Zuständen
 - ❖ Organisation (Struktur- und Interaktionsgestaltung) vs. Planung
 - Konsequenz: Dynamisches Verhalten
 - ❖ Verzicht auf scheinbare „Sicherheit“ (determinierte Lösung)
 - ❖ „Hoffen“ auf Emergenz erwünschten Verhaltens



Exkurs: Logistische Systeme

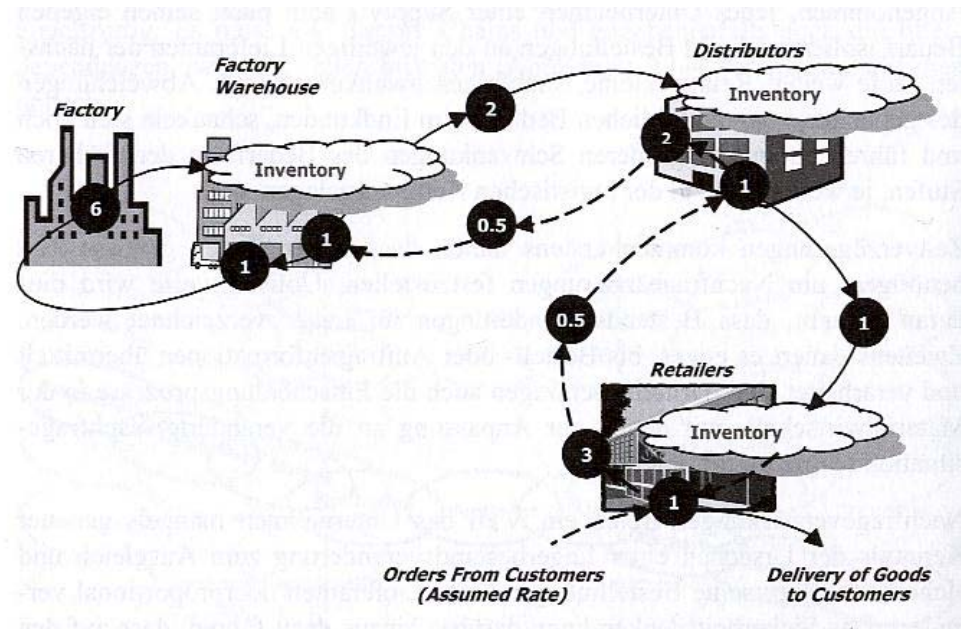
- Moderne Konzeption der Logistik
 - Planung und Steuerung des Materialflusses und der verbundenden Informationsflüsse im Unternehmen (Koordination der Leistungserstellung)
 - Offenes, dynamisches System (Interaktion mit Lieferanten, Lohnfertigern und Kunden)
- Koordination der kompletten Supply Chain (Lieferkette)
 - Zentrale Steuerung in einem Totalmodell nicht möglich
 - ❖ Machtpolitisch nur durch organisatorische Integration erreichbar (nicht gewünscht)
 - ❖ Technisch zu komplex



Die unkoordinierte Supply Chain

- Arbeiten von J. W. Forrester: „Industrial Dynamics“ (1958, 1961)
 - Simulation einer Supply Chain
 - Systemeigenschaften:
 - ❖ Lineare Struktur (kein Informationsaustausch über die Stufen hinweg)
 - ❖ Timelag zwischen den Stufen
 - Exogene Größe: Kundennachfrage
 - ❖ Leichte Schwankung

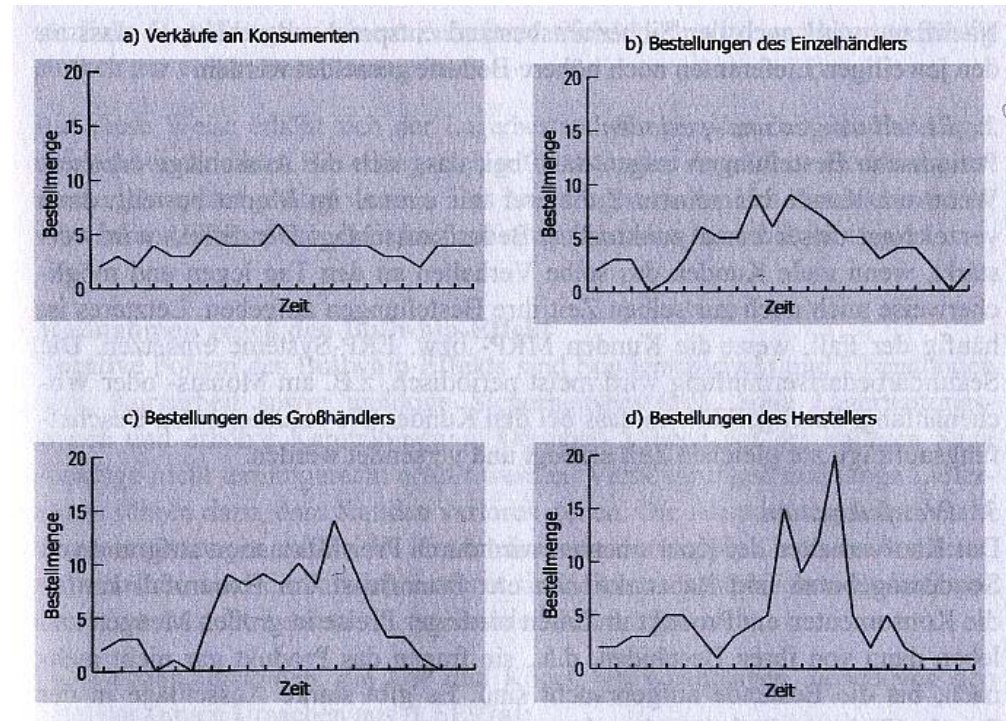
- Spätere Arbeiten
 - WORLD2-Model (World Dynamics, Forrester 1971)
 - WORLD3-Model (The Limits to Growth, Meadows 1972)





Die unkoordinierte Supply Chain

- Beobachtung:
 - Bestände und Auftragsgrößen in der Lieferkette unterliegen höheren Schwankungen, als dies durch die ursprüngliche Kundennachfrage gerechtfertigt war. Je ferner die Stufe vom Kunden, je ausgeprägter fielen die Schwankungen aus
 - „Bullwhip-Effekt“:
Lee H.L., Padmanabhan V., Whang S. (1997)



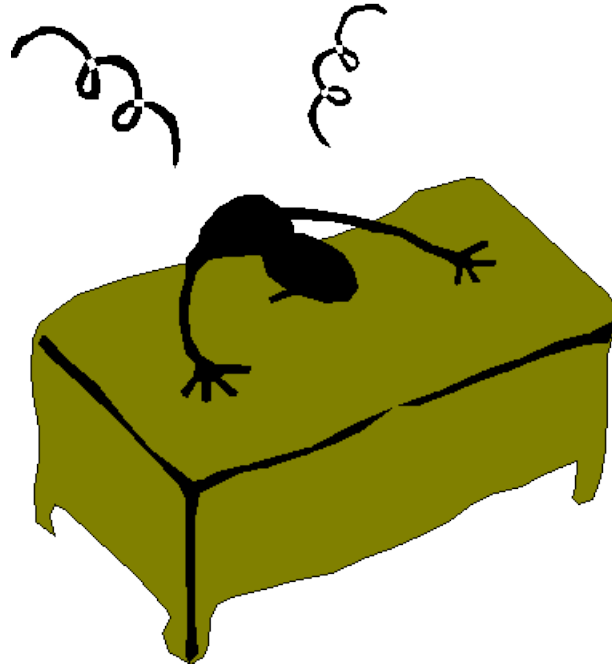


Maßnahmen gegen den Bullwhip-Effekt (Systemgestaltung)

- Verringerung der Cycle Times
 - In großem Maße erfolgt (im Vergleich zu 1958)
 - Keine strukturelle Änderung, begrenzte Auswirkungen
- Stufenübergreifender Informationsfluss
 - Strukturelle Änderung
 - Sehr erfolgreich (Anwendungen z.B. im Bereich „Quick Response“, VMI)
- Einfluss auf das Kundenverhalten
 - Glättung der Nachfrage (Vermeiden von Preisaktionen, Hamsterkäufen, ...)
 - Lee et al. (1997)



Fragen?





Quellen und weiterführende Links

- Rundreisen, etc.
 - Applegate, D. L. et al.: The Traveling Salesman Problem: A Computational Study, 2007.
- Genetische und evolutionäre Algorithmen
 - Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs, 3. Aufl., Berlin Heidelberg New York 1996.
- Ant Colony Systems
 - Dorigo, M., Stützle, T.: Ant Colony Optimization, 2004.
- Tabu Suche
 - Glover, F., Laguna, M.: Tabu Search, Dordrecht 1997.
- Bullwhip Effekt
 - Forrester J.W. (1958), Industrial Dynamics – a major breakthrough for decision makers, in Harvard Business Review, July-August 1958, S. 37-66
 - Lee H.L., Padmanabhan V., Whang S. (1997), The Bullwhip Effect in Supply Chains, in Sloan Management Review, Spring 1997, S. 93-102



Zeitschriften

- **Swarm Intelligence**
- Editor-in-Chief: Marco Dorigo
- ISSN: 1935-3812
- Springer.
- Start: 2007