Methoden und Anwendungen der Optimierung (MAO)

Kapitel 4: Lokale Suche

Univ.-Prof. Dr. Michael Schneider
Christian Schröder

Deutsche Post Chair – Optimization of Distribution Networks (DPO)
RWTH Aachen University

schroeder@dpo.rwth-aachen.de

WS 2017/18

Gesamtgliederung

Agenda

Lokale Suche

Ziele des Kapitels:

- wissen, was eine Nachbarschaft ist und wie lokale Suche funktioniert
- grundlegende Begriffe zur Beschreibung und Analyse von Nachbarschaften kennen
- wünschenswerte Eigenschaften für Nachbarschaften insbesondere am Beispiel des STSP erklären können

Agenda

Lokale Suche I

Merke: lokale Suche

Die Idee der *lokalen Suchverfahren* (kurz: *"Lokale Suche"*) besteht darin, eine gegebene zulässige Lösung durch elementare Operationen in eine andere zulässige Lösung zu transformieren, die bzgl. des Zielfunktionswerts eine bessere Bewertung aufweist.
Findet man eine solche bessere Lösung, so wechselt man zu dieser und beginnt wiederum mittels der Elementaroperationen mit der Suche nach einer nochmals verbessernden Lösung. Die lokale Suche wird solange fortgesetzt, bis keine elementare Operation mehr zu einer Verbesserung führt (hill climbing, greedy local search).

Lokale Suche II

Die Menge aller zulässigen Lösungen, welche durch Anwendung einer Elementaroperation auf $x \in X$ entstehen, bilden die *Nachbarschaft* $\mathcal{N}(x)$.

Nachbarschaft (Definition)

Eine Abbildung $\mathcal{N} : X \to Pot(X), x \mapsto \mathcal{N}(x) \subset X$ heißt *Nachbarschaftsabbildung* und $\mathcal{N}(x)$ die *Nachbarschaft von* x (bzgl. $\mathcal{N}$).

"Jeder zulässigen Lösung x wird eine Menge $\mathcal{N}(x)$ von zulässigen Lösungen zugeordnet."

verbessernder Nachbar (Definition)

Es sei z die zu minimierende Zielfunktion. Ein $x' \in \mathcal{N}(x)$ mit $z(x') < z(x)$ heißt *verbessernder Nachbar*.

Pseudocode Lokale Suche

Algorithmus 1 : Lokale Suche

`// Input: Lösung` x `und Nachbarschaft` $\mathcal{N}$

SETZE $t := 0$ und $x^0 := x$

repeat

  Durchsuche die Nachbarschaft $\mathcal{N}(x^t)$ nach einem verbessernden Nachbarn $x' \in \mathcal{N}(x^t)$

  if *verbessernder Nachbar* x' *gefunden* **then**

    SETZE $x^{t+1} := x'$ und $t := t + 1$

until *kein verbessernder Nachbar gefunden*

`// Output: lokal optimale Lösung` x^t

Suche nach verbessernden Nachbarn

Die *Suche nach verbessernden Nachbarn* kann in unterschiedlicher Weise ausgestaltet werden:

Erstensuche (=First-Improvement) Durchsuche die Elemente der Nachbarschaft nacheinander (in einer definierten Reihenfolge) und stoppe, wenn der erste verbessernde Nachbar gefunden wurde

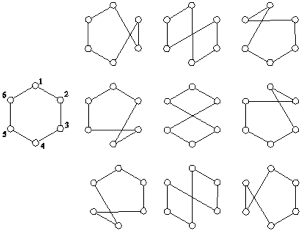
Bestensuche (Best-Improvement) Durchsuche die komplette Nachbarschaft nach verbessernden Nachbarn und wähle einen Nachbarn mit größter Verbesserung (i. A. nicht eindeutig)

ℓ-Erstensuche ($\ell > 1$) Durchsuche die Nachbarschaft bis ℓ verbessernde Nachbarn gefunden sind (oder fest steht, dass es keine ℓ verbessernden Nachbarn gibt). Wähle unter den gefundenen verbessernden Nachbarn einen mit größter Verbesserung

Beispiel: 2-opt Nachbarschaft TSP I

Für das symmetrische TSP (=STSP) besteht die *2-Opt-Nachbarschaft* $\mathcal{N}_{2Opt}(x)$ einer Tour x aus allen Touren $x' \neq x$, welche sich durch *Entfernen von zwei Kanten* (nicht benachbart!) und *Einfügen von zwei anderen Kanten* bilden lassen.

Beispiel: Für das symmetrische $n = 6$-Städte-TSP (STSP) besteht die 2-Opt-Nachbarschaft der Tour $x = (1, 2, 3, 4, 5, 6, 1)$ aus folgenden Elementen:



Beispiel: 2-opt Nachbarschaft TSP II

Allgemein enthält die 2-Opt-Nachbarschaft einer Instanz des STSP mit n Städten $n(n-3)/2$ Elemente:

$$|\mathcal{N}_{2Opt}(x)| = \frac{n(n-3)}{2}$$

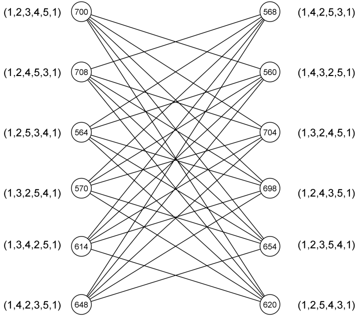
Beispiel: 2-opt Nachbarschaft TSP III

Beispiel: Für das $n = 5$ Städte STSP zur Distanzmatrix

$$D = \begin{pmatrix} 0 & 233 & 152 & 118 & 227 \\ 233 & 0 & 95 & 115 & 90 \\ 152 & 95 & 0 & 30 & 93 \\ 118 & 115 & 30 & 0 & 115 \\ 227 & 90 & 93 & 115 & 0 \end{pmatrix}$$

ist die Nachbarschaft rechts abgebildet.

Es gibt $(n-1)!/2 = 12$ verschiedene STSP Touren, jede hat $|\mathcal{N}_{2Opt}(x)| = n(n-3)/2 = 5$ Nachbarlösungen



Agenda

Nachbarschaftsbegriffe I

Nachbarschaftsgraph (Definition)

Der *Nachbarschaftsgraph* $(X, A_{\mathcal{N}})$ enthält als Knoten alle zulässigen Lösungen X und hat Bögen (x, x'), falls $x' \in \mathcal{N}(x)$ gilt.

Obige Abbildung zeigt den Nachbarschaftsgraphen eines 5 Städte STSP und die 2-Opt-Nachbarschaft. Da die 2-Opt-Nachbarschaft *symmetrisch* ist (d.h. $x' \in \mathcal{N}(x)$ genau dann wenn $x \in \mathcal{N}(x')$), ist der Graph *ungerichtet* gezeichnet.

Erreichbarkeit, starker Zusammenhang (Definition)

Eine Lösung $x \in X$ *ist von* $y \in X$ *erreichbar*, falls es einen Weg von y zu x im Nachbarschaftsgraphen gibt, d.h. es gibt eine Folge $(x_0 = y, x_1, x_2 \dots, x_p, x_{p+1} = x)$ von Lösungen in X mit $x_{i+1} \in \mathcal{N}(x_i)$ für $i = 1, 2, \dots, p$. Eine Nachbarschaft $\mathcal{N}$ heißt *stark zusammenhängend*, falls für je zwei Lösungen $x, y \in X$ gilt, dass x von y aus erreichbar ist.

Nachbarschaftsbegriffe II

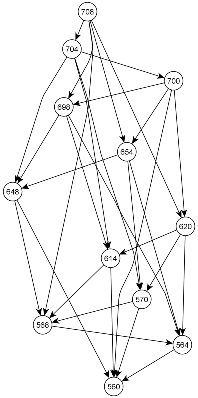
Transitionsgraph (Definition)

Der *Transitionsgraph* $(X, A_{\mathcal{N}}^{trans})$ enthält als Knoten alle zulässigen Lösungen X und hat Bögen (x, x'), falls x' ein verbessernder Nachbar von x ist.

Bemerkungen:

1. Der Transitionsgraph ist ein gerichteter und kreisfreier (=azyklischer) Teilgraph des Nachbarschaftsgraphen.
2. Lokal optimale Lösungen entsprechen Knoten des Transitionsgraphen mit Außengrad 0.

Transitionsgraph für $\mathcal{N}_{2Opt}$ im Beispiel:



Nachbarschaftsbegriffe III

exakte Nachbarschaft (Definition)

Eine Nachbarschaft $\mathcal{N}$ heißt *exakt*, wenn jedes lokale Optimum auch ein globales Optimum ist.

Beispiel: Es sei $X = \{1, 2, 3, 4, 5, 6, 7\}$. Die Kosten $c(x)$ und Nachbarschaft $\mathcal{N}(x)$ einer Lösung $x \in X$ sind in folgender Tabelle geben:

x	1	2	3	4	5	6	7
$c(x)$	2	3	1	2	2	1	2
$\mathcal{N}(x)$	$\{2\}$	$\{1,3\}$	$\{2,4\}$	$\{3\}$	$\{6\}$	$\{5,7\}$	$\{6\}$

Welche Eigenschaften hat diese Nachbarschaft?

Nachbarschaftsbegriffe IV

Durchmesser einer Nachbarschaft (Definition)

Für eine Nachbarschaft $\mathcal{N}$ heißt

$$\max_{x,y \in X} \{\text{Länge eines kürzesten Wegs von } x \text{ zu } y\}$$

(=Anzahl der Bögen des Wegs im Nachbarschaftsgraphen $(X, A_{\mathcal{N}})$) der *Durchmesser* der Nachbarschaft.

Bemerkungen:

1. Ist die Nachbarschaft nicht stark zusammenhängend, so ist der Durchmesser ∞.
2. Für das 5-Städte STSP und die 2-Opt-Nachbarschaft $\mathcal{N}_{2Opt}$ ist der Durchmesser 3.
3. Man kann allgemein zeigen, dass beim STSP der Durchmesser der 2-Opt-Nachbarschaft zwischen $n/2$ und $n-1$ beträgt (Hoos und Stützle, 2005, S. 205).

Wünschenswerte Eigenschaften einer Nachbarschaft I

1. **Starker Zusammenhang:** Je zwei Lösungen $x, y \in X$ sollten durch eine Folge von Elementaroperationen (=Moves) ineinander zu überführen sein.
2. **Symmetrie:** $x' \in \mathcal{N}(x) \Leftrightarrow x \in \mathcal{N}(x')$
3. **Effiziente Berechnung des Zielfunktionswerts:** Lokale Suche (Alg. 1) erfordert die Bestimmung von $c(x')$ für alle $x' \in \mathcal{N}(x^t)$ bei gegebenem x^t. Möglichkeiten:
   - Berechnung von $c(x')$ direkt ("from scratch")
   - Berechnung von $c(x')$ mittels Gewinn $\Delta(x', x^t) = c(x^t) - c(x')$
   - Berechnung von $\Delta(x', x^t)$ mittels vorab bestimmter Hilfswerte

Wünschenswerte Eigenschaften einer Nachbarschaft II

4. **Effiziente Konstruktion der Nachbarn:**
   Jede Nachbarlösung $x' \in \mathcal{N}(x^t)$ sollte sich effizient aus $x^t \in X$ konstruieren lassen.
5. **Effiziente Prüfung der Zulässigkeit:**
   Ist die Nachbarschaft über eine Menge von Moves $m \in M$ gegeben, so überführen manche Moves ein gegebenes $x^t \in X$ in ein *unzulässiges* $x' = m(x^t) \notin X$. Diese Fälle sollten effizient erkannt werden können.
6. **Effiziente Suche nach zulässigen verbessernden Nachbarn:** 3.+4.+5. sollen zusammen so effizient wie möglich ablaufen.

Agenda

Lokale Suche und das asymmetrische TSP

Bemerkungen zum asymmetrischen TSP (=ATSP):

1. Durch Löschen von zwei Bögen aus einer Tour entstehen zwei (gerichtete) Wege mit jeweils mindestens zwei Knoten. Diese werden auch als Segmente bezeichnet. Ein 2-Opt-Austauschschritt kehrt die Orientierung eines Segments um.
2. Im ATSP gibt es zu zwei entfernten Bögen daher immer genau zwei neue Touren, welche sich nur in ihrer Orientierung unterscheiden.
3. Die Berechnung des Gewinns muss die Richtung der entfernten und eingefügten Bögen sowie die Invertierung eines Segments mit berücksichtigen.

Nachfolgend betrachten wir ausschließlich symmetrische TSP (=STSP).

Lokale Suche und das (S)TSP I

Effiziente Bestimmung des Zielfunktionswerts beim STSP: Die Berechnung der *Länge* $c(x')$ *einer Tour* x', welche durch Kantentausch aus der Tour x entsteht, kann effizient durch $c(x') = c(x) - \Delta(x, x')$ bestimmt werden.

Der *Gewinn* $\Delta(x, x')$ (mit $\Delta(x, x') > 0$ für verbessernde Nachbarn x') ist

"Länge entfernte Kanten − Länge eingefügte Kanten"

Die *Bestimmung des Gewinns* erfordert nur konstante Laufzeit $\mathcal{O}(1)$, während die Neuberechnung der Tourlänge $c(x')$ ("from scratch") Aufwand $\mathcal{O}(n)$ hat.

Sequentielle Suche I

Unter *sequentieller Suche* wollen wir ein Kantenaustauschverfahren verstehen, das derart abwechselnd Kanten löscht und hinzufügt, dass der Endknoten der zuletzt gelöschten Kante gleichzeitig der Startknoten der nachfolgend hinzugefügten Kante ist und umgekehrt der Endknoten der hinzugefügten Kante der Startknoten der nächsten gelöschten Kante ist. Die entsprechend generierten Wege aus gelöschten und hinzugefügten Kanten bilden sog. *alternierende Wege*.

Merke: k-opt und sequentielle Suche

Die 2-Opt- und die 3-Opt-Nachbarschaft können mit sequentieller Suche vollständig durchsucht werden, für die k-Opt-Nachbarschaft mit $k \geq 4$ gilt dies nicht.

Sequentielle Suche II

Formal kann die sequentielle Suche für k-Opt folgendermaßen definiert werden: es werden k *Kanten gelöscht* und k *Kanten hinzugefügt*. Durch $2k$ Knoten

$$t_1, t_2, \ldots, t_{2k}$$

lassen sich dabei die *gelöschten Kanten* eindeutig durch

$$\{t_{2i-1}, t_{2i}\}$$

für $i = 1, ..., k$ und die *hinzugefügten Kanten* durch

$$\{t_{2i}, t_{2i+1}\}$$

für $i = 1, ..., k$ beschreiben (Verabredung: $t_{2k+1} := t_1$).

Sequentielle Suche III

Um zu einer effizienten Nutzung der sequentiellen Suche zu gelangen ist es notwendig, die potentielle Verkürzung einer Tour durch die Kantenaustauschoperationen zu untersuchen.

Für die 2-Opt-Nachbarschaft $\mathcal{N}_{2Opt}$ definieren wir den Gewinn des 1-ten und 2-ten Tauschs als die Differenz aus der Länge der gelöschten und der hinzugefügten Kante, also

$$g_1 := c_{t_1,t_2} - c_{t_2,t_3} \qquad \text{und} \qquad g_2 := c_{t_3,t_4} - c_{t_4,t_1}.$$

Damit durch den Kantenaustausch eine Verkürzung der Tour entsteht, muss der kumulierte Gewinn $g_1 + g_2$ positiv sein.

Sequentielle Suche IV

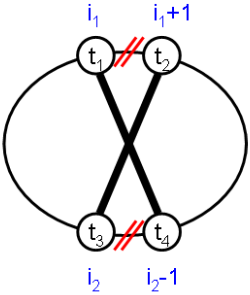
Speziell folgt: $g_1 + g_2 > 0 \qquad \Rightarrow \qquad g_1 > 0 \quad \text{oder} \quad g_2 > 0$

Wegen der Symmetrie in g_1 und g_2 bedeutet dies, dass wir uns bei der Suche nach einer Verbesserung immer auf solche Folgen beschränken können, deren Teilsummen $g_1 > 0$ und $g_1 + g_2 > 0$ sämtlich positiv sind. Aus der Sicht der sequentiellen Suche kann man formulieren, dass diese lediglich *mit den „richtigen Kanten“ begonnen werden* muss.

Sequentielle Suche für 2-Opt I

Algorithmus 2 : 2-Opt-Bestensuche

```
// Input: Tour x = (x_1, x_2 ..., x_n, x_1)
SETZE G* := 0
for i_1 = 1, ..., n und s ∈ {−1, +1} do
    SETZE t_1 := x_{i_1}, t_2 := x_{i_1+s}
    SETZE Bound := c_{t_1,t_2} − G*/2
    for t_3 ∈ N(t_2) mit c_{t_2,t_3} < Bound do
        SETZE i_2 := Position(t_3) und t_4 := x_{i_2−s}
        SETZE G := c_{t1,t2} − c_{t2,t3} + c_{t3,t4} − c_{t4,t1}
        if G > G* then
            SETZE G* := G
            SETZE t* := (t_1, t_2, t_3, t_4)
// Output: Gewinn G* und für G* > 0 die
   Knotenkombination t*
```



Sequentielle Suche für 2-Opt II

Bemerkung:

1. für jeden Knoten $t \in V$ sei $N(t)$ eine nach Abstand $c_{t,u}$ nicht-fallend geordnete Liste (→sog. Nachbarliste) aller anderen Knoten $u \in V \setminus \{t\}$
2. Vorteil der seq. Suche: Der Aufwand für einen Suchschritt reduziert sich deutlich, im Average-Case von quadratisch zu (etwas langsamer als) linear
3. Nachteil der seq. Suche: Vorab (aber nur einmalig) muss für alle Knoten die Nachbarliste aufgebaut werden; der Aufwand beträgt $\mathcal{O}\left(n^2 log(n)\right)$ Zeit

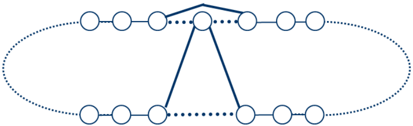
Das Prinzip der sequentiellen Suche lässt sich auch auf andere STSP Nachbarschaften anwenden:

- Relocation-Nachbarschaft
- Teile der k-Opt Nachbarschaft für $k \geq 4$

Weitere Nachbarschaften für das STSP

Relocation bezeichnet das Entfernen eines Knotens (=Stadt) aus ihrer aktuellen Position und ihr Einfügen an einer neuen Position. Relocation ist eine *Knotenaustauschnachbarschaft*. Dies ist ein spezieller 3-Opt-Schritt.

Beispiel:



Weitere Nachbarschaften für das STSP

Allgemein definiert man die *k-Opt-Nachbarschaft* durch ein Kantenaustauschverfahren, bei dem *k Kanten entfernt* und *k Kanten eingefügt* werden.

Merke: Suchaufwand beim TSP

Für große TSP ist der Aufwand für das Durchsuchen der k-Opt-Nachbarschaft ab $k \geq 4$ nicht mehr akzeptabel.

Zur weiteren Verbesserung von lokal optimalen Lösungen bzgl. 2-Opt bzw. 3-Opt können

1. Kantentauschverfahren mit variabler Suchtiefe entsprechend der *Lin-Kernighan-Nachbarschaft* oder *Edge-Ejection-Chain-Nachbarschaften*
2. oder *Metaheuristiken*

eingesetzt werden.

Zur Vertiefung...

- (Aarts und Lenstra, 1997) Kap. 1
- (Hoos und Stützle, 2005) Kap. 1, Abs. 5.1
- (Michalewicz und Fogel, 2000) Abs. 3.2
- (Michiels u. a., 2007) Kap. 1 und 2

[Aarts und Lenstra 1997] Aarts, E. ; Lenstra, J.K.: *Local Search in Combinatorial Optimization*. Chichester : Wiley, 1997

[Hoos und Stützle 2005] Hoos, H.H. ; Stützle, T.: *Stochastic Local Search Foundations and Applications*. San Francisco, CA : Morgan Kaufmann Publishers, Elsevier, 2005. – ISBN 1558608729

[Michalewicz und Fogel 2000] Michalewicz, Zbigniew ; Fogel, David B.: *How to solve it: Modern heuristics ; with 7 tables*. Corr. 2. printing. Berlin : Springer, 2000

[Michiels u. a. 2007] Michiels, Wil ; Aarts, Emile ; Korst, Jan: *Theoretical Aspects of Local Search*. Berlin, Heidelberg : Springer, 2007. – ISBN 978-3-540-35853-4