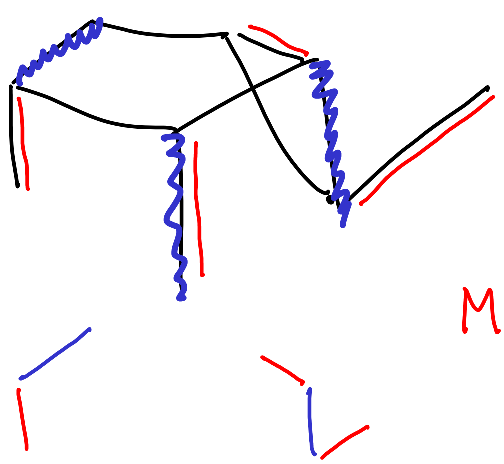
Näherungslösungen, Approximationsalgorithmen

- kein Algorithmus zur Berechnung einer Optimallösung verfügbar.
  - NP-schwer
  - exakter Algorithmus zu aufwändig
  - nicht bekannt.

A) Gütegarantie

Bsp. größte Paarung (ist in polynomieller Zeit berechenbar)

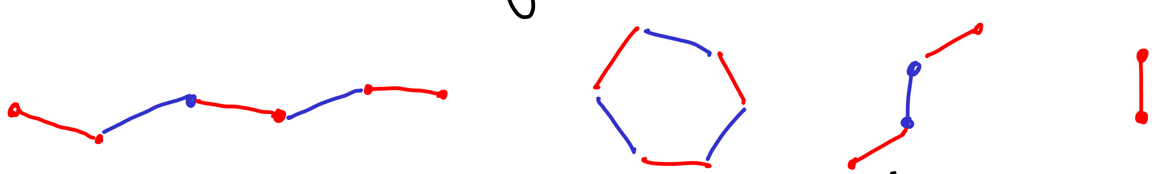
Gieriger Algorithmus findet eine maximale Paarung: G



Man kann keine Kante mehr hinzufügen.

M ... eine andere (oder größte) Paarung

Die Komponenten der symmetrischen Differenz von G und M sind alternierende Wege oder Kreise.



isolierte rote Kante nicht möglich

schlimmster Fall

in jeder Komponente: BLAU $\geq \frac{\text{ROT}}{2}$

B) ohne Gütegarantie

„Heuristik"

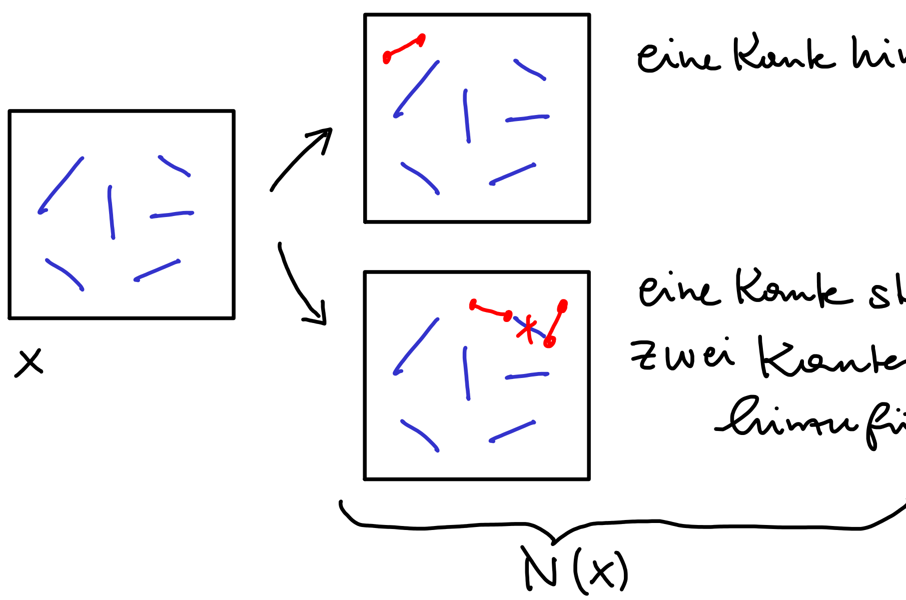
---

lokale Optimierung, steilster Abstieg

Versuche eine Lösung durch eine „kleine Änderung" zu verbessern.

<u>Nachbarschaft</u>: Zu jeder Lösung x ist eine Menge $N(x)$ von „ähnlichen" Lösungen definiert.

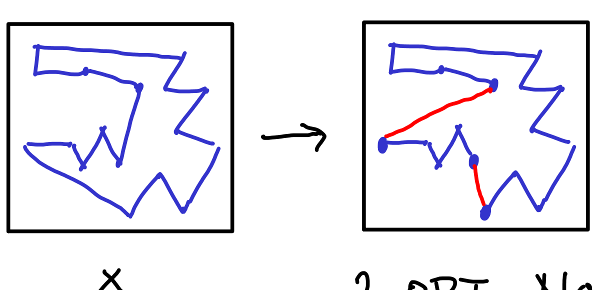
Bsp ① Paarungen



eine Kante hinzufügen

eine Kante streichen
zwei Kanten hinzufügen

Bsp ② Rundreisen



zwei Kanten streichen
zwei Kanten einfügen

2-OPT-Nachbarschaft

Lokale Optimierung:

- Menge S von möglichen Lösungen
- Zielfunktion $c: S \to \mathbb{R}$

AUFGABE: finde $x \in S$ mit $c(x) \to$ MIN (oder $\to$ MAX)

- Nachbarschaft $N(x) \subseteq S$ für alle $x \in S$

Algorithmus:

Beginne mit einer Startlösung $x = \underline{x_0}$

<u>Schleife</u>:

Wenn es ein $x' \in N(x)$ mit $c(x') < c(x)$ gibt:

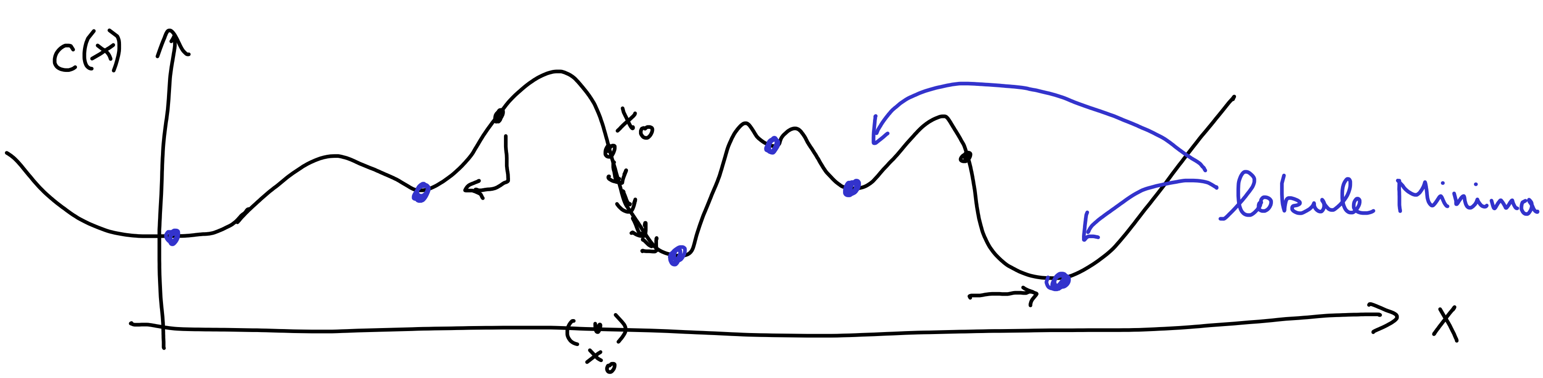
$x := x'$

Andernfalls: STOPP

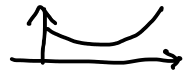
SATZ: Wenn S endlich ist, terminiert lokale Optimierung immer, und zwar in einem <u>lokalen Optimum</u>.

DEF: x ist <u>lokales Minimum</u>: $\forall\, x' \in N(x): \; c(x') \geq c(x)$

Muss kein (globales) Minimum sein!



[Bei einer <u>konvexen</u> Funktion $c: \mathbb{R}^n \to \mathbb{R}$ ist jedes lokale Minimum ein globales Minimum.]



größere Nachbarschaft $N(x)$

- $\to$ weniger lokale Minima
  $\to$ tendenziell bessere Lösungen
- größere Laufzeit zum Durchsuchen der Nachbarschaft.

Wiederholung mit verschiedenen Startlösungen x_0
(z.B. zufällig)

Variante <u>steilster Abstieg</u> (Gradientenverfahren):

- Wähle das $x' \in N(x)$ mit kleinstem $c(x')$ [falls $< c(x)$]

Laufzeit?

Wie kommt man aus einem lokalen Optimum wieder heraus?

simulated annealing

- Akzeptiere auch <u>Verschlechterungen</u> der Zielfunktion (aber mit geringerer Wahrscheinlichkeit)

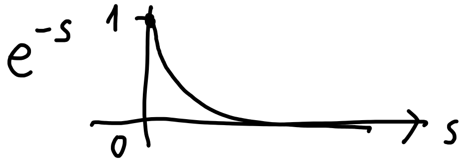
Schleife: $\boxed{T = \text{„TEMPERATUR"}}$

Wähle einen <u>zufälligen</u> Nachbarn $x' \in N(x)$

$\Delta c := c(x') - c(x)$

Wenn $\Delta c < 0$: $x := x'$

Wenn $\Delta c \geq 0$: <u>Setze $x := x'$</u> mit Wahrscheinlichkeit $e^{-\frac{\Delta c}{T}}$
„akzeptiere x'"



Δc groß $\rightarrow$ Wahrscheinlichkeit gering

$\Delta c \sim 0$ $\rightarrow$ " nahe bei 1

T groß $(T \to \infty)$: Alle Nachbarn werden akzeptiert.
Zielfunktion spielt keine Rolle

T klein $(T \to 0)$: Keine Verschlechterungen werden akzeptiert.
$\Rightarrow$ lokale Optimierung.

[festes T : Verteilung strebt gegen die Boltzmannverteilung:
Wahrscheinlichkeit für $x \in S$ ist proportional zu $e^{-\frac{c(x)}{T}}$.]

Heuristische Suchverfahren sind einfach und vielseitig.

- Das einzige, was man braucht ist eine Nachbarschaft.

- Komplizierte Nebenbedingungen lassen sich als „Strafterme" in die Zielfunktion einbauen.