Definition (TSP, Problem des Handlungsreisenden)

gegeben: Gewichteter, gerichteter Graph.

gesucht: Eine Rundreise (= einfacher Kreis),

mit:

- Enthält alle Knoten.
- Summe der Kosten der betretenen Kanten minimal.

□

Graph durch Distanzmatrix gegeben. ($\infty \Leftrightarrow$ keine Kante). Etwa,

$$M = \big(M(u,v)\big)_{1 \le u,v \le 4}$$

$$\begin{array}{r c l}
 & & \text{Zeile} \\
M = & \begin{pmatrix} \infty & 10 & 15 & 20 \\ 5 & \infty & 9 & 10 \\ 6 & 13 & \infty & 12 \\ 8 & 8 & 9 & \infty \end{pmatrix} & \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} \\
\text{Spalte} \to & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} &
\end{array}$$

$4 \xrightarrow{9} 3$

Die Kosten der Rundreise $R = (1,2,3,4,1)$ sind

$$K(R) = M(1,2) + M(2,3) + M(3,4) + M(4,1)$$

$$= 10 + 9 + 12 + 8 = 39$$

Für $R' = (1,2,4,3,1)$ ist $K(R') = 35$.

Ist das minimal?

1. Versuch: Knoten 1 festhalten, alle $(n-1)!$ Permutation auf $2, \ldots, n$ aufzählen, jeweils Kosten ermitteln.

Zeit

$$\Omega((n-1)! \cdot n) = \Omega(n!) \geq \left(\frac{n}{2}\right)^{\frac{n}{2}}$$

$$= 2^{((\log n) - 1) \cdot \frac{n}{2}}$$

$$= 2^{\frac{(\log n) - 1}{2} \cdot n}$$

$$= 2^{\Omega((\log n) \cdot n)} \gg 2^n$$

Beachte:

$$n^n = 2^{(\log n) \cdot n}$$

$$n! \geq 2^{\frac{\log n - 1}{2} \cdot n}$$

$\frac{1}{2}$ im Exponenten!

$$n! \ll n^n.$$

2. Versuch: Greedy. Gehe zum jeweils nächsten Knoten, der noch nicht vorkommt (Analog Prim).

Im Beispiel mit Startknoten 1

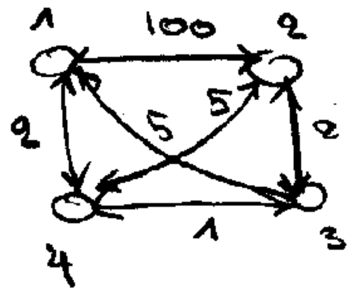
$$R = (1,2,3,4,1), \quad K(R) = 39$$

nicht optimal. Mit Startknoten 4

$$R' = (4,2,1,3,4), \quad K(R') = 35,$$

optimal.

Aber: Greedy geht es zwingend in die Irre:



greedy:

(2, 3, 4, 1, 2) Kosten = 105

(3, 4, 1, 2, 3) "

(4, 3, 2, 1, 4) "

(1, 4, 3, 2, 1) "

Immer ist $1 \xrightarrow[100]{} 2$ dabei. Optimal ist (1, 4, 2, 3, 1) Kosten 14. Hier ist $4 \longrightarrow 2$

nicht greedy, sondern mit Voraussicht (!) gewählt. Tatsächlich für TSP nur Exponentialzeitalgorithmen bekannt.

Weiter führt man backtracking. Wir verzweigen nach dem Vorkommen einer Kante in der Rundreise. Das ist korrekt, denn: Entweder eine optimale Rundreise enthält diese Kante $\overset{v}{\circ} \to \overset{w}{\circ}$ oder dann nicht.

Gehen wir auf unser Beispiel von S. 10.61. Wir hatten die aktuelle Matrix

$$M = \begin{pmatrix} \infty & 10 & 15 & 20 \\ 5 & \infty & 9 & 10 \\ 6 & 13 & \infty & 12 \\ 8 & 8 & 9 & \infty \end{pmatrix}.$$

Wählen wir jetzt zum Beispiel

(10.65)

die Kante $\overset{2}{\circ}\rightarrow\overset{3}{\circ}$, dann

gilt:
- keine Kante $\overset{u}{\circ}\rightarrow\overset{3}{\circ}$, $u \neq 2$

  ~~braucht~~ muß noch betrachtet werden
- keine $\overset{2}{\circ}\rightarrow\overset{v}{\circ}$ $v \neq 3$
- Nicht $\overset{3}{\circ}\rightarrow\overset{2}{\circ}$.

D.h.: ~~dass~~ suchen eine optimale Reise in

$$M' = \begin{pmatrix} \infty & 10 & \infty & 20 \\ \infty & \infty & 9 & \infty \\ 6 & \infty & \infty & 12 \\ 8 & 8 & \infty & \infty \end{pmatrix}$$

Alle nicht mehr benötigten Kanten stehen auf ∞.

Ist dagegen $\overset{2}{\circ}\rightarrow\overset{3}{\circ}$ nicht gewählt, dann

$$M' = \begin{pmatrix} \infty & 10 & 15 & 20 \\ 5 & \infty & \infty & 10 \\ 6 & 13 & \infty & 12 \\ 8 & 8 & 9 & \infty \end{pmatrix}.$$

Backtracking für unser Beispiel

M von S. 10.61

$(2,3)$ — Nicht $(2,3)$

M' von oben — M'' von oben ...

$(1,4)$ — Nicht $(1,4)$

$$\begin{pmatrix} \infty & \infty & \infty & 20 \\ \infty & \infty & 9 & \infty \\ 6 & \infty & \infty & \infty \\ \infty & 8 & \infty & \infty \end{pmatrix} \qquad \begin{pmatrix} \infty & 10 & \infty & \infty \\ \infty & \infty & 9 & \infty \\ 6 & \infty & \infty & 12 \\ 8 & 8 & \infty & \infty \end{pmatrix}$$

Rundreise gefunden. Kosten 43.

Rundreise $< \infty \Leftrightarrow$ Pro Zeile und Spalte genau ein Wert $< \infty$.

$(3,2)$ — Nicht $(3,2)$

...

$$\begin{pmatrix} \infty & 10 & \infty & \infty \\ \infty & \infty & \infty & \infty \\ 6 & \infty & \infty & 12 \\ 8 & 8 & \infty & \infty \end{pmatrix}$$

4

keine Rundreise $< \infty$ mehr möglich.

(10.67)

Algorithmus (Backtracking für TSP)

Eingabe: $M = (M(u,v))_{1 \le u,v \le n}$

Ausgabe: Eine optimale Rundreise dargestellt als Modifikation von M.

TSP(M) // Bei Aufruf $O(n^2)$ Zeit.
// M wird übergeben.
// (wirklich nötig?)

1. if M stellt Rundreise dar
   return (M, Kosten von M)

2. if M hat keine Rundreise $< \infty$
   return (M, ∞)
   // Etwa eine Zeile
   // voller ∞, eine Spalte
   // voller ∞.

3. Wähle (u,v), $u \neq v$ mit $M(u,v) < \infty$,
   wobei Zeile von u oder Spalte von v mindestens einen Wert $\neq \infty$.

4. $M' := M$ modifiziert, so daß $\overset{u}{\circ}\!\!\longrightarrow\!\!\overset{v}{\circ}$ gewählt // Vgl. S. 10.65

5. $M'' := M$ modifiziert, daß $M(u,v) = \infty$.

6. Führe TSP (M') aus;

7. Führe TSP (M'') aus.

7. Vergleiche die Kosten;
~~liefere~~ (M, k), wobei ~~M~~ die Entscheidung der kleineren Kosten und k diese Kosten sind.

Korrektheit: Induktion über die # Einträge in $M = \infty$ (nimmt jedesmal zu.) Zeit, zunächst $O\left(2^{n(n-1)}\right)$, $2^{n(n-1)} = 2^{\Theta(n^2)}$ >>m!!

Verbesserung des backtracking durch branch-and-bound.

Dazu: Für Knoten M (= Knoten im Baum):

$S(M)$ = untere(!) Schranke an die Kosten aller Rundreisen, die sich unter M ergeben,

d.h. $S(M) \le$ Kosten aller Rundreisen unter M im Baum.

Prinzip: Haben wir eine Rundreise der Kosten K gefunden, dann keine Auswertung (Expansion) der Knoten M mit $S(M) \geq K$ mehr.

Allgemein ist es immer so:

$S(k)$ untere Schranke

bei Minimierungsproblemen

$S(k)$ obere Schranke

bei Maximierungsproblemen,

↑

Etwa Max-k-Sat

wobei k Knoten im backtracking Baum.

Wir geben einige mögliche $S(k)$'s

~~Es folgen einige konkrete~~ $S(M)$'s

bei unseren backtrackingproblemen an:

~~für unser TSP-backtracking:~~

1) $S_1(M)$ = minimale Kosten einer Rundreise unter M.

(Das ist die beste (d.h. größte) Schranke, aber nicht polynomial zu ermitteln.)

(10.71)

2) $f_2(M)$ = Summe der Kosten der bei M gewählten Kanten.

Ist untere Schranke bei $M(u,v) \geq 0$. (Das können wir aber hier annehmen, ein Unterschied zu den kürzesten Wegen (warum?))

Ist M = Wurzel des backtracking Baums, dann im allgemeinen $f_2(M) = 0$, (sofern nicht) $M = \begin{pmatrix} \infty\ \infty\ \infty\ \dots\ \infty\ \infty \end{pmatrix}$ oder analog für Spalte.

3) $f_3(M) = \frac{1}{2} \cdot \sum_{v} \min \{ M(u,v) + M(v,w) \mid u,w \in V \}$,

wobei M = Matrix beim Knoten k ist.

Jedesmal das Minimum, um durch den Knoten v zu gehen: $u \longrightarrow v \longrightarrow w$ lokales Minimum, um durch einen Knoten v zu gehen.

10.142

Wieso ist $\hat{f}_3(M)$ untere Schranke?

Sei R eine Rundreise unter k.

Dann gilt

$$k(R) \overset{R=(1,v_1,\dots,v_{m-1},1)}{=} M(1,v_1)+M(v_1,v_2)+\dots+M(v_{n-1},1)$$

$$= \frac{1}{2}\big(M(1,v_1)+M(1,v_1)+M(v_1,v_2)+M(v_1,v_2) \dots + M(v_{n-1},1)+M(v_{n-1},1)\big)$$

schiebe um 1 nach rechts.

$$= \frac{1}{2}\cdot\big(M(v_{n-1},1)+M(1,v_1) + M(1,v_1)+M(v_1,v_2) + M(v_1,v_2)+M(v_2,v_3) + \dots + M(v_{n-2},v_{n-1})+M(v_{n-1},1)\big)$$

(15.78)

$$\geq \frac{1}{2}\cdot\sum_v \min\{M(u,v)+M(v,w) \mid u,w\in V\}$$

Also $\xi_3(M)$ korrekte Schranke.

Es ist $S_3(M)=\infty \Leftrightarrow$ Eine Zeile oder Spalte voller ∞.

(Ebenso für $S_2(M)$).

Betrachten wir M von S. 15.61 und 15.61

Wir ermitteln $\xi_2(M)$.

Kante 1 zu $5+10$ $\left(\overset{2}{\circ}\rightarrow\overset{1}{\circ}\rightarrow\overset{2}{\circ}\right.$, nicht zulässig, macht aber nichts$\left.\right)$

$v=2$ $8+5$ $\left(\overset{4}{\circ}\rightarrow\overset{2}{\circ}\rightarrow\overset{1}{\circ}\right)$

$v=3$ $9+6$ $\left(\overset{2}{\circ}\rightarrow\overset{3}{\circ}\rightarrow\overset{1}{\circ}\right.$ oder $\left.\overset{4}{\circ}\rightarrow\overset{3}{\circ}\rightarrow\overset{1}{\circ}\right)$

$v=4$ $10+8$ $\left(\overset{2}{\circ}\rightarrow\overset{4}{\circ}\rightarrow\overset{1}{\circ}\right.$ oder $\left.\overset{2}{\circ}\rightarrow\overset{4}{\circ}\rightarrow\overset{2}{\circ}\right)$

Also $S_3(M)=\frac{1}{2}\cdot 61=30{,}5$.

Wegen ganzzahligen Kosten folgt (10.84)
$f_3^1(M) \geq$ nächstgrößere ganzzahlige Kosten

≥ 31 für jede Reise R.

Ist $\overset{2}{\circ} \to \overset{3}{\circ}$ kein Höchstgewichtskante,
dann

$$M' = \begin{array}{cc} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} & \\ \begin{pmatrix} \infty & 10 & \infty & 20 \\ \infty & \infty & 9 & \infty \\ 6 & \infty & \infty & 12 \\ 8 & 8 & \infty & \infty \end{pmatrix} & \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} \end{array}$$

für $v = 1$ ist $\geq 6 + 10$ $\left(\overset{3}{\circ} \to \overset{1}{\circ} \to \overset{2}{\circ}\right)$

$v = 2$ $\quad 9 \geq 8 + 9$ $\left(\overset{4}{\circ} \to \overset{2}{\circ} \to \overset{3}{\circ}\right)$

$v = 3$ $\quad 3 \geq 9 + 6$ $\left(\overset{2}{\circ} \to \overset{3}{\circ} \to \overset{1}{\circ}\right)$

$v = 4$ $\quad 4 \geq 12 + 8$ $\left(\overset{3}{\circ} \to \overset{4}{\circ} \to \overset{2}{\circ}\right.$
oder
$\left.\overset{3}{\circ} \to \overset{4}{\circ} \to \overset{1}{\circ}\right)$

Dann für $f_3^1(M')$ (i.e. für ... 3))

$f_3(M') = \frac{1}{2}(16 + 17 + 15 + 20) = 34$,

(15.7.6)

vorgegeben $f_2(M') = 9$. Immer

wird gezeigt $f_1(M) \le f_2(M)$

4) Betrachten wir eine quadratische Matrix

$M = (M(u,v))_{1 \le u,v \le m}$. Alle Einträge ≥ 0.

im Raum alle Einträge ≥ 0.

Ist kostenminimaler Durchschnitt von M bzgl. ...

... die Kosten

von R als

Alle $M(u,v) \ge 0$

$k(R) = z_1 + z_2 + \dots + z_m$,

wobei z_i geeigneter Werte aus Zeile i,

alternativ

$k(R) = s_1 + s_2 + \dots + s_m$,

wobei s_i = geeigneter Wert aus ~~Spalte~~ i,

Jeweils ein Wert pro Zeile und Spalte

10.16

Dann gilt wieder

$$f_4'(M) = \min\{\overbrace{M(1,1), M(1,2), \dots, M(1,n)}^{\text{Zeile 1}}\}$$
$$+ \min\{M(2,1), M(2,2), \dots, M(2,n)\}$$
$$\vdots$$
$$+ \min\{M(n,1), M(n,2), \dots, M(n,n)\}$$

nimmt aus jeder Zeile den kleinsten Wert. Ist korrekte Schranke. Wenn jetzt noch nicht alle Spalten vorkommen können wir noch besser werden. Und zwar so: Sei also für $1 \leq u \leq n$

$$M_u = \min\{M(u,1), M(u,2), \dots, M(u,n)\}$$
$$= \text{kleinster Wert in Zeile } u.$$

Wir setzen

$$\hat{M} = \begin{pmatrix} M(1,1)-M_1 & \dots & M(1,n)-M_1 \\ M(2,1)-M_2 & & M(2,n)-M_2 \\ & \vdots & \\ M(n,1)-M_n & \dots & M(n,n)-M_n \end{pmatrix}$$

Alles ≥ 0. Pro Zeile ein Eintrag $= 0$.

(10.57)

Es gilt für jede Rundreise R bzgl. M

$$K_M(R) = K_{\hat{M}}(R) + \mathcal{S}_4'(M) \geq \mathcal{S}_4'(M).$$

K_M: Kosten in M; $K_{\hat{M}}$: Kosten in $\hat{M}$; $\mathcal{S}_4'(M)$: zu $\hat{M}$ alles ≥ 0.

Ist in $\hat{M}$ in jeder Spalte eine 0

so gibt uns $\mathcal{S}_4'(M)$ = die minimalen

Kosten einer Rundreise. Ist

das nicht der Fall, verweisen wir

den Schritt mit den Spalten von $\hat{M}_u$.

Ist also

$$s_u = \min\{\overbrace{\hat{M}(1,u), \hat{M}(2,u), \ldots, \hat{M}(n,u)}^{\text{Spalte } u \text{ von } \hat{M}}\},$$

dann

$$\hat{\hat{M}} := \begin{pmatrix} \hat{M}(1,1)-s_1 & & \hat{M}(1,n)-s_n \\ \hat{M}(2,1)-s_1 & \cdots & \hat{M}(2,n)-s_n \\ \vdots & & \vdots \\ \hat{M}(n,1)-s_1 & & \hat{M}(n,n)-s_n \end{pmatrix}$$

Alles ≥ 0. Pro Spalte eine 0, pro Zeile eine 0.

(0.78)

Für jede Rundreise R durch $\hat{M}$ gilt

$$k_{\hat{M}}(R) \geq k_{\hat{M}}(R) + \delta_1 + \dots + \delta_m$$

Also haben wir insgesamt:

$$k_M(R) = k_{\hat{M}}(R) + \underbrace{M_1 + \dots + M_m}_{\text{aus } \hat{M}}$$

$$= \overbrace{k_{\hat{M}}(R) + \delta_1 + \dots + \delta_m}$$

im $\hat{M}$ alles ≥ 0

$$\overbrace{} + \underbrace{M_1 + \dots + M_m}_{\text{aus } M}$$

$$\geq M_1 + \dots + M_m + \delta_1 + \dots + \delta_m .$$

Wir definieren offiziell:

$$S_4(M) = M_1 + \dots + M_m + \delta_1 + \dots + \delta_m .$$

$$\mathrm{E} \dots S_4(M) \geq S_3(M) \dots$$

Aufgabe: $S_1(M) \geq S_4(M) \geq S_3(M) \geq S_2(M)$.

($S_4(M) \geq S_3(M)$ ist nicht ganz so einfach offensichtlich.)

Machen Sie diese!

(10.79)

In unserem Eingangsbeispiel von S.1061

$$M = \begin{pmatrix} \infty & 10 & 15 & 20 \\ 5 & \infty & 9 & 10 \\ 6 & 13 & \infty & 12 \\ 8 & 8 & 9 & \infty \end{pmatrix} \quad \begin{matrix} m_1 = 10 \\ m_2 = 5 \\ m_3 = 6 \\ m_4 = 8 \end{matrix}$$

$$\tilde{M} = \begin{pmatrix} \infty & 0 & 5 & 10 \\ 0 & \infty & 4 & 5 \\ 0 & 7 & \infty & 6 \\ 0 & 0 & 1 & \infty \end{pmatrix}$$

$s_1 = 0 \quad s_2 = 0 \quad s_3 = 1 \quad s_4 = 5$

$$\hat{\tilde{M}} = \begin{pmatrix} \infty & 0 & 4 & 10 \\ 0 & \infty & 3 & 0 \\ 0 & 7 & \infty & 1 \\ 0 & 0 & 0 & \infty \end{pmatrix}$$

Also $S_4(M) = 35 \geq S_3(M) = 31$.

Die Schranken lassen sich für jede Durchlaufreihenfolge des backtracking Baums zum Rausschneiden weiterer Teile verwenden.

(10.80)

Algorithmus (Offizielles branch-and-bound)

Breitensuche (Wurzel) Aufspaltung

Front des aktuellen Teils

Raum modifiziert

das backtracking Baums

im Heap geordnet nach $S(M)$

($S(M)$ = die verwendete Schranke).

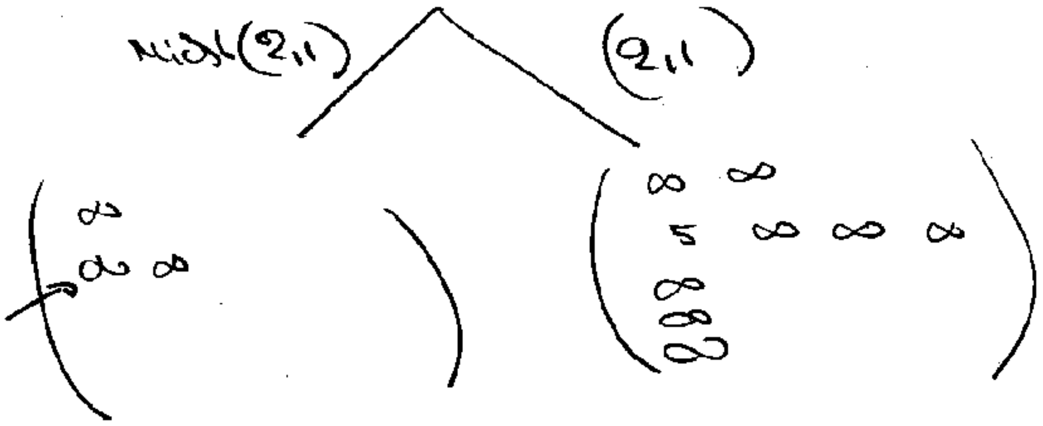
Immer an dem M mit $S(M)$ minimal weitermachen, Minimale gefundene Rundreise vermerken.

Anhalten, wenn $f(M) \geq$ Kosten der minimalen Reise für M $S(M)$ minimal im Heap. □

Offizielles branch-and-bound
mit Beispiel von § 10.61 und Schranke S_4:

$$\begin{pmatrix} \infty & 10 & 15 & 20 \\ 5 & \infty & 9 & 10 \\ 6 & 13 & \infty & 12 \\ 8 & 8 & 9 & \infty \end{pmatrix}$$

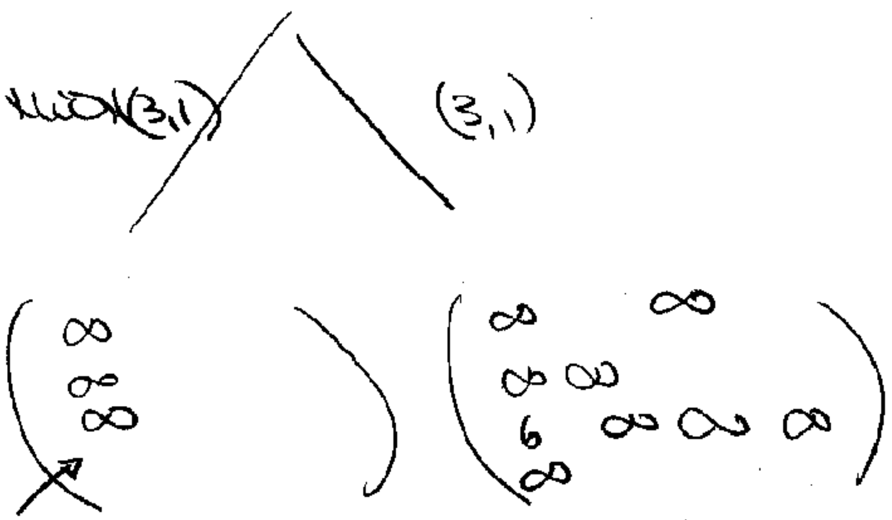
Schranke = 35



Schranke 34

Schranke = 40

Nehmen weiter 35!



Schranke 39

Schranke 34

⋮ Unten weiter

⋮

$$\begin{pmatrix} \infty & & & \\ \infty & \infty & & \\ 6 & \infty & \infty & \infty \\ \infty & & & \end{pmatrix}$$

nicht (2,1) oder (3,1)

Schranke = 34
Wähle 35

nicht (4,2) | (4,2)

$$\begin{pmatrix} \infty & 10 & \infty & 20 \\ \infty & \infty & 9 & 10 \\ 6 & \infty & \infty & \infty \\ \infty & \infty & 9 & \infty \end{pmatrix} \qquad \begin{pmatrix} \infty & \infty & \infty & 20 \\ \infty & \infty & 9 & 10 \\ 6 & \infty & \infty & \infty \\ \infty & 8 & \infty & \infty \end{pmatrix}$$

Schranke = 35
$(10+9+6+9+1 = 35)$

Schranke = 43
$(20+9+6+8 = 43)$

nicht 2,4 | (2,4)

$$\begin{pmatrix} \infty & 10 & \infty & 20 \\ \infty & \infty & 9 & \infty \\ 6 & \infty & \infty & \infty \\ \infty & \infty & 9 & \infty \end{pmatrix} \qquad \begin{pmatrix} \infty & 10 & \infty & \infty \\ \infty & \infty & \infty & 10 \\ 6 & \infty & \infty & \infty \\ \infty & \infty & 9 & \infty \end{pmatrix}$$

Schranke = 45 | Schranke = 35

~~Datenebene~~

Lösung nicht möglich, wg. Schranke.

Aufgabe: Zeigen bei festem Abstand

Menge M, M' im Baum, $d(B)$ (gilt so nicht für f_4!)

$f_2(M') \geq f_2(M)$, $S_3(M') \geq S_3(M)$.