Strukturelle Modelle in der Bildverarbeitung
Submodulare MinSum Probleme

D. Schlesinger – TUD/INF/KI/IS

– Submodulare MinSum Probleme

– Submodulare OrAnd Probleme, Relaxation Labelling

– Binäre submodulare MinSum Probleme

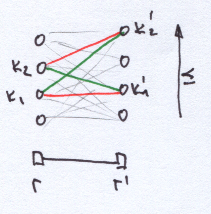
– MinSum Probleme $\rightarrow$ MinCut

Submodulare MinSum Probleme

Zur Erinnerung:

- Im allgemeinen Fall sind MinSum Probleme NP-vollständig.
- Alle Labelling Probleme sind für partielle k-Bäume polynomiell lösbar.

Submodulare Aufgaben: der Graph ist nicht eingeschränkt,
die paarweisen Funktionen $g_{rr'}$ besitzen aber eine bestimmte Eigenschaft:



Sei die Menge der Label K vollständig geordnet,
eine transitive, reflexive, antisymmetrische Relation „$\preceq$“,
im Weiteren sei $K = \{1, 2, \ldots |K|\}$,

sei $k_1 \preceq k_2$ in einem Knoten r
und $k_1' \preceq k_2'$ in einem anderen Knoten r'.

Die Funktion $g_{rr'}$ heißt submodular, wenn

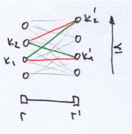
$$g(k_1, k_1') + g(k_2, k_2') \leq g(k_1, k_2') + g(k_2, k_1')$$

für alle derartige Viertupel k_1, k_2, k_1', k_2' gilt.

Eine MinSum Aufgabe (zweiter Ordnung) heißt submodular,
wenn alle Funktionen $g_{rr'}$ submodular sind.

Die Funktionen q_r sind dabei beliebig.

Submodulare OrAnd Probleme



Sehr ähnlich (bis auf Operationen):

$$g(k_1, k_2') \wedge g(k_2, k_1') \Rightarrow g(k_1, k_1') \wedge g(k_2, k_2')$$

Interessant:
Relaxation Labelling liefert für submodulare OrAnd Probleme exakte Lösung.

Zur Erinnerung:
nach dem Anhalten des Algorithmus sind alle q_r und $g_{rr'}$ **lokal konsistent**.

Beweis:

- Sei k_1 das „niedrigste" nicht weggestrichene Label im r,
  ähnlich, sei k_1' das „niedrigste" nicht weggestrichene Label im r'.
- Da das Label k_1 überlebte, gibt es ein Label k_2' im r' so, dass $g_{rr'}(k_1, k_2') = 1$,
  ähnlich, gibt es einen Partner k_2 im r für k_1', d.h. $g_{rr'}(k_2, k_1') = 1$.
- Aus der Submodularität folgt, dass $g_{rr'}(k_1, k_1') = 1$ gilt,
  d.h. so gewählten Labels sind mit einander **global konsistent**.

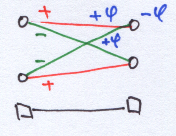
$\Rightarrow$ In der Situation, wenn in allen Knoten evtl. mehr als zwei Labels nicht weggestrichen sind, wähle man in jedem Knoten jeweils das „niedrigste" (oder jeweils das „höchste") Label. Das so ermittelte Labelling ist global konsistent.

Man betrachte für eine Kante (r, r') und Viertupel k_1, k_2, k'_1, k'_2 die Zahl

$$\alpha_{rr'}(k_1, k_2, k'_1, k'_2) = g_{rr'}(k_1, k'_1) + g_{rr'}(k_2, k'_2) - g_{rr'}(k_1, k'_2) - g_{rr'}(k_2, k'_1)$$

Dementsprechend ist eine Aufgabe submodular, gdw. alle $\alpha \leq 0$.

Wichtig: Die α sind invariant zur äquivalenten Transformationen – äquivalente Transformationen ändern die Submodularität nicht.



Beweis: nach der Anwendung einer äquivalenten Transformation Φ:

$$\begin{aligned}
&\alpha^{new}_{rr'}(k_1, k_2, k'_1, k'_2) = \\
&\quad = g^{new}_{rr'}(k_1, k'_1) + g^{new}_{rr'}(k_2, k'_2) - g^{new}_{rr'}(k_1, k'_2) - g^{new}_{rr'}(k_2, k'_1) = \\
&\quad = g^{old}_{rr'}(k_1, k'_1) + \phi_{rr'}(k_1) + \phi_{r'r}(k'_1) + \\
&\quad + g^{old}_{rr'}(k_2, k'_2) + \phi_{rr'}(k_2) + \phi_{r'r}(k'_2) - \\
&\quad - g^{old}_{rr'}(k_1, k'_2) - \phi_{rr'}(k_1) - \phi_{r'r}(k'_2) - \\
&\quad - g^{old}_{rr'}(k_2, k'_1) - \phi_{rr'}(k_2) - \phi_{r'r}(k'_1) = \\
&\quad = g^{old}_{rr'}(k_1, k'_1) + g^{old}_{rr'}(k_2, k'_2) - g^{old}_{rr'}(k_1, k'_2) - g^{old}_{rr'}(k_2, k'_1) = \\
&\alpha^{old}_{rr'}(k_1, k_2, k'_1, k'_2) =
\end{aligned}$$

Binäre submodulare MinSum Probleme

Gegeben sei ein binäres submodulares MinSum Problem, d.h.

$$\alpha_{rr'} = g_{rr'}(0,0) + g_{rr'}(1,1) - g_{rr'}(0,1) - g_{rr'}(1,0) \leq 0$$

(es gibt nur ein α pro Kante).

Wird es in die kanonische Form überführt, d.h.

$$E(y) = \ldots + \sum_{rr'} \beta_{rr'} \cdot \mathbb{I}(y_r \neq y_{r'}),$$

so gilt

$$\begin{aligned}
\alpha_{rr'} &= 0 + 0 - \beta_{rr'} - \beta_{rr'} \\
\beta_{rr'} &= -\alpha_{rr'}/2 = -\Big(g_{rr'}(0,0) + g_{rr'}(1,1) - g_{rr'}(0,1) - g_{rr'}(1,0)\Big)/2 \geq 0
\end{aligned}$$

$\Rightarrow$ Wird diese Aufgabe in das entsprechende MinCut Problem weiter überführt, so sind alle Kantenkosten des MinCut nichtnegativ

$\Rightarrow$ durch MaxFlow polynomiell lösbar.

Kette:

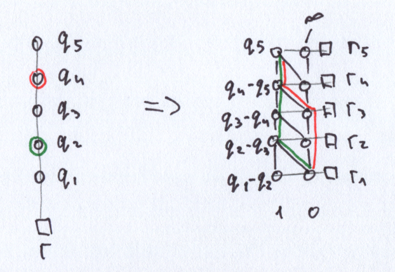
MinSum Problem → **Binäres MinSum Problem** → MinCut (→ MaxFlow)

Die Überführung „K-MinSum → 2-MinSum“:

– Für jedes Label der ursprünglichen Aufgabe wird eine binäre Variable eingeführt (einen Knoten im Graphen der neuen Aufgabe).
– Die neuen Knoten werden mit „speziellen“ Kanten so verbunden, dass eine eineindeutige Relation zwischen den Labellings der neuen und den Labellings der alten Aufgabe entsteht.

  ⇒ alle Labellings werden mit binären Variablen neu kodiert.
– Die Funktionen q und g der neuen Aufgabe werden so definiert, dass auch die Energien der entsprechenden Labellings gleich sind.

Weiterhin gilt: die entsprechenden Aufgaben (2-MinSum und K-MinSum) sind entweder beide submodular oder beide nichtsubmodular.

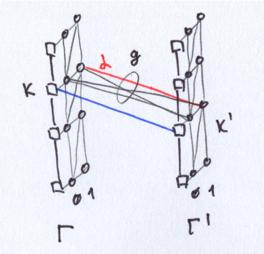


- Jedem Label k wird eine binäre Variable r_k zugeordnet.
- Die in der Ordnung benachbarten Variablen werden mit den Kanten verbunden, die Funktionen g sind „Z-Prädikate“ $g_{r_k,r_{k+1}}(1,0) = \infty$, sonst 0.
  Diese Funktion g ist **submodular**!!!
- Jedem Label k entspricht ein Labelling $(0 \ldots 0, 1 \ldots 1)$, d.h. $y_{r'_k} = 0$ für $k' < k$, alle anderen Labellings haben unendlich große Energie.
- Funktionen q sind:

$$
\begin{aligned}
&q_{r_k}(0) = 0, \quad q_{r_k}(1) = q(k) - q(k+1) \quad \text{für} \quad k < K, \\
&q_{r_K}(0) = \infty, \quad q_{r_K}(1) = q(K).
\end{aligned}
$$

K-MinSum → 2-MinSum, g

Man betrachte zwei Knoten r und r', die in der ursprünglichen Aufgabe benachbart sind.



Die neuen Kanten verbinden
jede eingeführte binäre Variable in r mit jeder in r'

Die neuen Funktionen g dafür sind:

$$g_{r_k, r_{k'}}(1,1) = \alpha_{rr'}(k, k{+}1, k', k'{+}1), \quad 0 \ \text{ sonst}$$

(+ unäre Terme, die in q einfließen).

Es lässt sich zeigen, dass die „g-Anteile“ der Energien in beiden Aufgaben gleich sind.

Ist die ursprüngliche Aufgabe submodular, so sind alle α-s nichtpositiv
$\Rightarrow$ die so konstruierte binäre Aufgabe ist auch submodular.

Weiter in die Kanonische Form → MinCut → MaxFlow – polynomiell lösbar.