13. Approximation algorithms for \((\text{Min}_+,+)\)-problems

We still consider the task

\[
S^* \in \arg \min_{S \in K} U(S) = \arg \min_{S \in K} \left[ \sum_{i \in V} u_i(S_i) + \sum_{ij \in E} u_{ij}(S_i, S_j) \right]
\]

A. Heuristic descent

- define a family of neighbourhoods \(N_m(S) \subseteq K, m=1, \ldots, M\)
- repeatedly solve the restricted problem

\[
S^{(m)} = \arg \min_{S \in N_m(S)} U(S)
\]

until no further improvement is possible, i.e.

\[
S^{(t)} = \arg \min_{S \in N_m(S^{(t-1)})} U(S) + m = 1, \ldots, M
\]

\(\alpha\)-Expansions (Beykov et al. 2001)

For each label \(\alpha \in K\) define the neighbourhood

\[
N_\alpha(S) = \{ S' \in K \mid s_i = \alpha \text{ if } s_i + s_i + i \in V \}
\]

Their sizes are exponential, i.e. \(|N_\alpha(S)| \sim 2^{1V}\)

Is the task \(\arg \min_{S \in N_\alpha(S)} U(S)\) solvable in polynomial time?

Yes, if \(u_{ij}(k, k') + u_{ij}(\alpha, \alpha') \leq u_{ij}(k, \alpha') + u_{ij}(\alpha, k')\) holds for \(i, j \in E\) and \(k, k' \in K \setminus \alpha\). This can be seen by constructing a binary valued \((\text{Min}_+,+)\)-problem that is equivalent to the considered restricted optimisation task

\[
V' = \{ i \in V \mid s_i + 0 \}, \quad E' = \{ ij \in E \mid i, j \in V' \}
\]

\(y_i = 0,1\) encodes \(S \in N_\alpha(S')\), i.e.

\[
s_i = S_i \iff y_i = 0 \quad \text{and} \quad s_i = \alpha \iff y_i = 1
\]
The pairwise functions of this equivalent problem are submodular if the condition given above holds.

**Example 1** Consider the Potts model $U_{ij}(k, k') = q_{ij} \cdot (1 - \delta_{kk'})$, $q_{ij} > 0$. It is not submodular if $|k| > 2$, however, it fulfills the above conditions.

**Theorem 1** (w/o proof)

Let $\mathcal{S}$ be a fixpoint of $\alpha$-expansions $+ \alpha \in K$. Then $U(\mathcal{S}) \leq 2C \min_{\mathcal{S} \in K^V} U(\mathcal{S})$, where $C$ is defined by

$$C = \max_{\mathcal{S} \in \mathcal{E}} \frac{\max_{k \in K} \min_{k' \in K} U_{ij}(k, k')}{\min_{k \in K} \min_{k' \in K} U_{ij}(k, k')}.$$  

**αβ Swaps (Boykov et al. 2001)**

Define neighbourhoods $N_{\alpha \beta}$ for each pair of labels $\alpha, \beta \in K$:

$$N_{\alpha \beta}(S') = \left\{ S \in K^V \mid S_i = \begin{cases} S'_i & \text{if } S_i \neq \alpha, \beta \\ \alpha, \beta & \text{otherwise} \end{cases} \right\}$$  

The reduced task argmin $U(\mathcal{S})$ is tractable if the restriction of every $U_{ij} : K^2 \rightarrow \mathbb{R}$ to $\mathcal{L}_{\alpha \beta} \subseteq K^2$ is submodular $+ \alpha, \beta \in K$.

**Example 2** Consider the truncated metric on $K \subseteq \mathbb{Z}$

$$U_{ij}(k, k') = a_{ij} \cdot \min\left(C, |k-k'|\right), \quad a_{ij} > 0.$$  

It is not submodular. It allows $\alpha \beta$-swaps, but does not allow $\alpha$-expansions.
Remark 1: Another class of approximation algorithms construct submodular upper bounds (instead of considering restricted problems). I.e. given \( S^{(t)} \), construct a submodular upper bound \( \tilde{U}_t(S) \) s.t.
\[
\tilde{U}_t(S) \geq U(S) \quad \forall S \in K^v \quad \text{and} \quad \tilde{U}_t(S^{(t)}) = U(S^{(t)}).
\]
Then, solve
\[
S^{(t+1)} \in \arg \max \tilde{U}_t(S) \quad \forall S \in K^v.
\]

3. Algorithms based on LP-relaxations

Loopy belief propagation (aka message passing): Apply equivalent transformations which resemble dynamic programming on trees, until convergence. Not well grounded. See next section.

More principled: Start from an LP-relaxation of the discrete optimization problem
\[
U(S) = \sum_{i \in V} u_i(S_i) + \sum_{(i,j) \in E} U_{ij}(S_i, S_j) \rightarrow \min_{S \in K^v}
\]
A lower bound is given by
\[
\sum_{i \in V} \min_{k \in K} u_i(k) + \sum_{(i,j) \in E} \min_{k,k' \in K} U_{ij}(k,k') \leq \min_{S \in K^v} U(S)
\]
Combine it with equivalent transformations and maximise the lower bound w.r.t. them
\[
B(\Psi) = \sum_{i \in V} \min_{k \in K} \left[ u_i(k) - \sum_{j \in V_i} \Psi_{ij}(k) \right] +
\]
\[
+ \sum_{(i,j) \in E} \min_{k,k' \in K} \left[ \Psi_{ij}(k) + U_{ij}(k,k') + \Psi_{ji}(k') \right] \rightarrow \max_{\Psi}
\]
This can be expressed as a linear optimisation task by introducing additional variables:

$$\sum_{i \in V} C_i + \sum_{ij \in E} C_{ij} \rightarrow \max \quad \psi_{i,C}$$

s.t.  
$$C_i + \sum_{j \in V_i} \psi_{ij}(k) \leq \psi_{i,j_i}(k) \quad \forall i \in V, \quad \forall k \in K$$

$$C_{ij} - \psi_{ij}(k) - \psi_{ij}(k') \leq \psi_{ij}(k,k') \quad \forall j \in E, \quad \forall k,k' \in K$$

Notice that this LP-task is dual to the following direct relaxation of the discrete optimisation task. Encode the label $$s_i \in K$$ by 1-out-of-K encoding with components denoted as $$\lambda_i(k) = 0,1$$, and similarly for edges, by $$\lambda(k,k') = 0,1$$

$$\sum_{i \in V} \sum_{k \in K} \lambda_i(k) k_i(k) + \sum_{i \in E} \sum_{k,k' \in K} \lambda_{ij}(k,k') \psi_{ij}(k,k') \rightarrow \min \quad \lambda > 0$$

s.t.  
$$\lambda_i(k) = \sum_{k' \in K} \lambda_{ij}(k,k') \quad \forall j \in E, \quad \forall k \in K$$

$$\sum_{k \in K} \lambda_i(k) = 1 \quad \forall i \in V$$

$$\sum_{k,k' \in K} \lambda_{ij}(k,k') = 1 \quad \forall i,j \in E$$

Relaxing the integrality constraints $$\lambda_i(k) = 0,1$$, $$\lambda_{ij}(k,k') = 0,1$$ makes this an LP-task.

Find suitable algorithms for solving the primal task, or the dual task, or both simultaneously.