Cost Saturation for Multiple Sequence Alignment

Mika Skjelnes, PhD Student, Linköping University
IDA, Machine Reasoning Lab
Supervisors: Jendrik Seipp (LiU) & Daniel Gnad (LiU)

Motivation & Research Goals

- **Cost Partitioning** enables combining admissible heuristics additively, which is state-of-the-art in automated planning.
- We want to expand its scope by investigating problems outside of planning, such as Multiple Sequence Alignment (MSA).

→ Successfully used Cost Partitioning for MSA, and investigated theoretical relations to earlier approaches.

- Next we want to examine **cost saturation** for MSA, as this turns out to be a non-trivial task unlike in a typical planning setting.

Cost Partitioning for MSA

- Given a collection of sequences $S = \{s_1, s_2, \ldots, s_n\}$, find an optimal alignment. We can formulate this as a cheapest-path problem $\mathcal{T}_S$.
- The cost of a transition is defined by the pairwise substitutions that it defines. We call a unique pairwise substitution a **cost component**.

→ A subproblem $P \subseteq S$ under-approximates the optimal cost of any state of $\mathcal{T}_S$.

S

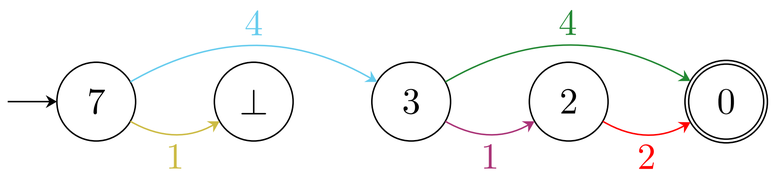
s_1:	T	T	A
s_2:	G	C	
s_3:	A	C	

A

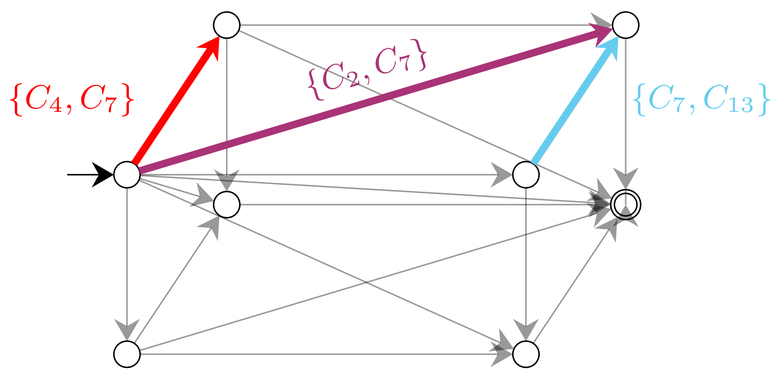
A_1:	T	T	–	A
A_2:	–	G	–	C
A_3:	A	–	C	–

	A	C	T	G	–
A	0	4	2	2	3
C		1	4	3	3
T			0	6	3
G				1	3
–					0

Previously, we showed that the cost partitioning algorithm *post-hoc optimization* dominates the strongest admissible heuristic for MSA, called all-k [1]. The next step is to look into the *saturated* family of cost partitioning algorithms. Cost saturation is trivial in a typical planning context, but for MSA it turns out to be slightly more complicated.



Cost saturation is easy in this transition system because one color make up one label!



In this transition system $\mathcal{T}$, multiple cost components make up a label. How do we do cost saturation here? We can introduce variables $x_1, \ldots, x_n$ where x_i is the remaining cost of cost component C_i. For the red label, we get the linear constraint

$$x_4 + x_7 \leq b_r$$

where b_r is the remaining cost of the red label. In general, for each label of the transition system, we get one linear constraint. The objective is to maximize the total remaining cost.

$$\begin{aligned} \text{Maximize} \quad & c^T x \\ \text{s.t.} \quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$

Cost Saturation LP

ISSUE: The size of the LP grows too aggressively. Even simple MSA instances run out of memory when building the LP.
OBSERVATION: The LP seems to have at most two variables per constraint, for any subproblem of three sequences, and a predictable number of single-variable constraints. **This must be exploitable!**
IDEA: Solve the LP implicitly through *Dantzig-Wolfe decomposition*, declaring the set of two-variable constraints as the *complicating constraints* $Ax \leq b$, and the single-variable constraints as the *easy constraints* $Dx \leq d$.

EXAMPLE: Below is the LP for cost saturation of all cost components in transition system $\mathcal{T}$. The corresponding constraints for the colored labels are highlighted. Constraints of the form $a_i x \leq 0$ and zero-valued variables have been pruned.

$$\begin{aligned} \text{Maximize} \quad & c^T x \\ \text{s.t.} \quad & \left.\begin{aligned} & x_4 + x_7 \leq 10 \\ & x_1 + x_8 \leq 10 \\ & x_1 + x_4 \leq 8 \\ & x_2 + x_5 \leq 8 \\ & x_2 + x_7 \leq 10 \end{aligned}\right\} Ax \leq b \\ & \left.\begin{aligned} & x_1 \leq 8 \\ & x_2 \leq 8 \\ & x_4 \leq 4 \\ & x_5 \leq 4 \\ & x_7 \leq 6 \\ & x_8 \leq 6 \end{aligned}\right\} Dx \leq d \\ & x \geq 0 \end{aligned}$$

Future Work & Questions

- How can we exploit the known structure of our LP? Can the LP be avoided altogether?
- Can we compute optimal cost partitioning reasonably fast in MSA?
- How do we select a promising set of subproblems, assuming that selecting the pool of all subproblems is infeasible?

References

[1] Mika Skjelnes, Daniel Gnad, and Jendrik Seipp. Cost partitioning for multiple sequence alignment. In Ulle Endriss and Francisco S. Melo, editors, *Proceedings of the 27th European Conference on Artificial Intelligence (ECAI 2024)*. IOS Press, 2024.

LINKÖPING UNIVERSITY
WASP WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM