# Cost Partitioning For Multiple Sequence Alignment

Mika Skjelnes, Daniel Gnad, Jendrik Seipp

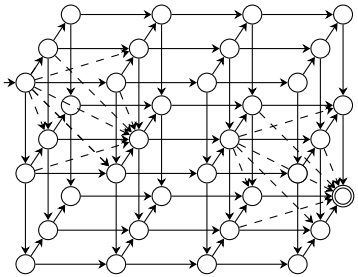
Linköping University

# Multiple Sequence Alignment

Find the cheapest alignment of $n$ sequences $\rightarrow$ non-trivial search problem.

| | $S$ | | |
|---|---|---|---|
| $s_1$: | 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 | – |



# Problem Formulation

Previous research: Solve MSA with A* using **admissible** heuristics.

Best approaches **combine multiple heuristics** from **small subproblems** in a semifixed manner.

We formulate MSA using planning concepts, and produce **stronger and more flexible heuristics** with **cost partitioning**.

# Key Contributions

MSA heuristics from patterns + **cost partitioning**

Stronger heuristics than previous approaches on MSA.

Given $n$ sequences, find a cheapest alignment of the sequences, based on a character substitution cost function.

$S$

| | | | |
|---|---|---|---|
| $s_1$: | T | T | A |
| $s_2$: | G | C | |
| $s_3$: | A | C | |

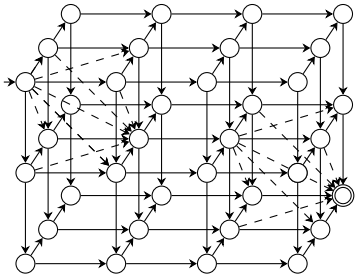
$A$

| | | | | |
|---|---|---|---|---|
| $A_1$: | 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 |

# Naïvely Solving MSA

Why not solve the problem directly, e.g. with dynamic programming?
$\prod_{s_i \in S} |s_i|$ states, and to $2^n - 1$ outgoing transitions per state.
$\Rightarrow$ Grows infeasible



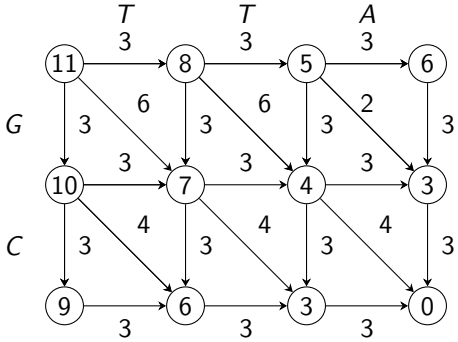
# Solving MSA subproblems

Exploit optimal values from subproblems!

Combine multiple subproblems admissibly?

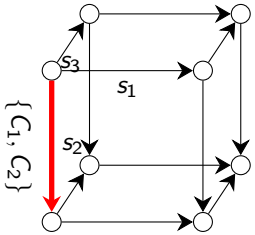
| | $S'$ | | |
|---|---|---|---|
| $s_1$: | T | T | A |
| $s_2$: | G | C | |

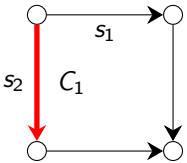
$\rightarrow$

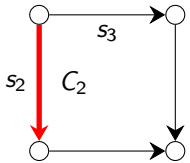


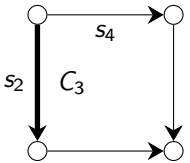
# Cost Components

MSA subproblems are patterns, with a "slight" difference.
Cost Components determine the transition cost.



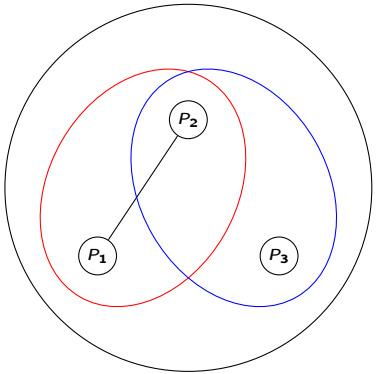



# Admissibility

Two patterns $P_1$ and $P_2$ conflict if they have a cost component in common.
No conflicting patterns in a pattern collection $\mathcal{P} \to h^{\mathcal{P}}$ is admissible.

$$P_1 = (s_1, s_2, s_3), P_2 = (s_2, s_3), P_3 = (s_2, s_4)$$



# Cost Partitioning

Given admissible heuristics $h_1, \dots h_n$ a cost partitioning $c_1, \dots c_n$ distributes the cost $c(o)$ of each operator $o$ among the heuristics so that $\sum_{i=1,\dots,n} c_i(o) \leq c(o)$. This ensures that no more than the full cost of each operator is used when combining the heuristics $\rightarrow$ sum admissible heuristics.

# MSA PhO

$$h_{\text{PhO}}(s) = \text{maximize} \sum_{i=1}^{n} w_i \cdot h_i(s) \text{ s.t.}$$

$$\sum_{h_i \in \mathcal{H}: h_i \text{ uses } \ell} w_i \leq 1 \text{ for all } \ell \in L(\mathcal{T})$$

$$w_i \geq 0 \text{ for all } h_i \in \mathcal{H}.$$

$$h^{\mathcal{P}}_{\text{PhO}}(s) = \text{maximize} \sum_{i=1}^{n} w_i \cdot h^{P_i}(s) \text{ s.t.}$$

$$\sum_{P_i \in \mathcal{P}'} w_i \leq 1 \text{ for all strictly conflicting } \mathcal{P}' \subseteq \mathcal{P}$$

$$w_i \geq 0 \text{ for all } P_i \in \mathcal{P}.$$

## MSA heuristic all,k

all,k combines the heuristic estimate from all patterns of size k, scaling the contribution of each heuristic with $\frac{1}{\binom{n-2}{k-2}}$.

Let $S = \{s_1, s_2, s_3, s_4\}$ and $k = 3$. Then we have

$$h^{\text{all},k} = \frac{1}{\binom{2}{1}} \cdot \left(h^{(s_1,s_2,s_3)} + h^{(s_1,s_2,s_4)} + h^{(s_1,s_3,s_4)} + h^{(s_2,s_3,s_4)}\right)$$

## Dominance

*PhO* dominates $all, k$, and the relation is strict.

$$h^{\text{all},k} = w_1 \cdot (h^{(s_1,s_2,s_3)} + w_2 \cdot h^{(s_1,s_2,s_4)} + w_3 \cdot h^{(s_1,s_3,s_4)} + w_4 \cdot h^{(s_2,s_3,s_4)})$$

where $w_i = \frac{1}{\binom{2}{1}}$.

In general

$$h^{\text{all},k} = \sum_{i=1}^{\binom{n}{k}} w_i \cdot h^{P_i}, \text{ where } w_i = \frac{1}{\binom{n-2}{k-2}}.$$

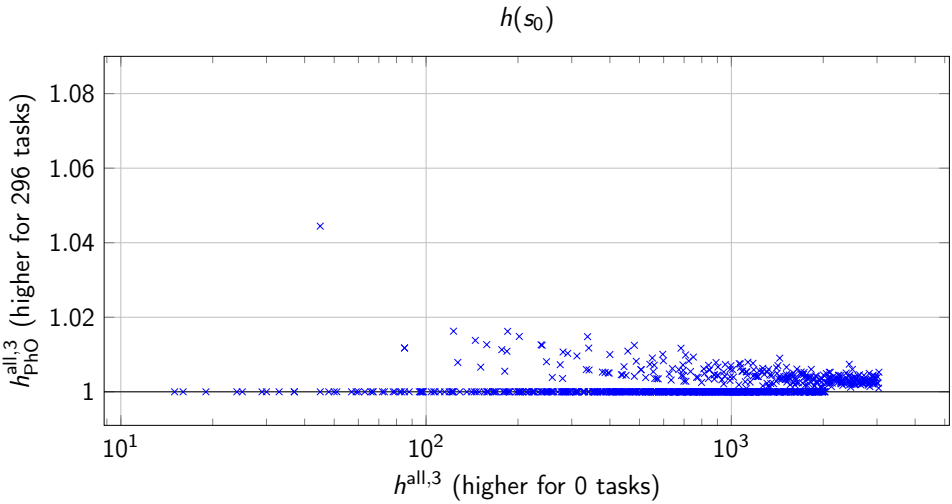
$$\sum_{P_i \in \mathcal{P}'} w_i \leq 1 \text{ for all strictly conflicting } \mathcal{P}' \subseteq \mathcal{P} \text{ holds.}$$

$$w_i \geq 0 \text{ for all } P_i \in \mathcal{P} \text{ holds.}$$

We have a similar analysis with UCP, and show that UCP is equal to all,k. item But like PhO, UCP can be computed for any pattern collection.

$$w_P(C) = \begin{cases} \frac{1}{|\{P \in \mathcal{P} | C \text{ exists in } \mathcal{T}_P\}|} & \text{if } C \text{ exists in } \mathcal{T}_P \\ 0 & \text{otherwise.} \end{cases}$$

# Experiments



# Future Work

SCP, SPHO, OUCP

Investigating MSA pattern selection.

# Key Takeaways

We adapt planning heuristics for MSA

The flexibility of weighted pattern selection, allowed by PhO, indeed yields stronger heuristics. But maybe we can do even better!

There are many more aspects of MSA to investigate further, such as other cost partitioning schemes or pattern selection.