Partitioning into Colorful Components by Minimum Edge Deletions

Sharon Bruckner[1] Falk Hüffner[2]
Christian Komusiewicz[2] Rolf Niedermeier[2] Sven Thiel[3]
Johannes Uhlmann[2]

[1]Institut für Mathematik, Freie Universität Berlin

[2]Institut für Softwaretechnik und Theoretische Informatik, TU Berlin

[3]Institut für Informatik, Friedrich-Schiller-Universität Jena

3 July 2012

International Max Planck Research School for Computational Biology and Scientific Computing

Multiple Sequence Alignment

T_1 A_2 C_3 G_4 T_5 A_6

T_1 A_2 G_3 T_4 A_5

T_1 A_2 C_3 G_4 T_5 G_6 A_7

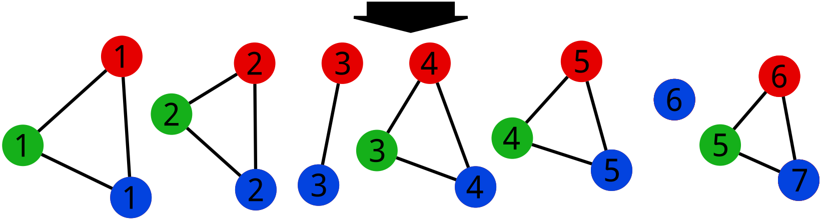
International Max Planck Research School for Computational Biology and Scientific Computing

Multiple Sequence Alignment

T_1	A_2	C_3	G_4	T_5		A_6
T_1	A_2		G_3	T_4		A_5
T_1	A_2	C_3	G_4	T_5	G_6	A_7

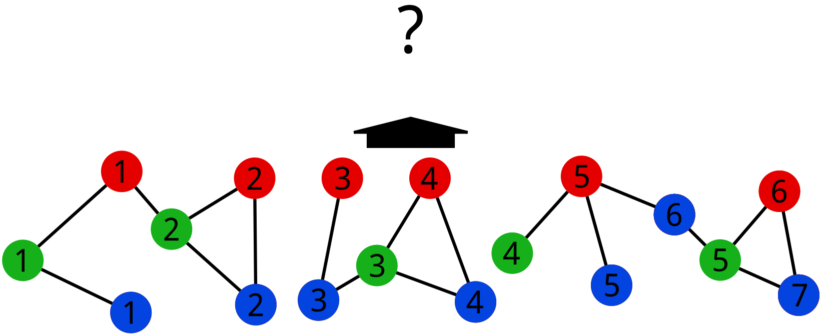
Multiple Sequence Alignment

T_1	A_2	C_3	G_4	T_5		A_6
T_1	A_2		G_3	T_4		A_5
T_1	A_2	C_3	G_4	T_5	G_6	A_7



International
Max Planck Research School
for Computational Biology
and Scientific Computing

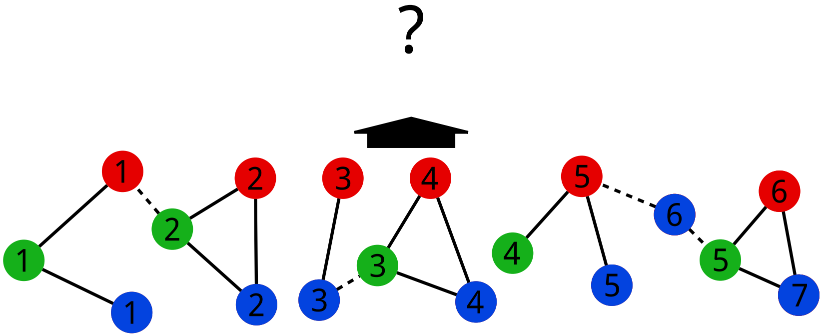
Multiple Sequence Alignment



Idea

Use alignment graph constructed by local alignment to reconstruct global alignment.

Multiple Sequence Alignment

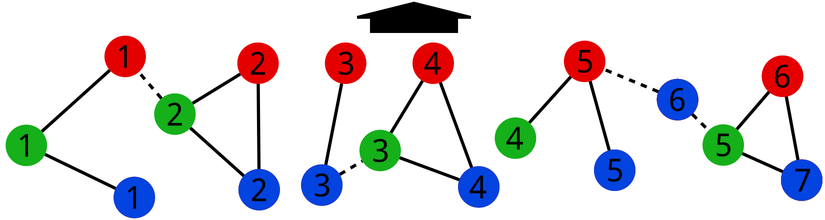


Idea

Use alignment graph constructed by local alignment to reconstruct global alignment.

Multiple Sequence Alignment

T_1	A_2	C_3	G_4	T_5		A_6
T_1	A_2		G_3	T_4		A_5
T_1	A_2	C_3	G_4	T_5	G_6	A_7



Idea

Use alignment graph constructed by local alignment to reconstruct global alignment.

Colorful Components

Part of a Multiple Sequence Alignment pipeline suggested by Corel, Pitschi & Morgenstern (Bioinformatics 2010).

International Max Planck Research School for Computational Biology and Scientific Computing

Colorful Components

Part of a Multiple Sequence Alignment pipeline suggested by Corel, Pitschi & Morgenstern (Bioinformatics 2010).

COLORFUL COMPONENTS

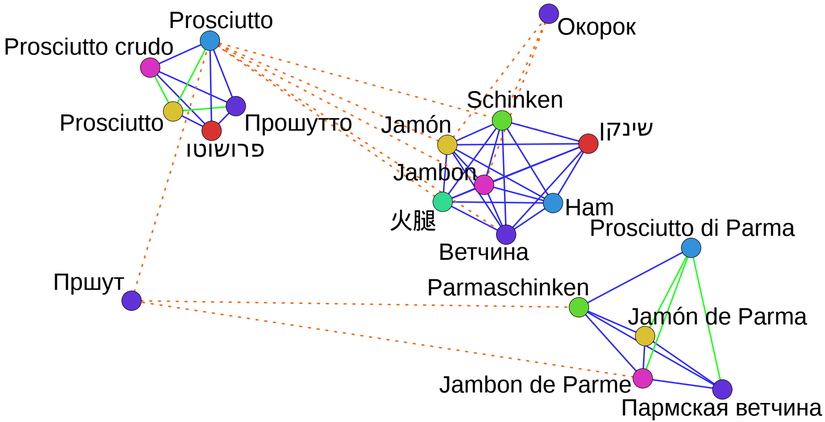
Instance: An undirected graph $G = (V, E)$ and a coloring of the vertices $\chi : V \to \{1, \ldots, c\}$.
Task: Delete a minimum number of edges such that all connected components are *colorful*, that is, they do not contain two vertices of the same color.

International Max Planck Research School for Computational Biology and Scientific Computing

Other application: Wikipedia interlanguage links

Other application: Wikipedia interlanguage links

Wikipedia interlanguage link graph example



International Max Planck Research School for Computational Biology and Scientific Computing

Complexity of Colorful Components

- COLORFUL COMPONENTS with two colors can be solved in $O(\sqrt{n}m)$ time by matching techniques.

Complexity of Colorful Components

- COLORFUL COMPONENTS with two colors can be solved in $O(\sqrt{n}m)$ time by matching techniques.
- COLORFUL COMPONENTS is NP-hard already with three colors.

International Max Planck Research School for Computational Biology and Scientific Computing

Complexity of Colorful Components

- COLORFUL COMPONENTS with two colors can be solved in $O(\sqrt{n}m)$ time by matching techniques.
- COLORFUL COMPONENTS is NP-hard already with three colors.
- COLORFUL COMPONENTS is NP-hard on trees.

International Max Planck Research School for Computational Biology and Scientific Computing

Complexity of Colorful Components

- COLORFUL COMPONENTS with two colors can be solved in $O(\sqrt{n}m)$ time by matching techniques.
- COLORFUL COMPONENTS is NP-hard already with three colors.
- COLORFUL COMPONENTS is NP-hard on trees.
- COLORFUL COMPONENTS on trees with c colors can be solved in $2^c \cdot n^{O(1)}$ time.

International Max Planck Research School for Computational Biology and Scientific Computing

Fixed-parameter algorithm

Observation

COLORFUL COMPONENTS can be seen as the problem of destroying by edge deletions all bad paths, that is, simple paths between equally colored vertices.

Fixed-parameter algorithm

Observation

Colorful Components can be seen as the problem of destroying by edge deletions all bad paths, that is, simple paths between equally colored vertices.

Observation

Unless the graph is already colorful, we can always find a bad path with at most c edges, where c is the number of colors.

International Max Planck Research School for Computational Biology and Scientific Computing

Fixed-parameter algorithm

Observation

COLORFUL COMPONENTS can be seen as the problem of destroying by edge deletions all bad paths, that is, simple paths between equally colored vertices.

Observation

Unless the graph is already colorful, we can always find a bad path with at most c edges, where c is the number of colors.

Theorem

COLORFUL COMPONENTS can be solved in $O(c^k \cdot m)$ time, where k is the number of edge deletions.

Improved fixed-parameter algorithm

Theorem

Colorful Components can be solved in $O((c-1)^k \cdot m)$ time, where k is the number of edge deletions.

Improved fixed-parameter algorithm

Theorem

Colorful Components can be solved in $O((c-1)^k \cdot m)$ time, where k is the number of edge deletions.

Proof.

If there is a degree-3 or higher vertex v, find a bad path with at most $(c-1)$ edges by BFS from v. Otherwise, the instance is easy. □

International Max Planck Research School for Computational Biology and Scientific Computing

Limits of fixed-parameter algorithms

Question

How much further can we improve this algorithm?

Limits of fixed-parameter algorithms

Question

How much further can we improve this algorithm?

Exponential Time Hypothesis (ETH)

For all $x \geqslant 3$, x-SAT, which asks whether a Boolean input formula in conjunctive normal form with n variables and m clauses and at most x variables per clause is satisfiable, cannot be solved within a running time of $2^{o(n)}$ or $2^{o(m)}$.

International Max Planck Research School for Computational Biology and Scientific Computing

Limits of fixed-parameter algorithms

Question

How much further can we improve this algorithm?

Exponential Time Hypothesis (ETH)

For all $x \geqslant 3$, x-SAT, which asks whether a Boolean input formula in conjunctive normal form with n variables and m clauses and at most x variables per clause is satisfiable, cannot be solved within a running time of $2^{o(n)}$ or $2^{o(m)}$.

Theorem

Colorful Components with three colors cannot be solved in $2^{o(k)} \cdot n^{O(1)}$ unless the ETH is false.

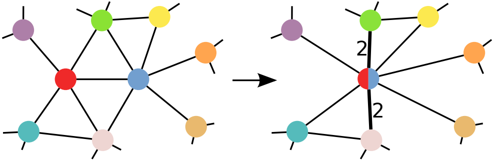
Weighted version

Problem

If we know that two vertices must belong to the same colorful component, we want to be able to simplify the instance by merging them.

Idea

Introduce color *sets* per vertex and edge weights.



International
Max Planck Research School
for Computational Biology
and Scientific Computing

Uses of the merge operation

Edge branching

Can branch into two cases: delete an edge, or merge its endpoints.

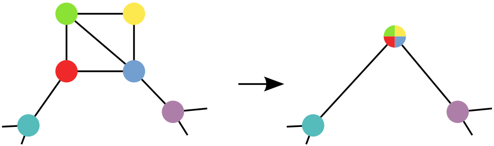
Uses of the merge operation

Edge branching

Can branch into two cases: delete an edge, or merge its endpoints.

Data reduction

Let $V' \subseteq V$ be a colorful subgraph. If the cut between V' and $V \setminus V'$ is at least as large as the connectivity of V', then merge V' into a single vertex.



International Max Planck Research School for Computational Biology and Scientific Computing

Merge-based heuristic

Idea

Repeatedly merge the two vertices "most likely" to be in the same component, while immediately deleting edges connecting vertices with intersecting color sets.

Merge-based heuristic

Idea

Repeatedly merge the two vertices "most likely" to be in the same component, while immediately deleting edges connecting vertices with intersecting color sets.

We always merge the endpoints of the edge that maximizes cut cost minus merge cost.

- *merge cost*: weight of the edges that would need to be deleted when merging
- *cut cost*:

$$3w(\{u,v\}) + \sum_{w \in V | \{\{u,w\},\{v,w\}\} \subseteq E} \min\{w(\{u,w\}), w(\{v,w\})\}$$

International Max Planck Research School for Computational Biology and Scientific Computing

Data

- We generated one COLORFUL COMPONENTS instance for each multiple alignment instance from the BAliBASE 3.0 benchmark.
- We restricted the experiments to the 135 of instances that have at most 10 colors.

Data reduction: Largest connected component

(1) originally
(2) after data reduction in the COLORFUL COMPONENTS formulation
(3) after data reduction in the weighted formulation

	(1)			(2)			(3)		
	n	*m*	*c*	*n*	*m*	*c*	*n*	*m*	*c*
average	504	921	6.2	407	697	4.7	354	607	5.3
median	149	232	6	46	90	5	42	58	5

International Max Planck Research School for Computational Biology and Scientific Computing

Branching algorithms: running time

	< 1 s	1 s to 10 min	> 10 min
bad-path branching	61	6	68
merging branching	70	9	56

International Max Planck Research School for Computational Biology and Scientific Computing

Branching algorithms: running time

	$< 1\,\text{s}$	$1\,\text{s}$ to $10\,\text{min}$	$> 10\,\text{min}$
bad-path branching	61	6	68
merging branching	70	9	56

Note

In ongoing research, we are able to solve several more instances to optimality with integer linear programming (ILP) based approaches.

Heuristics: relative error

	min.	max.	avg.	med.
min-cut heuristic [1]	0 % (1)	70.0 %	29.2 %	27.8 %
merging heuristic	0 % (76)	12.7 %	0.6 %	0 %

[1] Corel, Pitschi & Morgenstern (Bioinformatics 2010)

Sequence alignment quality

DIALIGN with several methods for solving the COLORFUL COMPONENTS subproblem:

	TC score
min-cut heuristic	53.6 %
merge heuristic	55.1 %
exact algorithm	56.6 %

International Max Planck Research School for Computational Biology and Scientific Computing

Sequence alignment quality

DIALIGN with several methods for solving the COLORFUL COMPONENTS subproblem:

	TC score
min-cut heuristic	53.6 %
merge heuristic	55.1 %
exact algorithm	56.6 %

DIALIGN with the min-cut heuristic is about 10 percentage points worse than current state-of-the-art multiple alignment methods. Hence, an improvement of 3 percentage points is a sizable step towards closing the gap between DIALIGN and these methods.

International Max Planck Research School for Computational Biology and Scientific Computing

Outlook

- ILP-based solutions
- Application to network alignment
- Relaxation of the colorfulness constraint

International Max Planck Research School for Computational Biology and Scientific Computing

Acknowledgments

Sharon Bruckner

Freie Universität Berlin

International Max Planck Research School for Computational Biology and Scientific Computing

nanopoly

Falk Hüffner, Christian Komusiewicz, Rolf Niedermeier, Johannes Uhlmann

TU berlin

DFG

Sven Thiel

seit 1558