The MAX-CMO problem and its representation

Improving an efficiency of the heuristic search

Paweł Widera
pawel.widera@cs.nott.ac.uk

SUPERVISORS
Natalio Krasnogor, Jonathan Garibaldi

The University of Nottingham

5th East Midlands Proteomics Workshop

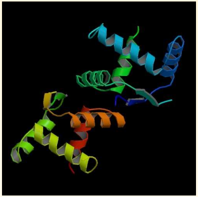
2006-11-15

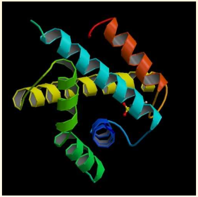
Outline

1. **Problem definition**
   - Biological background
   - Measure of similarity
2. **Dual representation**
3. **Metaheuristic search efficiency**
   - Search method outline
   - Possible improvements
4. **Summary**

Protein comparison

Is it so hard indeed?





Structure comparison

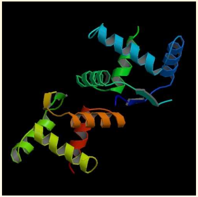
- root mean square distance
- difference of distance matrices
- alignment of contact maps

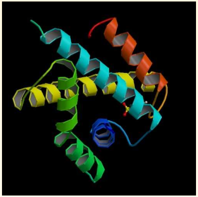
Contact map

Mathematical construct capturing the **proximity relation** between residues.

Protein comparison

Is it so hard indeed?





Structure comparison

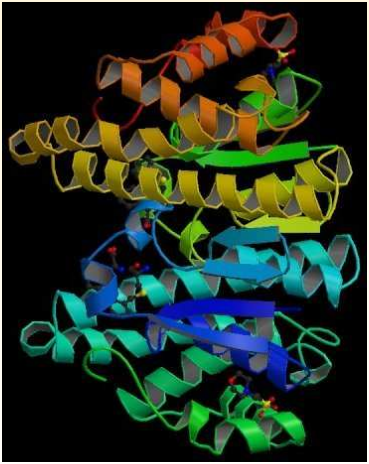
- root mean square distance
- difference of distance matrices
- alignment of contact maps

Contact map

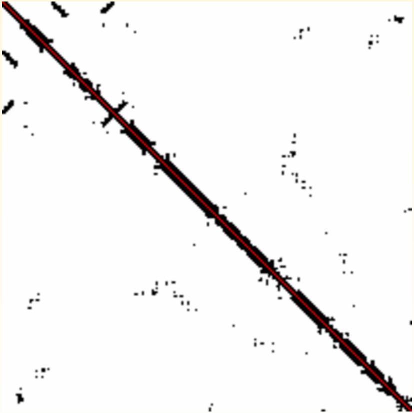
Mathematical construct capturing the **proximity relation** between residues.

Contact map

Proximity relation depicted as 2D binary matrix



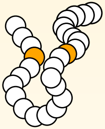
10GS (GLUTATHIONE S-TRANSFERASE P1-1)



CM for 10GS:A (210), $\alpha = 6.5Å$

Contact map

Proximity relation depicted as graph

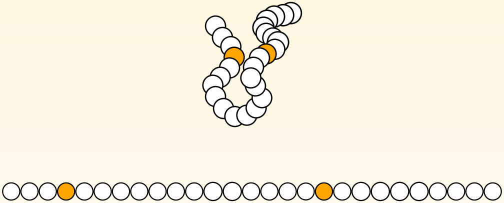


Contact map graph

- $r_i \mathbf{R} r_j \Leftrightarrow \delta(r_i, r_j) \leqslant \alpha$, $\alpha \in [2Å, 9Å]$
- node ⇆ residue
- edge ⇆ contact

Contact map

Proximity relation depicted as graph

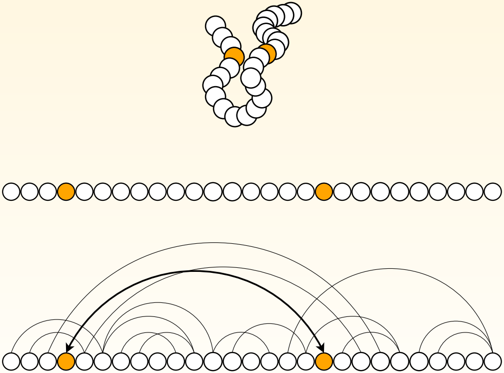


Contact map graph

- $r_i \mathbf{R} r_j \Leftrightarrow \delta(r_i, r_j) \leqslant \alpha$, $\alpha \in [2\mathring{A}, 9\mathring{A}]$
- node ⇆ residue
- edge ⇆ contact

Contact map

Proximity relation depicted as graph

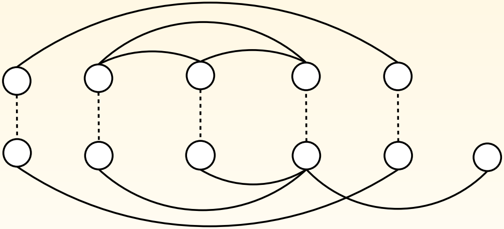


Contact map graph

- $r_i \mathbf{R} r_j \Leftrightarrow \delta(r_i, r_j) \leqslant \alpha$, $\alpha \in [2Å, 9Å]$
- node ⇆ residue
- edge ⇆ contact

MAX-CMO problem

Formal definition

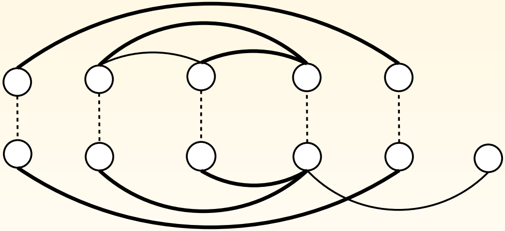


Definition

Maximum contact map overlap is an **alignment** of two proteins that **maximises** the structural **similarity**.

MAX-CMO problem

Formal definition



Definition

Maximum contact map overlap is an **alignment** of two proteins that **maximises** the structural **similarity**.

Graph isomorphism

A bit of graph theory

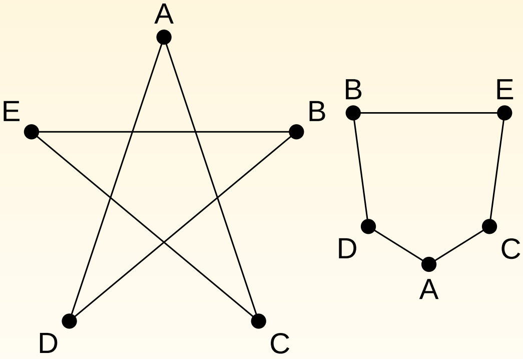


Figure: Isomorphic or not?

Graph isomorphism

Two graphs are isomorphic if there is a one-to-one correspondence between their nodes and there is an edge between two nodes of one graph if and only if there is an edge between the two corresponding nodes in the other graph.

Subgraph isomorphism

Is G1 isomorphic to a subgraph of G2?

Graph isomorphism

A bit of graph theory

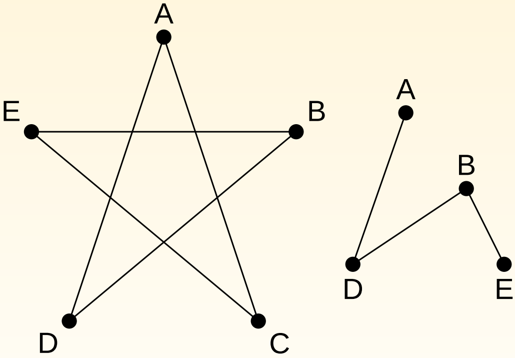


Figure: What about a subgraph?

Graph isomorphism

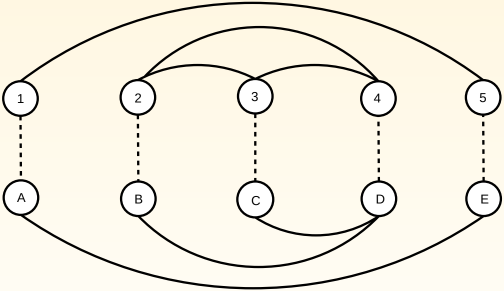
Two graphs are isomorphic if there is a one-to-one correspondence between their nodes and there is an edge between two nodes of one graph if and only if there is an edge between the two corresponding nodes in the other graph.

Subgraph isomorphism

Is G1 isomorphic to a subgraph of G2?

Domain of a problem

Classical approach

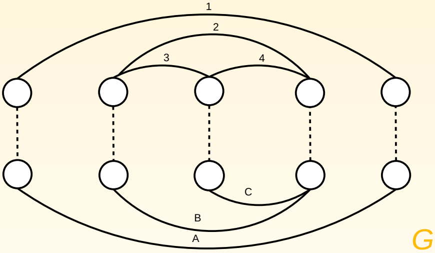


Representation

- protein alignment
- nodes matching

Domain of a solution

Line graph approach



Line graph

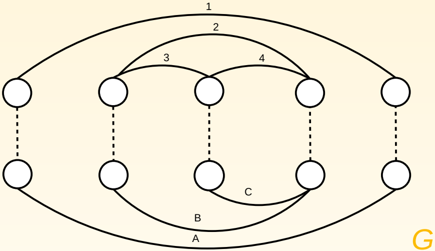
- a node of L(G) represents an edge of G
- two nodes of L(G) are adjacent if edges in G share a common node

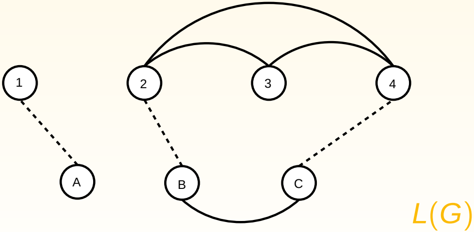
Representation

- graph isomorphism
- edges matching

Domain of a solution

Line graph approach





Line graph

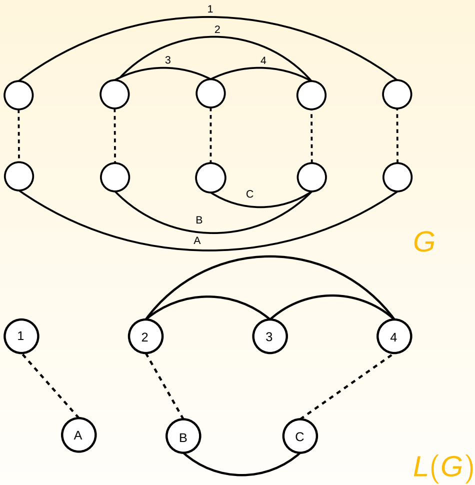
- a node of L(G) represents an edge of G
- two nodes of L(G) are adjacent if edges in G share a common node

Representation

- graph isomorphism
- edges matching

Domain of a solution

Line graph approach



Line graph

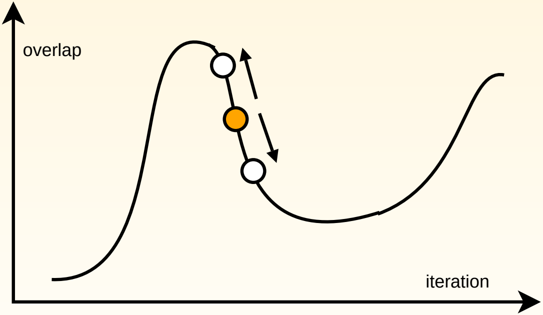
- a node of L(G) represents an edge of G
- two nodes of L(G) are adjacent if edges in G share a common node

Representation

- graph isomorphism
- edges matching

Search strategy

Beyond random walk

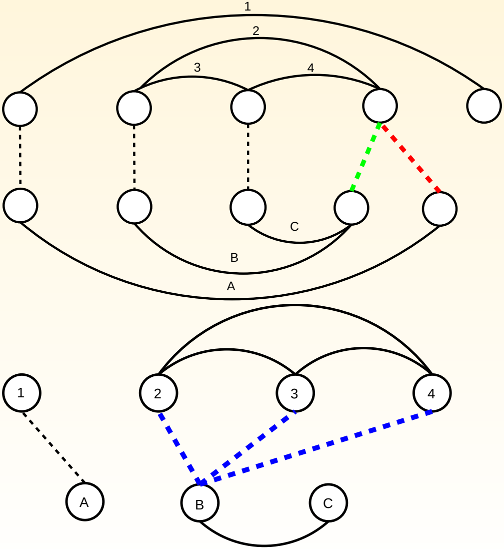


Key elements

- move application
- solution feasibility
- solution cost

How good the move is?

Determining the solution value



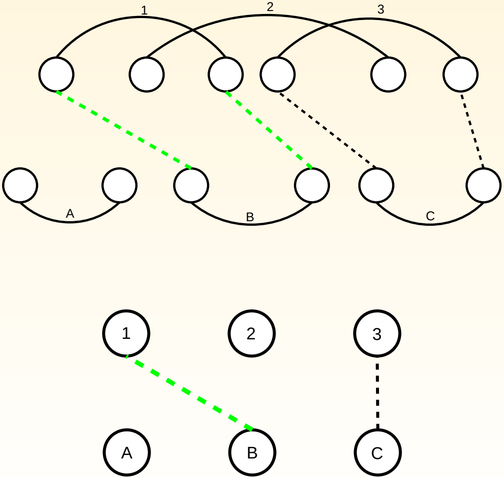
Complexity

- classic
  - $O(n * d(V))$ – full
  - $O(d(V))$ – only new
- dual
  - $O(m)$ – full
  - $O(1)$ – only new

where $m \approx n$

Is the move valid?

Checking the solution feasibility

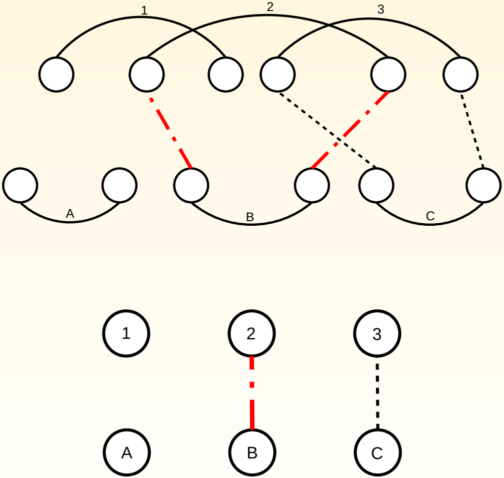


Complexity

- classic - $O(n)$
- dual - $O(n * d(V))$?

Is the move valid?

Checking the solution feasibility

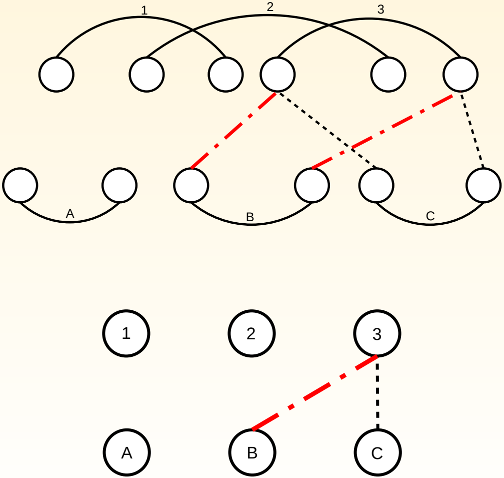


Complexity

- classic - $O(n)$
- dual - $O(n * d(V))$?

Is the move valid?

Checking the solution feasibility

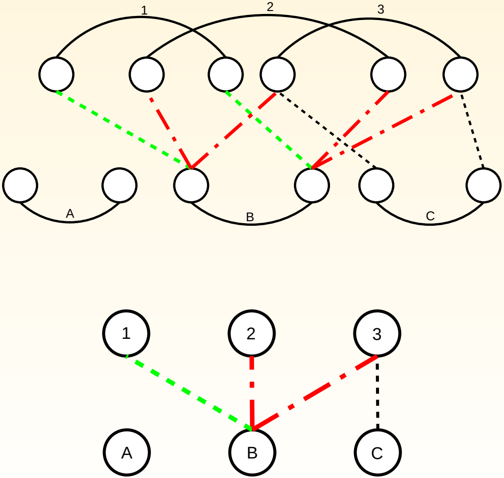


Complexity

- classic - $O(n)$
- dual - $O(n * d(V))$?

Is the move valid?

Checking the solution feasibility

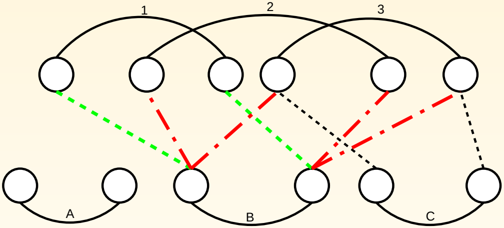


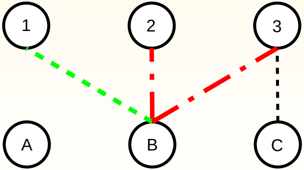
Complexity

- classic - $O(n)$
- dual - $O(n * d(V))$?

Is the move valid?

Checking the solution feasibility



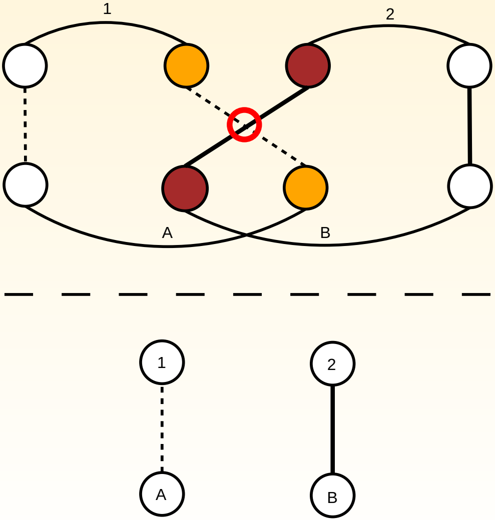


Complexity

- classic - $O(n)$
- dual - $O(n * d(V))$ **?**

Crossing conditions

Rules for feasibility checking



Consistent order

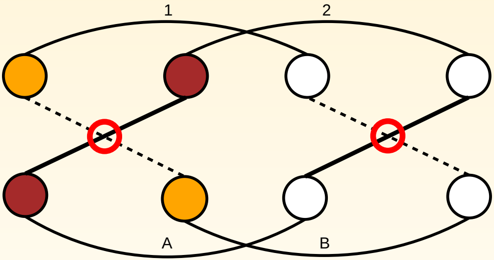
Relation of **succession** between **source** nodes should also hold for assigned **target** nodes.

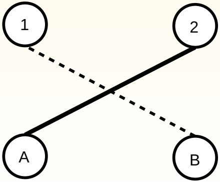
Crossing rules

- inner crossing
- outer crossing
- empirical prof of correctness

Crossing conditions

Rules for feasibility checking





Consistent order

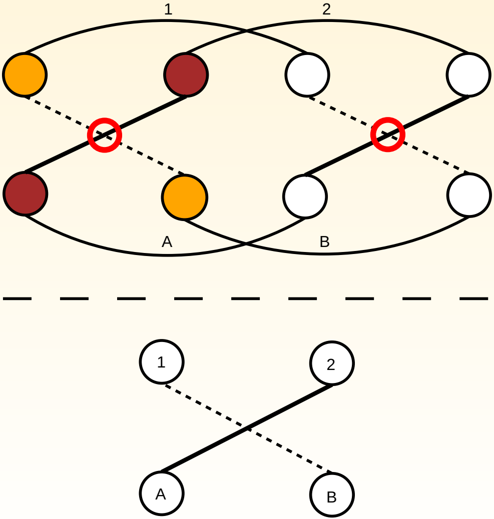
Relation of **succession** between **source** nodes should also hold for assigned **target** nodes.

Crossing rules

- inner crossing
- outer crossing
- empirical prof of correctness

Crossing conditions

Rules for feasibility checking



Consistent order

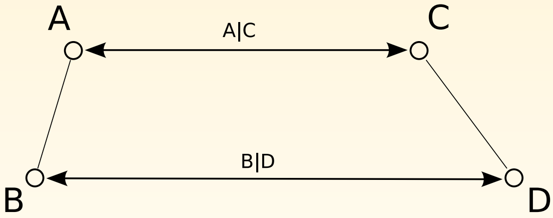
Relation of **succession** between **source** nodes should also hold for assigned **target** nodes.

Crossing rules

- inner crossing
- outer crossing
- empirical prof of correctness

The concept of limits

Improving the efficiency

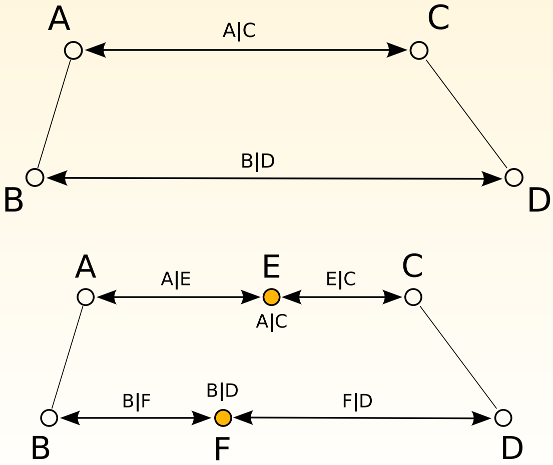


Complexity

- check – $O(1)$
- update – $O(m)$

The concept of limits

Improving the efficiency



Complexity

- check – $O(1)$
- update – $O(m)$

Summary

New algorithm vs. classical aproach

Computational complexity comparison

evaluation of solution $O(1)$ vs. $O(d(V))$

feasibility check $O(m^2 * O(1) + O(m))$ vs. $O(n^2 * O(n))$

overall gain $O(n)$ times faster

Future work

- test on reference data sets
- make available in *ProCKSi* server

Summary

New algorithm vs. classical aproach

Computational complexity comparison

evaluation of solution $O(1)$ vs. $O(d(V))$

feasibility check $O(m^2 * O(1) + O(m))$ vs. $O(n^2 * O(n))$

overall gain $O(n)$ times faster

Future work

- test on reference data sets
- make available in *ProCKSi* server

Summary

New algorithm vs. classical aproach

Computational complexity comparison

evaluation of solution	$O(1)$ vs. $O(d(V))$
feasibility check	$O(m^2 * O(1) + O(m))$ vs. $O(n^2 * O(n))$
overall gain	$O(n)$ times faster

Future work

- test on reference data sets
- make available in *ProCKSi* server

Summary

New algorithm vs. classical aproach

Computational complexity comparison

evaluation of solution $O(1)$ vs. $O(d(V))$

feasibility check $O(m^2 * O(1) + O(m))$ vs. $O(n^2 * O(n))$

overall gain $O(n)$ times faster

Future work

- test on reference data sets
- make available in *ProCKSi* server

Thank you!

Acknowledgements

This work was supported by **Marie Curie Action** MEST-CT-2004-7597 under the **Sixth Framework Programme** of the European Community.

SIXTH FRAMEWORK PROGRAMME

MARIE CURIE ACTIONS

Contact

pawel.widera@cs.nott.ac.uk