Habilitation à Diriger des Recherches

Stéphane Vialette
vialette@univ-mlv.fr

LIGM Université Paris-Est Marne-la-Vallée

01/06/10

Outline

Topics

Organization of the manuscript

- Structures
- Pattern matching in graphs
- Comparative genomics
- Additional material.

Description

Topics

Organization of the manuscript

- Structures
- Pattern matching in graphs
- Comparative genomics
- Additional material.

Description

- 2-intervals
- Linear graphs
- Arc-annotated sequences

Topics

Organization of the manuscript

- Structures
- Pattern matching in graphs
- Comparative genomics
- Additional material.

Description

- Graph homomorphisms-like aspects
- Topology-free patterns
- Softwares

Topics

Organization of the manuscript

- Structures
- Pattern matching in graphs
- Comparative genomics
- Additional material.

Description

- Genome rearrangement with duplicate genes
- Exact algorithms
- Heuristics

Topics

Organization of the manuscript

- Structures
- Pattern matching in graphs
- Comparative genomics
- Additional material.

Description

- Selenocysteine-like insertion
- Exemplar common subsequences
- How many words are needed to build up all words ?

Outline

Structures: objects of interest

Structures

- High-order intervals, *i.e*, *d*-intervals and variants
- Linear graphs
- Permutations
- Arc-annotated sequences

"Well, what are those (not so) linear structures?"

". . . all those combinatorial objects that I can draw from left to right, align and search for a pattern in".

More precisely . . .

". . . all those combinatorial objects that fit well under my $\mathcal{M} = \{<, \sqsubset, \between\}$ framework".

Structures: objects of interest

Structures

- High-order intervals, *i.e*, *d*-intervals and variants
- Linear graphs
- Permutations
- Arc-annotated sequences

“Well, what are those (not so) linear structures?”

“. . . all those combinatorial objects that I can draw from left to right, align and search for a pattern in”.

More precisely . . .

“. . . all those combinatorial objects that fit well under my $\mathcal{M} = \{<, \sqsubset, \between\}$ *framework”*.

Structures: objects of interest

Structures

- High-order intervals, *i.e*, *d*-intervals and variants
- Linear graphs
- Permutations
- Arc-annotated sequences

“Well, what are those (not so) linear structures?”

“. . . all those combinatorial objects that I can draw from left to right, align and search for a pattern in”.

More precisely . . .

“. . . all those combinatorial objects that fit well under my $\mathcal{M} = \{<, \sqsubset, \between\}$ *framework”*.

d-intervals

Definition (*Trotter, and Harary, 1979; Griggs, and West, 1979*)

A *d*-interval is a set of the real line which can be written as the union of *d* disjoint closed intervals $[a_i, b_i]$.

The intersection graph of a family of *d*-intervals is a *d*-interval graph.

Definition (*Gyárfás, 2003*)

A *d*-track interval is a union of *d* intervals, one each from *d* parallel lines

A graph is a *d*-track interval graph if it is the intersection graph of *d*-track intervals.

Definition

A *d*-box is the Cartesian product of intervals $[a_i, b_i]$, $1 \leq i \leq d$.

A graph is a *d*-box graph if it is the intersection graph of *d*-boxes.

d-intervals

Definition (*Trotter, and Harary, 1979; Griggs, and West, 1979*)

A *d*-interval is a set of the real line which can be written as the union of *d* disjoint closed intervals $[a_i, b_i]$.

The intersection graph of a family of *d*-intervals is a *d*-interval graph.

Definition (*Gyárfás, 2003*)

A *d*-track interval is a union of *d* intervals, one each from *d* parallel lines

A graph is a *d*-track interval graph if it is the intersection graph of *d*-track intervals.

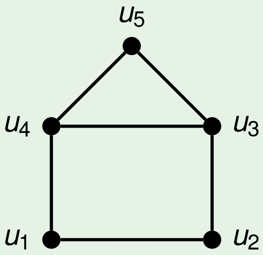
Definition

A *d*-box is the Cartesian product of intervals $[a_i, b_i]$, $1 \leq i \leq d$.

A graph is a *d*-box graph if it is the intersection graph of *d*-boxes.

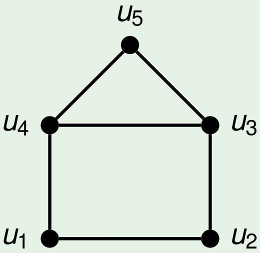
d-intervals: $d = 2$

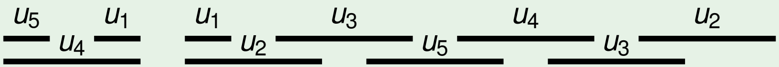
Example



d-intervals: $d = 2$

Example

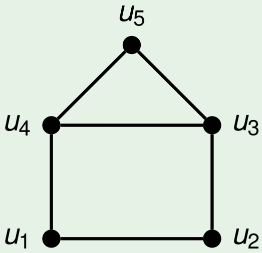


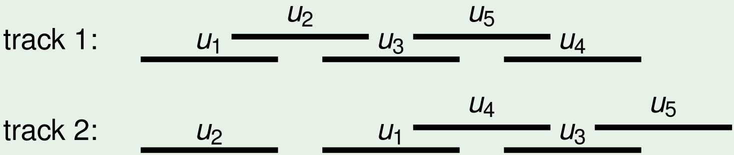


2-interval representation

d-intervals: $d = 2$

Example





2-track interval representation

Restricted d-intervals

Definition

- A d-interval $I = (I_1, I_2, \ldots, I_d)$ is balanced if $|I_1| = |I_2| = \ldots = |I_d|$.
- A d-interval $I = (I_1, I_2, \ldots, I_d)$ is unit if it is composed of d intervals of length 1.
- A d-interval $I = (I_1, I_2, \ldots, I_d)$ with integer endpoints is type $(l_1, l_2, \ldots, l_d)$ if $|I_i| = l_i$ for all $1 \leq i \leq d$.

Definition

The depth of a family of d-intervals is the maximum number of intervals that share a common point.

d-intervals

Recognizing d-interval and d-track interval graphs

Type	d-interval graphs	d-track interval graphs
UNRESTRICTED	**NP**-complete *[WS]*	**NP**-complete *[GW]*
BALANCED	**NP**-complete *[GV]*	**NP**-complete *[GV, J]*
UNIT	?	**NP**-complete *[J]*
$(2, 2, \ldots, 2)$	?	**NP**-complete *[J]*
DEPTH-2	? $(+1$ approximation$)$	**NP**-complete *[J]*
DEPTH-2, UNIT	linear-time *[J]*	**NP**-complete *[J]*

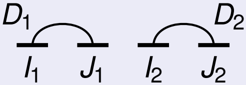
[WS] D. West and S. Shmoys, Discrete Applied Mathematics, 1984.
[GW] A. Gyárfás and D. West, Congressus Numerantium, 1995.
[GV] P. Gambette and S. Vialette, WG, LNCS, 2007.
[J] M. Jiang, FAW, LNCS, 2010.

2-Intervals: Introducing binary relations

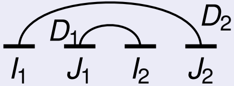
Definition

Let $D_1 = (I_1, J_1)$ and $D_2 = (I_2, J_2)$ be two 2-intervals. We write

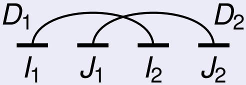
- $D_1 < D_2$ (D_1 *precedes* D_2), if $I_1 \prec J_1 \prec I_2 \prec J_2$,



- $D_1 \sqsubset D_2$ (D_1 is *nested* in D_2), if $I_2 \prec I_1 \prec J_1 \prec J_2$, and



- $D_1 \between D_2$ (D_1 *crosses* D_2), if $I_1 \prec I_2 \prec J_1 \prec J_2$,



2-Intervals and models

Definition (Model)

A non-empty subset $\mathcal{M} \subseteq \{<, \sqsubset, \between\}$ is called a model.

A collection of disjoint 2-interval $\mathcal{D}$ is said to be type $\mathcal{M}$ for some model $\mathcal{M}$ if any two 2-intervals of C are comparable for some relation $R \in \mathcal{M}$.

Example

2-Intervals and models

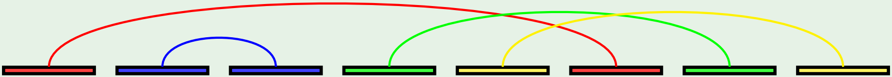
Definition (Model)

A non-empty subset $\mathcal{M} \subseteq \{<, \sqsubset, \between\}$ is called a model.

A collection of disjoint 2-interval $\mathcal{D}$ is said to be type $\mathcal{M}$ for some model $\mathcal{M}$ if any two 2-intervals of C are comparable for some relation $R \in \mathcal{M}$.

Example

$$\mathcal{M} = \{<, \sqsubset, \between\}$$



2-Intervals and models

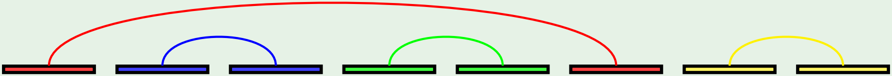
Definition (Model)

A non-empty subset $\mathcal{M} \subseteq \{<, \sqsubset, \between\}$ is called a model.

A collection of disjoint 2-interval $\mathcal{D}$ is said to be type $\mathcal{M}$ for some model $\mathcal{M}$ if any two 2-intervals of C are comparable for some relation $R \in \mathcal{M}$.

Example

$$\mathcal{M} = \{<, \sqsubset\}$$



2-Intervals and models

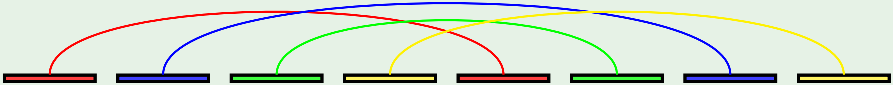
Definition (Model)

A non-empty subset $\mathcal{M} \subseteq \{<, \sqsubset, \between\}$ is called a model.

A collection of disjoint 2-interval $\mathcal{D}$ is said to be type $\mathcal{M}$ for some model $\mathcal{M}$ if any two 2-intervals of C are comparable for some relation $R \in \mathcal{M}$.

Example

$$\mathcal{M} = \{\sqsubset, \between\}$$



2-Intervals and models

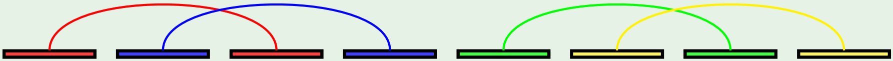
Definition (Model)

A non-empty subset $\mathcal{M} \subseteq \{<, \sqsubset, \between\}$ is called a model.

A collection of disjoint 2-interval $\mathcal{D}$ is said to be type $\mathcal{M}$ for some model $\mathcal{M}$ if any two 2-intervals of C are comparable for some relation $R \in \mathcal{M}$.

Example

$$\mathcal{M} = \{<, \between\}$$



Restricted stability

Finding patterns type $\mathcal{M}$ in 2-intervals

$\mathcal{M}$	Interval Ground Set	
	Unlimited, Balanced, Unit	Disjoint (*i.e.*, Linear graphs)
$\{<, \sqsubset, \between\}$	**APX**-hard *[BYal]*	$O(n\sqrt{n})$ *[MV]*
$\{<, \between\}$	**NP**-complete *[BFV]*	**NP**-complete *[LL]*
$\{\sqsubset, \between\}$	**APX**-hard *[V]*	$O(n \log n + \mathcal{L})$ *[CYY]*
$\{<, \sqsubset\}$	$O(n \log n + nd)$ *[CYY]*	
$\{<\}$	$O(n \log n)$ *[V]*	
$\{\sqsubset\}$	$O(n \log n)$ *[BFV]*	
$\{\between\}$	$O(n \log n + \mathcal{L})$ *[CYY]*	

[BYal] *R. Bar-Yehuda, M. Halldorsson, J. Naor, H. Shachnai and I. Shapira, SODA, 2002.*
[BFV] *G. Blin, F. Fertin, S. Vialette, Theoretical Computer Science, 2007.*
[MV] *S. Micali and V.V. Vazirani, FOCS, 1980.*
[CYY] *E. Chen, L. Yang and H.. Yuan, Journal of Combinatorial Optimization, 2007.*
[LL] *S. Li and M. Li, Theoretical Computer Science, 2009.*
[BFV] *S. Vialette, Theoretical Computer Science, 2004.*

Restricted stability

Finding patterns type $\mathcal{M}$ in 2-intervals

$\mathcal{M}$	Interval Ground Set			
	Unlimited	Balanced	Unit	Disjoint
$\{<, \sqsubset, \between\}$	4 [BYal]	4 [Cal]	3 [BYal]	N/A
$\{\sqsubset, \between\}$	4 [BYal]	4 [Cal]	3 [Cal]	N/A
$\{<, \between\}$	PTAS [J] (or effective 2 [J])			

[BYal] *R. Bar-Yehuda, M. Halldorsson, J. Naor, H. Shachnai and I. Shapira, SODA, 2002.*

[Cal] *M. Crochemore, D. Hermelin, G. Landau, D. Rawitz, and S. Vialette, Theoretical Computer Science, 2008.*

[J] *M. Jiang, COCOA, 2007.*

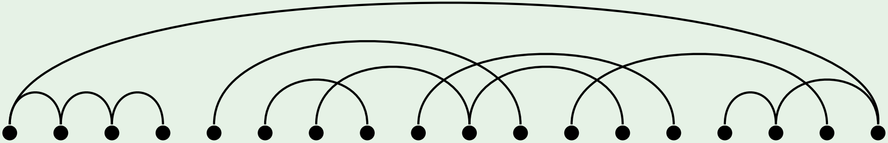
[J2] *M. Jiang, Journal of Combinatorial Optimization, 2007.*

Linear graphs

Definition (Linear graphs)

A *linear graph* of order n is a vertex-labeled graph where each vertex is labeled by a distinct label from $\{1, 2, \ldots, n\}$.

Example: Replacing labels by the natural left-to-right order



Definition (Linear matching)

A *linear matching* is an edge-disjoint linear graph.

Linear graphs: binary relations

Definition

Let $e = (i, j)$ and $e' = (i', j')$ be two disjoint edges in a linear graph or a linear matching G. We write:

- $e < e'$ (e *precedes* e') if $i < j < i' < j'$,
- $e \sqsubset e'$ (e is *nested* in e') if $i' < i < j < j'$, and
- $e \between e'$ (e and e' *cross*) if $i < i' < j < j'$.

Definition

- Two edges e and e' are *R-comparable*, for some $R \in \{<, \sqsubset, \between\}$, if eRe' or $e'Re$.
- For a subset $\mathcal{M} \subseteq \{<, \sqsubset, \between\}$, $\mathcal{M} \neq \emptyset$, edges e and e' are said to be *$\mathcal{M}$-comparable* if e and e' are R-comparable for some $R \in \mathcal{M}$.
- A linear matching whose edge set is $\mathcal{M}$-comparable (*i.e.*, any pair of distinct edges are $\mathcal{M}$-comparable) is said to be *type* $\mathcal{M}$.

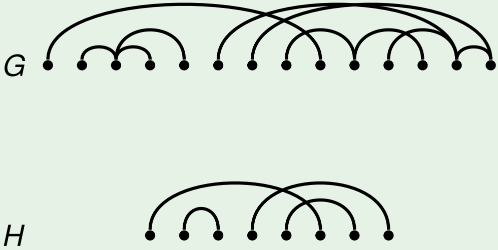
Linear graphs: Pattern matching

Pattern Matching

Input: A pattern in the form of a linear matching and a target linear graph.

Question: Does there exist an occurrence of the pattern in the target?

Example



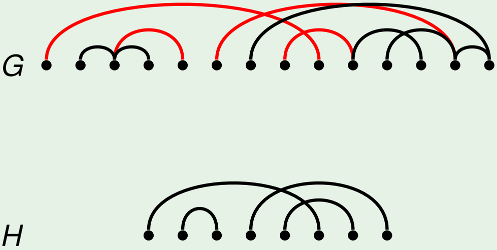
Linear graphs: Pattern matching

Pattern Matching

Input: A pattern in the form of a linear matching and a target linear graph.

Question: Does there exist an occurrence of the pattern in the target?

Example



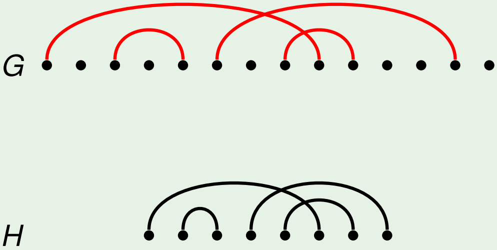
Linear graphs: Pattern matching

PATTERN MATCHING

Input: A pattern in the form of a linear matching and a target linear graph.

Question: Does there exist an occurrence of the pattern in the target?

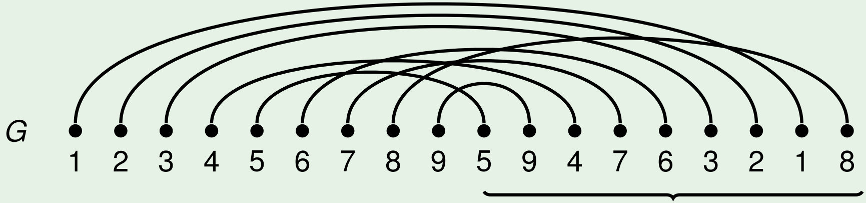
Example



Permutations

- It is well-known that linear matchings type $\{\sqsubset, \between\}$ are in bijection with permutations.
- Pattern matching for linear matchings type $\{\sqsubset, \between\}$ is the bottleneck.

Example: From linear matchings type $\{\sqsubset, \between\}$ to permutations



$\pi_G = 5\,9\,4\,7\,6\,3\,2\,1\,8$

Linear graphs: Permutations

PERMUTATION PATTERN

Input: Two permutations σ and π.

Question: Decide whether $\sigma \preceq \pi$, *i.e.*, there exists a subsequence of entries of π that has the same relative order as σ?

Example

3215674 contains the pattern 132 since the subsequence 154 is ordered in the same way as 132.

3215674 does not contain the pattern 1324.

Theorem (*Bose, Buss and Lubiw, 1998*)

PERMUTATION PATTERN *is* **NP***-hard.*

Linear graphs: Focusing on pattern avoiding permutations

Some positive results

- PERMUTATION PATTERN is solvable in $O(n^{0.47k+o(k)})$ time [*Albert, Aldred, Atkinson, and Holton, 2001*]
- PERMUTATION PATTERN is polynomial-time solvable if σ is *separable*, [*Bose, Buss and Lubiw, 1998*].
- PERMUTATION PATTERN is solvable in $O(n \log\log(n))$ time if $\sigma = 1 \ldots k$ or $\sigma = k \ldots 1$ [*Hunt, Szymanski, 1977*].

Theorem (*Guillemot, and V., 2009*)

PERMUTATION PATTERN *is solvable in* $O(k^2 n^6)$ *time in case both* π *and* σ *are* 321*-avoiding.*

If only σ *is required to be* 321*-avoiding,* PERMUTATION PATTERN *is* **NP***-complete but is solvable in* $O(kn^{4\sqrt{k}+12})$ *time.*

Permutations: The big (algorithmic) question

Question

Is PERMUTATION PATTERN fixed-parameter tractable for its standard parameterization, *i.e.*, solvable in $f(k)\, n^{O(1)}$ time, where f is an arbitrary function depending only on k?

Remarks

- I would go for yes.
- Proving fixed-parameter tractability is likely to require strong new results for pattern avoiding permutations.
- Many weaker questions are still unanswered.

Linear graphs: finding common restricted patterns

Maximum Common Structured Pattern (MCSP)

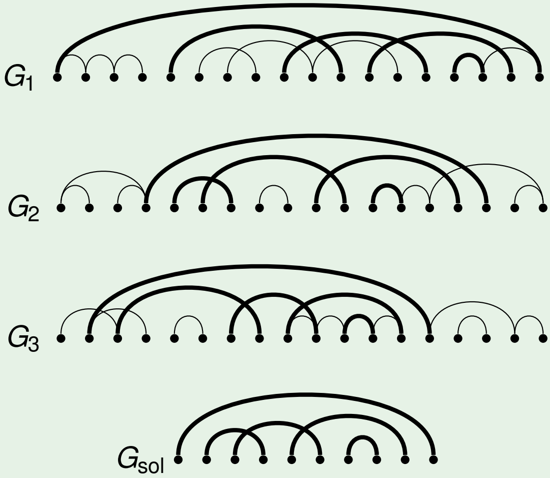
Input: A family of linear graphs $\mathcal{G} = \{G_1, G_2, \ldots, G_n\}$ and a non-empty subset $\mathcal{M} \subseteq \{<, \sqsubset, \between\}$.

Solution: A common structured pattern G_{sol} type $\mathcal{M}$ of $\mathcal{G}$, *i.e.*, a linear matching type $\mathcal{M}$ that occurs in each input linear graph of $\mathcal{G}$.

Measure: The size of G_{sol}, *i.e.*, $|\mathbf{E}(G_{\mathrm{sol}})|$.

Linear graphs: finding common restricted patterns

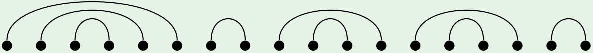
Example



Finding common patterns type $\{<, \sqsubset\}$

Some convenient names: Sequence, towers, sequence of towers

- A linear matching type $\{<\}$ (resp. $\{\sqsubset\}$) is called a *sequence* (resp. *tower*).
- A linear matching type $\{<, \sqsubset\}$ with the additional property that any two maximal towers in it do not share an edge is called a *sequence of towers*.



Theorem (*Kubica, Rizzi, V., and Waleń, 2010*)

MCSP *for structured patterns type* $\{<, \sqsubset\}$ *is* **NP**-*hard even if each input linear matching is a sequence of towers of height at most* 2*.*

Finding common patterns type $\{<, \sqsubset\}$

Theorem (*Kubica, Rizzi, V., and Waleń, 2010*)

MCSP *for structured patterns type* $\{<, \sqsubset\}$ *is approximable within ratio* $O(\log k)$ *in* $O(nm^2)$ *time, where* k *is the size of an optimal solution,* $n = |\mathcal{G}|$*, and* m *is the maximum size of any linear graph in* $\mathcal{G}$*.*

Remarks

- Improve previous $O(\log^2(k))$ ratio by Davydov and Batzoglou [*Davydov, and Batzoglou, 2006*].
- We are not aware of any better approximation ratio for sequences of towers.
- MCSP for structured patterns type $\{<, \sqsubset\}$ is polynomial-time solvable in case the number of input linear graphs is a fixed integer [*Kubica, Rizzi, V., and Waleń, 2010*].

Finding common patterns type $\{<, \sqsubset, \between\}$

Theorem (*Kubica, Rizzi, V., and Waleń, 2010*)

The MCSP *problem for patterns type* $\{<, \sqsubset, \between\}$ *is approximable*

- *within ratio* $O(k^{2/3})$ *in* $O(nm^{1.5})$ *time,*
- *within ratio* $O(\sqrt{k \log^2(k)})$ *in* $O(nm^2)$ *time, and*
- *within ratio* $O(\sqrt{k \log(k)})$ *in* $O(nm^{3.5} \log m)$ *time,*

where k *is the size of an optimal solution,* $n = |\mathcal{G}|$*, and* $m = \max_{G \in \mathcal{G}} |E(G)|$.

Theorem (*Kubica, Rizzi, V., and Waleń, 2010*)

Let G *be a linear matching type* $\{<, \sqsubset, \between\}$ *of size* k*. Then* G *contains either a tower or a balanced sequence of staircases of size* $\Omega\left(\sqrt{k/\log(k)}\right)$.

Finding common patterns type $\{<, \sqsubset, \between\}$

Theorem (*Kubica, Rizzi, V., and Waleń, 2010*)

The MCSP *problem for patterns type* $\{<, \sqsubset, \between\}$ *is approximable*

- *within ratio* $O(k^{2/3})$ *in* $O(nm^{1.5})$ *time,*
- *within ratio* $O(\sqrt{k\log^2(k)})$ *in* $O(nm^2)$ *time, and*
- *within ratio* $O(\sqrt{k\log(k)})$ *in* $O(nm^{3.5}\log m)$ *time,*

where k *is the size of an optimal solution,* $n = |\mathcal{G}|$*, and* $m = \max_{G\in\mathcal{G}} |E(G)|$.

Theorem (*Kubica, Rizzi, V., and Waleń, 2010*)

Let G *be a linear matching type* $\{<, \sqsubset, \between\}$ *of size* k*. Then* G *contains either a tower or a balanced sequence of staircases of size* $\Omega\left(\sqrt{k/\log(k)}\right)$.

Finding common patterns

Closing remarks

- Approximating MCSP for structured patterns type $\{\sqsubset, \between\}$ remains the bottleneck:

  > *Using families of linear matchings type* $\{\sqsubset, \between\}$ *to probe the input graphs, no approximation guarantee better than* $O(\sqrt{k})$ *for maximum common structured patterns type* $\{\sqsubset, \between\}$ *can be possibly achieved.*

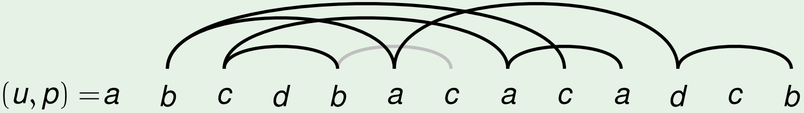
- MCSP is strongly related to finding common structures in contact maps [*Goldman, Istrail, and Papadimitriou, 1999*]. What about 3-dimensional self-avoid walks?
- MCSP is strongly related to finding common structures in 2-pages linear structures [*Evans, 2007*]. It has been argued that 2-pages linear structures capture most RNA pseudoknotted structures.
- Biologically sounding models [*Herrbach, abd V., 2005*].

Arc-annotated sequences

Definition (Arc-annotated sequence)

An *arc-annotated sequence* over alphabet $\mathcal{A}$ is a pair (u, P), where u (the *sequence*) is a string over $\mathcal{A}^*$ and P (the *annotation*) is a set of arcs $\{(i, j) : 1 \leq i < j \leq |u|\}$.

Occurrences



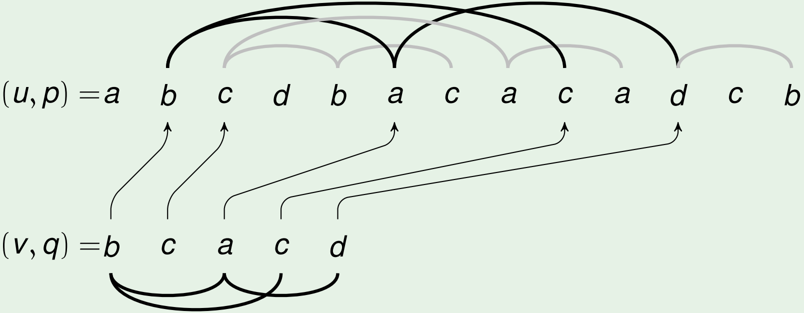
$(u, p) = a \; b \; c \; d \; b \; a \; c \; a \; c \; a \; d \; c \; b$

Arc-annotated sequences

Definition (Occurrence)

Let (u, P) and (v, Q) be two arc-annotated sequences. The arc-annotated sequence (v, Q) *occurs* in (u, P) if (v, Q) can be obtained from (u, P) by letter deletions.

Example



$(u, p) = a \; b \; c \; d \; b \; a \; c \; a \; c \; a \; d \; c \; b$

$(v, q) = b \; c \; a \; c \; d$

Arc-annotated sequences and binary relations

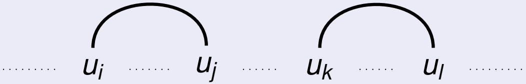
Hierarchy

Evans has introduced a five level hierarchy for arc-annotated sequences: **UNLIMITED**, **CROSSING**, **NESTED**, **CHAIN** and **PLAIN**.

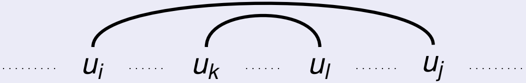
PLAIN $\subset$ **CHAIN** $\subset$ **NESTED** $\subset$ **CROSSING** $\subset$ **UNLIMITED**.

Precedence, inclusion and nesting

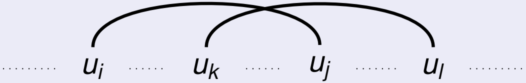
$(i,j) < (k,l)$



$(k,l) \sqsubset (i,j)$



$(i,j) \between (k,l)$



Arc-annotated sequences: APS

Arc-Preserving Subsequence (APS)

Input: Two arc-annotated sequences (u, p) and (v, Q).

Question: Does there exist an occurrence of (u, P) in (v, Q)?

Notation

For two subsets $\mathcal{M}, \mathcal{M}' \in \{<, \sqsubset, \between\}$, $\mathcal{M} \neq \emptyset$, $\mathcal{M}' \neq \emptyset$, we let LAPCS$(\mathcal{M}, \mathcal{M}')$ stand for the APS problem where (u, P) and (v, Q) are arc-annotated sequences type $\mathcal{M}$ and $\mathcal{M}'$, respectivelly.

Remark

APS$($**Plain**, **Plain**$)$ is the standard pattern matching problem.

Arc-annotated sequences: APS

Some key results

- APS$(\{<,\sqsubset,\between\},\{<\})$ is **NP**-complete [*Guo, 2002*].
- APS$(\{<,\sqsubset,\between\},\{<,\sqsubset,\between\})$ and APS$(\textsc{Unlimited},\textsc{Plain})$ are **NP**-complete [*Evans, 1999*; *Gramm, Guo and Niedermeier, 2006*].
- APS$(\{<,\sqsubset\},\{<,\sqsubset\})$ and APS$(\{<\},\textsc{Plain})$ are solvable in $O(nm)$ and $O(n+m)$ time, respectivelly [*Gramm, Guo and Niedermeier, 2006*].

Theorem (*Blin, Fertin, Rizzi and V., 2005*)

APS$(\{\sqsubset,\between\},\textsc{Plain})$ *and* APS$(\{<,\between\},\textsc{Plain})$ *are* **NP***-complete .*
APS$(\{\between\},\{\between\})$ *is solvable in* $O(nm^2)$ *time.*

Arc-annotated sequences: LAPCS

Longest Arc-Preserving Common Subsequence (LAPCS)

Input: Two arc-annotated sequences (u, p) and (v, Q).

Solution: An arc-annotated sequence (w, R) that occurs in both (u, P) and (v, Q).

Measure: The number of letters of (w, R), *i.e.*, $|w|$.

Notation

For two subsets $\mathcal{M}, \mathcal{M}' \in \{<, \sqsubset, \between\}$, $\mathcal{M} \neq \emptyset$, $\mathcal{M}' \neq \emptyset$, we let LAPCS$(\mathcal{M}, \mathcal{M}')$ stand for the LAPCS problem where (u, P) and (v, Q) are arc-annotated sequences type $\mathcal{M}$ and $\mathcal{M}'$, respectivelly.

Remark

LAPCS(**Plain**, **Plain**) is the standard longest common subsequence problem.

Arc-annotated sequences: LAPCS

Some key results

- LAPCS($\{<, \sqsubset, \between\}$, **Plain**) is **NP**-complete [*Evans, 1999*].
- LAPCS($\{<, \sqsubset\}, \{<\}$) is polynomial-time solvable [*Jiang:Lin:Ma:Zhang:2000*].
- LAPCS($\{<, \sqsubset\}, \{<, \sqsubset\}$) is **NP**-complete [*Lin, Chen, Jiang and Wen, 2002*].

Theorem (*Blin, Hamel and V., 2010*)

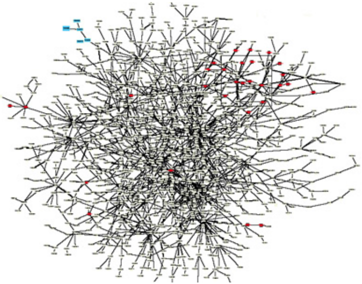
LAPCS($\{\sqsubset\}, \{\sqsubset\}$) *is* **NP***-complete.*

Outline

Pattern matching in graphs

graph-based algorithmic aspects of PPI networks

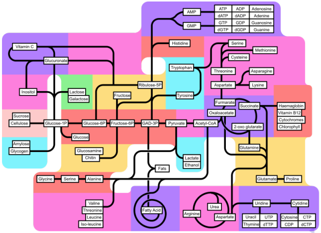
- Protein-protein interactions involve not only the direct-contact association of protein molecules but also longer range interactions.
- The interactions between proteins are important for very numerous - if not all - biological functions.



Pattern matching in graphs

graph-based algorithmic aspects of PPI networks

- Comparative analysis of protein-protein interaction graphs aims at finding complexes that are common to different species.
- Classical views include:
  - dense, clique-like interaction patterns, and
  - alignments of protein-protein interaction networks.
- We focus here on
  - graph homomorphisms-like aspects, and
  - functional approaches, *i.e.*, topology-free patterns.



Graph homomorphisms-like aspects

The big picture

- Edge-preserving pattern matching problems in graphs.
- Key element: each vertex of the motif (given in the form of a graph) is allowed to match to only few vertices of the target graph.

Injective list homomorphisms

Each vertex of the motif is associated with the list of vertices of the target graph it is allowed to match.

The goal is to find an injective mapping with respect to the lists that matches all (or at most as possible) edges.

Color matching

Vertices are associated to colors.

The goal is to find an injective mapping with respect to the colors that matches all (or at most as possible) edges.

Topology-free motifs

Motifs ?

There are two views of graph (or network) motifs:

- the topological view (where one basically ends up with certain subgraph isomorphism problems) [*Shen Orr, Mio, Mangan, and Alan, 2002*], and
- the functional approach where topology is of lesser importance [*Lacroix, Fernandes, and Sagot, 2006*].

Graph Motif

Input: A set of colors $\mathcal{C}$, a motif $\mathcal{M} = (\mathcal{C}, \text{mult})$, and a vertex colored graph (G, λ), where $\lambda : \mathbf{V}(G) \to \mathcal{C}$ is the coloring mapping.

Question: Does there exist a connected induced subgraph of G colored by $\mathcal{M}$, *i.e.*, a subset $V' \subseteq \mathbf{V}(G)$ such that (i) $G[V']$ is connected, and (ii) $\lambda(V') = \mathcal{M}$?

Topology-free motifs

Motifs ?

There are two views of graph (or network) motifs:

- the topological view (where one basically ends up with certain subgraph isomorphism problems) [*Shen Orr, Mio, Mangan, and Alan, 2002*], and
- the functional approach where topology is of lesser importance [*Lacroix, Fernandes, and Sagot, 2006*].

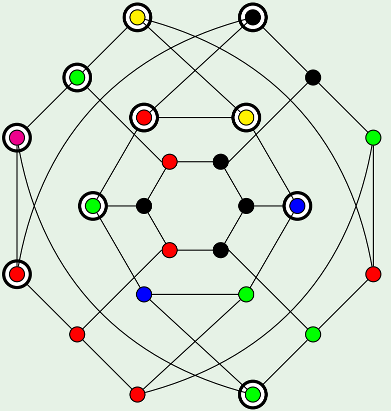
Graph Motif

Input: A set of colors $\mathcal{C}$, a motif $\mathcal{M} = (\mathcal{C}, \text{mult})$, and a vertex colored graph (G, λ), where $\lambda : \mathbf{V}(G) \rightarrow \mathcal{C}$ is the coloring mapping.

Question: Does there exist a connected induced subgraph of G colored by $\mathcal{M}$, *i.e.*, a subset $V' \subseteq \mathbf{V}(G)$ such that (i) $G[V']$ is connected, and (ii) $\lambda(V') = \mathcal{M}$?

Topology-free motifs

Example



$\mathcal{M} = \{$ 🔴 🔴 ⚫ 🟢 🟢 🟢 🔵 🟣 🟡 🟡 $\}$

Topology-free motifs: Standard complexity results

Theorem (*Lacroix, Fernandes, and Sagot, 2006*)

GRAPH MOTIF *is* **NP***-complete even if the target graph G is a tree.*

Theorem (*Fellows, Fertin, Hermelin, and V., 2010*)

The two following variants of GRAPH MOTIF *are* **NP***-complete:*

1. *the target G is a bipartite graph, $\Delta(G) = 4$, and λ is a proper 2-coloring of G, and*
2. *the target G is a tree, $\Delta(G) = 3$, each color occurs at most three times in G, and $\mathcal{M}$ is a colorful motif.*

Theorem (*Fellows, Fertin, Hermelin, and V., 2010*)

GRAPH MOTIF *is solvable in polynomial-time if the target G is a tree, each color occurs at most two times in G, and $\mathcal{M}$ is a colorful motif.*

Topology-free motifs: Standard complexity results

Theorem (*Lacroix, Fernandes, and Sagot, 2006*)

GRAPH MOTIF *is* **NP***-complete even if the target graph G is a tree.*

Theorem (*Fellows, Fertin, Hermelin, and V., 2010*)

The two following variants of GRAPH MOTIF *are* **NP***-complete:*

1. *the target G is a bipartite graph, $\Delta(G) = 4$, and λ is a proper 2-coloring of G, and*
2. *the target G is a tree, $\Delta(G) = 3$, each color occurs at most three times in G, and $\mathcal{M}$ is a colorful motif.*

Theorem (*Fellows, Fertin, Hermelin, and V., 2010*)

GRAPH MOTIF *is solvable in polynomial-time if the target G is a tree, each color occurs at most two times in G, and $\mathcal{M}$ is a colorful motif.*

Topology-free motifs: Parameterized complexity

Theorem (*Lacroix, Fernandes, and Sagot, 2006*)

GRAPH MOTIF *is fixed-parameter tractable when parameterized by the size of the motif (i.e., $|\mathcal{M}|$), in case the target graph is a tree.*

Theorem (*Fellows, Fertin, Hermelin, and V., 2010*)

GRAPH MOTIF *is solvable in $2^{O(k)}\ n^2\ \log(n)$ time, where $k = |\mathcal{M}|$ and n is the number of vertices in the target graph G.*

Proof.

Heavy use of

- Color-coding technique, and
- Perfect hash families

introduced in [*Alon:Yuster:Zwick:1995*]. □

Topology-free motifs: Parameterized complexity

Theorem (*Lacroix, Fernandes, and Sagot, 2006*)

GRAPH MOTIF *is fixed-parameter tractable when parameterized by the size of the motif (i.e., $|\mathcal{M}|$), in case the target graph is a tree.*

Theorem (*Fellows, Fertin, Hermelin, and V., 2010*)

GRAPH MOTIF *is solvable in $2^{O(k)}\, n^2 \log(n)$ time, where $k = |\mathcal{M}|$ and n is the number of vertices in the target graph G.*

Proof.

Heavy use of

- Color-coding technique, and
- Perfect hash families

introduced in [*Alon:Yuster:Zwick:1995*]. □

Topology-free motifs: Parameterized complexity

Theorem (*Fellows, Fertin, Hermelin, and V., 2010*)

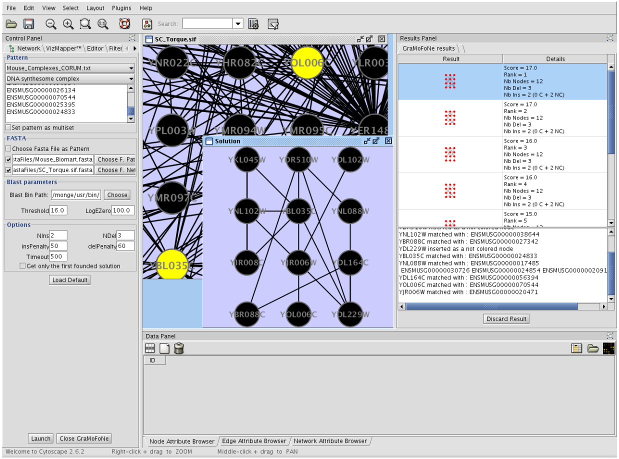
The GRAPH MOTIF *problem is in* **XP** *when parameterized by both the number of colors in the motif* $|\mathcal{M}|$ *and the treewidth of the target graph G, i.e., polynomial-time solvable when both these parameters are bounded by some constant.*

Theorem (*Fellows, Fertin, Hermelin, and V., 2010*)

The GRAPH MOTIF *problem, parameterized by the number of distinct colors c in the motif* $\mathcal{M}$*, is* **W[1]***-hard for trees.*

Topology-free motifs: Toolbox

GraMoFoNe [*Blin, Sikora, and V., 2010*]: a cytoscape integrated algorithmic toolbox to deal with the many flavors of GRAPH MOTIF.



Pattern

Mouse_Complexes_CORUM.txt

DNA synthesome complex

ENSMUSG00000026134
ENSMUSG00000070544
ENSMUSG00000025395
ENSMUSG00000024833

☐ Set pattern as multiset

FASTA

☐ Choose Fasta File as Pattern

☑ staFiles/Mouse_Biomart.fasta — Choose F. Pat

☑ astaFiles/SC_Torque.sif.fasta — Choose F. Net

Blast parameters

Blast Bin Path: /monge/usr/bin/ — Choose

Threshold 16.0 LogEZero 100.0

Options

NIns 2 NDel 3

insPenalty 50 delPenalty 60

Timeout 500

☐ Get only the first founded solution

Score = 17.0
Rank = 1
Nb Nodes = 12
Nb Del = 3
Nb Ins = 2 (0 C + 2 NC)

Score = 17.0
Rank = 2
Nb Nodes = 12
Nb Del = 3
Nb Ins = 2 (0 C + 2 NC)

Score = 16.0
Rank = 3
Nb Nodes = 12
Nb Del = 3
Nb Ins = 2 (0 C + 2 NC)

Score = 16.0
Rank = 4
Nb Nodes = 12
Nb Del = 3
Nb Ins = 2 (0 C + 2 NC)

Score = 15.0
Rank = 5

```
YNL102W matched with : ENSMUSG00000038644
YBR088C matched with : ENSMUSG00000027342
YDL229W inserted as a not colored node
YBL035C matched with : ENSMUSG00000024833
YNL088W matched with : ENSMUSG00000017485
 ENSMUSG00000030726 ENSMUSG00000024854 ENSMUSG0000002091
YDL164C matched with : ENSMUSG00000056394
YOL006C matched with : ENSMUSG00000070544
YJR006W matched with : ENSMUSG00000020471
```

Topology-free motifs: Extending the model

Graph Motif and variants

- The problem of finding a biconnected occurrence of $\mathcal{M}$ in G is **W[1]**-complete when the parameter is the size of the motif [*Betzler, Fellows, Komusiewicz, and Niedermeier, 2008*].
- Coloring vertices by lists.
- What about replacing the connectedness demand by modularity?

Turning Graph Motif into an optimization problem

- Minimizing the number of connected components in the occurrence,
- Maximizing the size of the occurrence,
- …

Topology-free motifs: Extending the model

Theorem (*Dondi, Fertin, and V., 2009*)

Maximum Motif *is* **APX***-hard even if the motif is colorful, the target graph is a tree with maximum degree* 3*, and each color occurs at most twice in the tree.*

Theorem (*Dondi, Fertin, and V., 2009*)

The Maximum Motif *problem for trees of size n can be solved in* $O(1.62^n \, \mathbf{poly}(n))$ *time. In case the motif is colorful, the time complexity reduces to* $O(1.33^n \, \mathbf{poly}(n))$.

Theorem (*Dondi, Fertin, and V., 2009*)

For any constant $\delta < 1$, Maximum Motif *for trees and colorful motifs cannot be approximated within performance ratio* $2^{\log^{\delta} n}$*, unless* $\mathrm{NP} \subseteq \mathrm{DTIME}[2^{\mathbf{poly}\log n}]$.

Outline

How many words are needed to build up all words?

Maximum Common Structured Pattern

Input: A set of strings S, a weight function $\omega : \mathcal{C}(S) \to \mathbb{Q}^+$, and an integer $\ell \geq 2$.

Solution: An ℓ-cover C of S. That is, a set of strings $C \subseteq \mathcal{C}(S)$, where for each $s \in S$ there exist $c_1, \ldots, c_p \in C$, $p \leq \ell$, with $s = c_1 \cdots c_p$.

Measure: The total weight of the cover, *i.e.*, $\omega(C) = \sum_{c \in C} w(c)$.

Example

Consider the set of strings $S = \{a, aab, aba\}$.

- $C_1 = \{a, b\}$ is a 3-cover of S, and
- $C_2 = \{a, ab\}$ is a 2-coverof S.

How many words are needed to build up all words?

Theorem (*Hermelin, Rawitz, Rizzi, and Vialette, 2008*)

Minimum Substring Cover *is* **NP**-*hard to approximate*

- *within ratio* $c\log(n)$ *for some constant c,*
- *within ratio* $\lfloor m/2 \rfloor - 1 - \varepsilon$ *for any* $\varepsilon > 0$*, and*
- *within some constant* c*, when m and* ℓ *are constant, and* ω *is either the unitary or the length-weighted function.*

Theorem (*Hermelin, Rawitz, Rizzi, and Vialette, 2008*)

With high probability, Minimum Substring Cover *is approximable within ratio* $O(m^{(\ell-1)^2/\ell}\log^{1/\ell}(n))$.

Jumping to numbers

h-GENERATING SET

Input: A set $A \subset \mathbb{N}^*$ and $h \in \mathbb{N}^*$.

Solution: A h-generating set X of A.

Measure: The size of X, *i.e.*, $|X|$.

Theorem (*Fagnot,Fertin, and Vialette, 2009*)

There exists an $O(5^{\frac{k^2(k+3)}{2}} k^2 \log k)$ *time algorithm for finding a minimum 2-generating set of any set* $A \subset \mathbb{N}^*$ *of* 2*-rank* k*.*

Theorem (*Rizzi, and Vialette, 2010*)

Deciding whether a set $A \subset \mathbb{N}^*$ *is* 2*-simplifiable is* **coNP***-hard .*

Theorem (*Rizzi, and Vialette, 2010*)

2-GENERATING SET *is strongly* **NP***-complete .*

Jumping to numbers

Som many questions are still open ...

- Let $S = \{s_i : 1 \leq i \leq n\} \subset \mathbb{N}^*$ and X be a minimum 2-generating set of S. There exist rationals $\alpha_{i,j} \in \{-1, -2^{-1}, 0, 2^{-1}, 1\}$, $1 \leq i \leq \mathrm{rk}_2(S)$ and $1 \leq j \leq n$, such that

$$X = \left\{ \sum_{j=1}^{n} \alpha_{i,j}\, s_j : 1 \leq i \leq \mathrm{rk}_2(S) \right\}.$$

Do we really need five rationals? What about 3-generatin sets? $O(1)$-generating sets?

- What about dense sets?
- What about set of consecutive integers?

Outline

Consecutive ones property

Definition

A matrix has the consecutive ones property for rows if there is a permutation of its columns that leaves the 1's consecutive in every row.

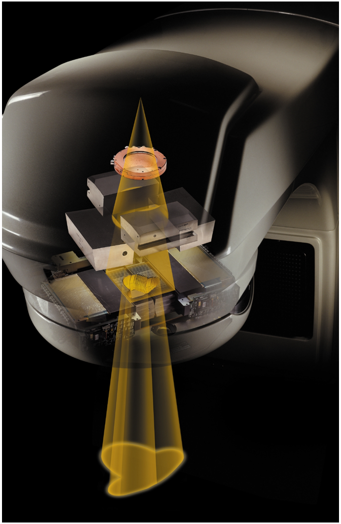
Example

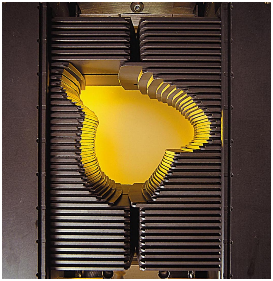
$$A = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \end{bmatrix} \qquad AP = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

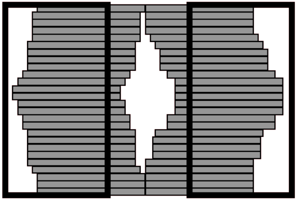
Focus

- Identifying Tucker configurations [*Blin, Rizzi, and V., 2010*].
- Identifying minimal conflicting sets [*Chauve, Stephen, Haus, and You, 2009*].

Radiotherapy: Multileaf collimators







Comparative genomics: partitions

Definition

The MINIMUM COMMON STRING PARTITION (MCSP) problem is to find a common partition of two strings *u* and *v* with the minimum number of blocks.

Example

$$u = abbbbbabcbaaaab$$
$$v = bbbabaaabcbaabb$$

Focus

- Is MCSP approximable within a constant ratio?
- What about approximating MINIMUM INDEPENDENT DOMINATING SET for 2-track interval graphs?

Comparative genomics: partitions

Definition

The Minimum Common String Partition (MCSP) problem is to find a common partition of two strings u and v with the minimum number of blocks.

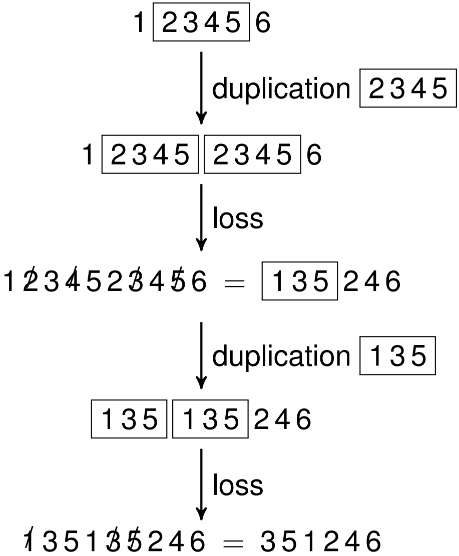
Example

$$u = (abb)_1\,(bbbab)_2\,(cba)_3\,(aaab)_4$$
$$v = (bbbab)_2\,(aaab)_4\,(cba)_3\,(abb)_1$$

Focus

- Is MCSP approximable within a constant ratio?
- What about approximating Minimum Independent Dominating Set for 2-track interval graphs?

Tandem duplication-random loss model



$$1\ \boxed{2\,3\,4\,5}\ 6$$

$$\downarrow \text{ duplication } \boxed{2\,3\,4\,5}$$

$$1\ \boxed{2\,3\,4\,5}\ \boxed{2\,3\,4\,5}\ 6$$

$$\downarrow \text{ loss}$$

$$1\,\cancel{2}\,3\,\cancel{4}\,5\,2\,\cancel{3}\,4\,\cancel{5}\,6 = \boxed{1\,3\,5}\,2\,4\,6$$

$$\downarrow \text{ duplication } \boxed{1\,3\,5}$$

$$\boxed{1\,3\,5}\ \boxed{1\,3\,5}\ 2\,4\,6$$

$$\downarrow \text{ loss}$$

$$\cancel{1}\,3\,5\,1\,\cancel{3}\,\cancel{5}\,2\,4\,6 = 3\,5\,1\,2\,4\,6$$