Parameterized Computation for Bio-molecule Folding

Liming Cai
RNA Informatics Laboratory
University of Georgia
Athens, Georgia, USA

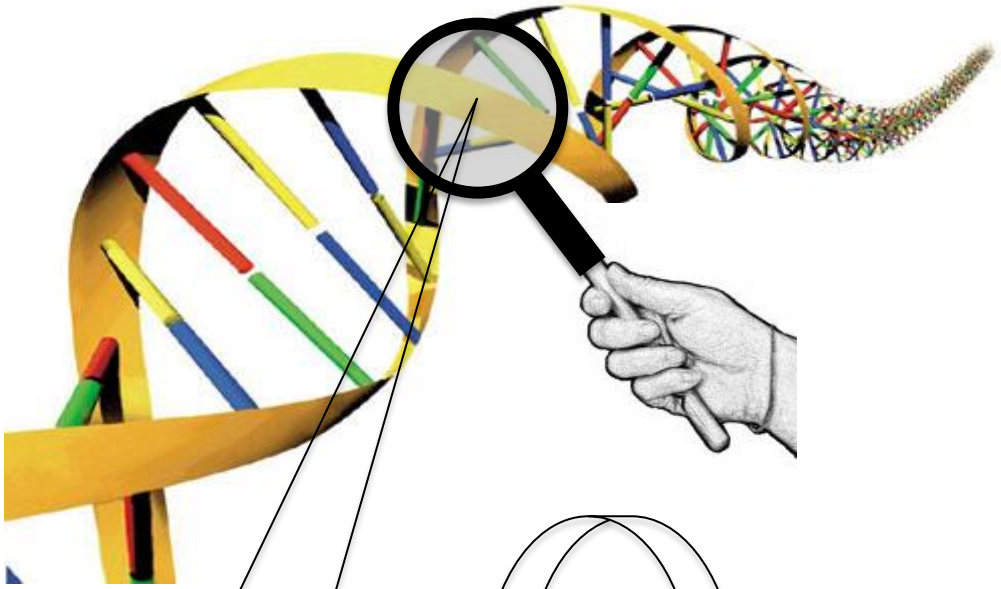
About this presentation

- New algorithmic graph algorithms needed for bio-molecule structure prediction
  - Involving graphs of small tree width
  - Parameterized computation
  - Engineering parameters
  - New applications for FPT framework

Outline

- Background
- Optimal subgraph isomorphism from *k*-trees
- Maximum spanning *k*-tree
- Additional applications

Background



GGGACTATAGTTTAGTTGGATATAACGACCGGCTACGAACCGGTAGGTCTCAGGTTCGACCCCTGATGGTCTCG

Single strand DNA

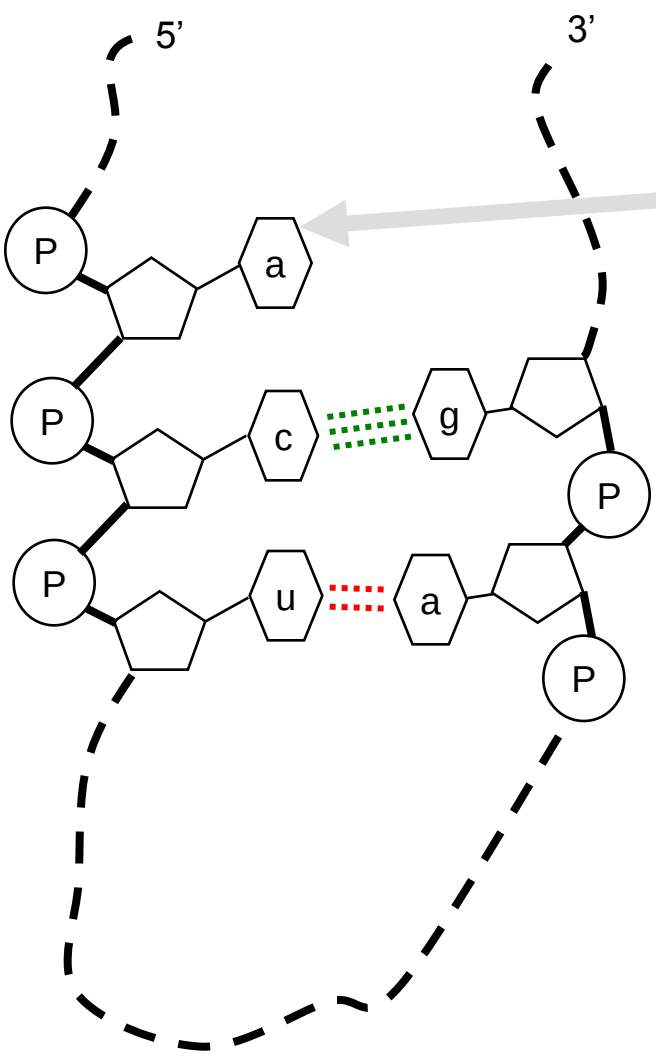
Transcribed to non-coding RNA sequence

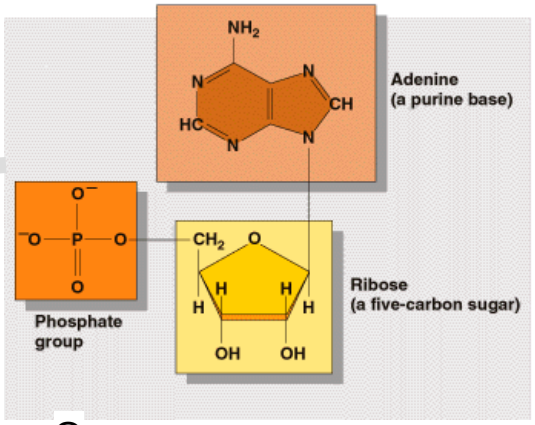
GGGACUAUAGUUUAGUUGGAUAUAACGACCGGCUACGAACCGGUAGGUCUCAGGUUCGACCCCUGAUGGUCUCG

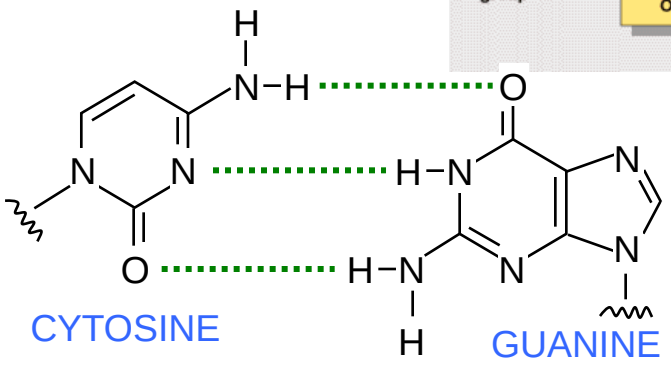
Folded into
3D structure

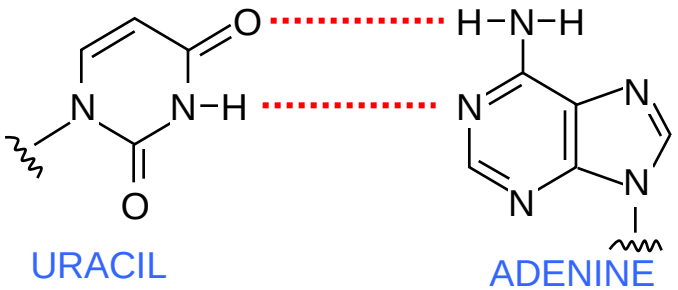
Background

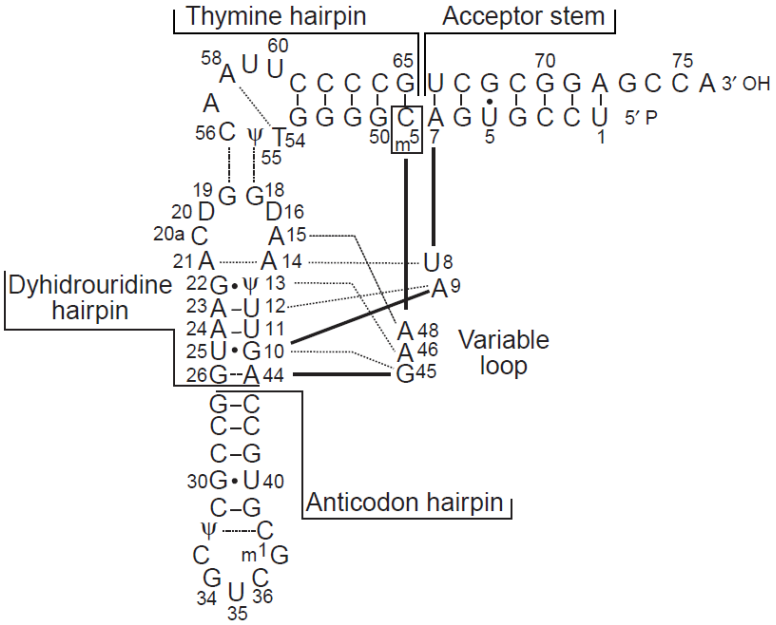
5’-u-u-c-c-g-a-a-g-c-u-c-a-a-c-g-g-g-a-a-a-u-g-a-g-c-u-3’

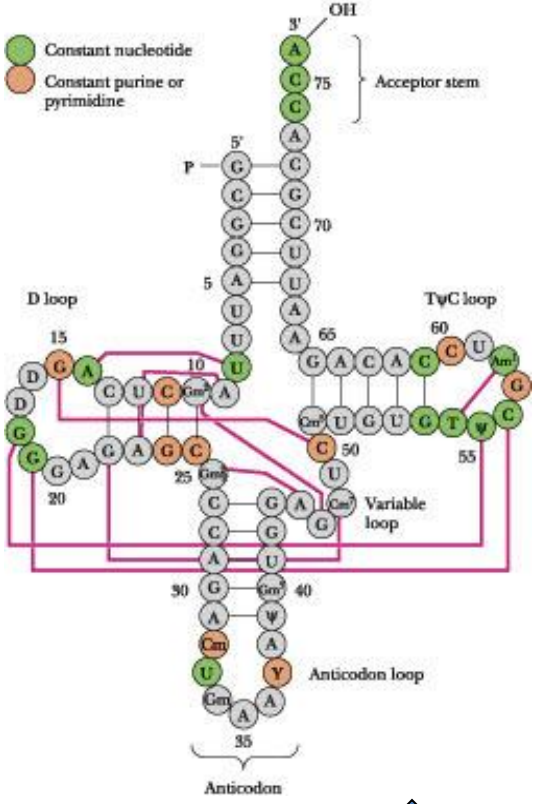


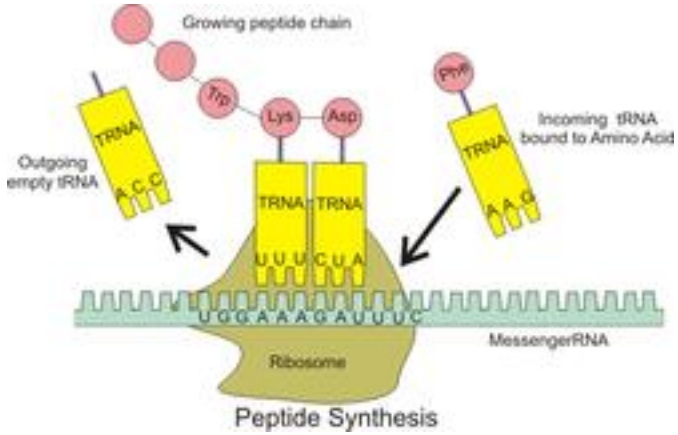


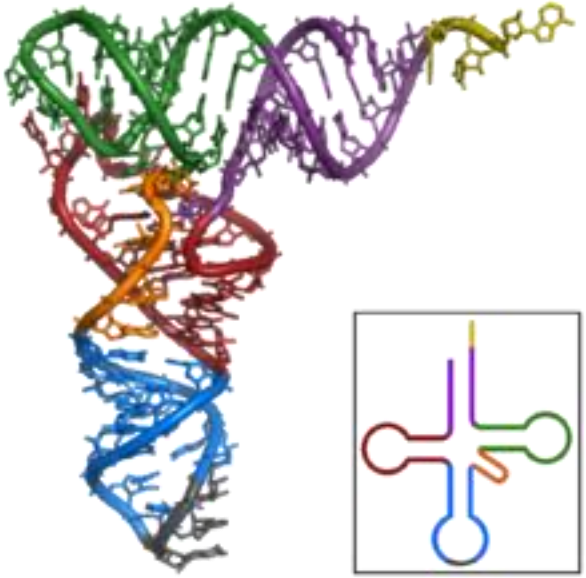





Thymine hairpin
Acceptor stem
Dyhidrouridine hairpin
Variable loop
Anticodon hairpin


Constant nucleotide
Constant purine or pyrimidine
Acceptor stem
D loop
TψC loop
Variable loop
Anticodon loop
Anticodon


Growing peptide chain
Outgoing empty tRNA
Incoming tRNA bound to Amino Acid
MessengerRNA
Ribosome
Peptide Synthesis



Background

- Tertiary structure:
  - Less understood non-canonical interactions
  - Only a small number of resolved structures

- Secondary structure:
  - (Well understood) canonical base pairs
  - Scaffolding tertiary structure
  - Well studied

Background

- Modeling structure with graphs

  To characterize interaction relationships between elements (e.g., residues) on the sequence (i.e., interaction topology)

- Need to model interactions:
  - Neighboring element connections through backbone chaining
  - Spatial contacts through energy potentials
  - Simplifications: pair-wise, non-geometric

Background

- Each topological structure of a molecule is modeled with ***backbone graph*** [Song *et al,* 2006]

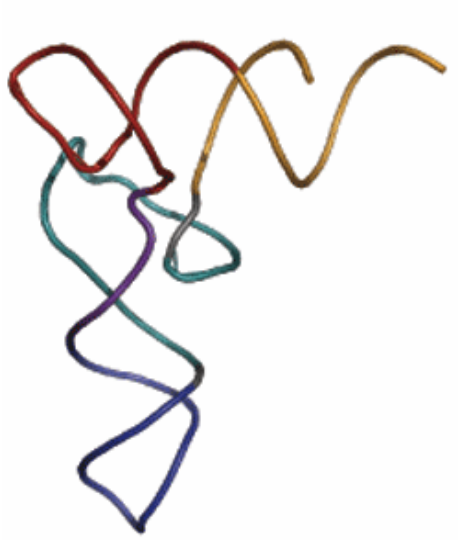
$$H = (V, E),\ E = D \cup A$$

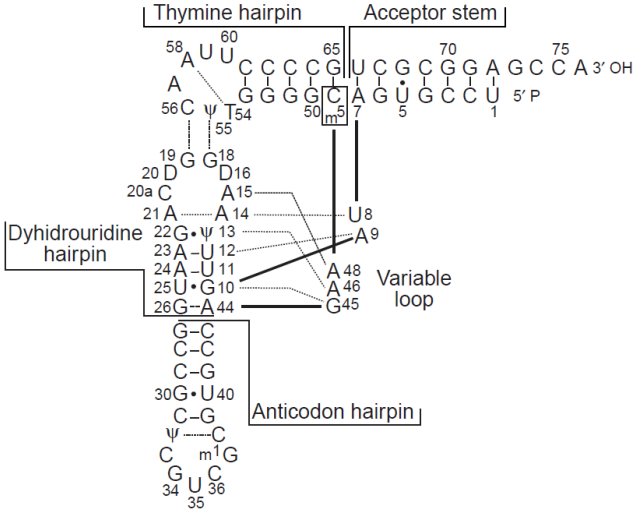
  - *V*: vertices for elements (often residues)
  - *D*: directed edges only for backbone connections
  - *A*: non-directed edges for spatial contacts

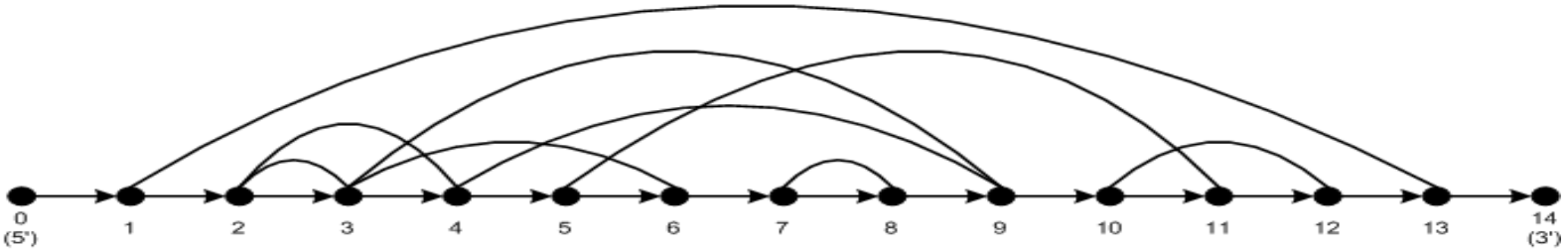
  D forms exactly a Hamiltonian path

Background

- A backbone graph example



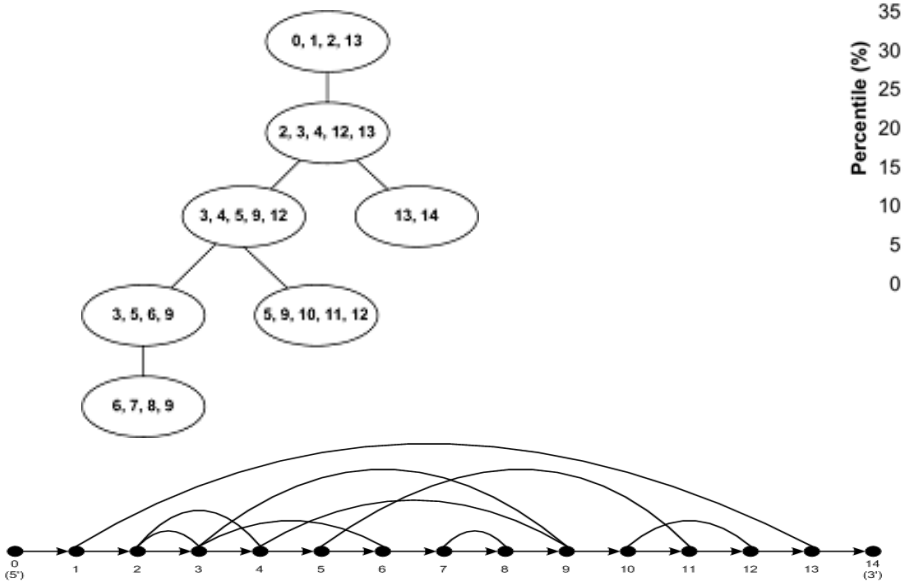


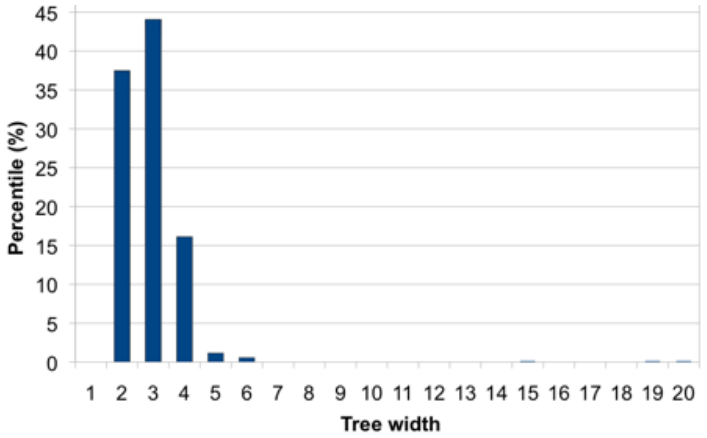


Backbone graph for tRNA tertiary structure (after residues are grouped)

Background

- Backbone graphs for bio-molecule structures are of small tree width



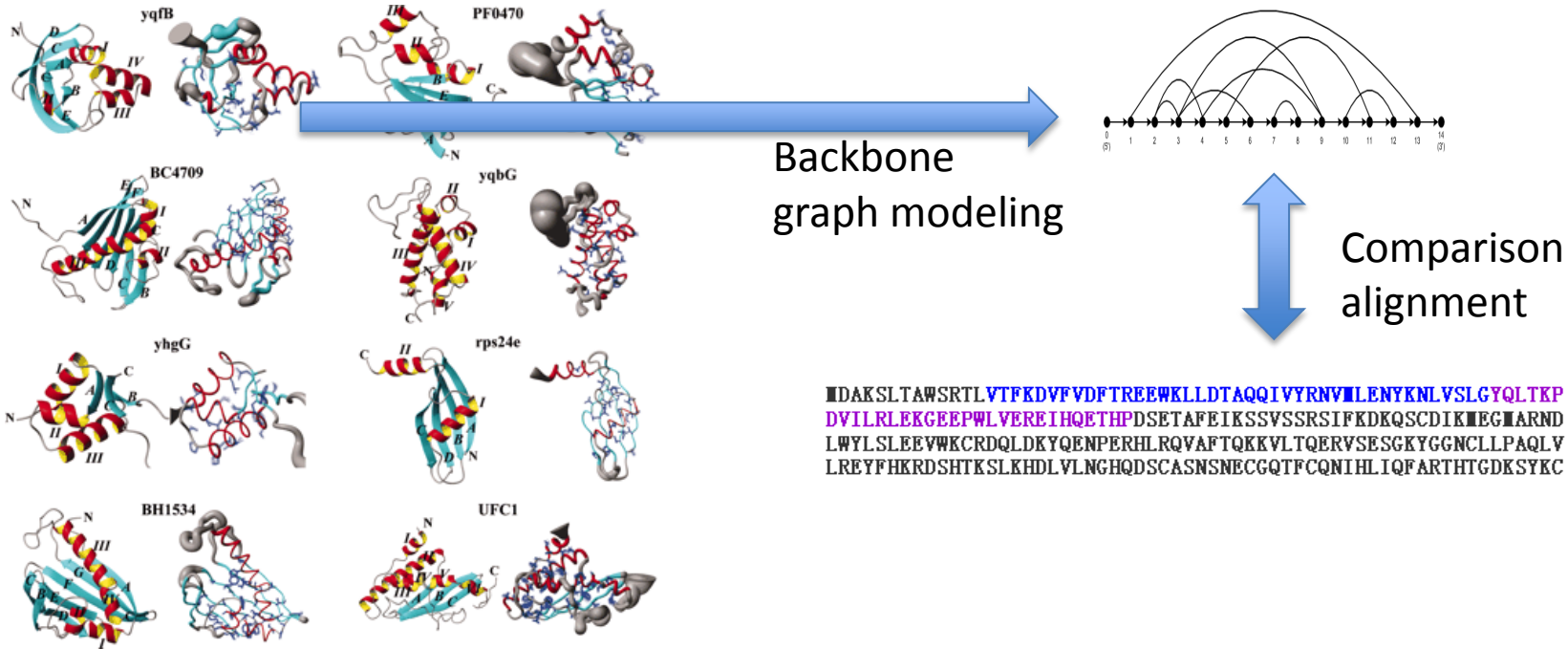


Tree width distribution of backbone graphs of 515 RNA resolved tertiary structures from PDB/NDB

Similar distributions hold for backbone graphs formulated with residues as vertices, and for 6,000+ protein tertiary structures.

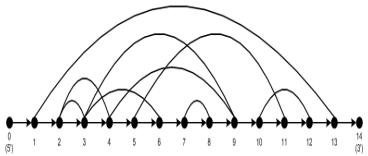
Background

- Structure prediction from molecule sequences
  1. Template based methods
     - number of structures is a small fraction of number of sequences
     - cannot predict new structures

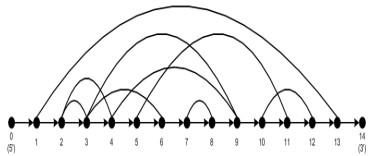


Background

- Structure prediction from molecule sequences
  1. Template based methods



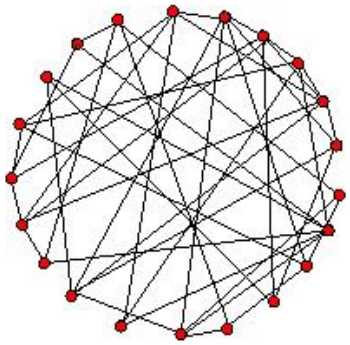
comparison
alignment



subgraph
isomorphism

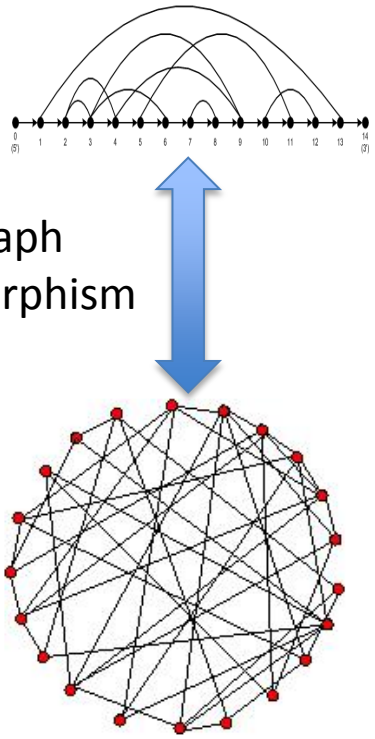
MDAKSLTAWSRTLVTFKDVFVDFTREEWKLLDTAQQIVYRNVMLENYKNLVSLGYQLTKP
DVILRLEKGEEPWLVEREIHQETHPDSETAFEIKSSVSSRSIFKDKQSCDIKMEGMARND
LWYLSLEEVWKCRDQLDKYQENPERHLRQVAFTQKKVLTQERVSESGKYGGNCLLPAQLV
LREYFHKRDSHTKSLKHDLVLNGHQDSCASNSNECGQTFCQNIHLIQFARTHTGDKSYKC

preprocessed



Background

- Structure prediction from molecule sequences
  1. Template based methods



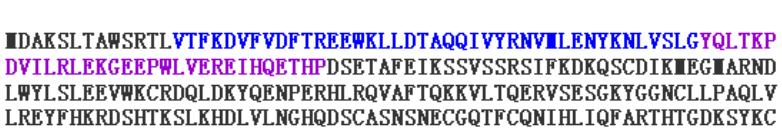
Input: backbone graph H of tree width $\leq k$,
mixed graph G, and functions g_1 and g_2,
Output: isomorphism $f : V_H \longrightarrow V_{G'}$, $G' \subseteq G$, such that

$$\sum_{v \in V_H} g_1(v, f(v)) + \sum_{(u,v) \in E_H} g_2(u, v, f(u), f(v))$$

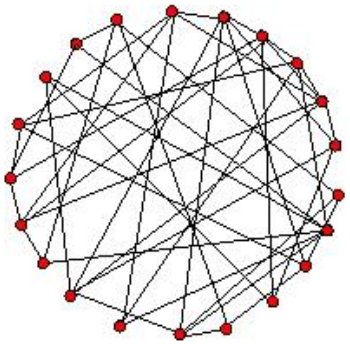
achieves the optimal.

Background

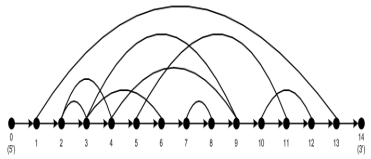
- Structure prediction from molecule sequences
  2. *ab initio* (*de novo*) methods
     - Prediction based on only the given sequence
     - Potentially can predict new structures



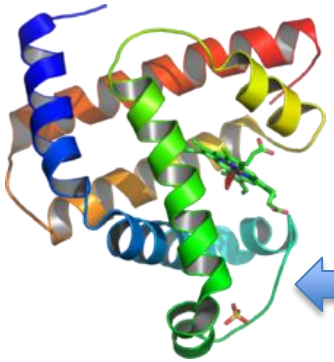
preprocessed



Extract the most plausible interaction topology

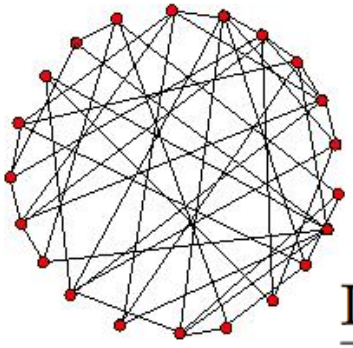


geometric fitting and shape refinement

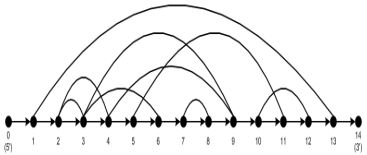


Background

- Structure prediction from molecule sequences
  2. *ab initio* (*de novo*) methods



Finding an optimal spanning k-tree as the most plausible Interaction topology



Input: backbone graph G of weight w, integer k,
Output: spanning k-tree H of G, such that

$$\sum_{(u,v)\in E_H} w(u,v)$$

achieves the optimal.

Background

- OSGI: Input: backbone graph H of tree width $\leq k$,
  mixed graph G, and functions g_1 and g_2,
  Output: isomorphism $f : V_H \longrightarrow V_{G'}$, $G' \subseteq G$, such that

$$\sum_{v \in V_H} g_1(v, f(v)) + \sum_{(u,v) \in E_H} g_2(u, v, f(u), f(v))$$

  achieves the optimal.

- MSkT: Input: backbone graph G of weight w, integer k,
  Output: spanning k-tree H of G, such that

$$\sum_{(u,v) \in E_H} w(u, v)$$

  achieves the optimal.

Subgraph isomorphism from *k*-trees

- *k*-tree, tree width, and tree decomposition are fundamental notions in coping with graph algorithm efficiency:

  e.g., Courcelle's theorem [Courcelle, 1990]:

  Monadic second order (MSO)-logic definable problems admit $O(f(k)n^c)$ -time algorithms on graphs of tree width ≤ *k*

Subgraph isomorphism from *k*-trees

- The theorem also includes: subgraph isomorphism from fixed source H of tree width ≤ *k* to host G can be solved in time $O(f(|H|, k)|G|)$

  There have been several lines of research extending this result for subgraph isomorphism.

Subgraph isomorphism from *k*-trees

- H is fixed, while G is planar:

  solvable in linear time [Eppstein , 1999]

- H has bounded degree but not fixed, while G has tree width ≤ *k*:

  solvable in time $O(|H|^{k+1}|G|)$ [Matousek and Thomas, 1992, Arnborg and Proskurowski, 1989]

- H is *k*-connected and a partial *k*-tree while G is another partial k-tree:

  solvable in time $O(n^{k+2})$ [Dessmark *et al*, 1999]

Subgraph isomorphism from *k*-trees

- Our OSGI problem is to find an optimal subgraph isomorphism from a backbone partial *k*-tree H to an arbitrary G.

Input: backbone graph H of tree width $\leq k$,
mixed graph G, and functions g_1 and g_2,
Output: isomorphism $f : V_H \longrightarrow V_{G'}$, $G' \subseteq G$, such that

$$\sum_{v \in V_H} g_1(v, f(v)) + \sum_{(u,v) \in E_H} g_2(u, v, f(u), f(v))$$

achieves the optimal.

Subgraph isomorphism from *k*-trees

- A subgraph isomorphism mapping *f* requires

  injection: $u \neq v \Rightarrow f(u) \neq f(v)$

  structure preserving:

  $$(u, v) \in E_H \Rightarrow (f(u), f(v)) \in E_G$$

- Using a tree decomposition for H, structure preserving can be checked along with a dynamic programming.

Subgraph isomorphism from *k*-trees

- With the Hamiltonian path constraint, the injection mapping property

$$u \neq v \Rightarrow f(u) \neq f(v)$$

  can be checked along with the dynamic programming.

  Proof by induction on the backbone distance between *u* and *v*.

Subgraph isomorphism from *k*-trees

- OSGI from backbone partial k-tree can be solved in tim $O(|V_G|^{k+c})$ f for some small integer $c > 0$ [Song *et al*, 2006]

Subgraph isomorphism from k-trees

- Backbone does not reduce the parameterized complexity of the problem.

  W[1]-hard, by a reduction from *k*-clique

Subgraph isomorphism from k-trees

- Additional engineering parameterization on the OSGI, with given candidate sets

$$\mathcal{M} : V_H \longrightarrow 2^{V_G}$$

$$\max\{|\mathcal{M}(v)| : v \in V_H\} \leq m$$

and bounded map-width *m.*

Subgraph isomorphism from k-trees

- OSGI problem, parameterized with *k* of (k-tree) and map width *m*, is solvable in time

$$O(m^{k+c}p(n))$$

for some constant *c* and polynomial *p*.

by following the result of Song *et al*, [2005].

Subgraph isomorphism from k-trees

- An alternative approach to the OSGI problem

  Is there a way for “non-MSO-definable” problems, such as subgraph isomorphism, to be redefined as (transformed to) MSO-definable sets over graphs of small tree width?

Subgraph isomorphism from k-trees

- An alternative approach to the OSGI problem

  E.g., subgraph isomorphism from H to G conveniently corresponds to a size $|V_H|$ clique over the product graph H x G , in which

$$V_{H\times G} = \{\, [v,x] \,:\, v \in V_H \text{ and } x \in V_G\}$$

$$E_{H\times G} = \{([v,x],[u,y]) \,:\, v \neq u, x \neq y, (v,u) \in E_H \Rightarrow (x,y) \in E_G\}$$

Subgraph isomorphism from k-trees

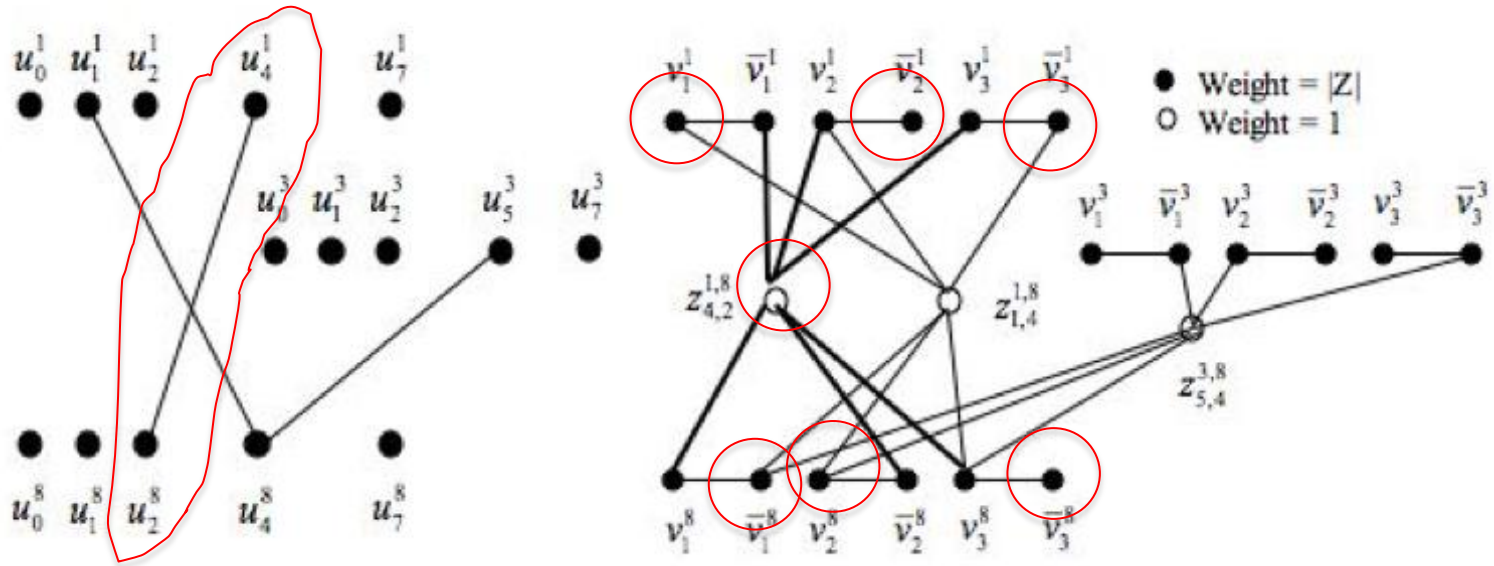
- Any clique would have to satisfy all conditions set for edges in E.

  But for SGI over backbone graphs, all conditions can be checked locally, we only need to focus on induced subgraph by each tree bag.

- Thus, the dense induced product subgraph may be replaced with a graph of smaller tree width.

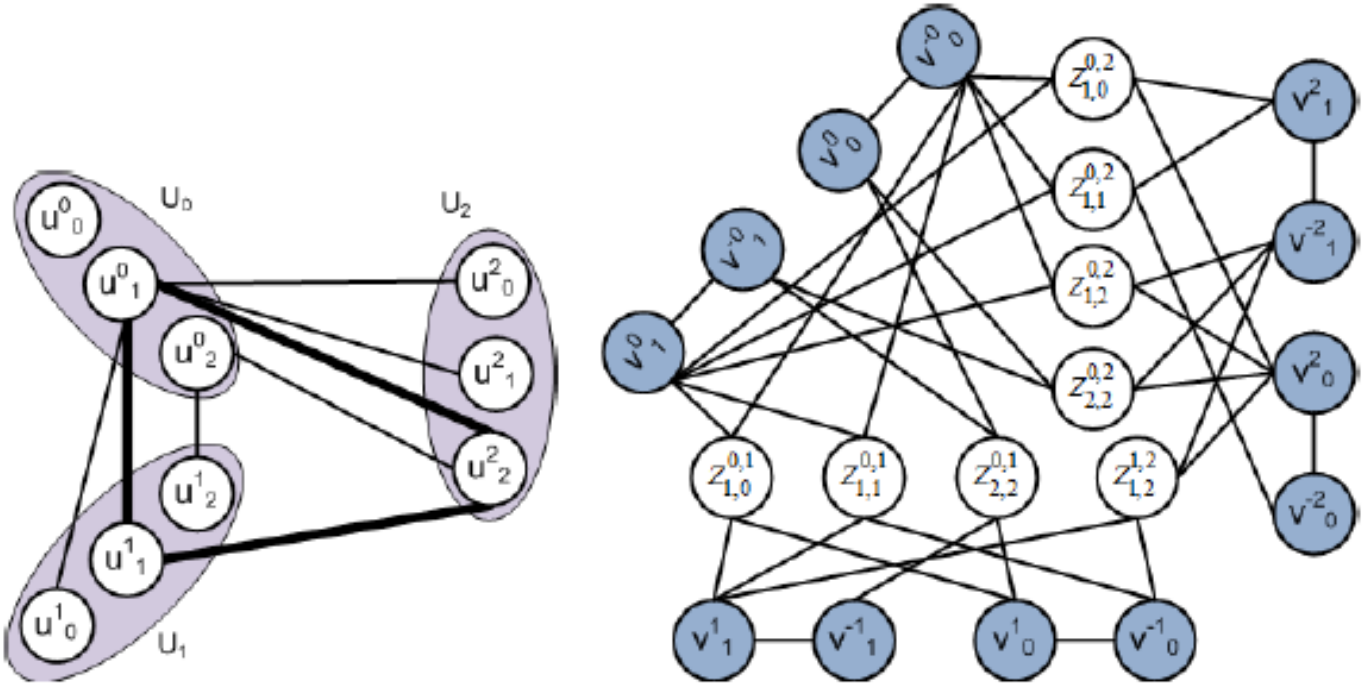
Subgraph isomorphism from k-trees

- Edge ([1, 4], [2, 8]) represented by the independent set involving 2log $|V_G|$ + 1 vertices



Subgraph isomorphism from k-trees

- The transformed product graph has tree width $\leq O((k+1)\log|V_G|)$ instead of $(k+1)|V_G|$.



(a) A 3-partite graph, G.

(b) Reduced graph G'.

Maximum spanning *k*-trees

- Problem definition (MS*k*T)

Input: backbone graph G of weight w, integer k,
Output: spanning k-tree H of G, such that

$$\sum_{(u,v)\in E_H} w(u,v)$$

achieves the optimal.

Maximum spanning *k*-trees

- Finding maximum spanning (partial) *k*-trees from a given graph,
- *K*=1, the same as minimum spanning tree
- NP-hard even for *k*=2 [Bern 1987]

Maximum spanning k-trees

- Remain NP-hard for many classes of restricted graphs [Leizhen Cai and Maffray 1993]
  - graphs of degree $\leq 3k + 2$
  - planar graphs
  - split graphs

Background

- **What type of graphs allow efficient algorithms for determining spanning *k*-tree?**
  - Decision problem on split-comparability graphs [Leizhen Cai and Maffray, 1993]
  - Not applicable to the optimization problem MS*k*T

Maximum spanning *k*-trees

- MS*k*T over backbone graphs can be solved in time $O(n^{k+2}4^k k)$ [Samad and Cai, 2012]
- It is not known if the efficiency can be further improved or it is W[1]-hard.

Maximum spanning *k*-trees

- The algorithm is dynamic programming, taking advantage of a number properties of MS*k*T on backbone graph.
- Properties are about child-parent relationships of (*k*+1)-cliques in a *k*-tree (in some ordering)

Maximum spanning *k*-trees

- **Main Property**:

  if $\mathcal{C} = \{x_0, x_1, \ldots, x_k\},\ x_i < x_{i+1}$

  then either all or none of $y,\ x_i < y < x_{i+1}$

  are in the subtree rooted at a single child (k+1)-clique of $\mathcal{C}$.

Maximum spanning *k*-trees

- **Theorem:** Any (*k*+1)-clique in a *k*-tree has at most (*k*+2) children.

- For dynamic programming, we use a canonical form of a (*k*+1)-clique sequence in a *k*-tree, such that each (*k*+1)-clique has at most two children.

Maximum spanning *k*-trees

- For every (*k*+1)-clique κ and every importable set I_κ , we compute the maximum spanning sub *k*-tree rooted at κ of I_κ .

When $I_\kappa \neq \phi$,

$$m(\kappa, I_\kappa) = \max \begin{cases} \max\limits_{\kappa'=\kappa|_y^x} m(\kappa', I_\kappa \backslash stretch(\kappa', x)) + \omega(x, \kappa) \\ \max\limits_{\kappa'=\kappa|_y^x, \mathcal{R}(I_1, I_2, I_\kappa, \kappa', x)} (m(\kappa', I_1) + m(\kappa, I_2)) \end{cases}$$

Maximum spanning *k*-trees

- The DP table has dimensions $O(n^{k+1}) \times O(2^{k+2})$

  Each entry requires a factor of time $O(n \times k)$

  for checking all *x*'s and *y*'s

  Each entry needs an additional factor of $O(2^{k+2})$

  for enumerating I_1, I_2

When $I_\kappa \neq \phi$,

$$m(\kappa, I_\kappa) = \max \begin{cases} \max\limits_{\kappa' = \kappa|_y^x} m(\kappa', I_\kappa \backslash stretch(\kappa', x)) + \omega(x, \kappa) \\ \max\limits_{\kappa' = \kappa|_y^x, \mathcal{R}(I_1, I_2, I_\kappa, \kappa', x)} (m(\kappa', I_1) + m(\kappa, I_2)) \end{cases}$$

Maximum spanning *k*-trees

- We do not know yet where MS*k*T stands in the W-hierarchy, though

  we suspect that it is at least W[1]-hard because

  using O(klog n) amount of nondeterminism to guess does not seem to solve the problem.

Maximum spanning *k*-trees

- Regardless where MS*k*T stands in the W-hierarchy, the time complexity $O(n^{k+2}4^k k)$ is too high;
- Is Maximum spanning *k*-path (MS*k*P) easier? e.g., solvable in time $O(n^{k-1}g(k))$ or better?

Maximum spanning *k*-trees

- If so, how about restricting the number of branches in the desired *k*-tree? (biologically still meaningful)
- Are there other engineering parameters making the computation problems easier?

Applications of MS*k*T solvers

1. Bio-molecule folding
   – *ab initio* structure prediction from single sequence

   - A complete graph can be formulated from a given molecule sequence, with edge weights for potentials of interactions between residues
   - Parameter value for *k* is chosen
   - MS*k*T answer gives a most plausible topological graph based on interaction potentials

Applications of MS*k*T solvers

1. Bio-molecule folding

   Note: the result of MSkT is not geometrical structure.

   - Incorporating geometric constraints into interaction potentials (non-pairwise, however)

   - From topology to geometry

Applications of MS*k*T solvers

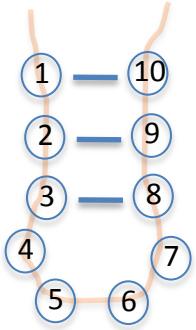
1. Bio-molecule folding

   Geometric modeling

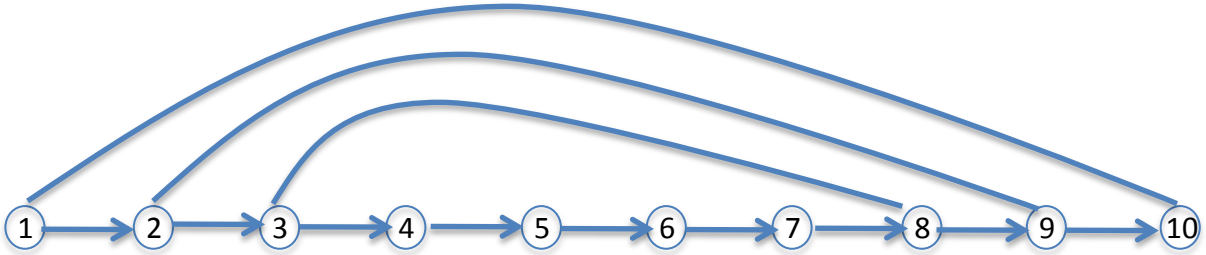
   – k=2, modeling (*k*+1)-cliques with triangles
   – suitable for secondary structure prediction

Applications of MS*k*T solvers

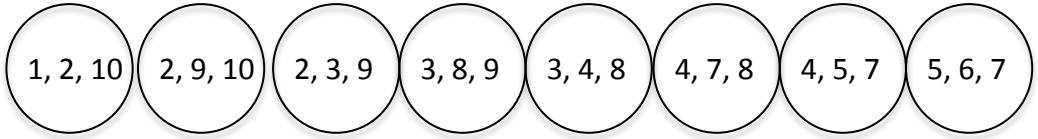
1. Bio-molecule folding (*k*=2, n=10)



RNA stem-loop
(not known)



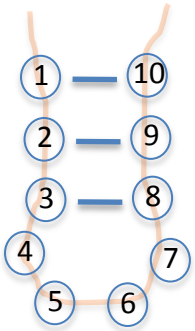
A predicted (partial) 2-path for k=2, n=10



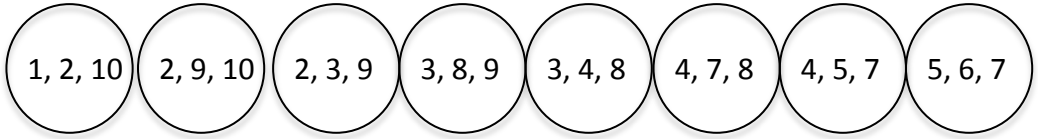
A path decomposition corresponding to the 2-path

Applications of MS*k*T solvers

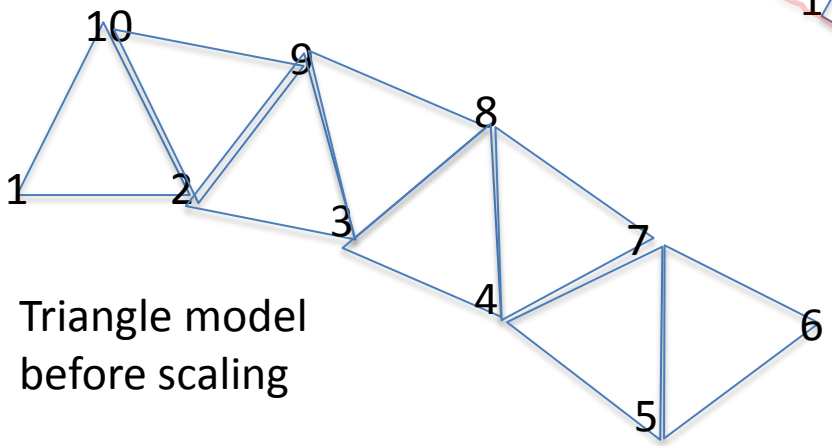
1. Bio-molecule folding (*k*=2, n=10)



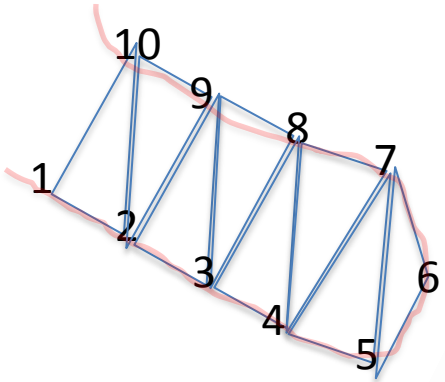
RNA stem-loop
(not known)



A path decomposition corresponding to the predicted 2-path



Triangle model
before scaling



Triangle model
after scaling

Applications of MS*k*T solvers

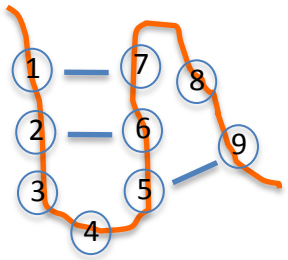
1. Bio-molecule folding

   Geometric modeling

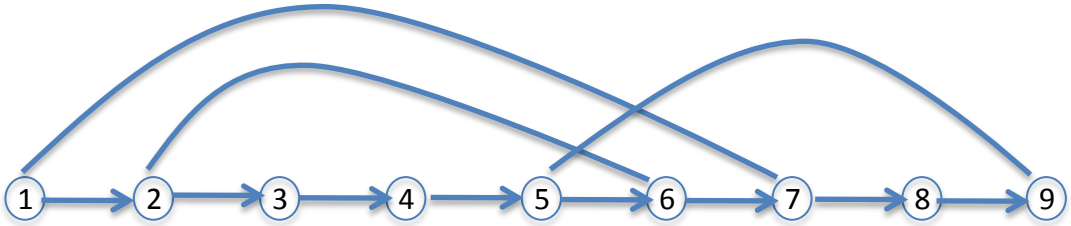
   – k=3, modeling (*k*+1)-cliques with tetrahedrons
   – can capture most tertiary structures

Applications of MS*k*T solvers

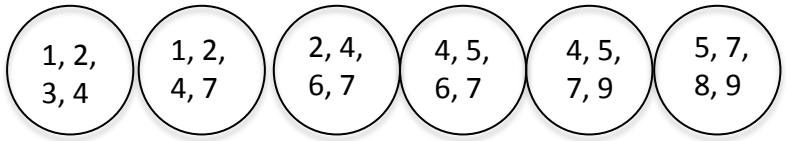
1. Bio-molecule folding (*k*=3, n=9)



RNA pseudoknot
(not known)



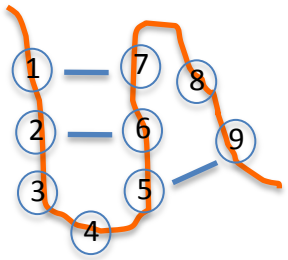
A predicted (partial) 3-path for k=3, n=9



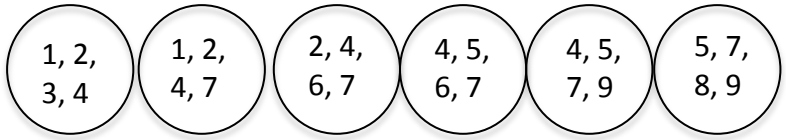
A path decomposition corresponding to the predicted 3-path

Applications of MS*k*T solvers

1. Bio-molecule folding (*k*=3, n=9)



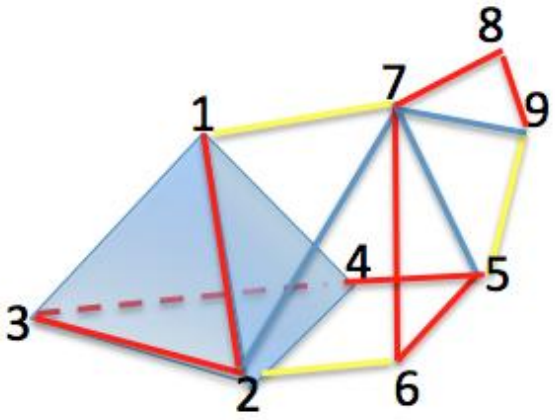
RNA pseudokn
(not known)



A path decomposition corresponding to the predicted 3-path

backbone

interaction



Tetrahedron modeling
Without scaling

Applications of MS*k*T solvers

2. Formal language theory

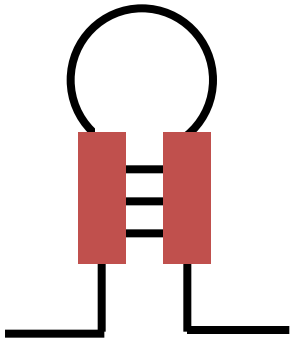
	Languages recognized	**Parsing process**	**Grammar rules**
CYK algorithm	Context-free	Tree	Context-free rules
Our algorithm	Mildly context-sensitive	k-tree	Mildly context-sensitive rules?
applications	molecule sequences	Molecule structures	

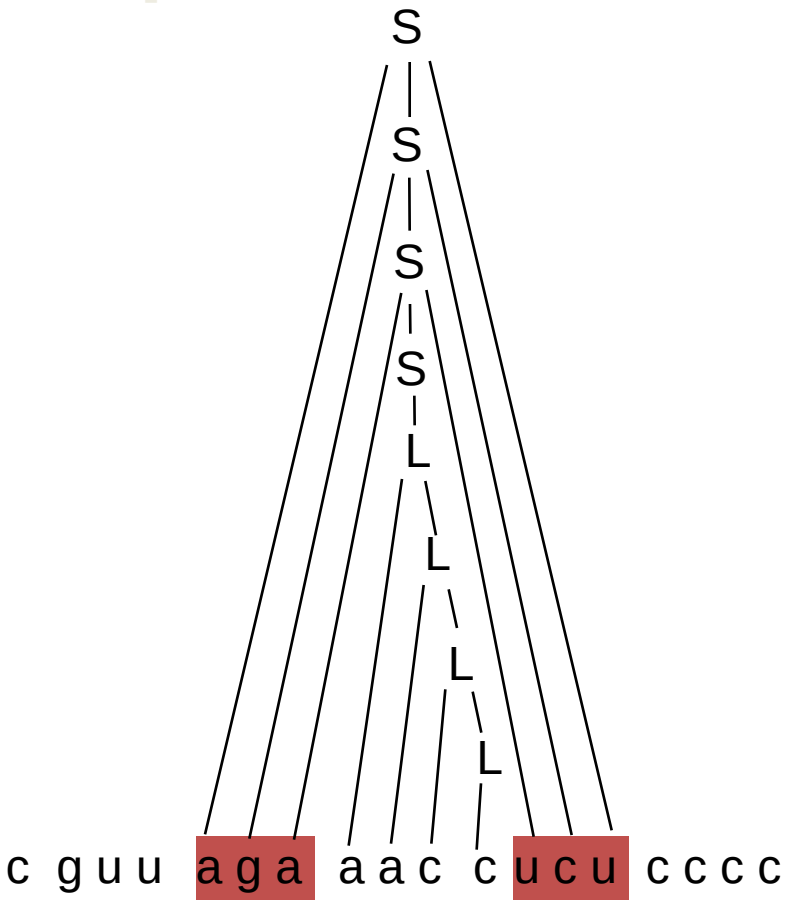
Applications of MS*k*T solvers

- **Stochastic Context-free Grammars (SCFGs)**

[Lari and Young'90, Sakakibara et al'94]

S → aSu	L → aL
S → uSa	L → cL
S → gSc	L → a
S → cSg	L → c
S → L	





- **Each derivation tree corresponds to a structure.**

Applications of MS*k*T solvers

S → aSu

S → cSg

S → gSc

S → uSa

S → a

S → c

S → g

S → u

S → SS

S ➔ aSu

➔ acSgu

➔ accSggu

➔ accuSaggu

➔ accuSSaggu

➔ accugScSaggu

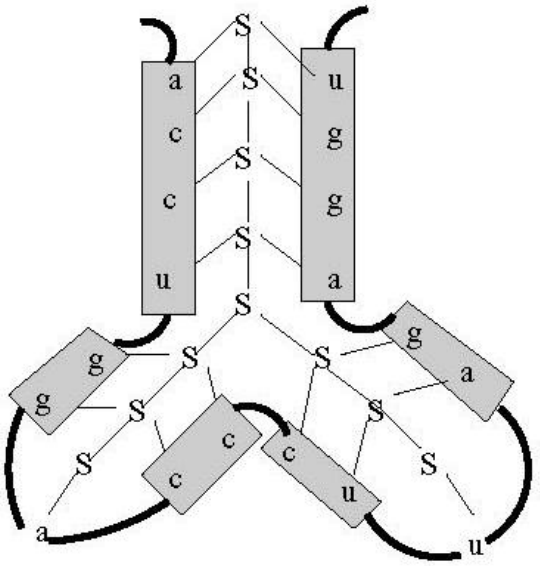
➔ accuggSccSaggu

➔ accuggaccSaggu

➔ accuggacccSgaggu

➔ accuggacccuSagaggu

➔ accuggacccuuagaggu



1. A CFG

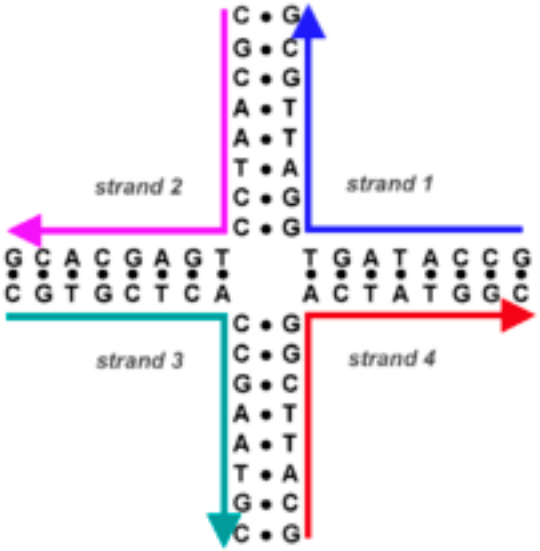
2. A derivation of "accuggacccuuagaggu"

3. Corresponding structure

Applications of MS*k*T solvers

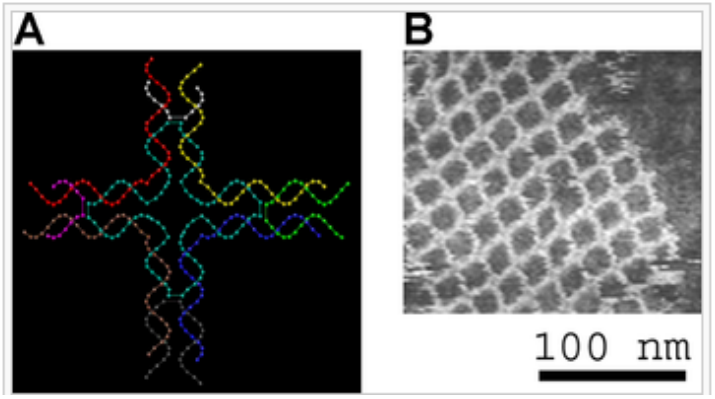
3. DNA nanotechnology

   Using DNA base pairs as basic construct to build complex with precisely controlled nanoscale features.

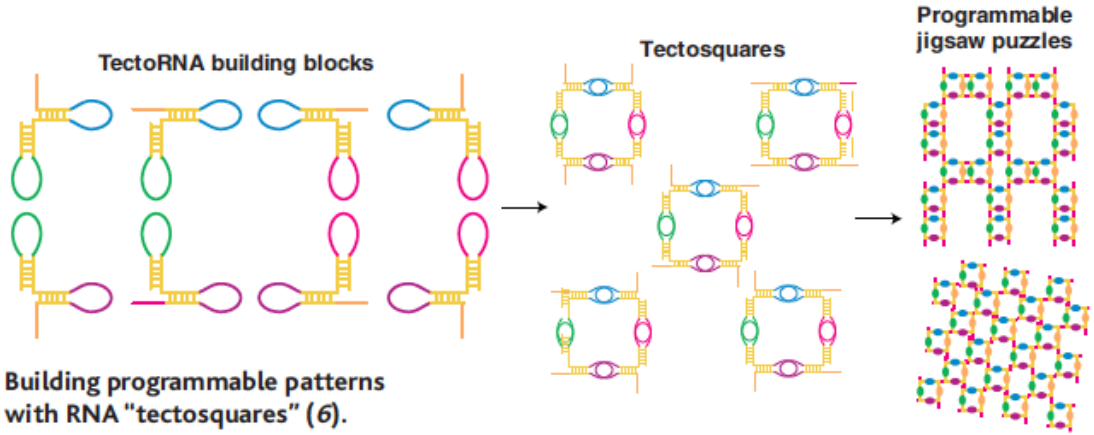


Applications of MS*k*T solvers

3. DNA nanotechnology



Can efficient algorithms for MS*k*T problem (and/or alike) serve critical roles for investigating DNA nanotechnology ?



Conclusion

- Introduced two types of parameterized computation problems on backbone graphs involving *k*-trees,
- Motivated by bio-molecule folding,
- Additional applications in formal language theory and DNA nanotechnology,
- Open problems, in particular, further engineering parameters for efficiency improvement.

Acknowledgement

- Abdul Samad

Pooya Shareghi

- Xiuzhen Huang

Russell Malmberg

- NSF

NIH