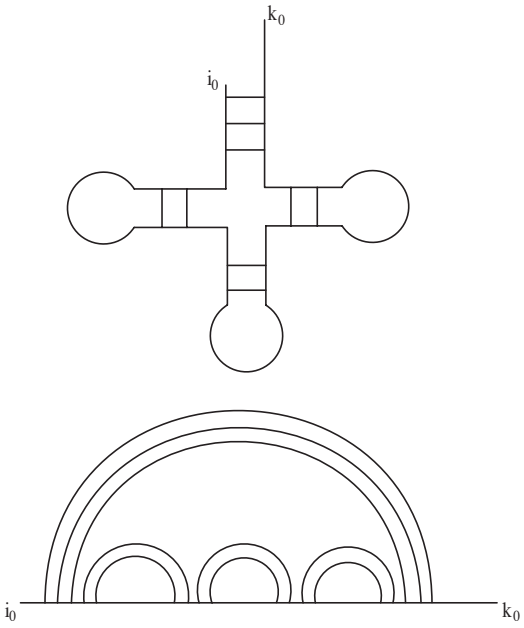
Structural Alignment of Pseudo-knotted RNA

RNA pseudo-knotted structures

RNA alignment problem has been solved for RNAs with a regular structure,
i.e. non-pseudo-knotted structures.

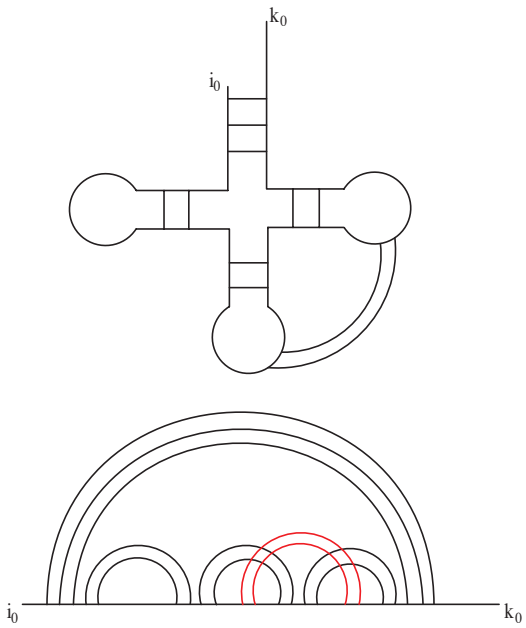
Regular structure:

All base pair are non-crossing.



Pseudo-knotted structure:

Some of the base pairs are crossing.



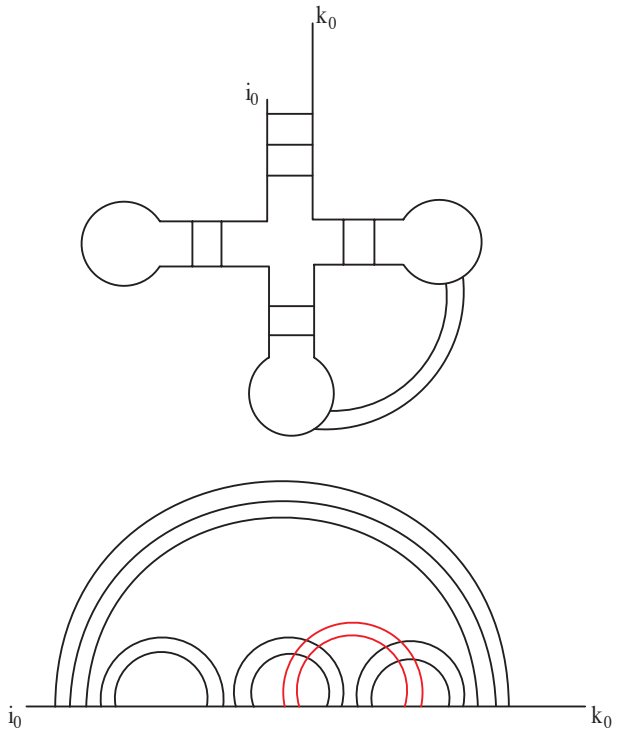
Solving problem for pseudo-knotted RNAs

Dynamic programming technique used to align subsequences.

Challenge: Aligning RNA with general pseudoknot sturctures is hard. (Jiang et. al JCB 2002).

Formal definition of pseudo-knots such that

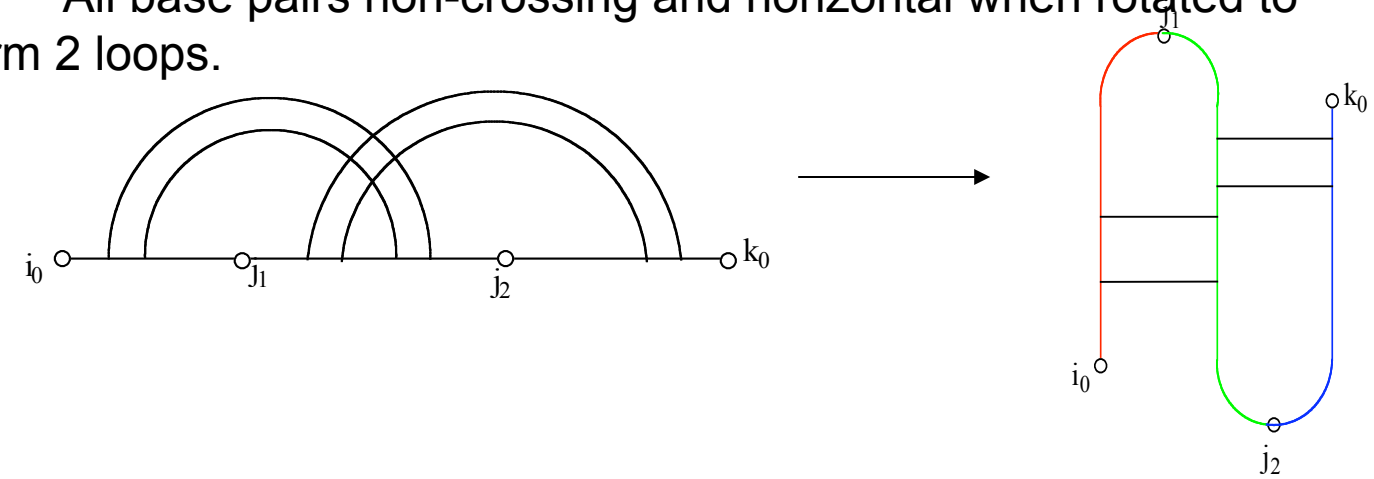
- To classify the pesudoknot strcutres so that most common pseudoknot is compuatable.
- computation is not very expensive
- biologically important



Definition: simple pseudo-knot

- How can we define a pseudo-knot?
- There are many pseudo-knot definitions: Akutsu [journal 2002?], Rivas&Eddy, ….
- For pediction.

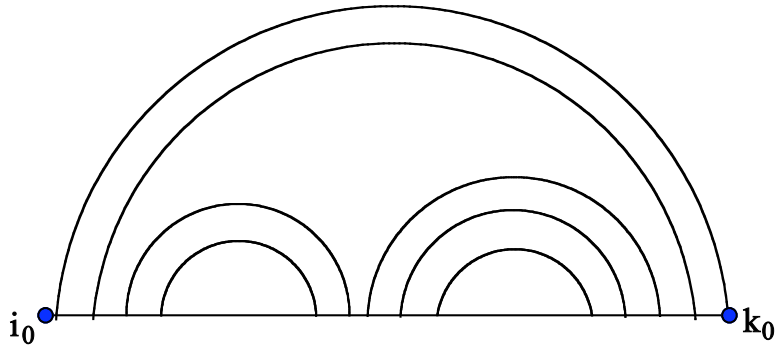
- We start with Akutsu's simple pseudo-knot formalism:
- All base pairs non-crossing and horizontal when rotated to form 2 loops.



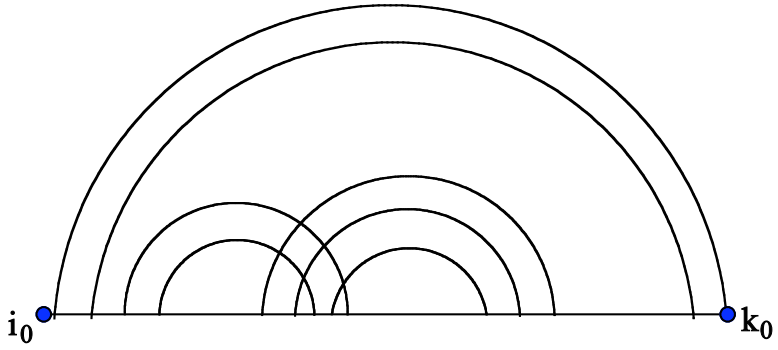
Sub-structure for a simple pseudo-knot

For DP algorithm, how to define sub-structure?

- Regular structure:
- continuous subintervals as
- substructure of recursion.



- Simple Pseudo-knot:
- can not use this substructure
- due to interweaving base pairs.



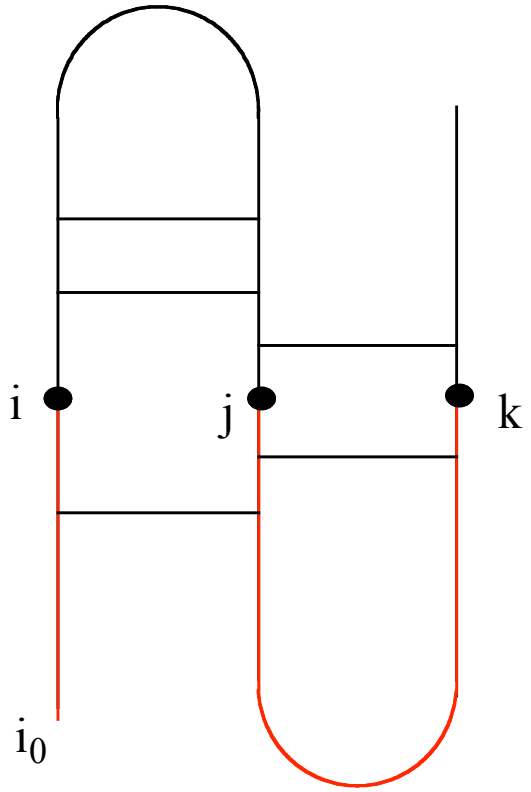
Sub-structure for a simple pseudo-knot

sub-pseudoknot P(i, j, k) as

the union of two subintervals

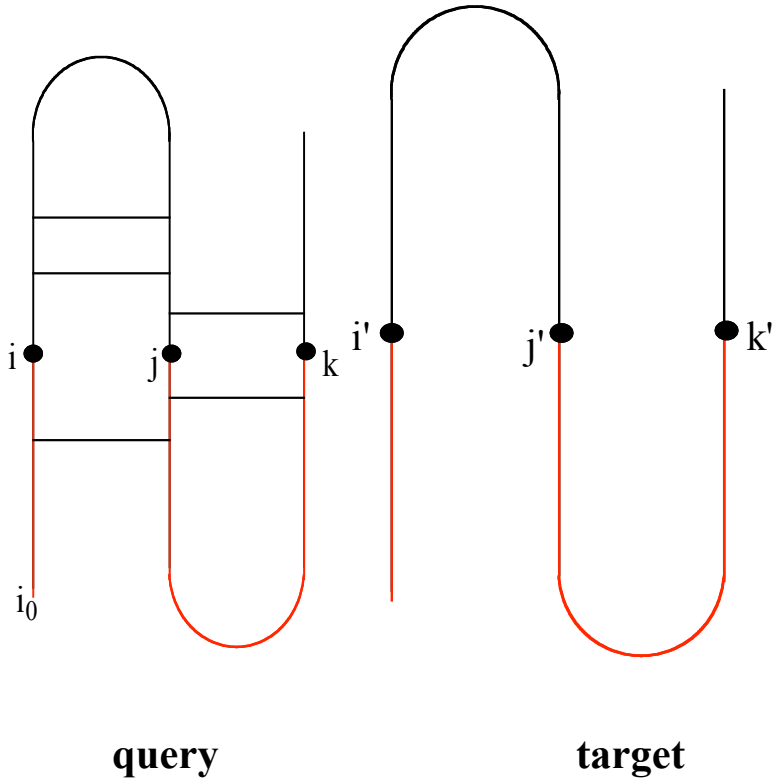
$P(i, j, k) = [i_0, i] \cup [j, k]$

frontier (i.j.k)



Naive approach

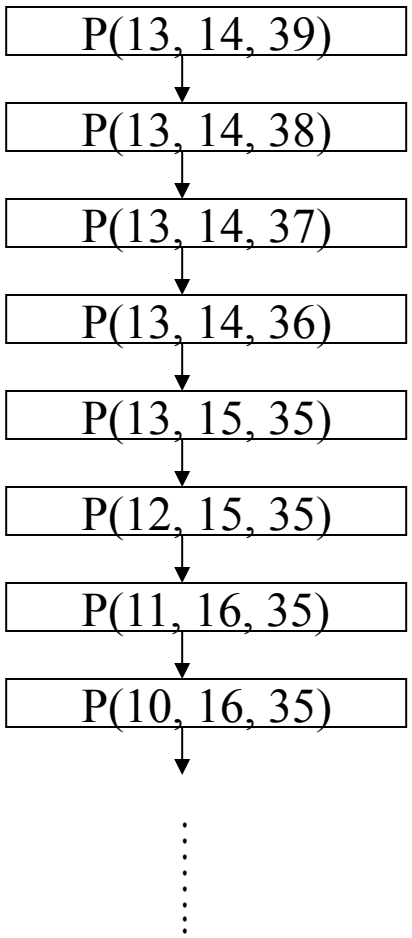
B[i, j, k, i', j' , k']: Optimal score of the alignment of the sub-pseudoknot P' (i' ,j' ,k') in target to sub-pseudoknot P(i,j,k) in query.

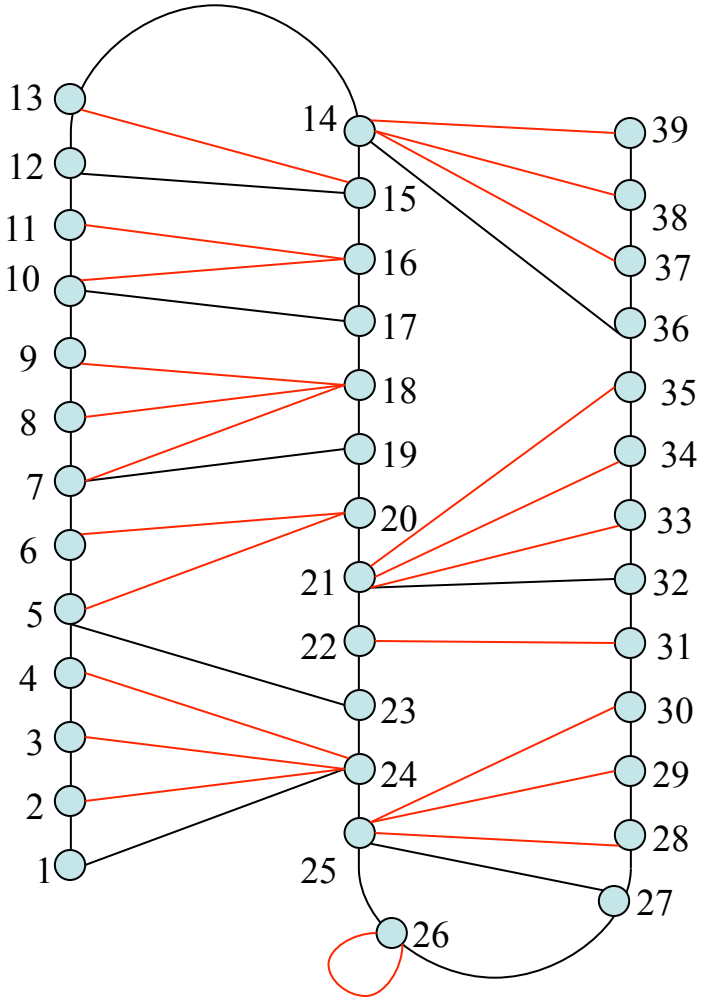


Compute B[i, j, k, i', j' , k']
$\Rightarrow O(m^3n^3)$ scores.
(m:query, n:target)

Instead of all triplets in the query, consider only the valid sub-pseudo-knots that will represent the simple pseudo-knot.

Use a chain of sub-pseudoknots to represent simple pseudo-knot

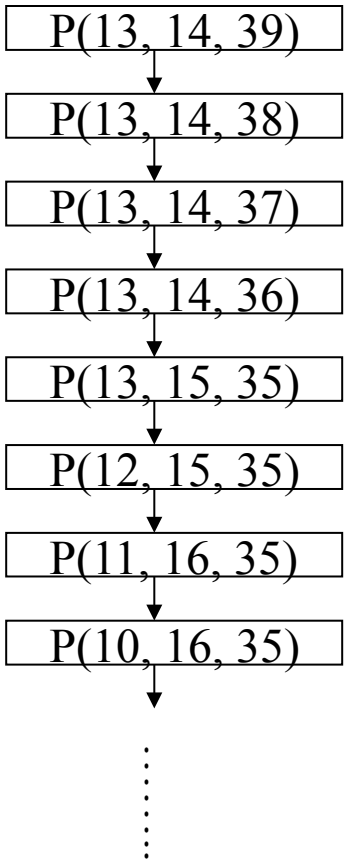


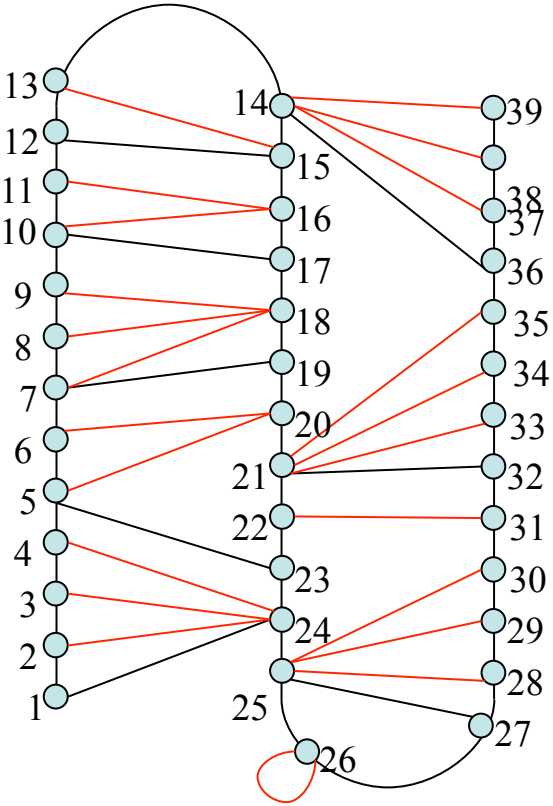


Why Chaining?

- DP: use sub-optimal solution of the child sub-structure to compute optimal score at each step.

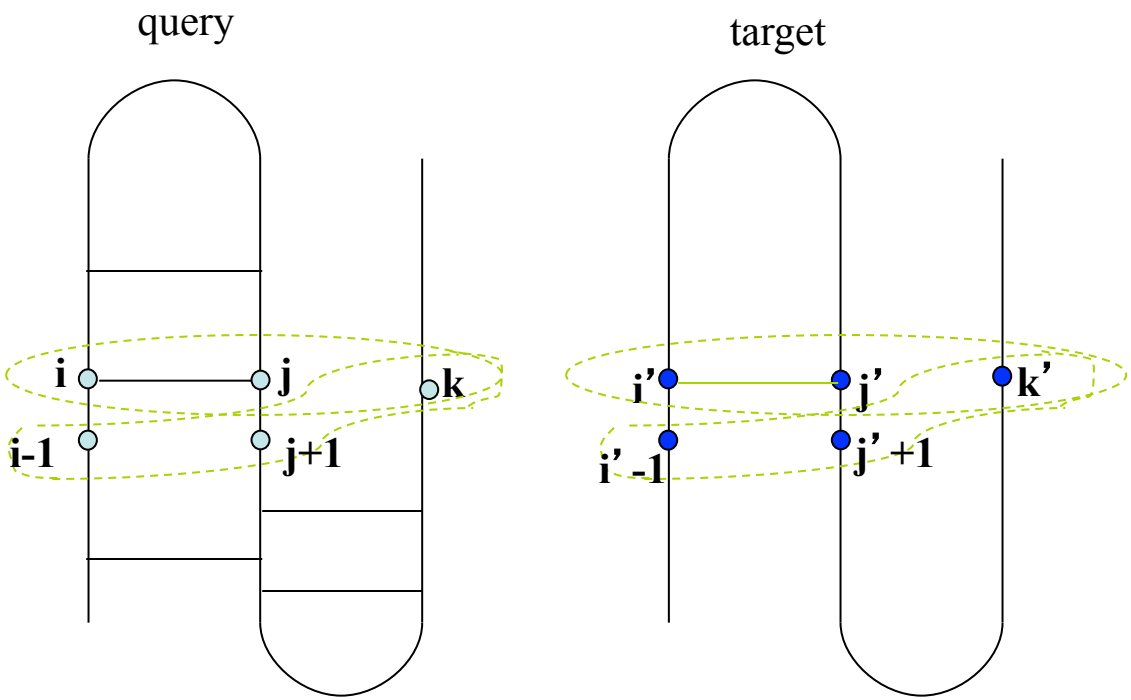
- compute B[i,j,k, i',j', k']
- => $O(mn^3)$ scores
- (m:query, n:target)





Alignment Algorithm Recursions: (i,j) is a base pair case

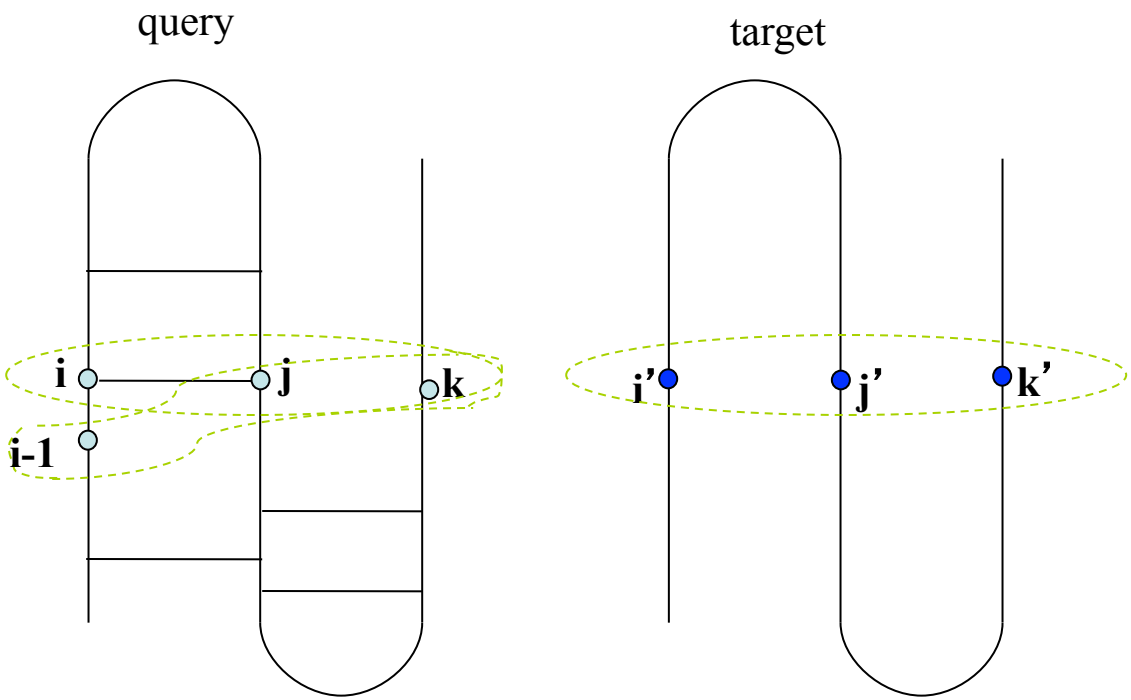
MATCH: (i,j) and (i’ ,j’) are corresponding pairs



- B[i, j, k , i’ , j’ , k’] = max {MATCH, INSERT, DELETE}

Alignment Algorithm Recursions: (i,j) is a base pair case

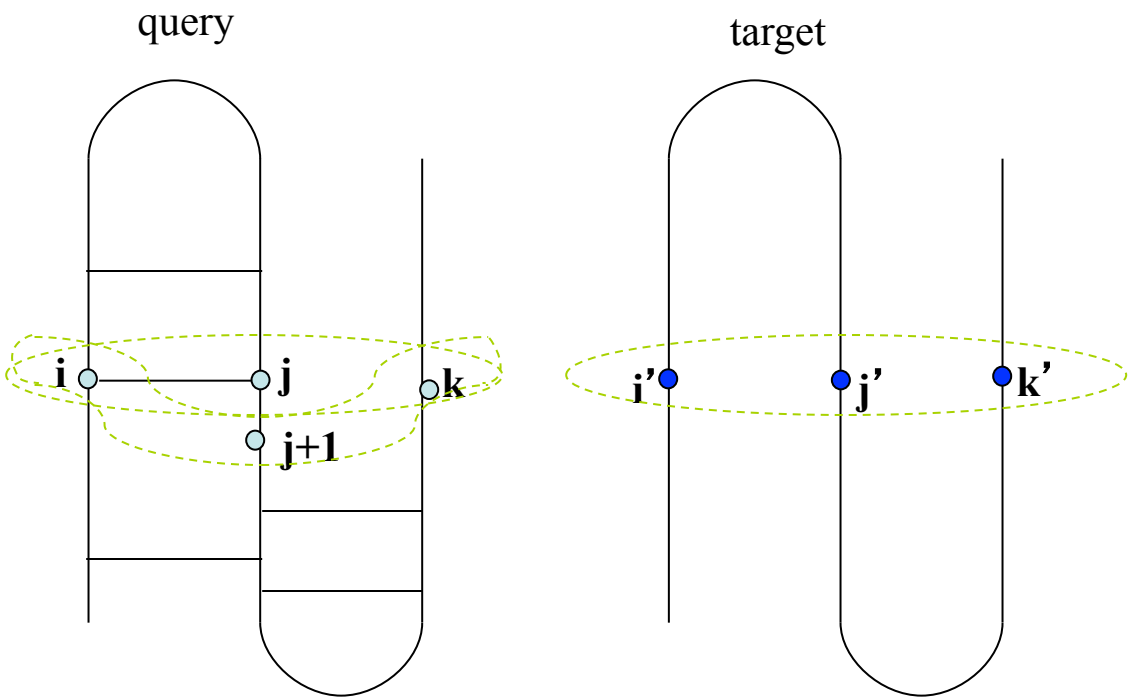
DELETION: i is deleted



- B[i, j, k , i' , j' , k'] = max {MATCH, INSERT, DELETE}

Alignment Algorithm Recursions: (i,j) is a base pair case

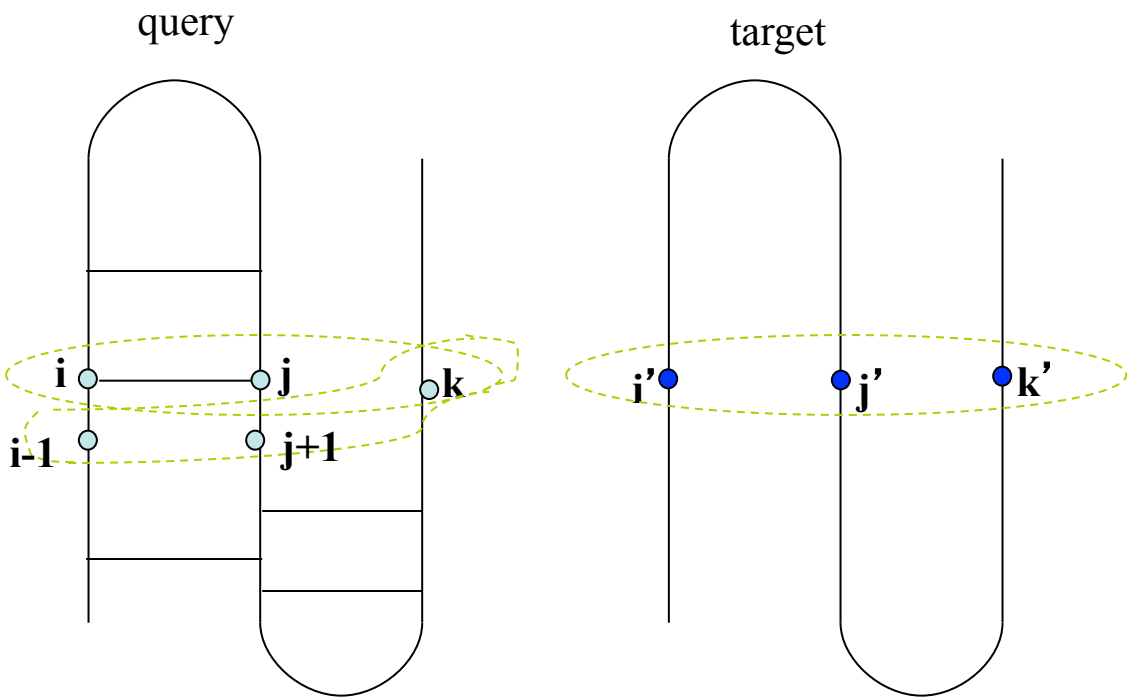
DELETION: j is deleted



- B[i, j, k , i' , j' , k'] = max {MATCH, INSERT, DELETE}

Alignment Algorithm Recursions: (i,j) is a base pair case

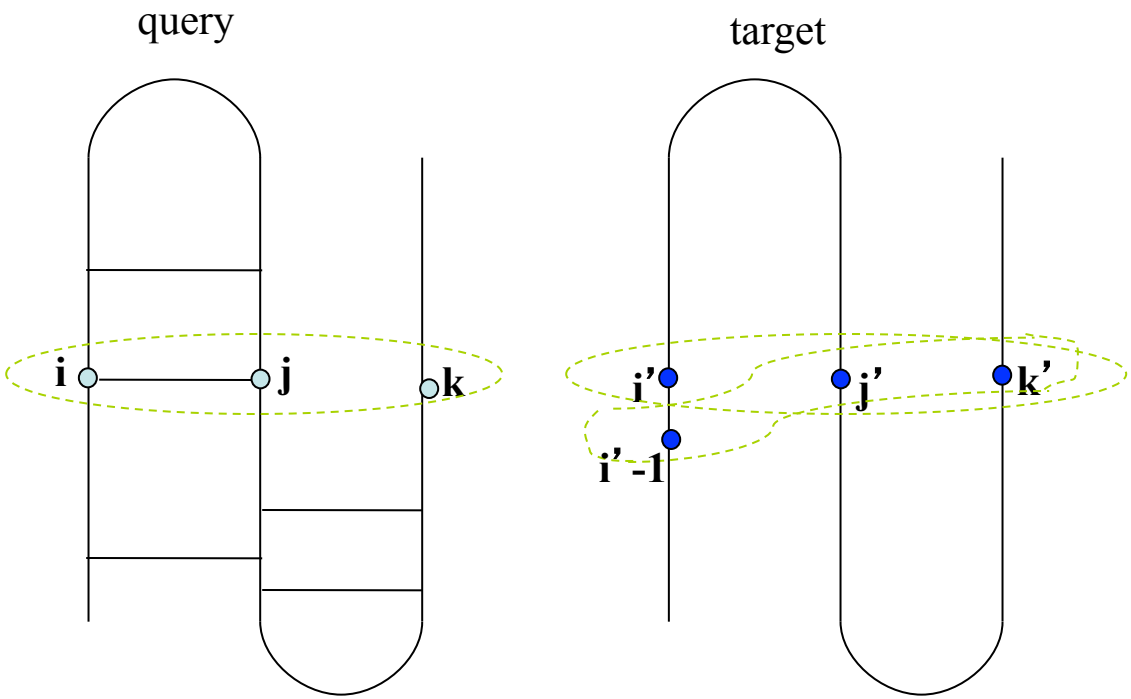
DELETION: i and j are deleted



- B[i, j, k , i' , j' , k'] = max {MATCH, INSERT, DELETE}

Alignment Algorithm Recursions: (i,j) is a base pair case

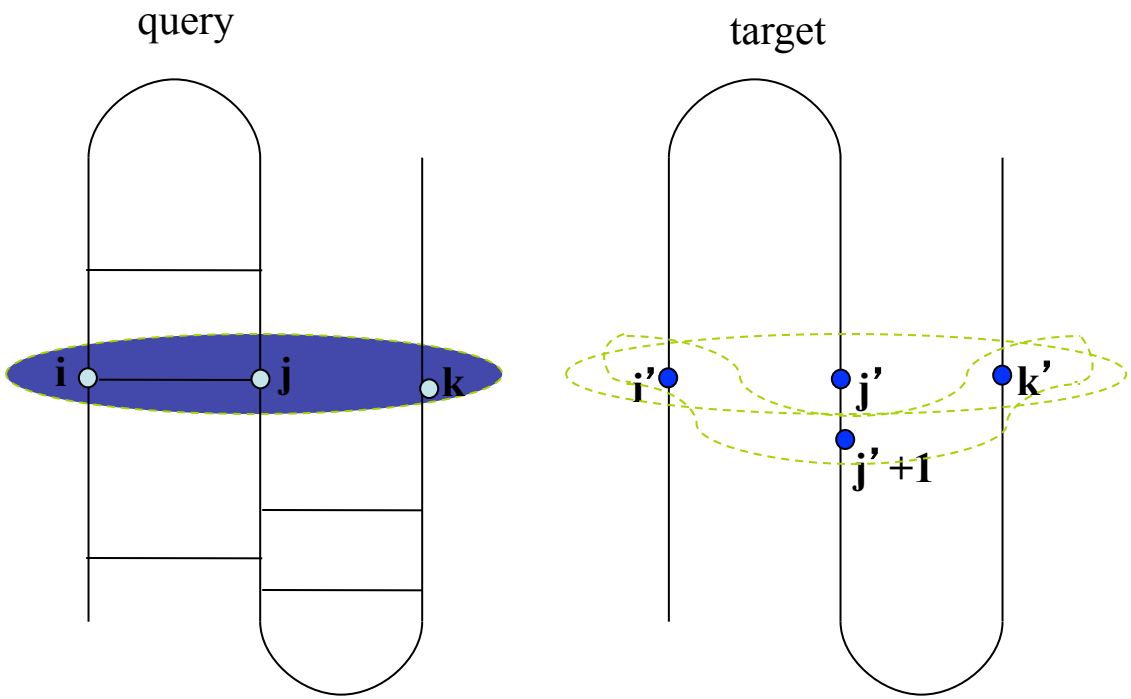
INSERTION: i’ is inserted



- B[i, j, k , i’ , j’ , k’] = max {MATCH, INSERT, DELETE}

Alignment Algorithm Recursions: (i,j) is a base pair case

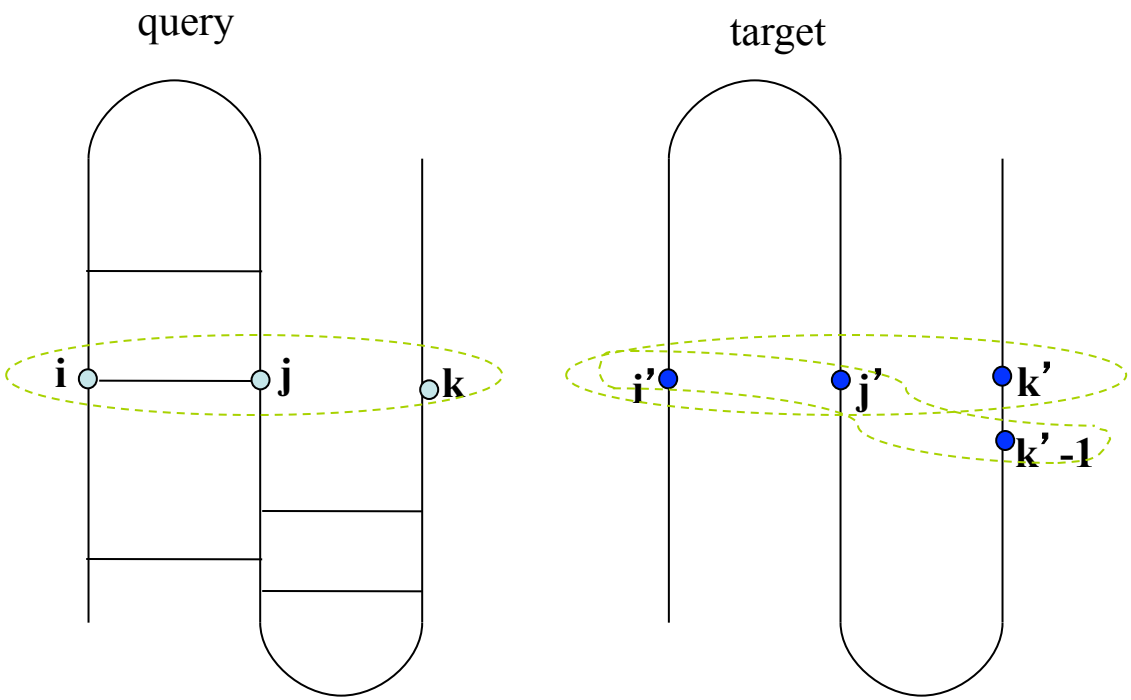
INSERTION: j’ is inserted



- B[i, j, k , i’ , j’ , k’] = max {MATCH, INSERT, DELETE}

Alignment Algorithm Recursions: (i,j) is a base pair case

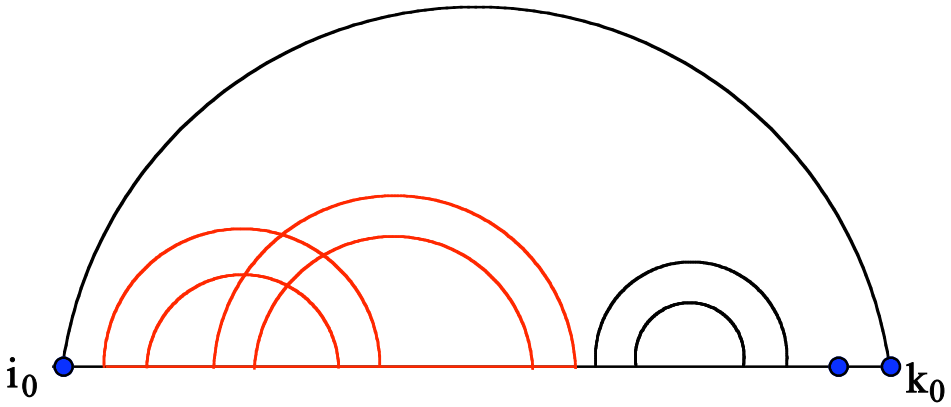
INSERTION: k’ is inserted

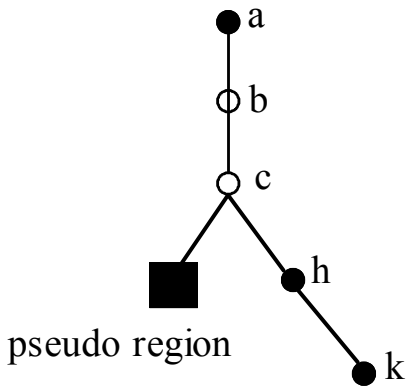


- B[i, j, k , i’ , j’ , k’] = max {MATCH, INSERT, DELETE}

Simple Pseudo-knot in a Regular Structure: S in R

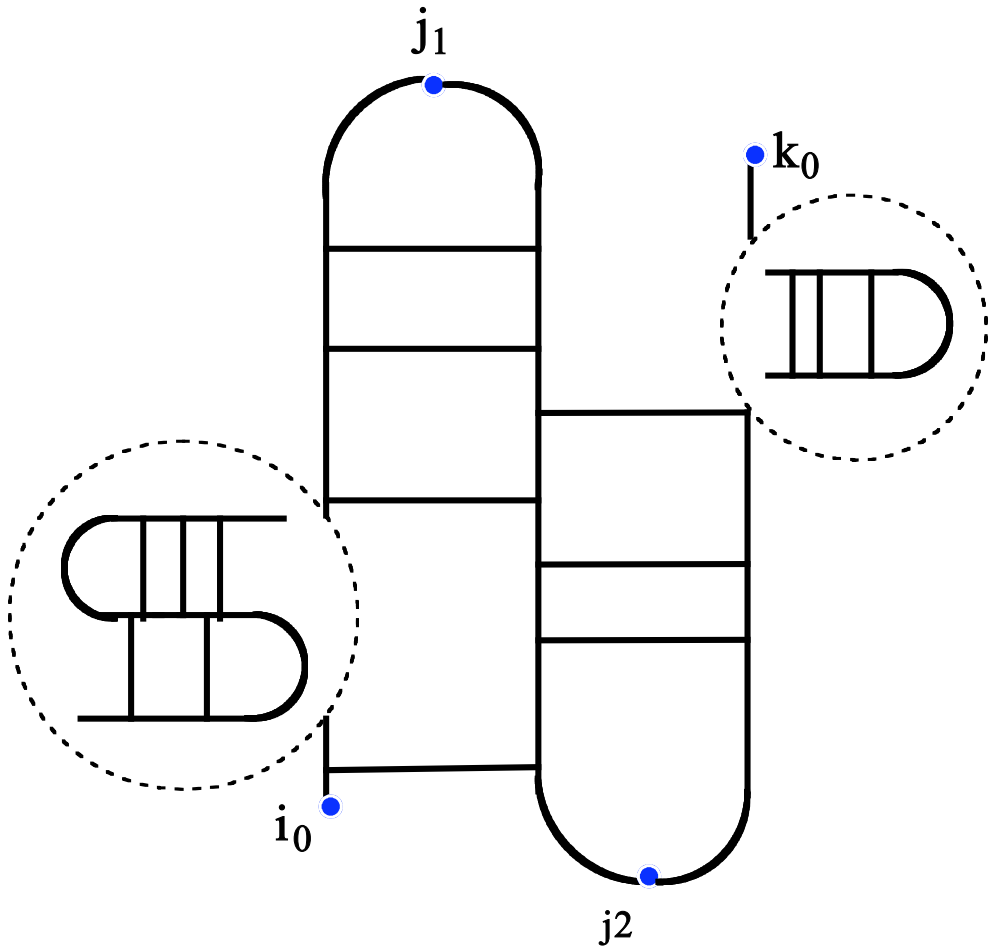
Use a binary tree to represent RNA
Solid circular nodes correspond to the actual base pairs.
Empty circular nodes correspond to unpaired bases.
Rectangular node correspond to sub-tree representing pseudo-knotted region





Simple pseudo-knot in a simple pseudo-knot: recursive simple pseudo-knot

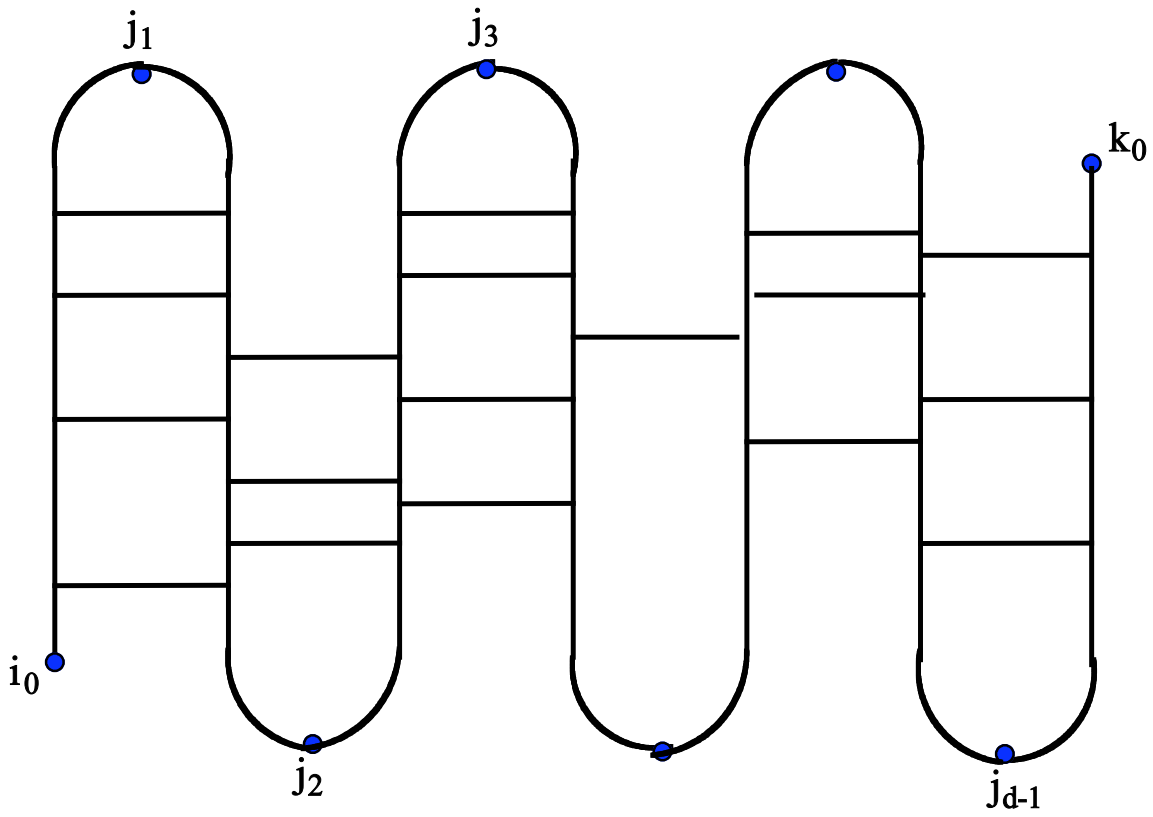
- S in S
- R in S



Which structures can we handle?

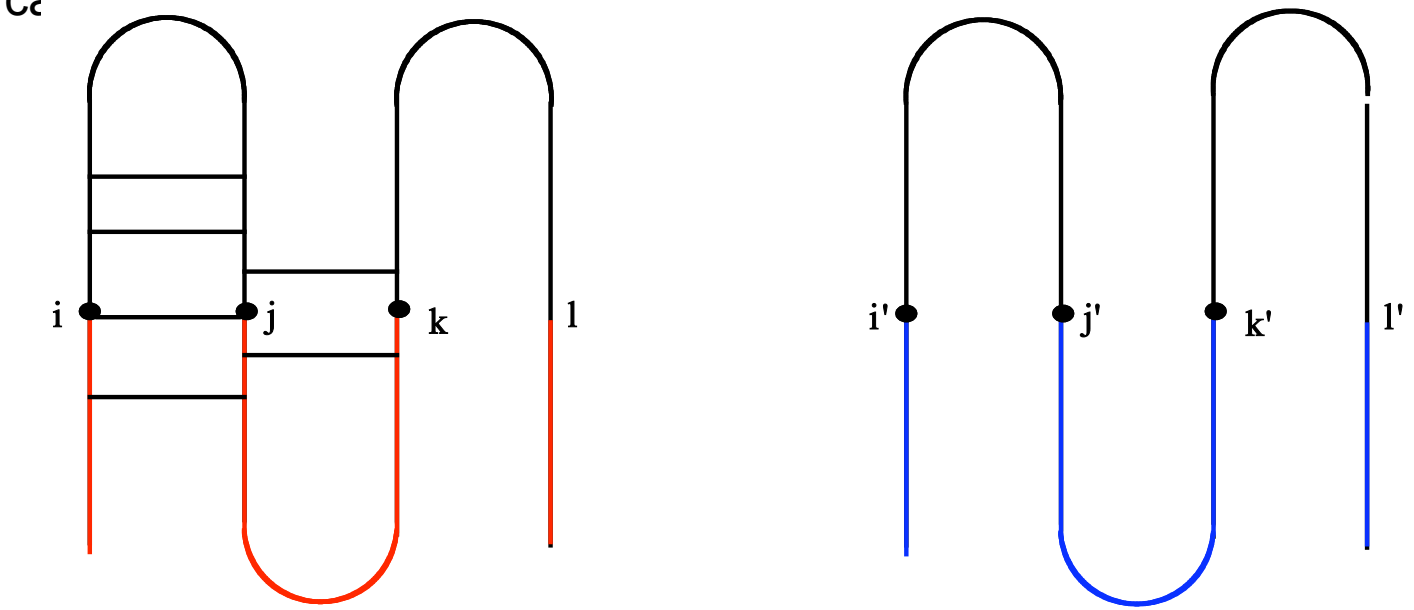
- Time complexity increases with the recursion depth of the pseudo-knotted region!
-
- R: regular structure
- S: simple pseudo-knot
-
- R: $O(mn^3)$
- S: $O(mn^4)$
- S in R: $O(mn^4)$
- R in S: $O(mn^5)$
- R in S in R: $O(mn^5)$ = S in S in R: $O(mn^5)$
- R in S in S in R = $O(mn^6)$
- ……..

Can we handle simple pseudo-knots with higher degree: standard pseudo-knots?



Can we handle simple pseudo-knots with higher degree: standard pseudoknots?

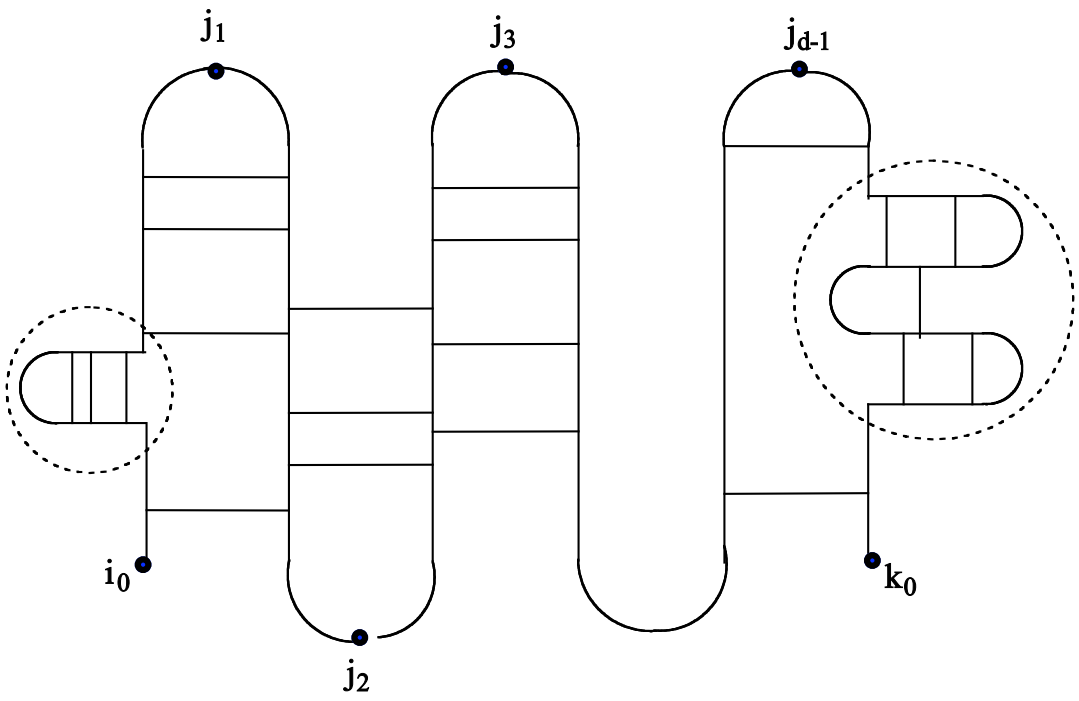
- Yes! By revising the sub-pseudoknot structure and the recursion cases accordingly



query target

Can we handle recursive standard pseudoknots?

Yes! Same reasoning with recursive simple pseudoknots.

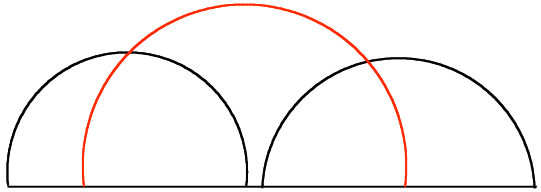


What is left? What can we NOT handle?

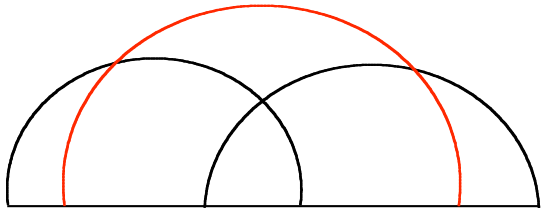
?We can handle the class of pseudoknots defined by Akutsu which is the second largest class currently defined. We can additionally handle standard and recursive standard pseudoknots which are defined by us.

A&U $\subseteq$ **A&U U {standard/recursive standard pseudoknots}** $\subseteq$ **R&E**

The largest class is defined by Rivas and Eddy. An example from this class we can not handle:



We can handle this!
(Standard pseudo-knot of degree 4)



We can NOT handle this!

Implementation: PAL

- C++ implementation of our algorithm.
  - input:
    - a query sequence with known structure (R/S/S in R)
    - a target sequence
  - output:
    - all high scoring local alignments in the target sequence

Testing

- Test Data:
- RFAM database, 6 RNA families with simple pseudo-knotted structures. (simple pseudo-knots in regular structure)

    - UPSK
    - Antizyme
    - Corona FSE
    - Corona pk3
    - Parecho CRE
    - IFN gamma

Test 1: Structure Prediction

- How good is PAL in inferring structure of the target sequence?
  - Pick 2 seed members of an RNA family as query and target.
  - Align them.
  - Compare the inferred structure of target with annotated structure in Rfam.

Test 1: Structure Prediction Results

- TP, FP, FN, Sensitivity, Specificity

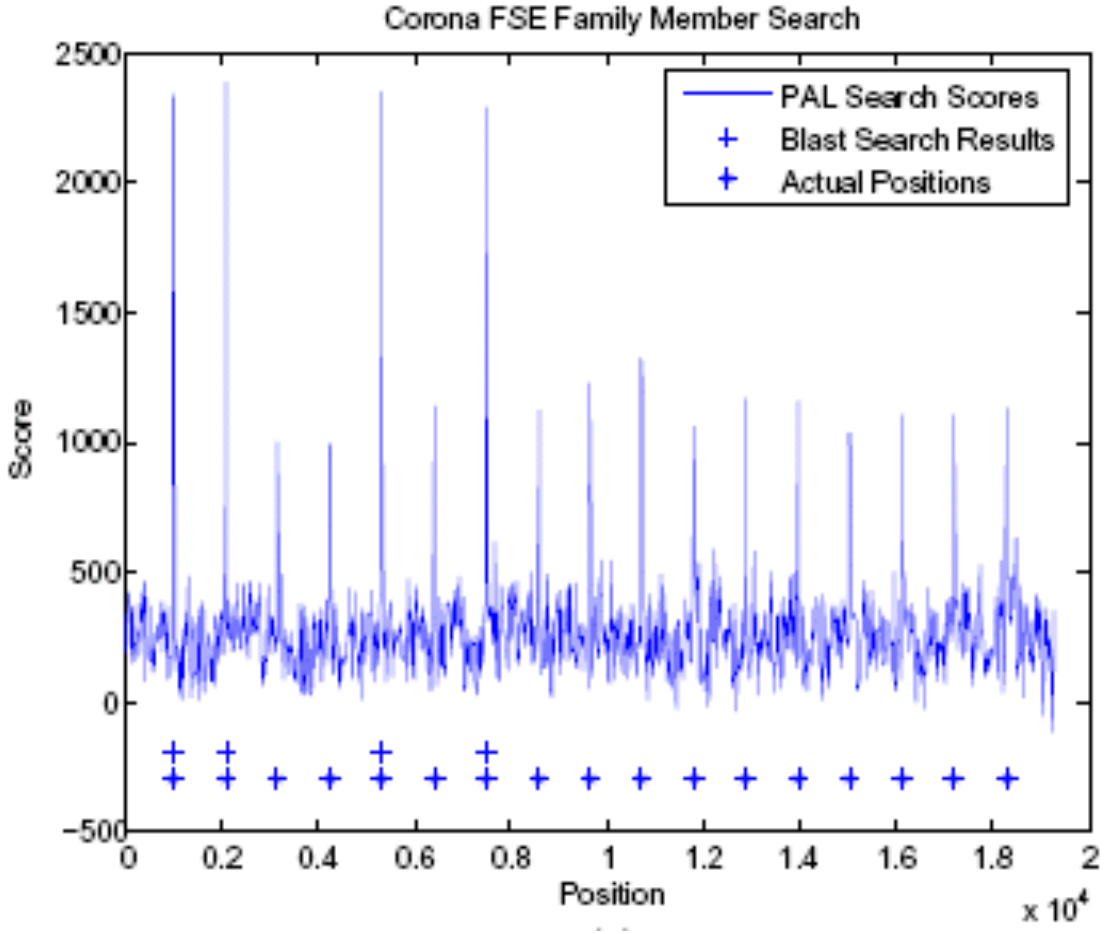
RNA Family	Specificity (Mean)	Sensitivity (Mean)
UPSK	1	1
Antizyme	0.99	0.99
Parecho	0.95	0.94
Corona FSE	0.94	0.94
Corona pk3	0.97	0.97
IFN Gamma	0.93	0.93

- Specificity = TP/(TP+FP)
- Sensitivity = TP/(TP+FN)
- Both measure is ~0.95
- PAL is a strong predictor of structure

Test 2: Homologue Search

- How well is PAL in finding the homologues of an RNA sequence?
  - Generate a random genome.
  - Insert the members of an RNA family.
  - Pick one of the members as a query.
  - Search for the homologues of the query.
  - Can we locate the members?

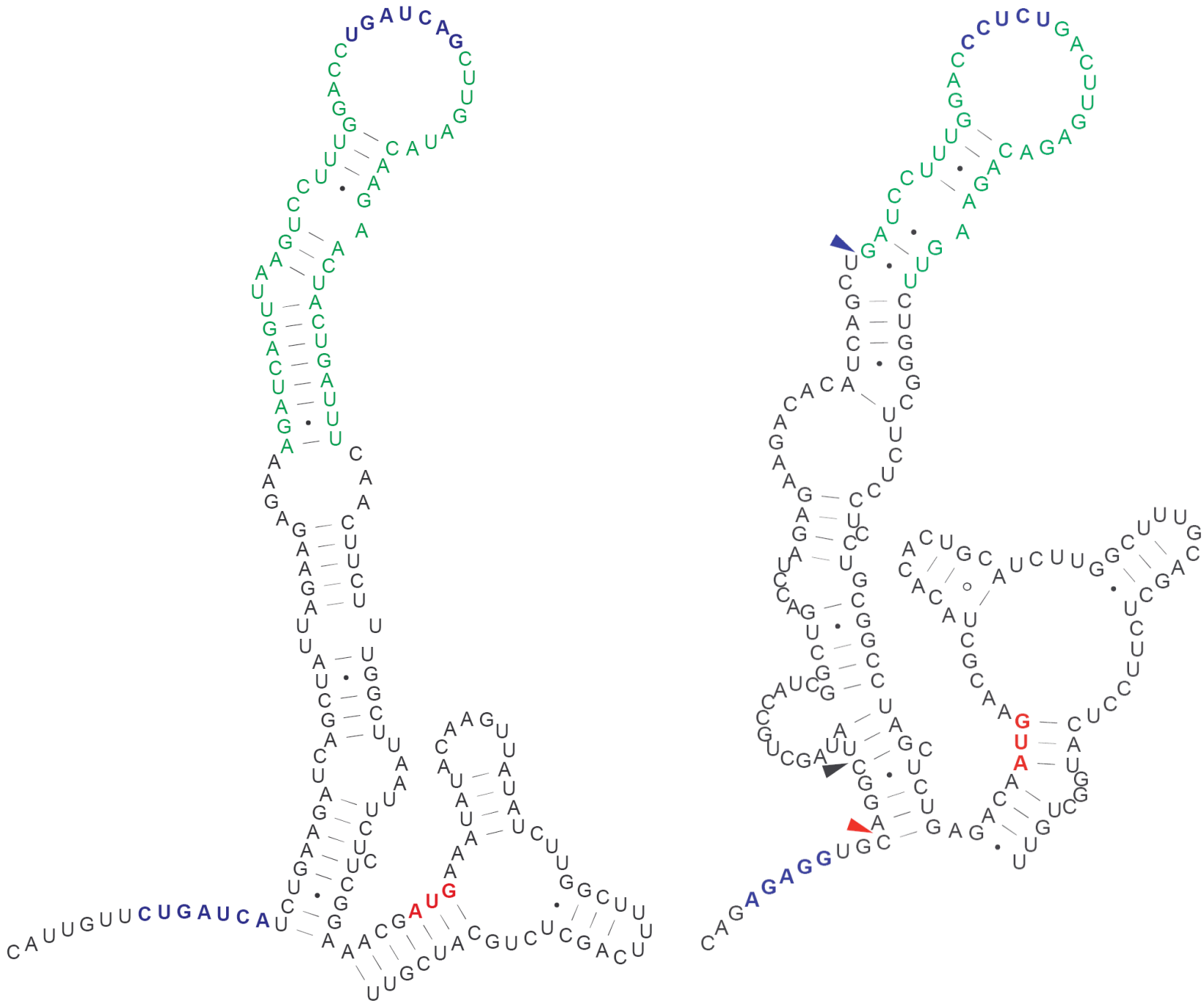
Test 2: Homologue Search Results



RNA Family	# Found	
	BLAST	PAL
UPSK	3	3
Antizyme	12	12
Parecho CRE	4	4
Corona-FSE	4	17
Corona-pk3	5	13
IFN-gamma	4	4

Novel Homolologues Search

- Searched mouse, rat and gerbil genomes for homologues of
- IFN-gamma RNA family.



Conclusion

- PAL is a viable tool in finding novel homologues and inferring structure.

- We hope PAL will help to understand and explore the impact of
- pseudo-knotted RNAs in cellular function.