A partition function algorithm for RNA-RNA interaction

Hamidreza Chitsaz

Raheleh Salari, Cenk Sahinalp, Rolf Backofen

Wayne State University

chitsaz@wayne.edu

Benasque RNA Meeting

July 27th, 2012

Mini biography

Robotics → RNA → Genome Assembly

- University of Illinois, Urbana-Champaign (Steven M. LaValle): PhD, Computer Science, 2008
- Simon Fraser University, Vancouver (Cenk Sahinalp): Postdoc, RNA algorithms, 2009
- University of California, San Diego (Pavel Pevzner): Postdoc, genome assembly, 2011
- Wayne State University, Detroit: Assistant professor, 2011-

Mini biography

Robotics → RNA → Genome Assembly

- University of Illinois, Urbana-Champaign (Steven M. LaValle): PhD, Computer Science, 2008
- Simon Fraser University, Vancouver (Cenk Sahinalp): Postdoc, RNA algorithms, 2009
- University of California, San Diego (Pavel Pevzner): Postdoc, genome assembly, 2011
- Wayne State University, Detroit: Assistant professor, 2011-

Mini biography

Robotics → RNA → Genome Assembly

- University of Illinois, Urbana-Champaign (Steven M. LaValle): PhD, Computer Science, 2008
- Simon Fraser University, Vancouver (Cenk Sahinalp): Postdoc, RNA algorithms, 2009
- University of California, San Diego (Pavel Pevzner): Postdoc, genome assembly, 2011
- Wayne State University, Detroit: Assistant professor, 2011-

Mini biography

Robotics → RNA → Genome Assembly

- University of Illinois, Urbana-Champaign (Steven M. LaValle): PhD, Computer Science, 2008
- Simon Fraser University, Vancouver (Cenk Sahinalp): Postdoc, RNA algorithms, 2009
- University of California, San Diego (Pavel Pevzner): Postdoc, genome assembly, 2011
- Wayne State University, Detroit: Assistant professor, 2011-

Single-cell bacterial genome and transcriptome assembly

nature biotechnology

ARTICLES

Efficient *de novo* assembly of single-cell bacterial genomes from short-read data sets

Hamidreza Chitsaz[1,6], Joyclyn L Yee-Greenbaum[2,6], Glenn Tesler[3], Mary-Jane Lombardo[2], Christopher L Dupont[2], Jonathan H Badger[2], Mark Novotny[2], Douglas B Rusch[4], Louise J Fraser[5], Niall A Gormley[5], Ole Schulz-Trieglaff[5], Geoffrey P Smith[5], Dirk J Evers[5], Pavel A Pevzner[1] & Roger S Lasken[2]

Whole genome amplification by the multiple displacement amplification (MDA) method allows sequencing of DNA from single cells of bacteria that cannot be cultured. Assembling a genome is challenging, however, because MDA generates highly

BIOINFORMATICS

Vol. 28 ISMB 2012, pages i188–i196
doi:10.1093/bioinformatics/bts219

SEQuel: improving the accuracy of genome assemblies

Roy Ronen[1,†], Christina Boucher[2,†], Hamidreza Chitsaz[3] and Pavel Pevzner[2,*]

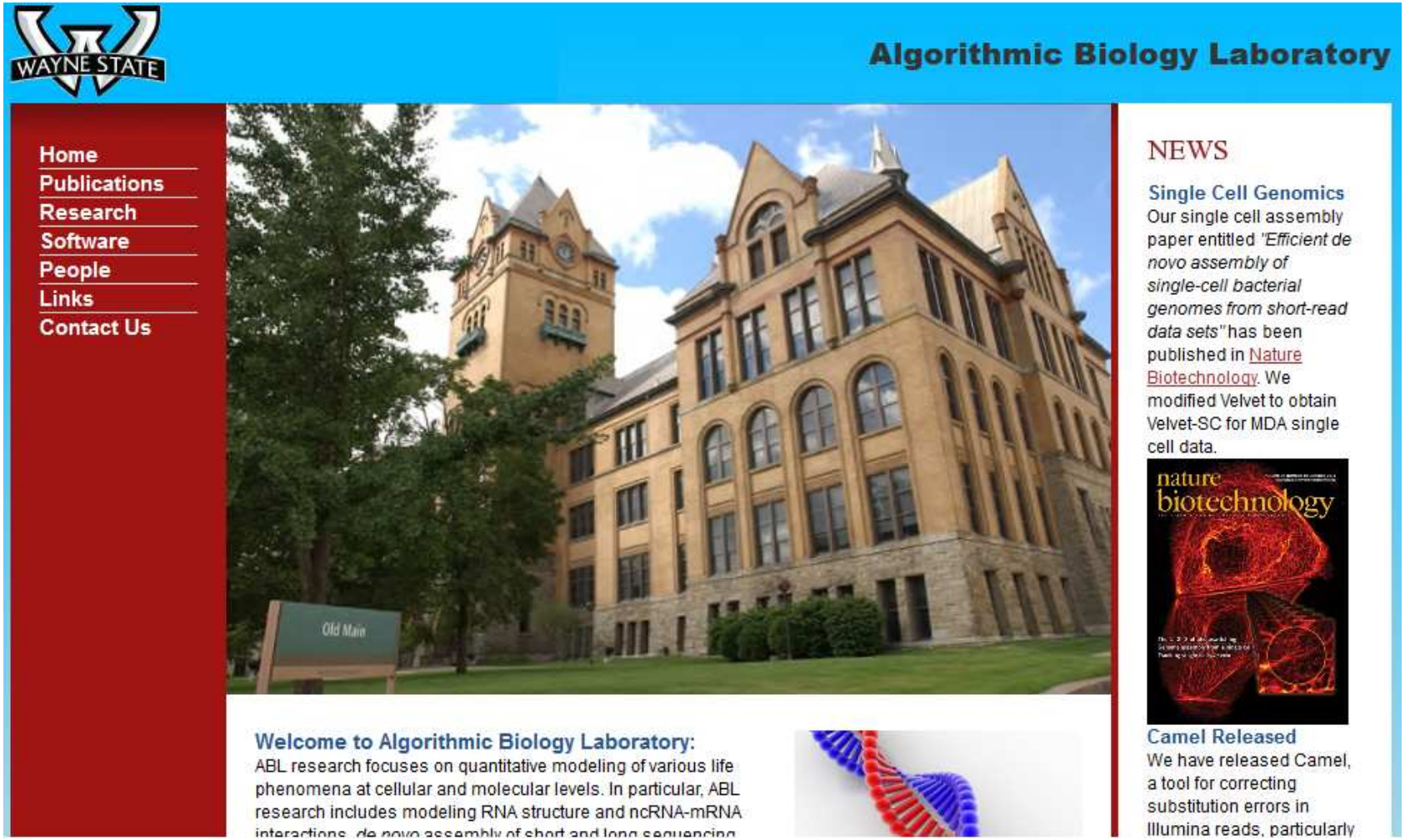
[1]Bioinformatics Graduate Program, [2]Department of Computer Science and Engineering, University of California, San Diego, La Jolla, CA 92093 and [3]Department of Computer Science, Wayne State University, Detroit, MI 48202, USA

ABSTRACT

Motivation: Assemblies of next-generation sequencing (NGS) data, finished genomes assembled using the previous technologies (Alkan, *et al.*, 2011). Earlier assembly algorithms developed for Sanger

Algorithmic Biology Laboratory

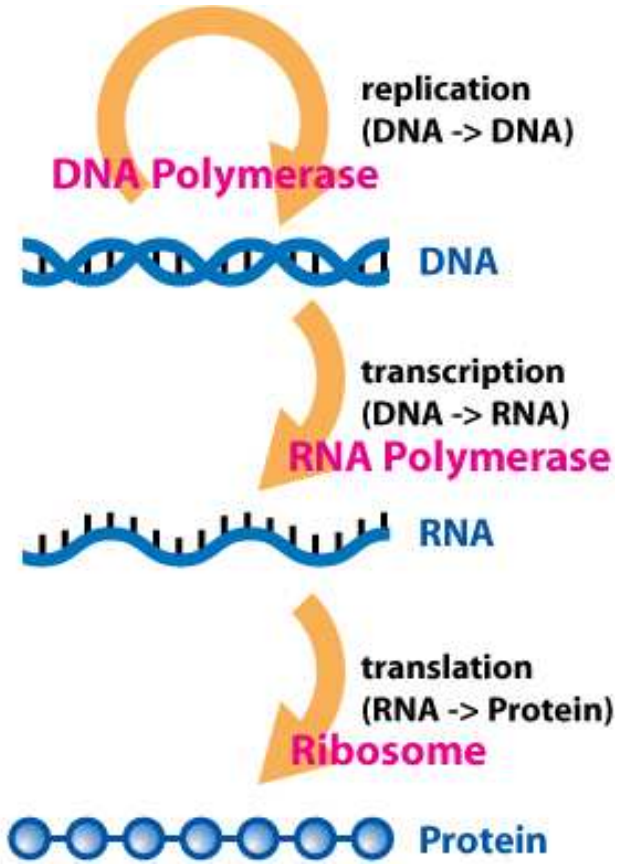
Wayne State University



http://compbio.cs.wayne.edu

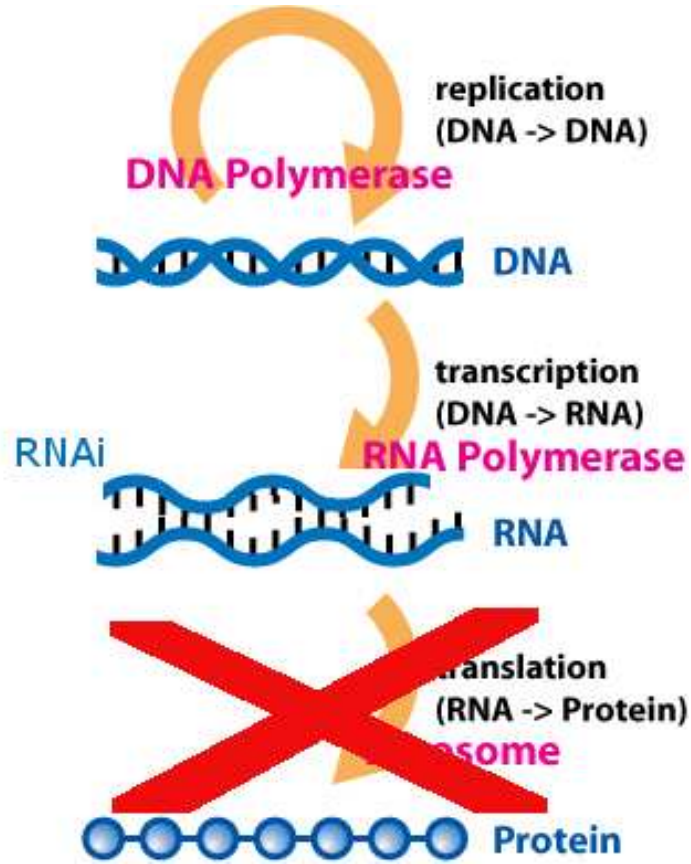
Central dogma

DNA $\rightarrow$ RNA $\rightarrow$ Protein



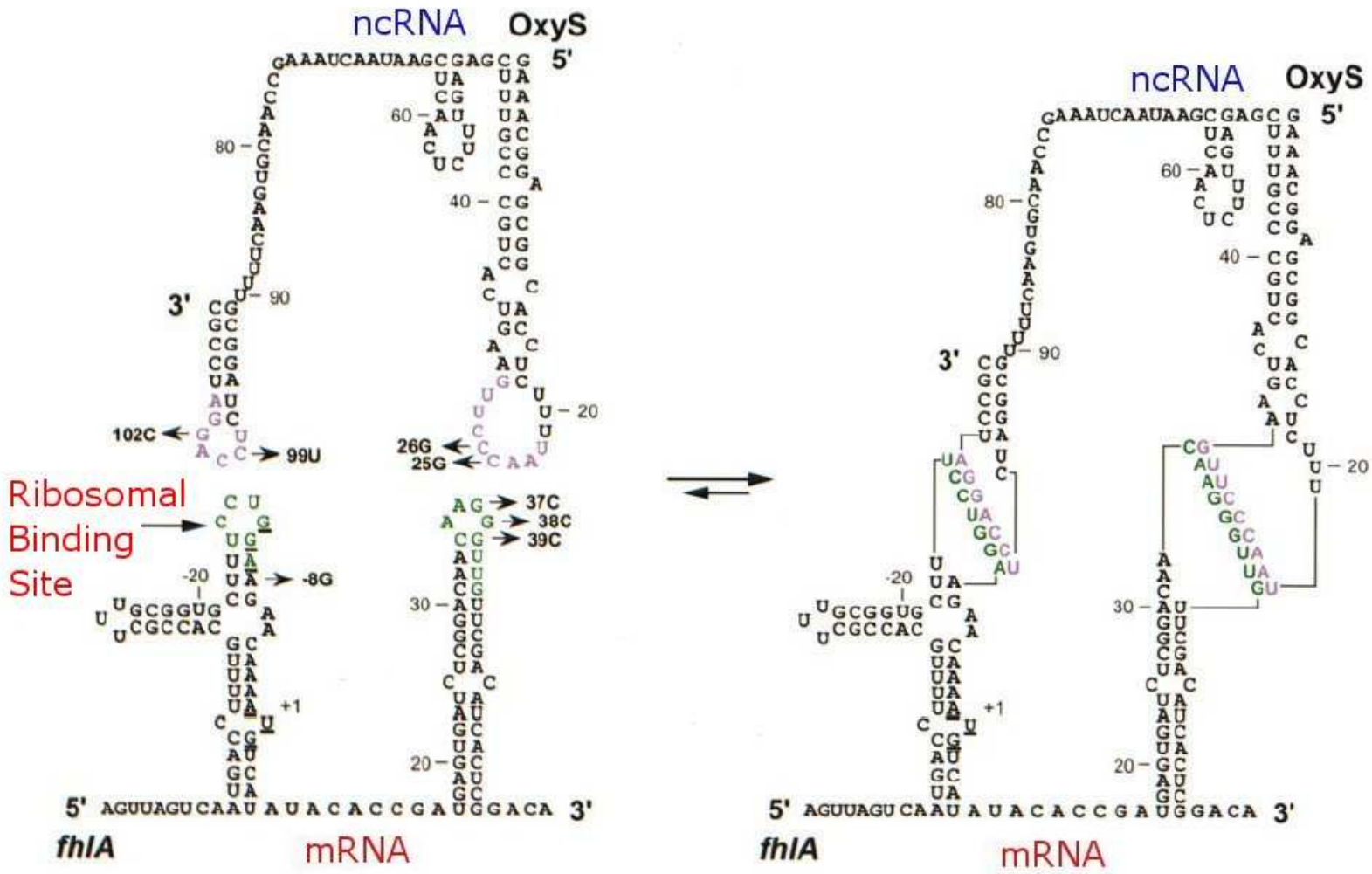
Motivation

Post-transcriptional regulation of gene expression



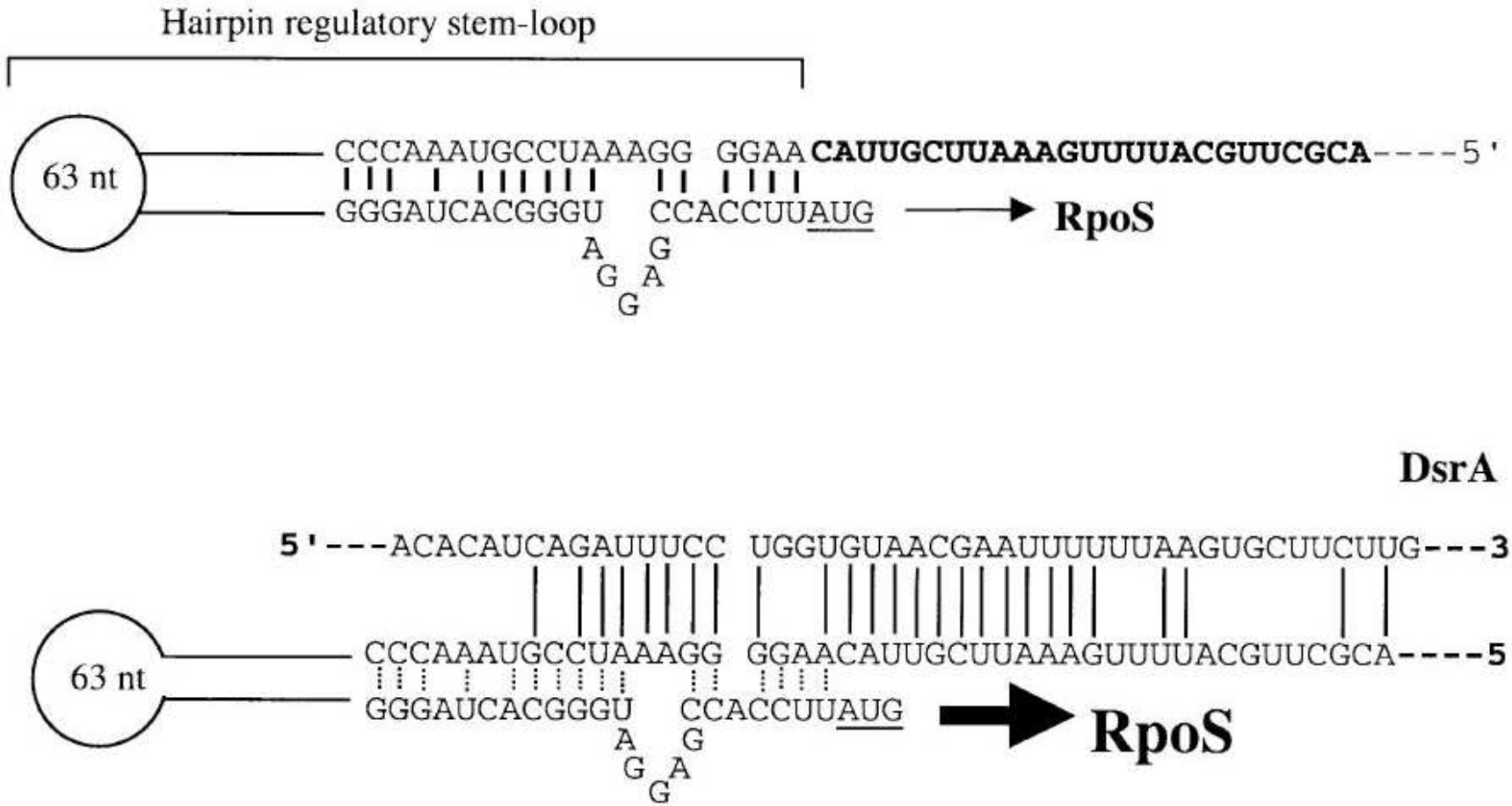
Regulatory RNA

Repression example (Argaman and Altuvia, J. Mol. Biol. 2000)



Regulatory RNA

Activation example (Repoila, Majdalani, and Gottesman, Mol. Microbiol. 2003)



Background

RNA-RNA MFE structure prediction

- Avoid intramolecular base pairing
  RNAhybrid (Rehmsmeier *et al.* 2004), **RNAduplex** (Bernhart *et al.* 2006), **UNAFold** (Markham *et al.* 2008)
  No internal structure
- Concatenate input sequences as a single strand; no pseudoknots
  PairFold (Andronescu *et al.* 2005), **RNAcofold** (Bernhart *et al.* 2006)
  No kissing hairpins
- Predict binding sites
  RNAup (Mückstein *et al.* 2008), **intaRNA** (Busch *et al.* 2008)
  Just one binding site not complete structure
- Concatenate input sequences; consider special pseudoknots
  NUPACK (Dirks *et al.* 2003,2007)

  Still no kissing hairpins!

Background

RNA-RNA MFE structure prediction

- Avoid intramolecular base pairing
  RNAhybrid (Rehmsmeier *et al.* 2004), **RNAduplex** (Bernhart *et al.* 2006), **UNAFold** (Markham *et al.* 2008)
  No internal structure
- Concatenate input sequences as a single strand; no pseudoknots
  PairFold (Andronescu *et al.* 2005), **RNAcofold** (Bernhart *et al.* 2006)
  No kissing hairpins
- Predict binding sites
  RNAup (Mückstein *et al.* 2008), **intaRNA** (Busch *et al.* 2008)
  Just one binding site not complete structure
- Concatenate input sequences; consider special pseudoknots
  NUPACK (Dirks *et al.* 2003,2007)

  Still no kissing hairpins!

Background

RNA-RNA MFE structure prediction

- Avoid intramolecular base pairing
  RNAhybrid (Rehmsmeier *et al.* 2004), **RNAduplex** (Bernhart *et al.* 2006), **UNAFold** (Markham *et al.* 2008)
  No internal structure
- Concatenate input sequences as a single strand; no pseudoknots
  PairFold (Andronescu *et al.* 2005), **RNAcofold** (Bernhart *et al.* 2006)
  No kissing hairpins
- Predict binding sites
  RNAup (Mückstein *et al.* 2008), **intaRNA** (Busch *et al.* 2008)
  Just one binding site not complete structure
- Concatenate input sequences; consider special pseudoknots
  NUPACK (Dirks *et al.* 2003,2007)

  Still no kissing hairpins!

Background

RNA-RNA MFE structure prediction

- Avoid intramolecular base pairing
  RNAhybrid (Rehmsmeier *et al.* 2004), **RNAduplex** (Bernhart *et al.* 2006), **UNAFold** (Markham *et al.* 2008)
  No internal structure
- Concatenate input sequences as a single strand; no pseudoknots
  PairFold (Andronescu *et al.* 2005), **RNAcofold** (Bernhart *et al.* 2006)
  No kissing hairpins
- Predict binding sites
  RNAup (Mückstein *et al.* 2008), **intaRNA** (Busch *et al.* 2008)
  Just one binding site not complete structure
- Concatenate input sequences; consider special pseudoknots
  NUPACK (Dirks *et al.* 2003,2007)

  Still no kissing hairpins!

Background (continued)

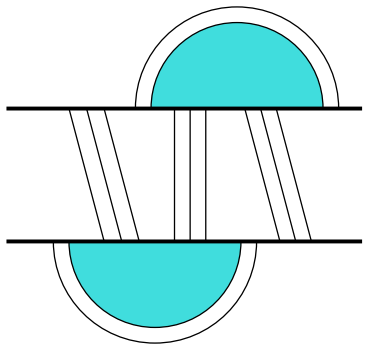
RNA-RNA MFE structure prediction

Consider inter- and intramolecular base pairing

IRIS (Pervouchine 2004), **inteRNA** (Alkan *et al.* 2005), **Grammatical Approach** (Kato *et al.* 2009)

Voilà, now we are talking business.

The problem is NP-Hard (Alkan *et al.* 2005); no surprise as pseudoknots are NP-Hard. Exclude *zigzags* and crossing interactions to lift the curse of complexity and obtain an exact $O(n^6)$-time $O(n^4)$-space DP algorithm (albeit for simple base-pair counting).



First order zigzag. A general zigzag involves an arbitrary number of kissing hairpins.

Ahhh...but MFE is often wrong!

Question: how about

1. computing base pairing probabilities,
2. sampling from the Boltzmann ensemble of interaction structures, clustering, centroids, etc.,
3. and computing equilibrium concentrations and melting temperature for RNA-RNA compounds?

Answer: the key enabling technology is the **partition function**. All of the above can be computed from the partition function.

Ahhh...but MFE is often wrong!

Question: how about

1. computing base pairing probabilities,
2. sampling from the Boltzmann ensemble of interaction structures, clustering, centroids, etc.,
3. and computing equilibrium concentrations and melting temperature for RNA-RNA compounds?

Answer: the key enabling technology is the **partition function**. All of the above can be computed from the partition function.

Ahhh...but MFE is often wrong!

Question: how about

1. computing base pairing probabilities,
2. sampling from the Boltzmann ensemble of interaction structures, clustering, centroids, etc.,
3. and computing equilibrium concentrations and melting temperature for RNA-RNA compounds?

Answer: the key enabling technology is the **partition function**. All of the above can be computed from the partition function.

Ahhh...but MFE is often wrong!

Question: how about

1. computing base pairing probabilities,
2. sampling from the Boltzmann ensemble of interaction structures, clustering, centroids, etc.,
3. and computing equilibrium concentrations and melting temperature for RNA-RNA compounds?

Answer: the key enabling technology is the **partition function**. All of the above can be computed from the partition function.

Partition function

$$Q(T) = \sum_{s \in S} e^{-G_s/RT},$$

$S =$ All considered interaction structures,

$$p(s) \propto e^{-G_s/RT},$$

and Q is the normalizing factor. Also other thermodynamic quantities can be derived from Q.

Partition function

$$Q(T) = \sum_{s \in S} e^{-G_s/RT},$$

$$S = \text{All considered interaction structures},$$

$$p(s) \propto e^{-G_s/RT},$$

and Q is the normalizing factor. Also other thermodynamic quantities can be derived from Q.

Partition function hardness $\geq$ MFE hardness

Partition function

$$\sum_{s \in S} e^{-G_s/RT}.$$

MFE secondary structure

$$\text{argmin}_{s \in S} G_s.$$

Transform a partition function algorithm to an MFE algorithm by

$$e^{-G_s} \to G_s$$

$$\times \to +$$

$$\sum \to \min$$

Partition function hardness $\geq$ MFE hardness

Partition function

$$\sum_{s \in S} e^{-G_s/RT}.$$

MFE secondary structure

$$\text{argmin}_{s \in S} G_s.$$

Transform a partition function algorithm to an MFE algorithm by

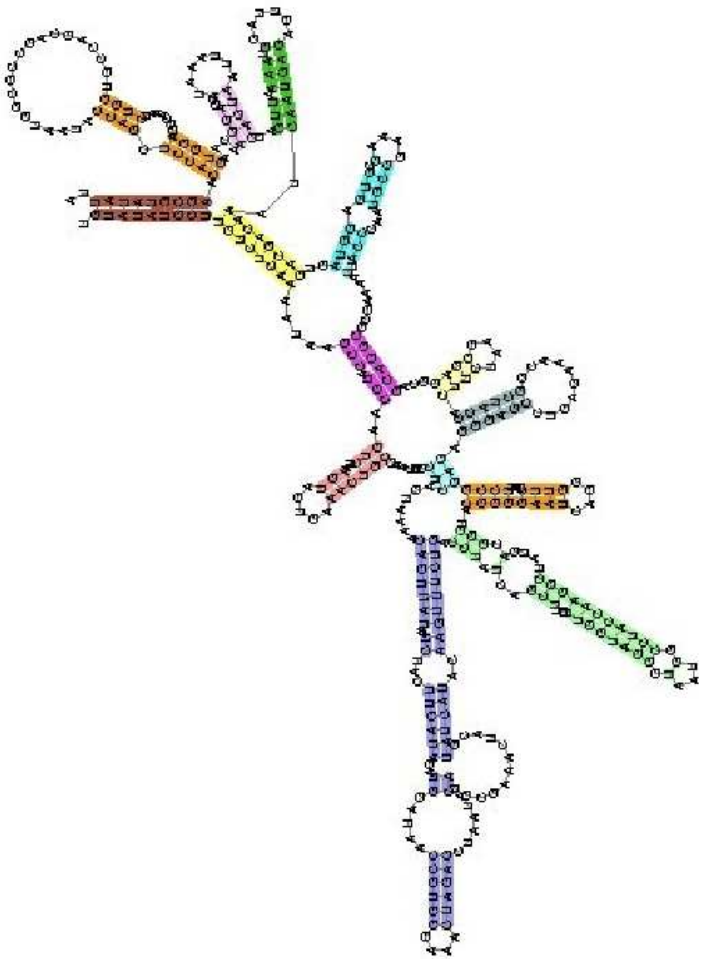
$$e^{-G_s} \rightarrow G_s$$

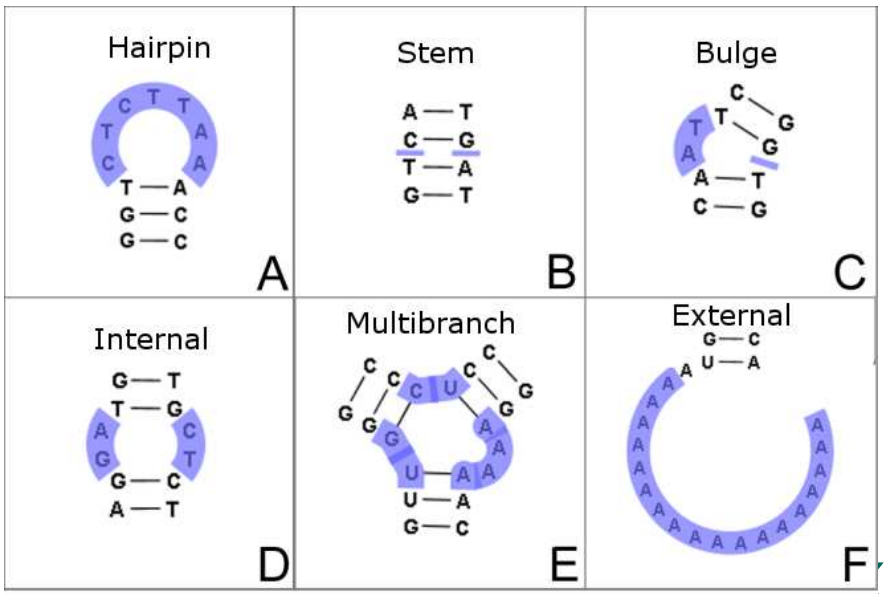
$$\times \rightarrow +$$

$$\sum \rightarrow \min$$

Turner energy model

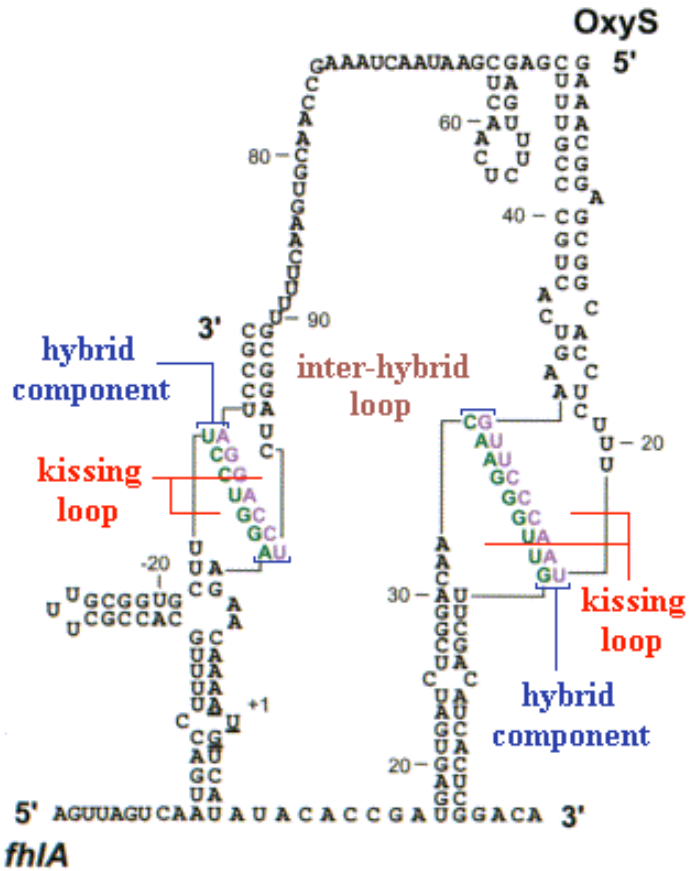
Mathews *et al.* 1999





Our extension of the Turner model

Chitsaz *et al.*, Bioinformatics 25(12): i365-i373



Hybrid component: as if intramolecular, with penalties.

Kissing loop: like multibranch loop.

Interaction partition function

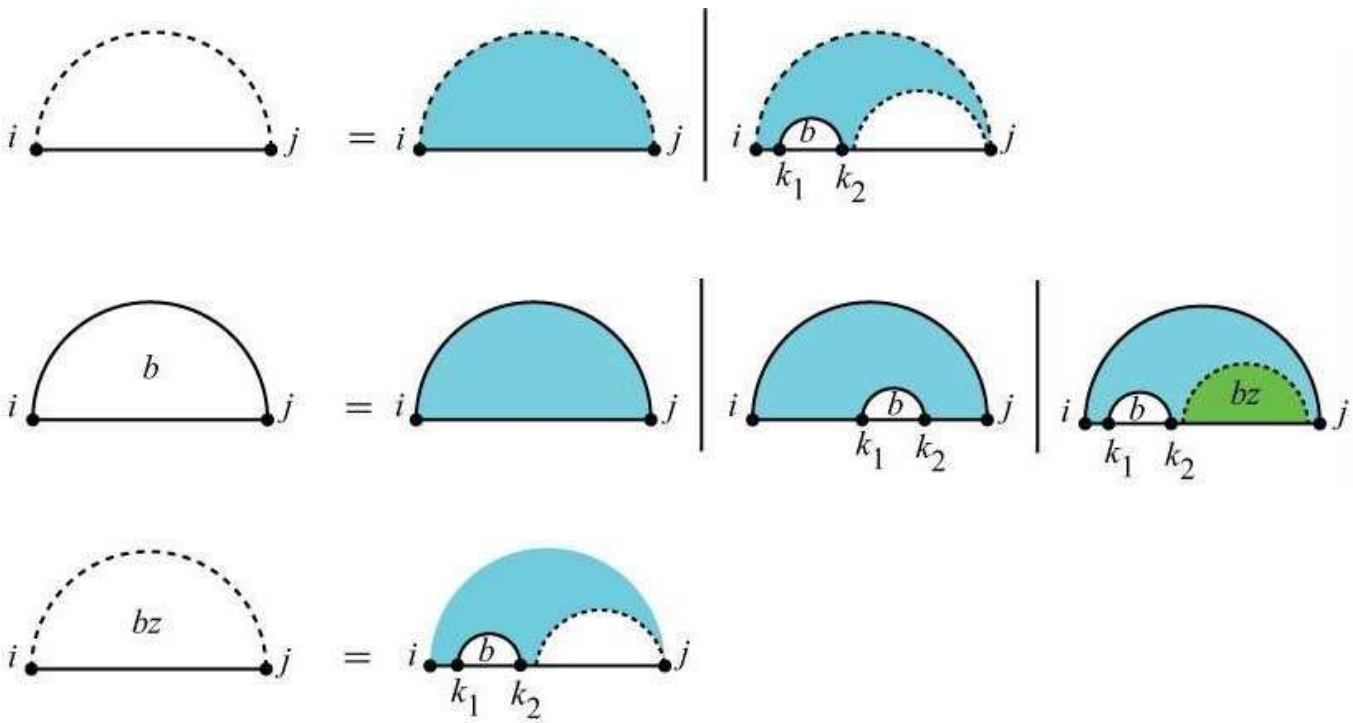
How?

Divide and conquer using dynamic programming:

$$
\begin{aligned}
Q(T) &= \sum_{s \in S} e^{-G_s/RT} \\
&= \sum_{s = s_a \cup s_b} e^{-(G_{sa} + G_{s_b})/RT} \\
&= [\sum_{s_a \in S_a} e^{-G_{sa}/RT}][\sum_{s_b \in S_b} e^{-G_{s_b}/RT}] \\
&= Q_a(T) Q_b(T).
\end{aligned}
$$

Partition function for single strand (McCaskill 1990)

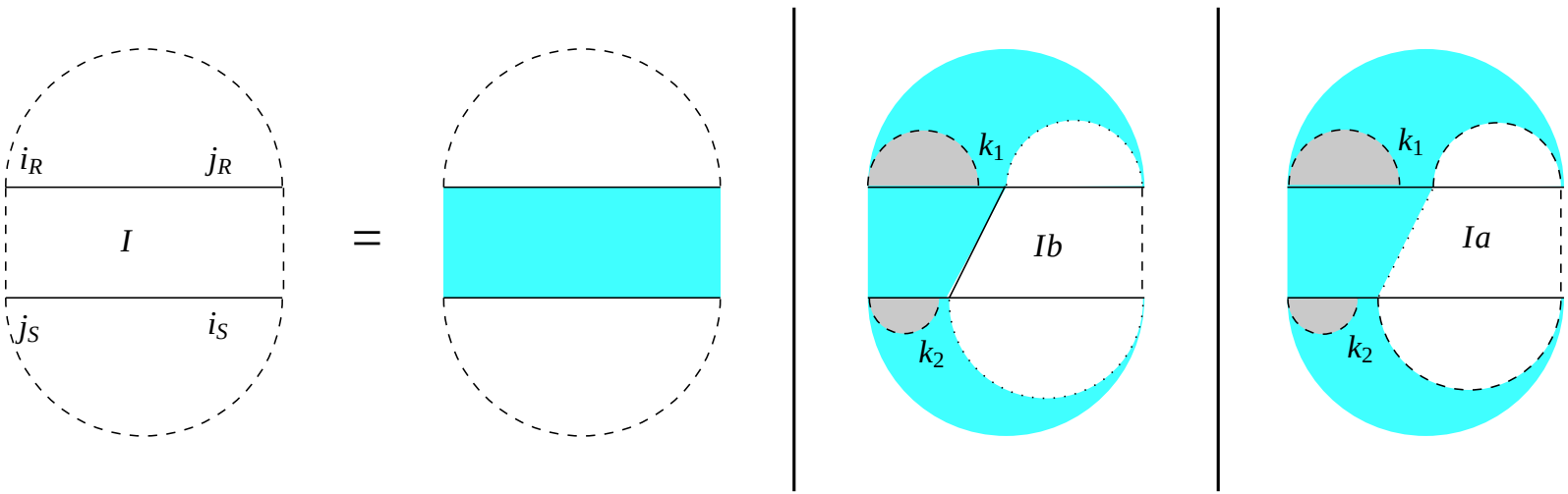
straight horizontal line: nucleotides indexed from 1 to n
solid arc: a base pair
dashed arc: can be base pair or not



white region: open to more recursions
blue region: finalized in the recursion, compute its energy contribution
green region: open to more recursions with multibranch loop energy

Partition function for two strands

straight vertical line: intermolecular bond
solid: a base pair
dotted: not a base pair
dashed: either of those two

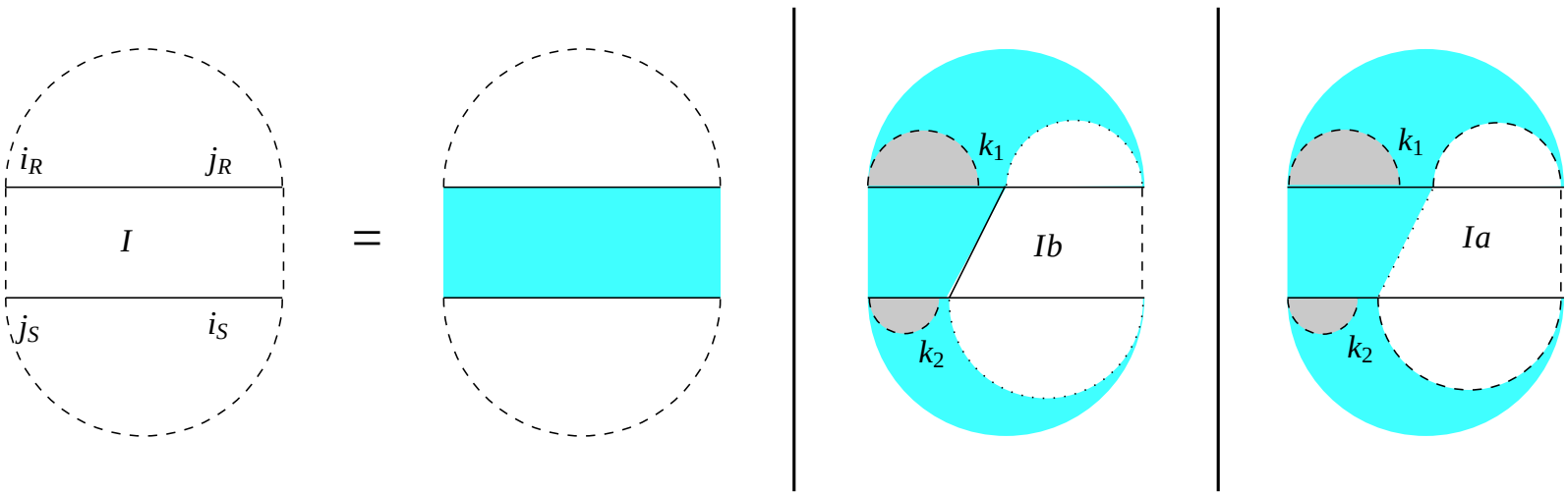


$$Q^{I}_{i_R,j_R,i_S,j_S} = Q_{i_R,j_R} Q_{i_S,j_S} + \sum_{\substack{i_R \le k_1 < j_R \\ i_S < k_2 \le j_S}} Q_{i_R,k_1-1} Q_{k_2+1,j_S} Q^{Ib}_{k_1,j_R,i_S,k_2} +$$

$$\sum_{\substack{i_R \le k_1 < j_R \\ i_S < k_2 \le j_S}} Q_{i_R,k_1-1} Q_{k_2+1,j_S} Q^{Ia}_{k_1,j_R,i_S,k_2}.$$

Partition function for two strands

straight vertical line: intermolecular bond
solid: a base pair
dotted: not a base pair
dashed: either of those two

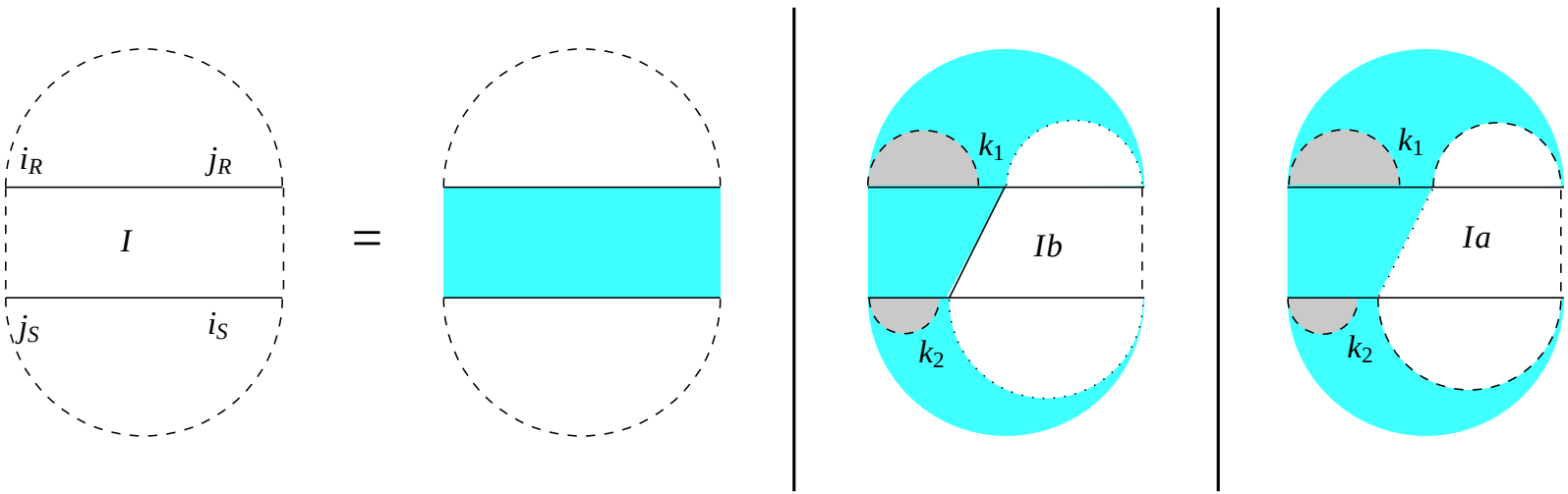


$$Q^{I}_{i_R,j_R,i_S,j_S} = Q_{i_R,j_R} Q_{i_S,j_S} + \sum_{\substack{i_R \le k_1 < j_R \\ i_S < k_2 \le j_S}} Q_{i_R,k_1-1} Q_{k_2+1,j_S} Q^{Ib}_{k_1,j_R,i_S,k_2} +$$

$$\sum_{\substack{i_R \le k_1 < j_R \\ i_S < k_2 \le j_S}} Q_{i_R,k_1-1} Q_{k_2+1,j_S} Q^{Ia}_{k_1,j_R,i_S,k_2}.$$

Partition function for two strands

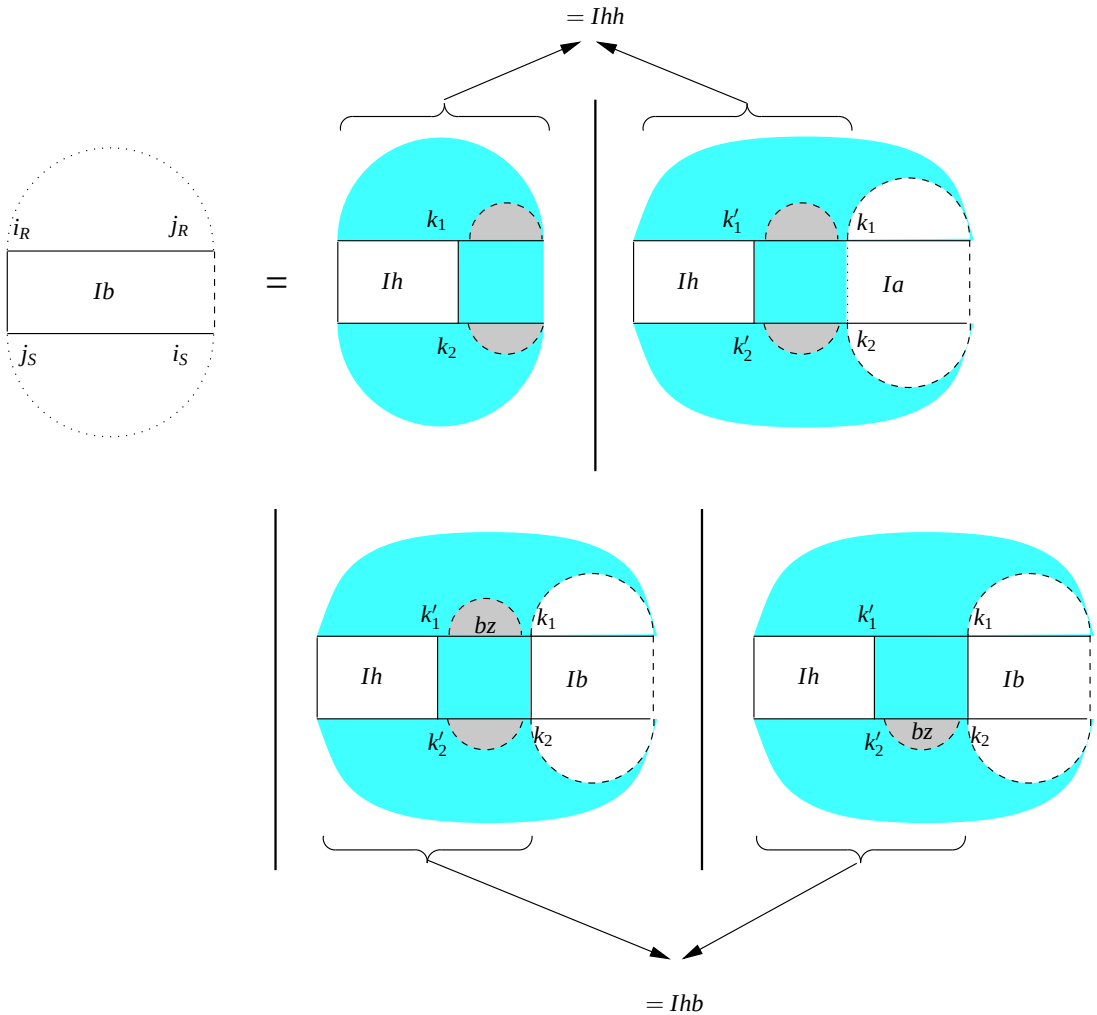
straight vertical line: intermolecular bond
solid: a base pair
dotted: not a base pair
dashed: either of those two



$$Q^{I}_{i_R,j_R,i_S,j_S} = Q_{i_R,j_R} Q_{i_S,j_S} + \sum_{\substack{i_R \le k_1 < j_R \\ i_S < k_2 \le j_S}} Q_{i_R,k_1-1} Q_{k_2+1,j_S} Q^{Ib}_{k_1,j_R,i_S,k_2} +$$

$$\sum_{\substack{i_R \le k_1 < j_R \\ i_S < k_2 \le j_S}} Q_{i_R,k_1-1} Q_{k_2+1,j_S} Q^{Ia}_{k_1,j_R,i_S,k_2}.$$

Q^{Ib}



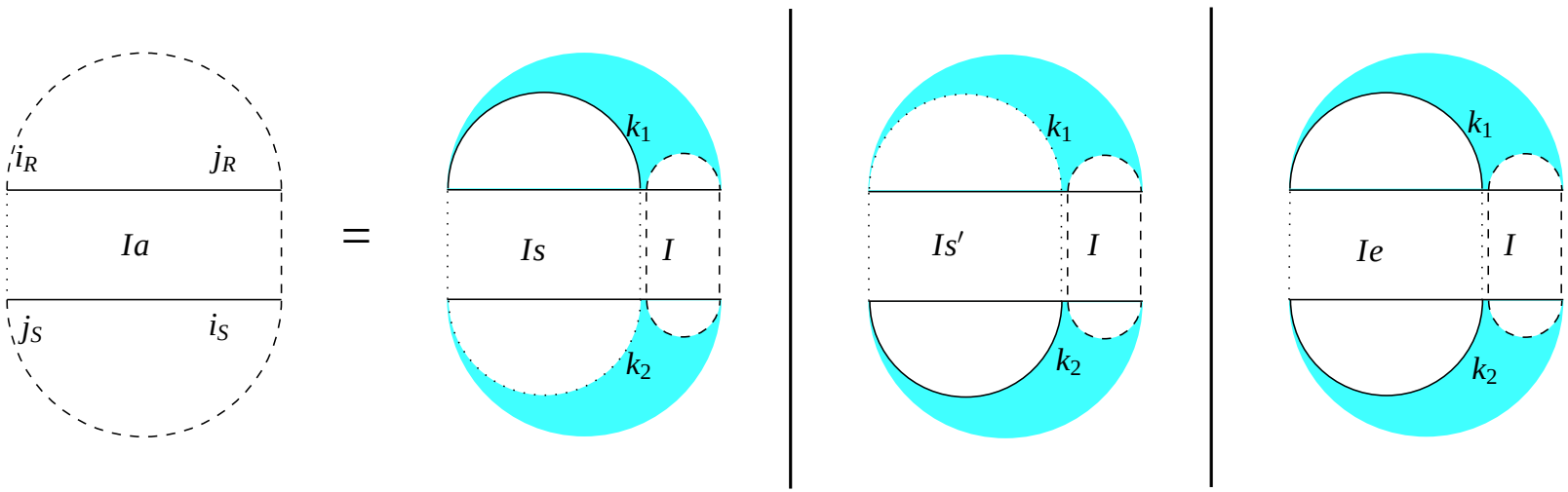
b: stands for bond

Q^{Ia}

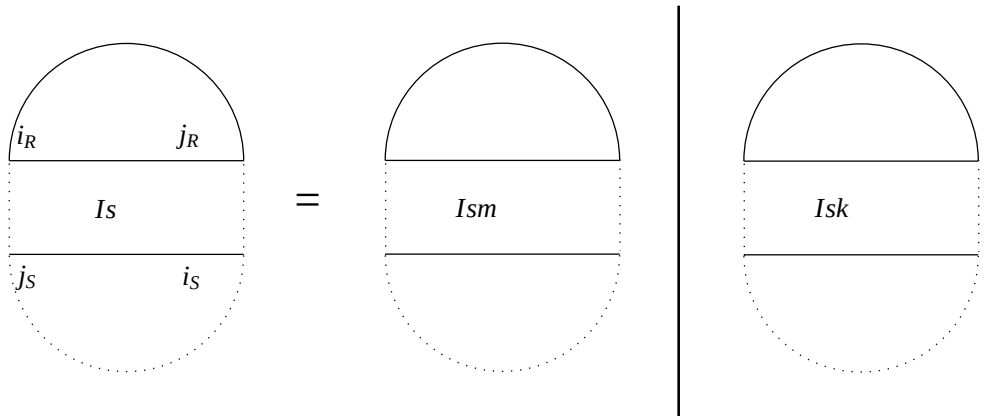
a: stands for arc

s: stands for subsume

e: stands for equivalent



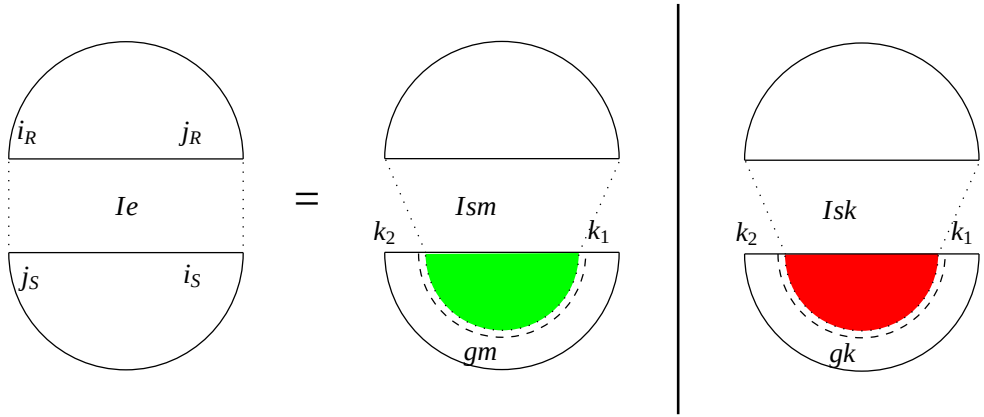
Q^{ls} and Q^{le}



s: stands for subsume

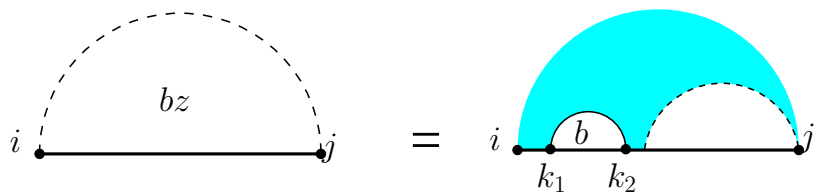
k: stands for kissing-loop

m: stands for multi-loop

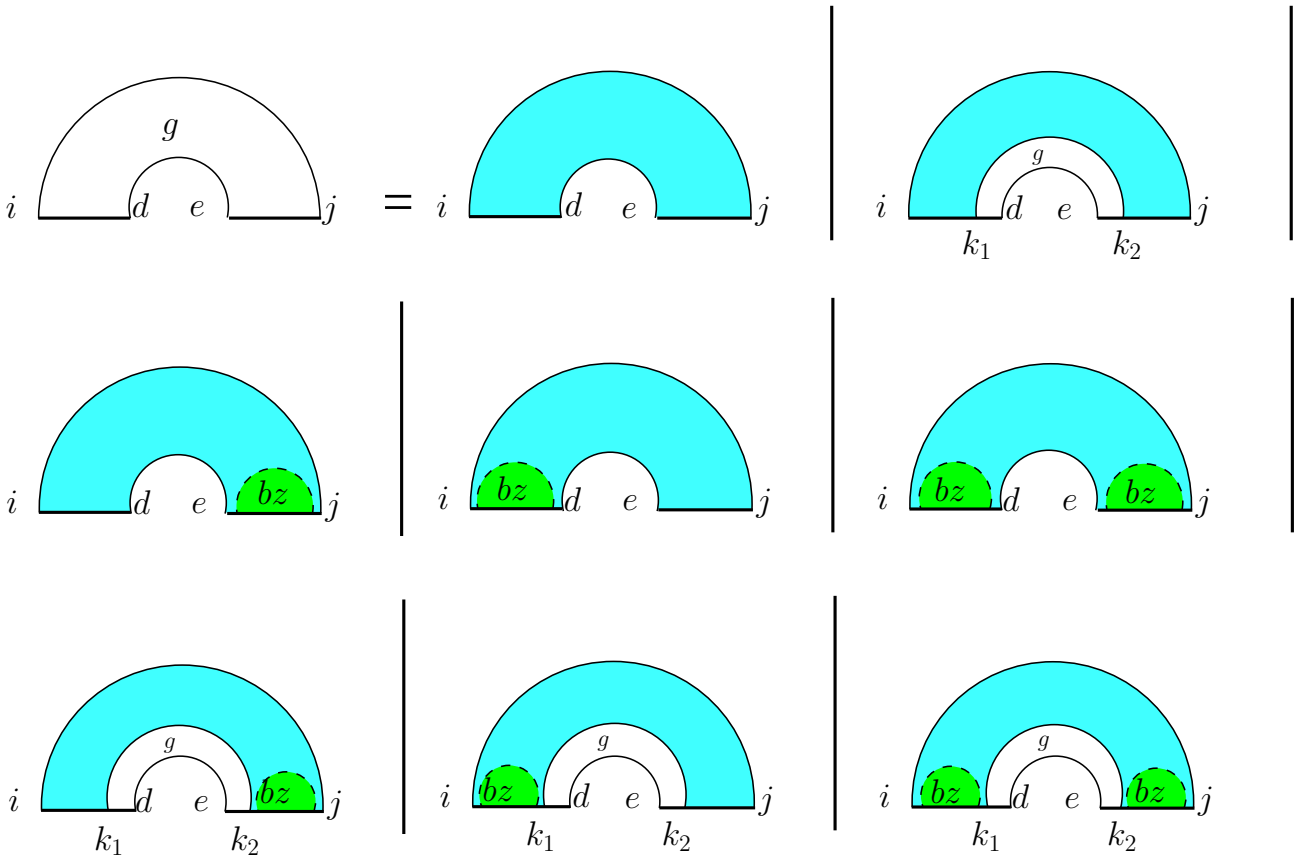


e: stands for equivalent

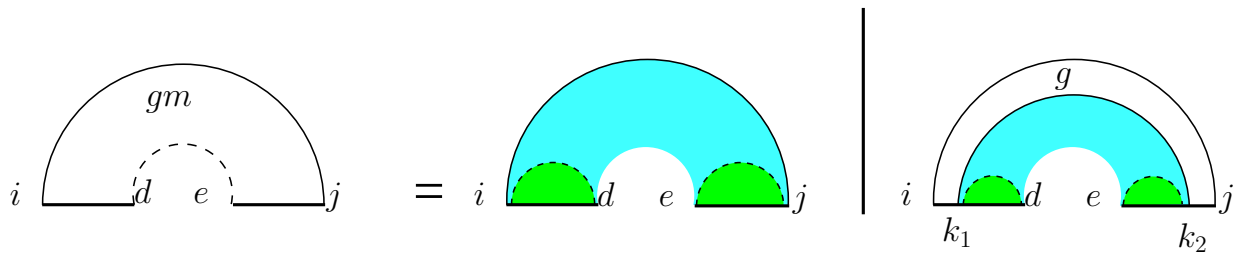
All tables



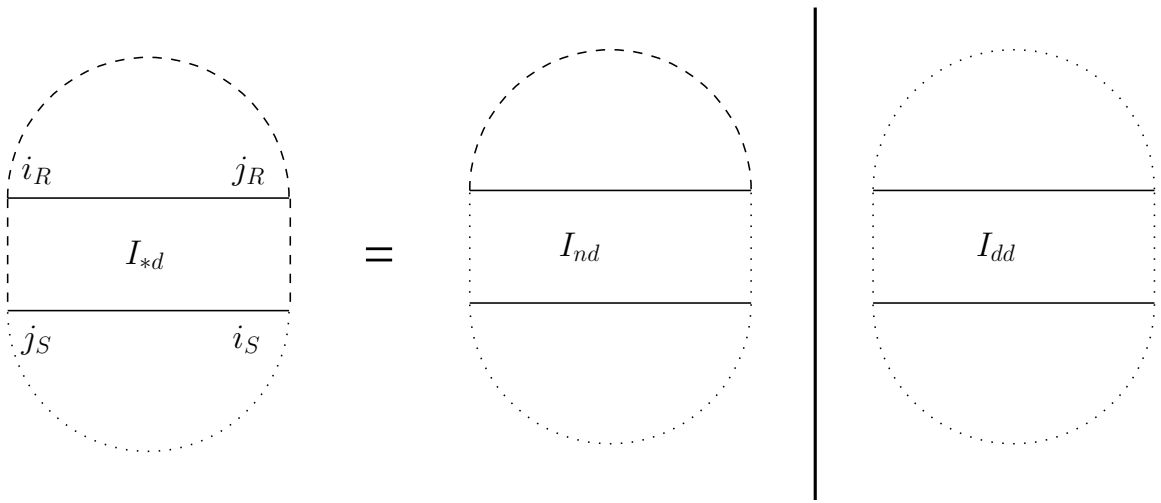
All tables



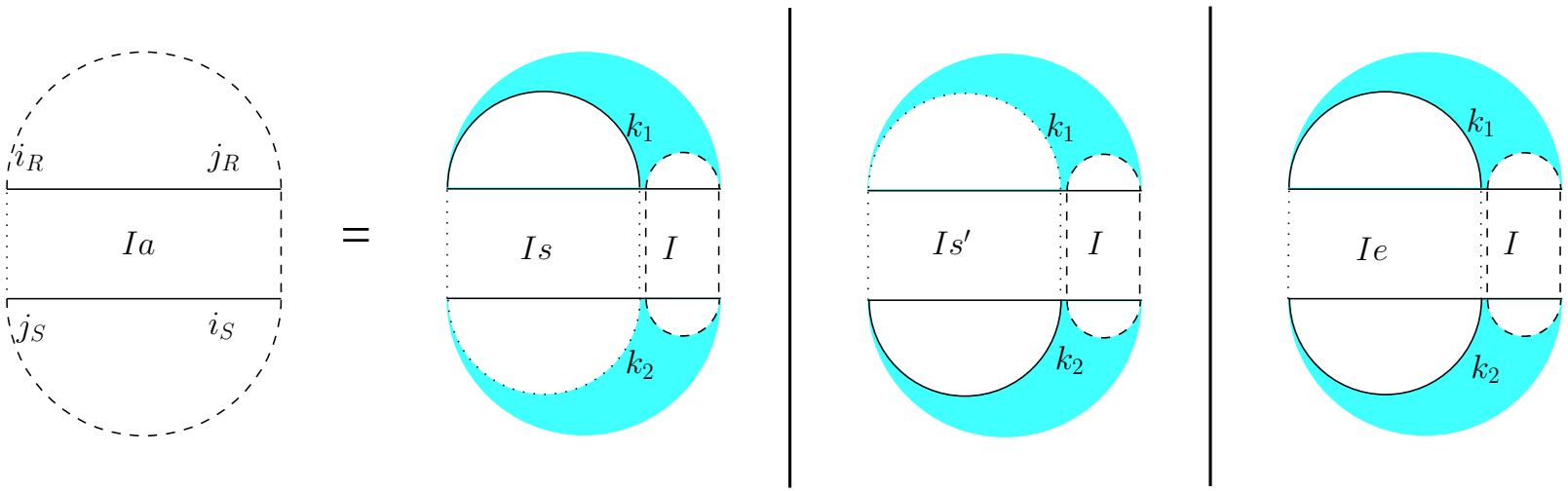
All tables



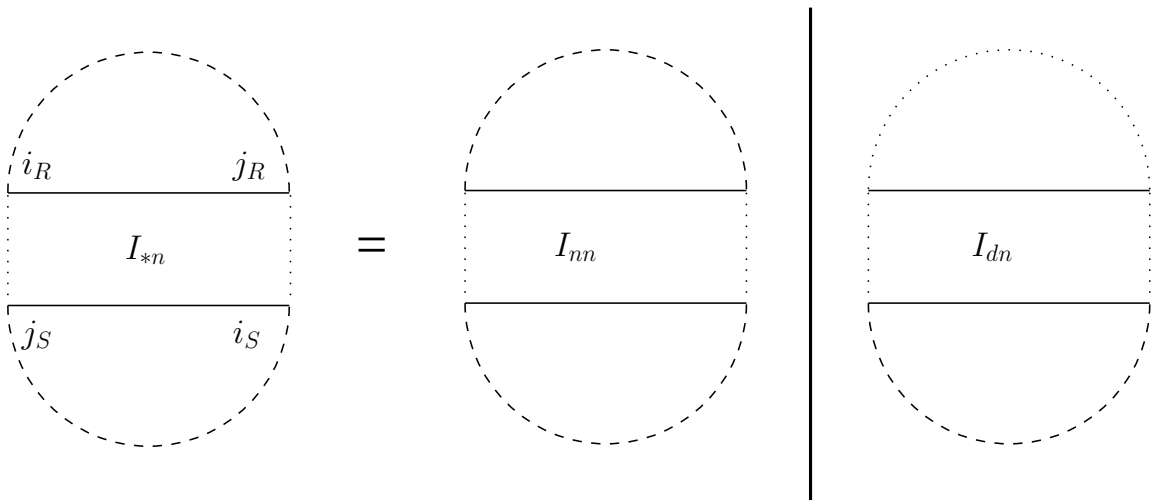
All tables



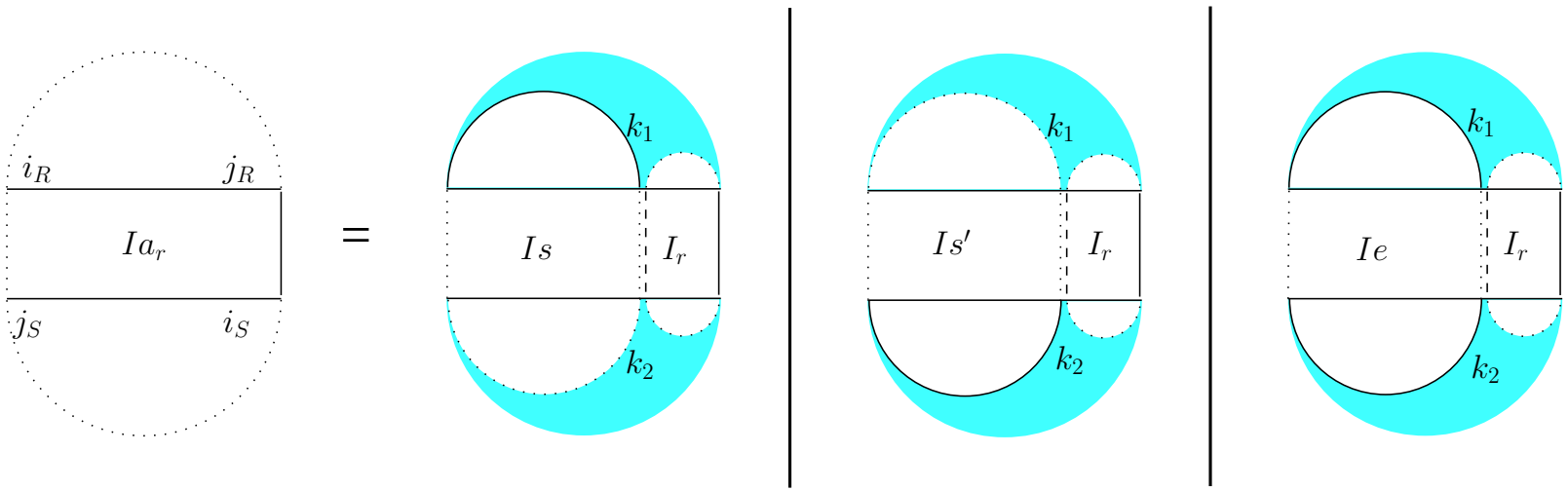
All tables



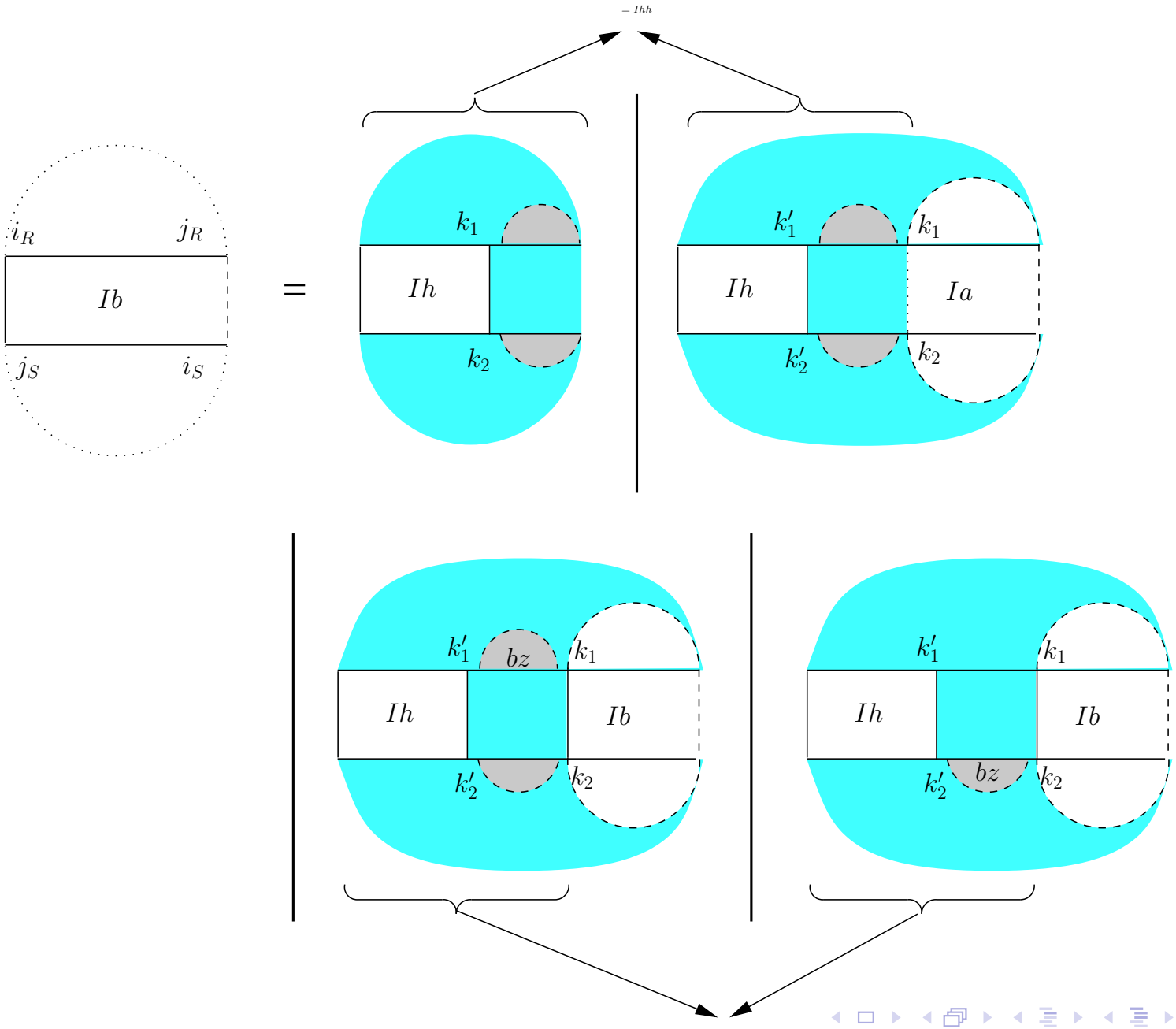
All tables



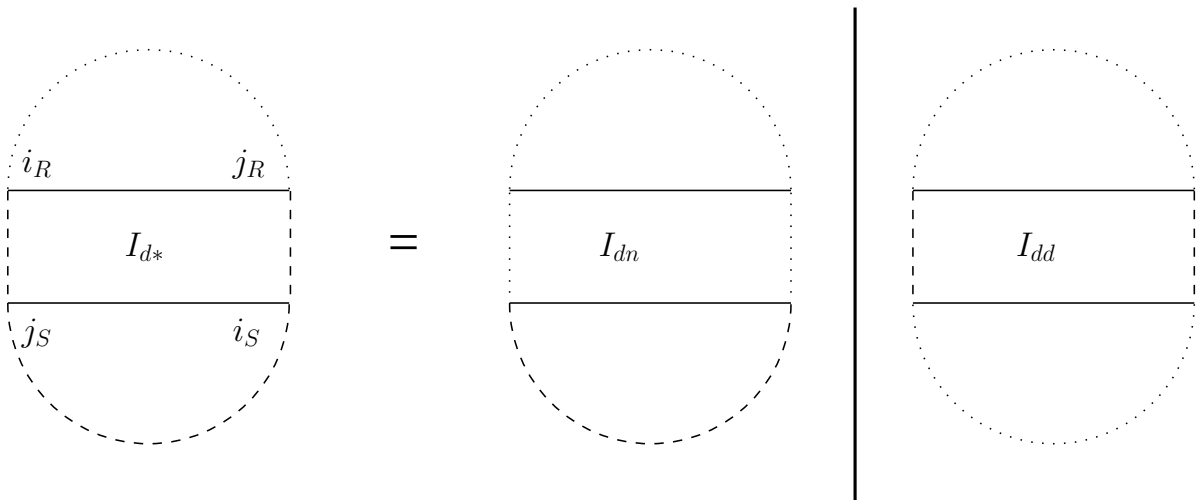
All tables



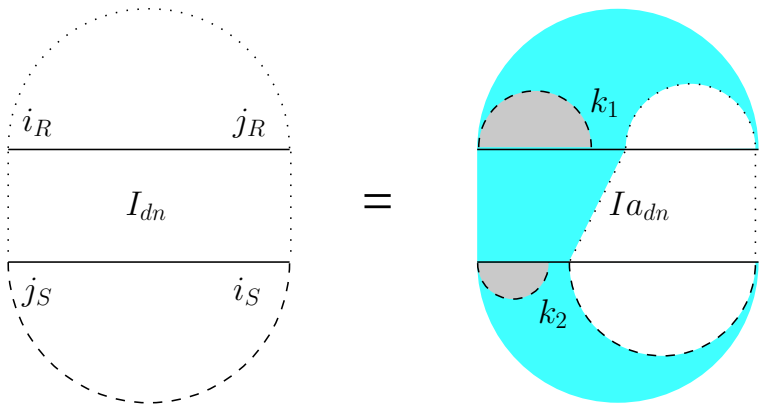
All tables


$= Ihh$
i_R
j_R
Ib
j_S
i_S
=
k_1
Ih
k_2
k_1'
k_1
Ih
Ia
k_2'
k_2
k_1'
bz
k_1
Ih
Ib
k_2'
k_2
k_1'
k_1
Ih
Ib
k_2'
bz
k_2

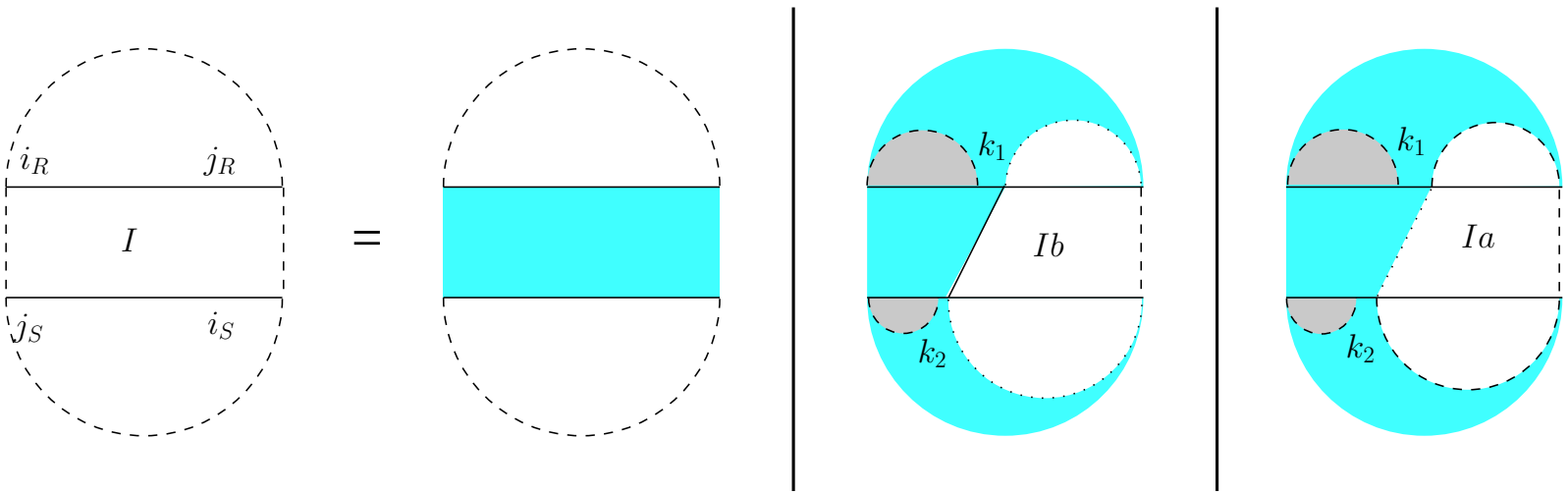
All tables



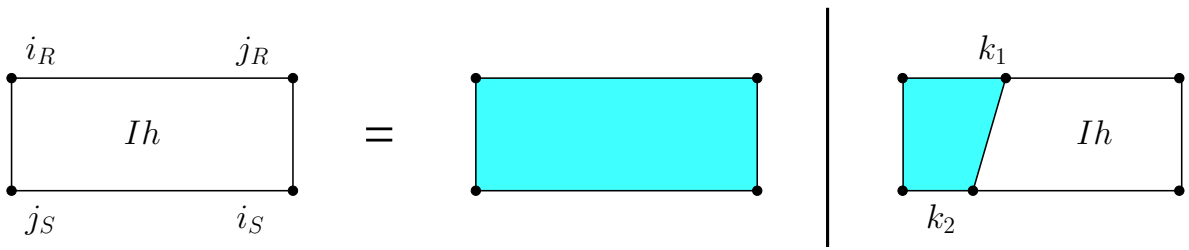
All tables



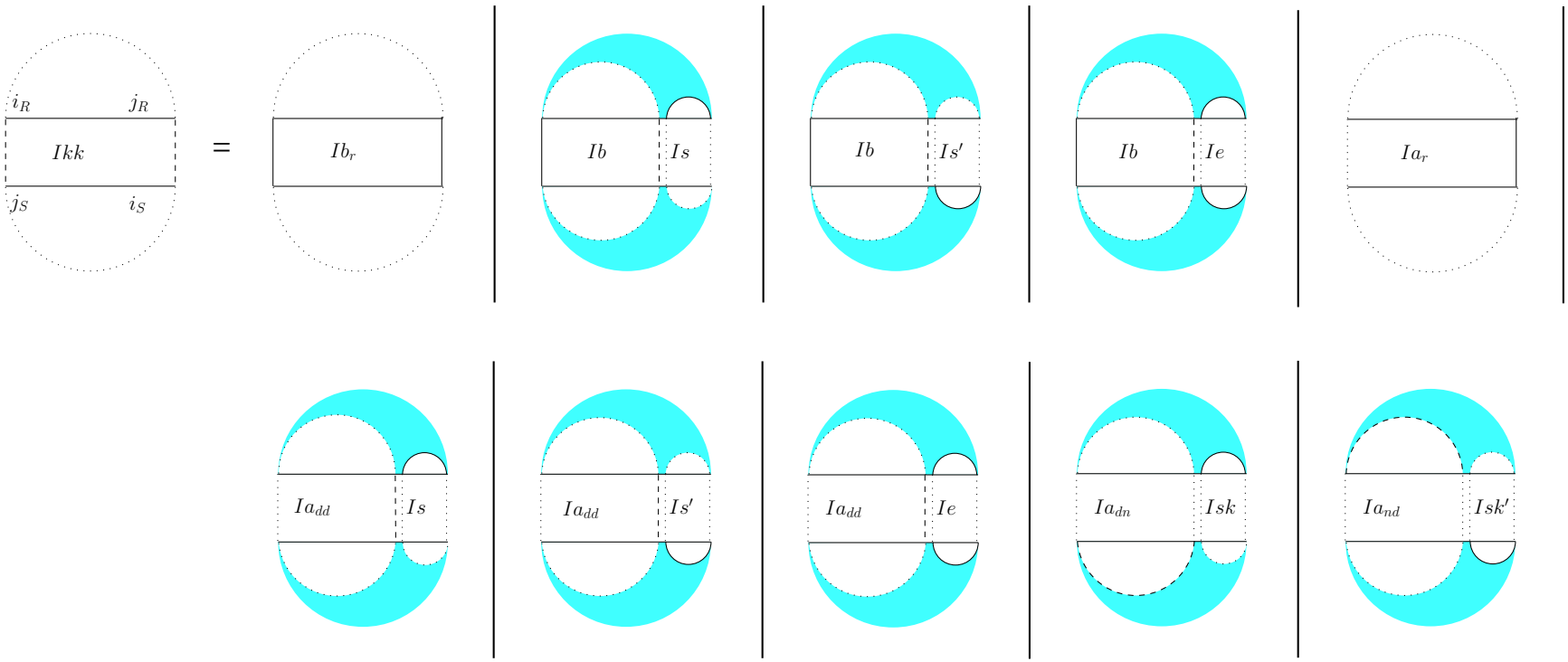
All tables



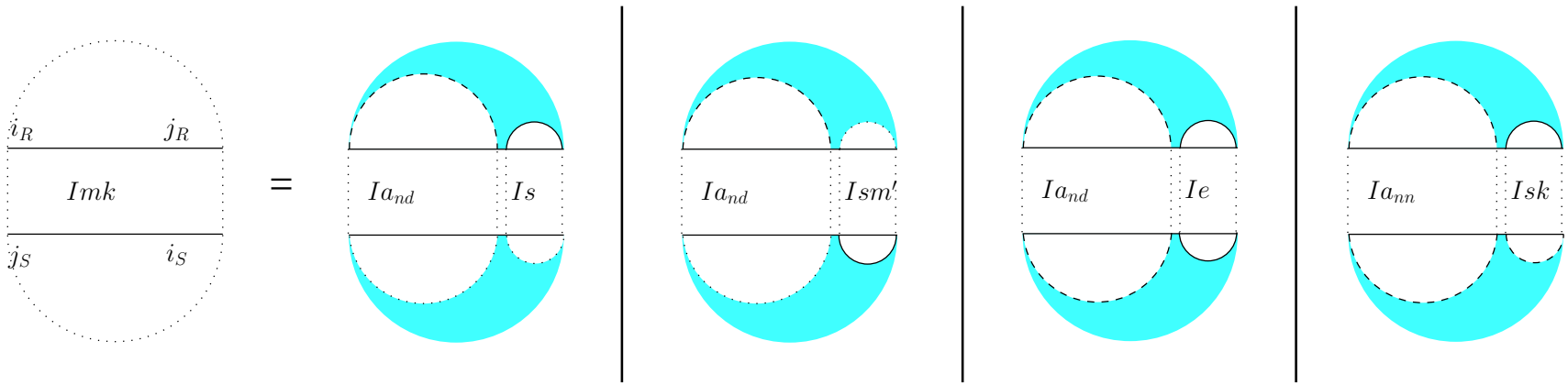
All tables



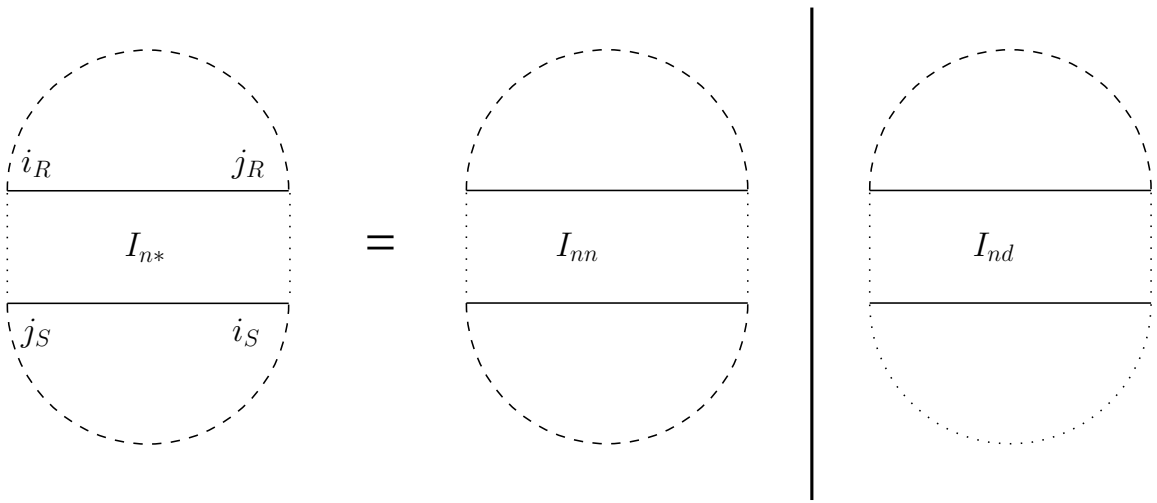
All tables


i_R
j_R
Ikk
j_S
i_S
=
Ib_r
Ib
Is
Ib
Is'
Ib
Ie
Ia_r
Ia_{dd}
Is
Ia_{dd}
Is'
Ia_{dd}
Ie
Ia_{dn}
Isk
Ia_{nd}
Isk'

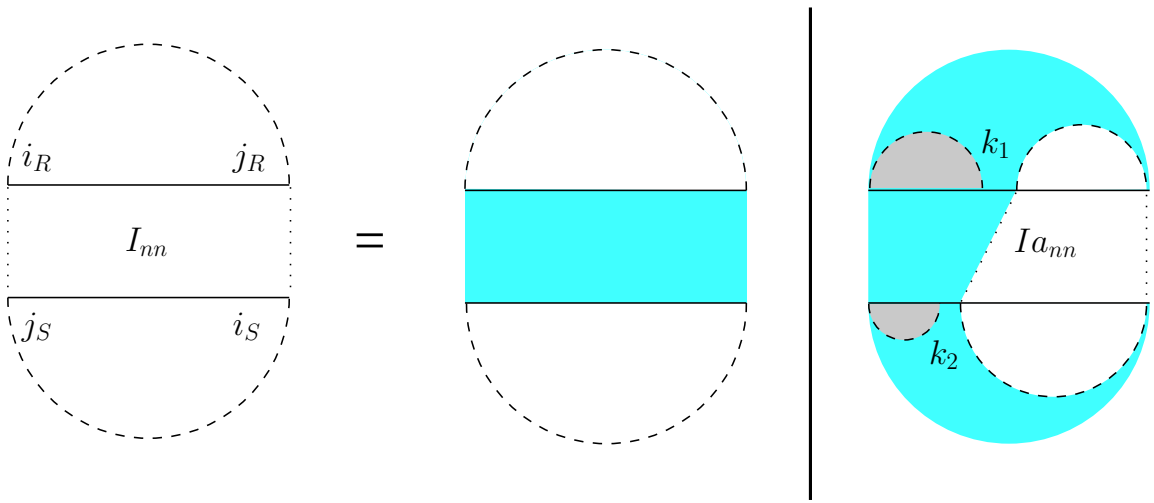
All tables



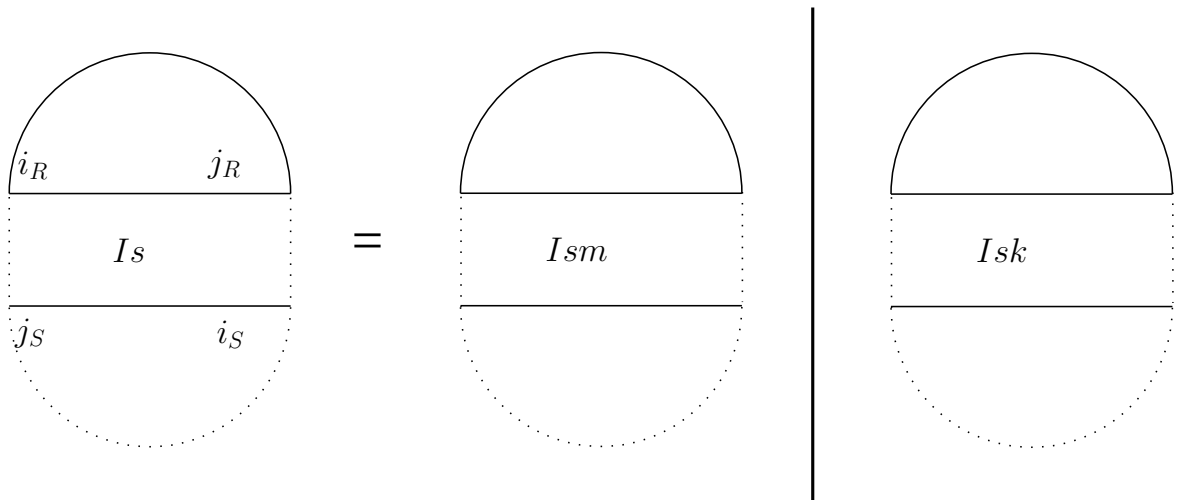
All tables



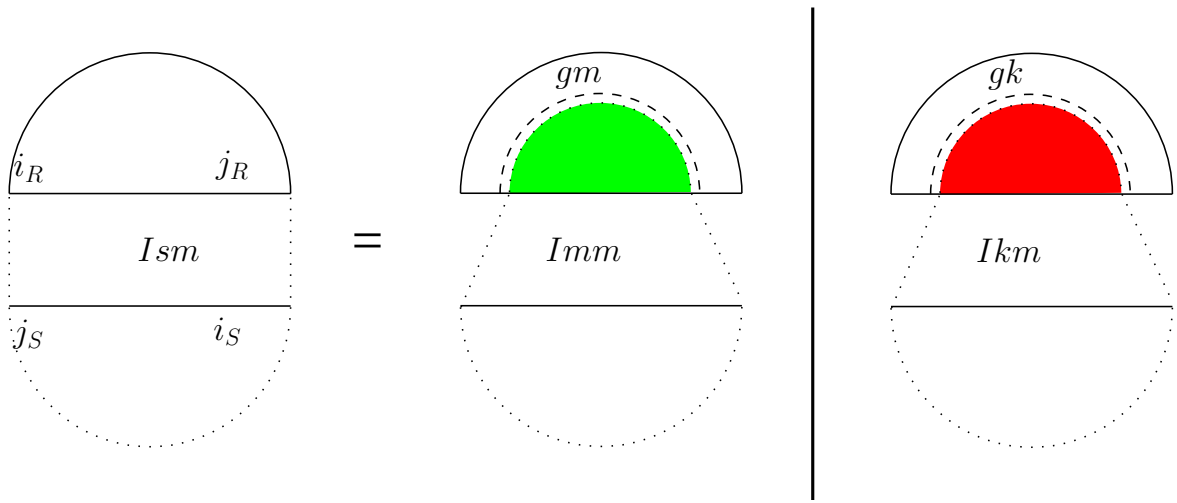
All tables



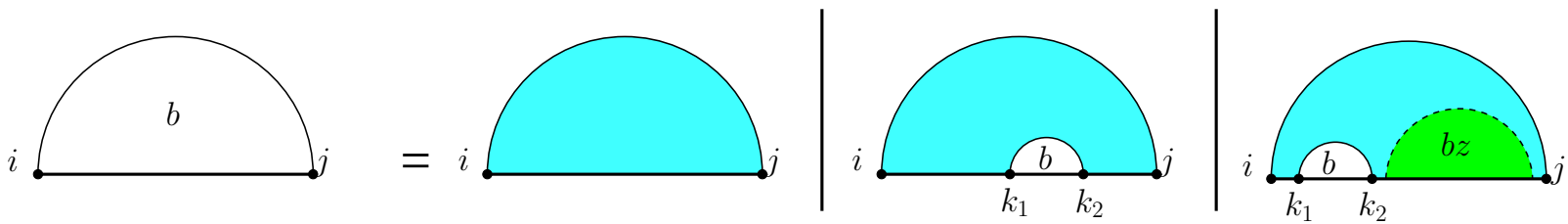
All tables



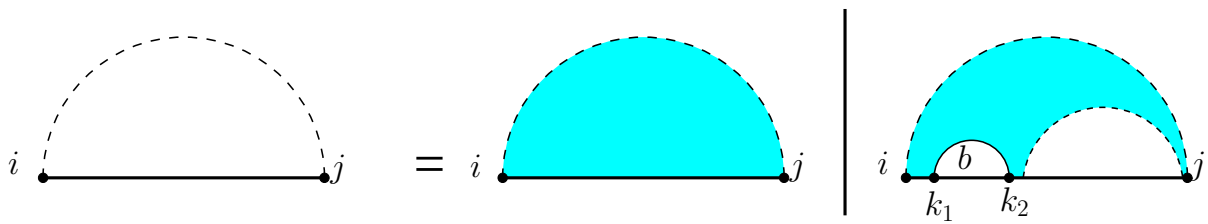
All tables



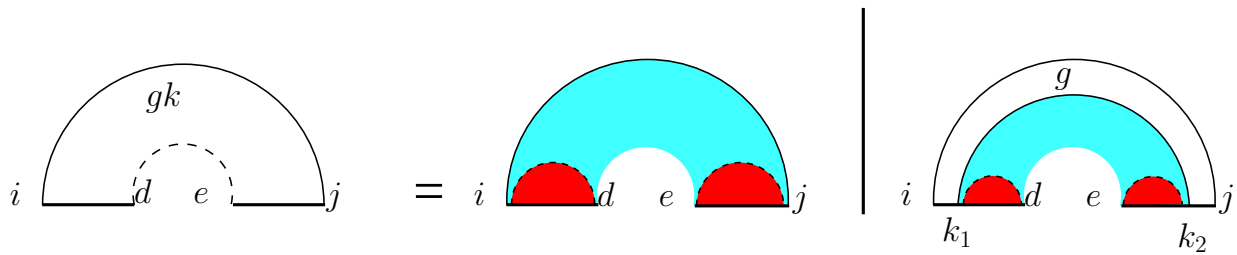
All tables



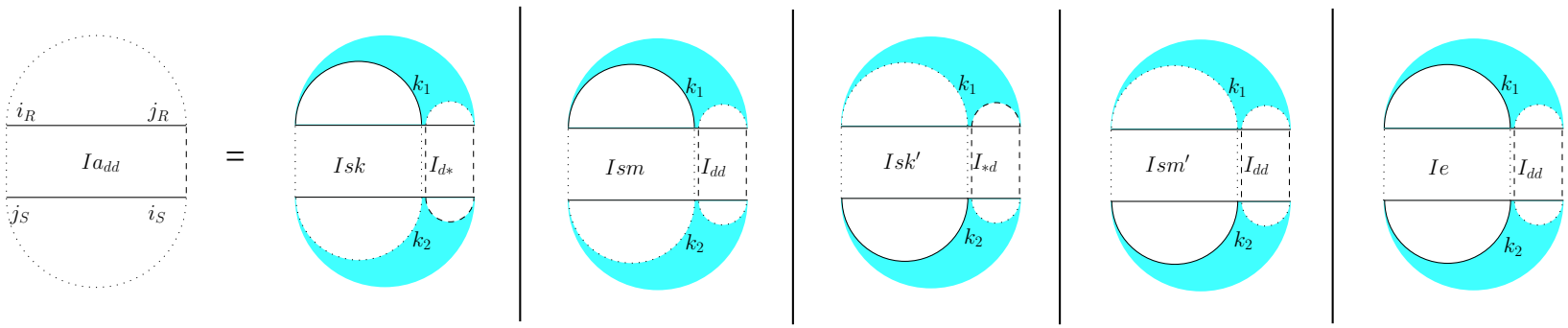
All tables



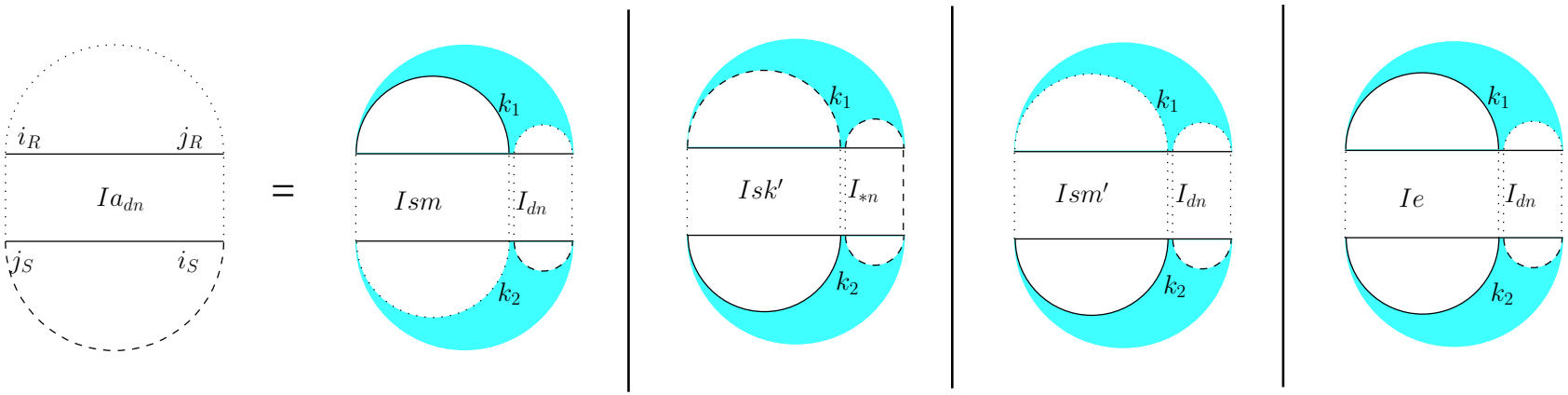
All tables



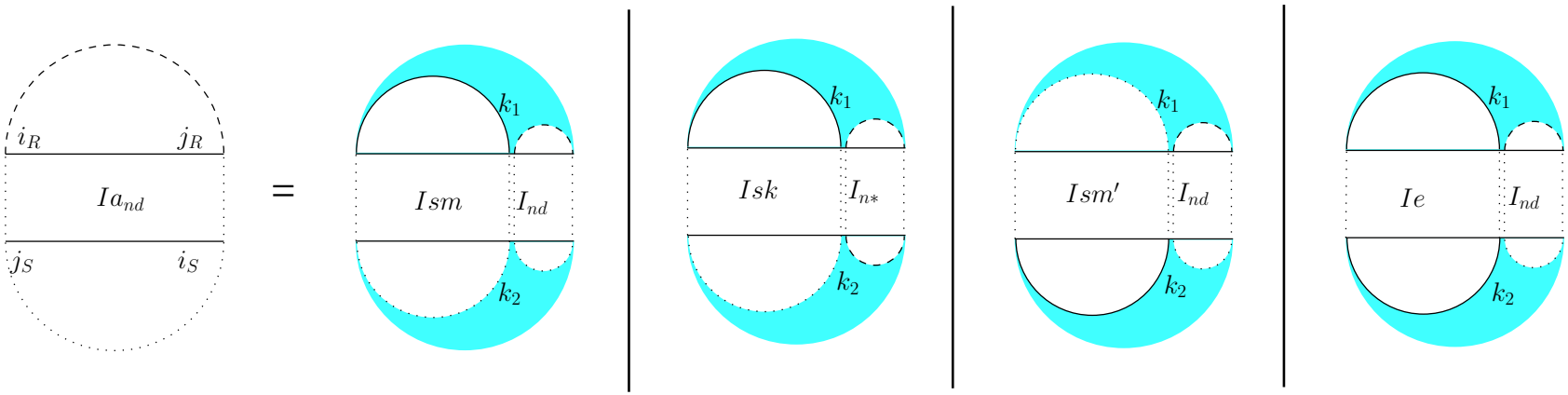
All tables


i_R
j_R
Ia_{dd}
j_S
i_S
=
k_1
Isk
I_{d*}
k_2
k_1
Ism
I_{dd}
k_2
k_1
Isk'
I_{*d}
k_2
k_1
Ism'
I_{dd}
k_2
k_1
Ie
I_{dd}
k_2

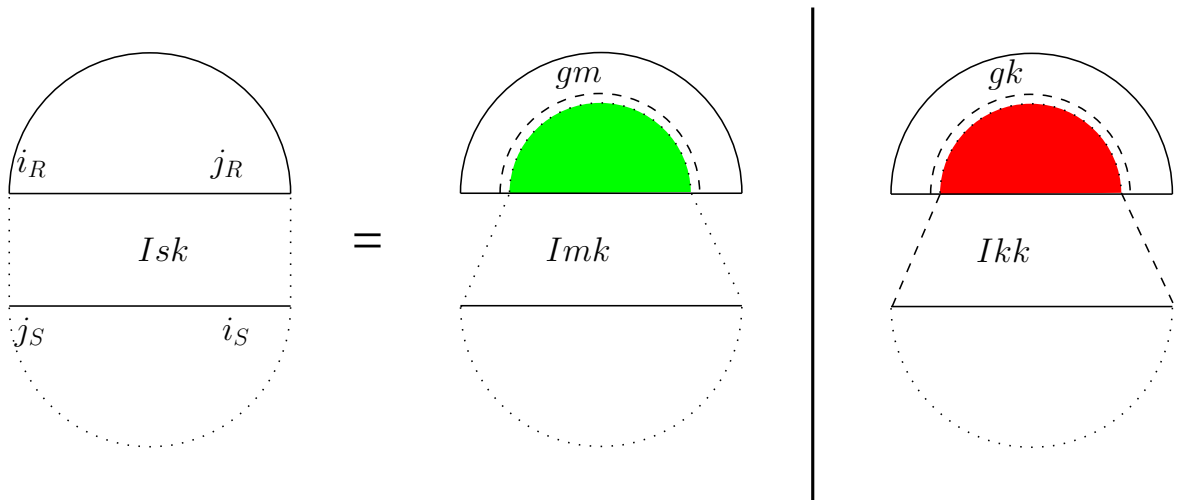
All tables


i_R
j_R
Ia_{dn}
j_S
i_S
=
k_1
Ism
I_{dn}
k_2
k_1
Isk'
I_{*n}
k_2
k_1
Ism'
I_{dn}
k_2
k_1
Ie
I_{dn}
k_2

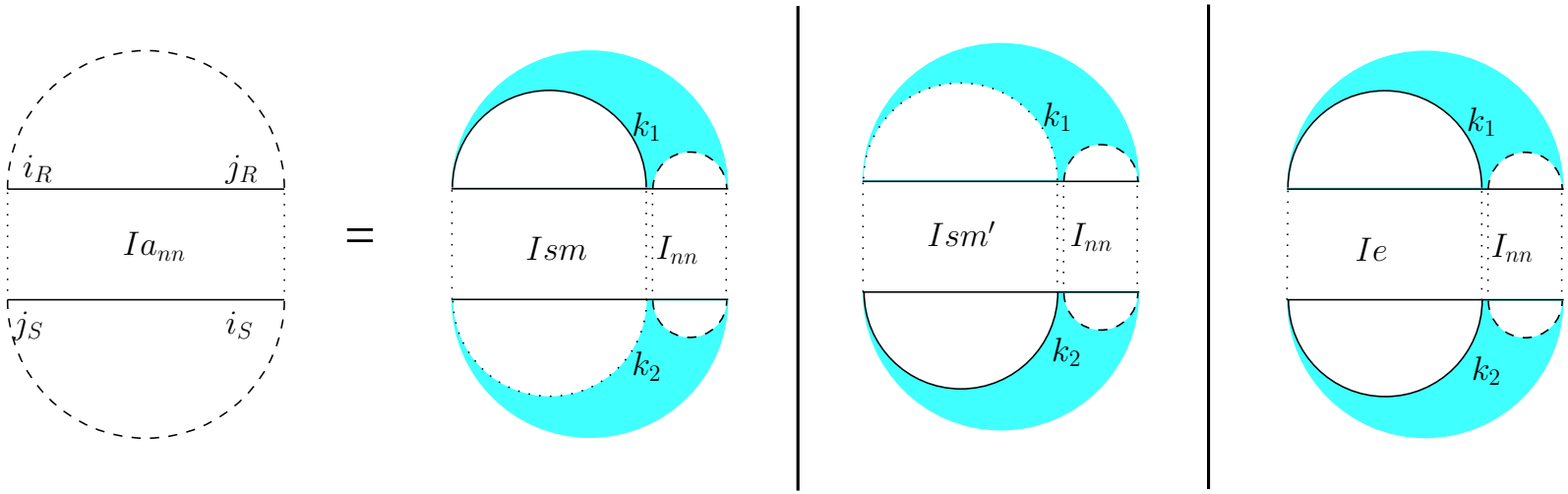
All tables


i_R
j_R
Ia_{nd}
j_S
i_S
=
k_1
Ism
I_{nd}
k_2
k_1
Isk
I_{n*}
k_2
k_1
Ism'
I_{nd}
k_2
k_1
Ie
I_{nd}
k_2

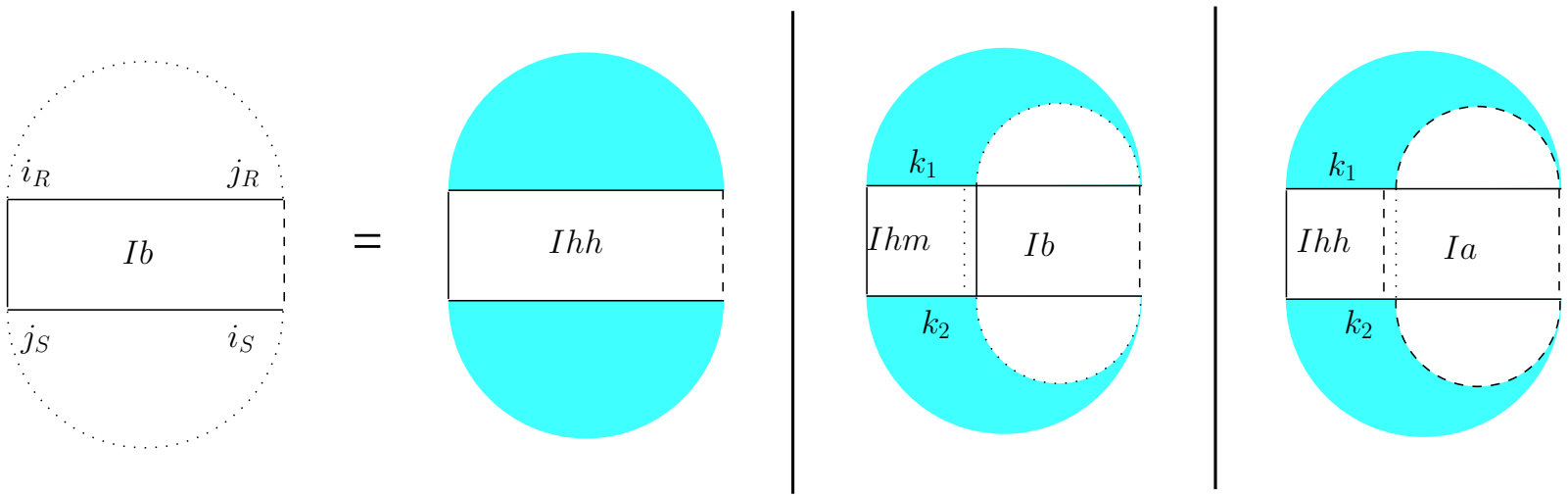
All tables



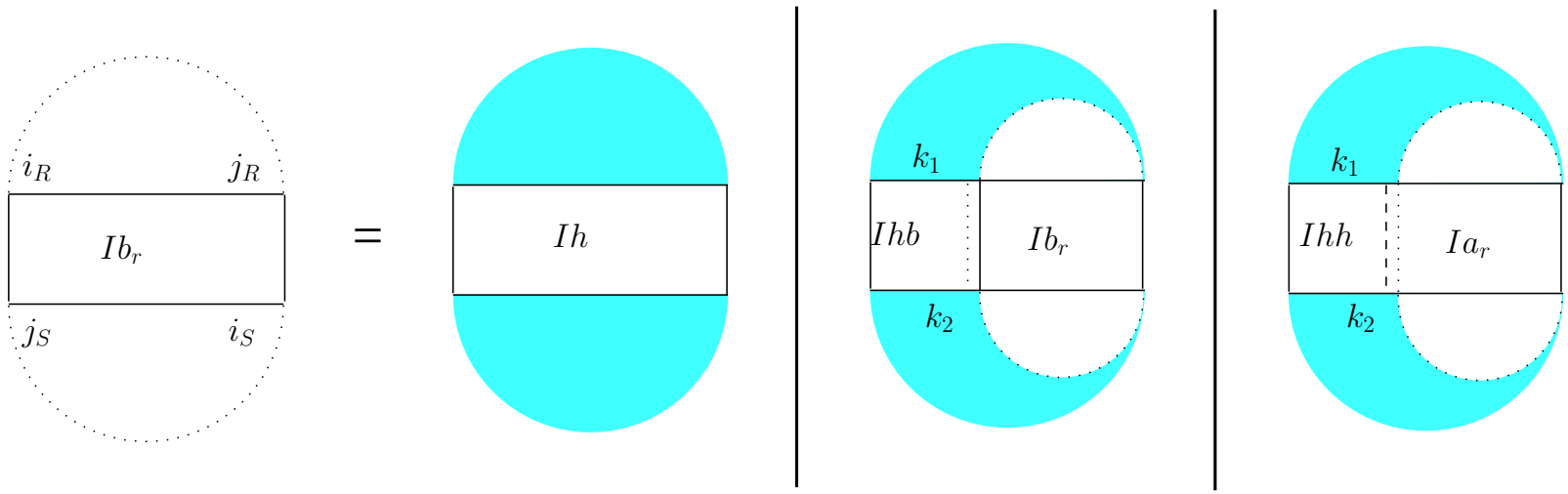
All tables



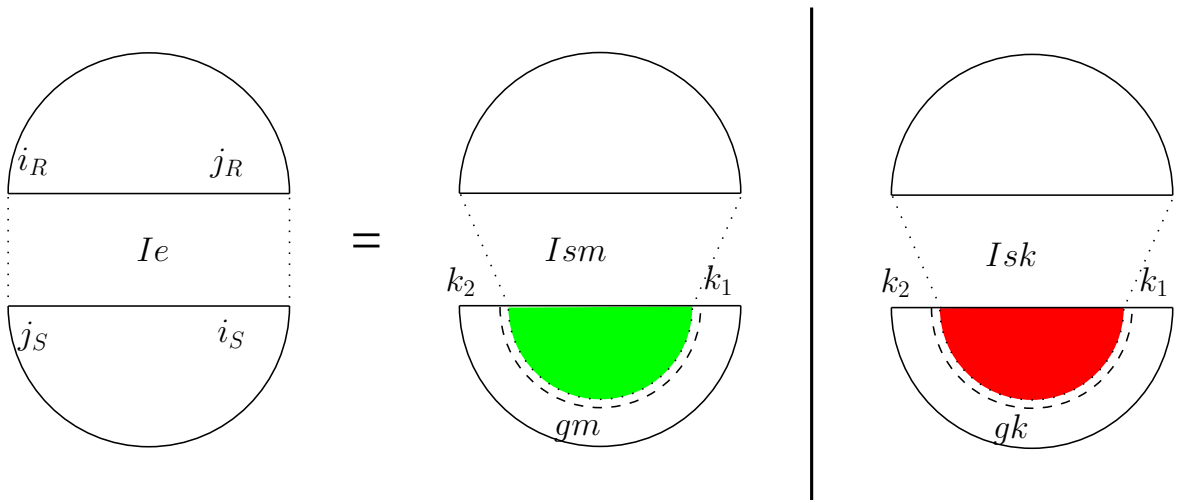
All tables



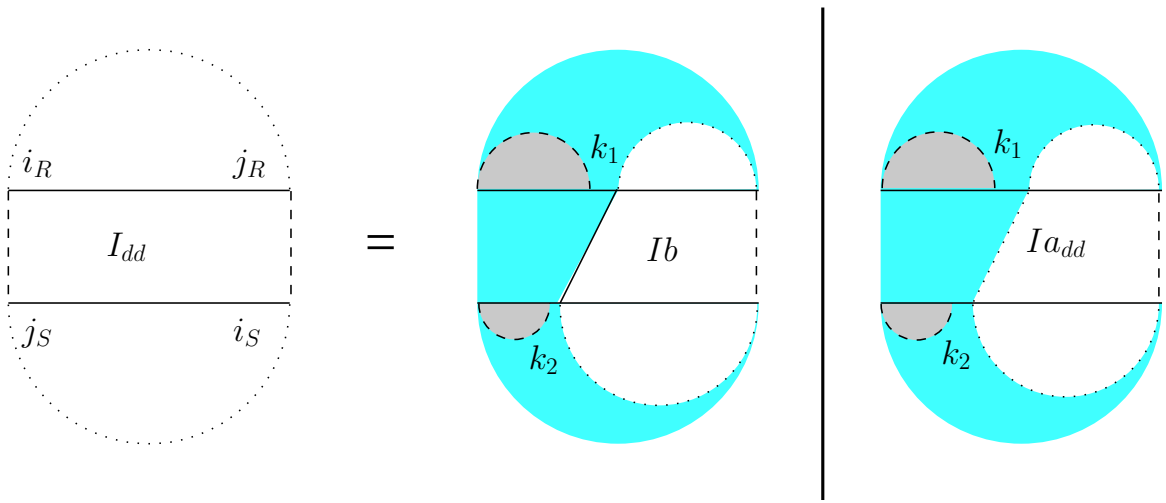
All tables



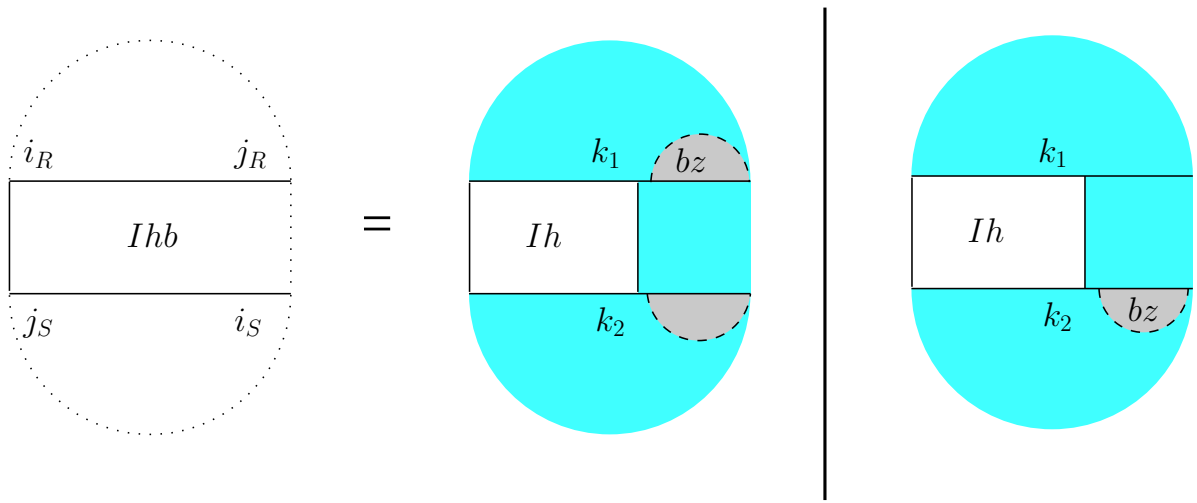
All tables



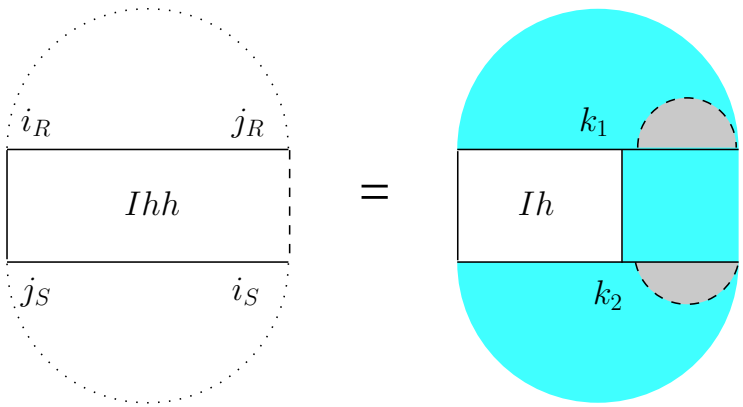
All tables



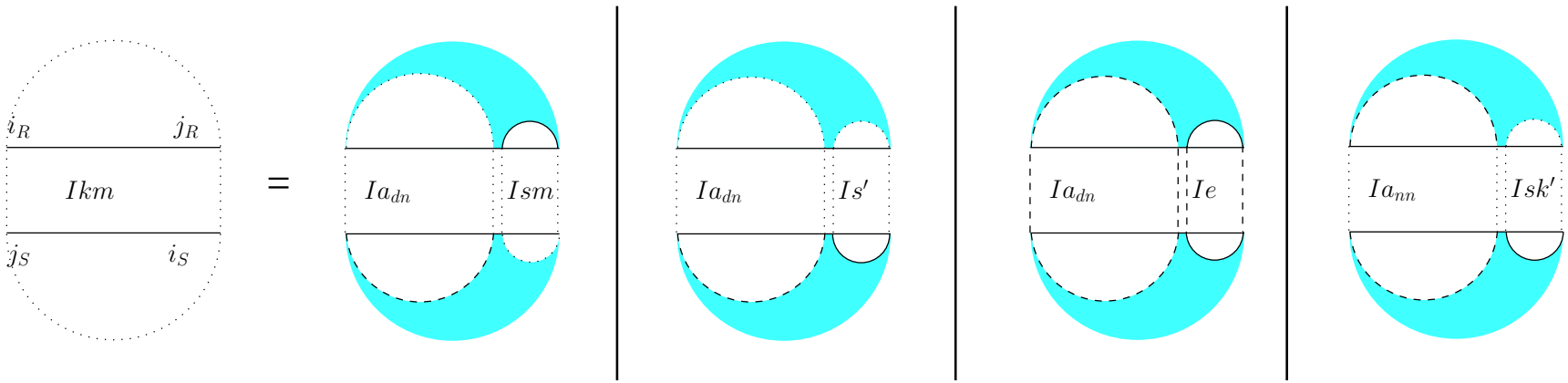
All tables



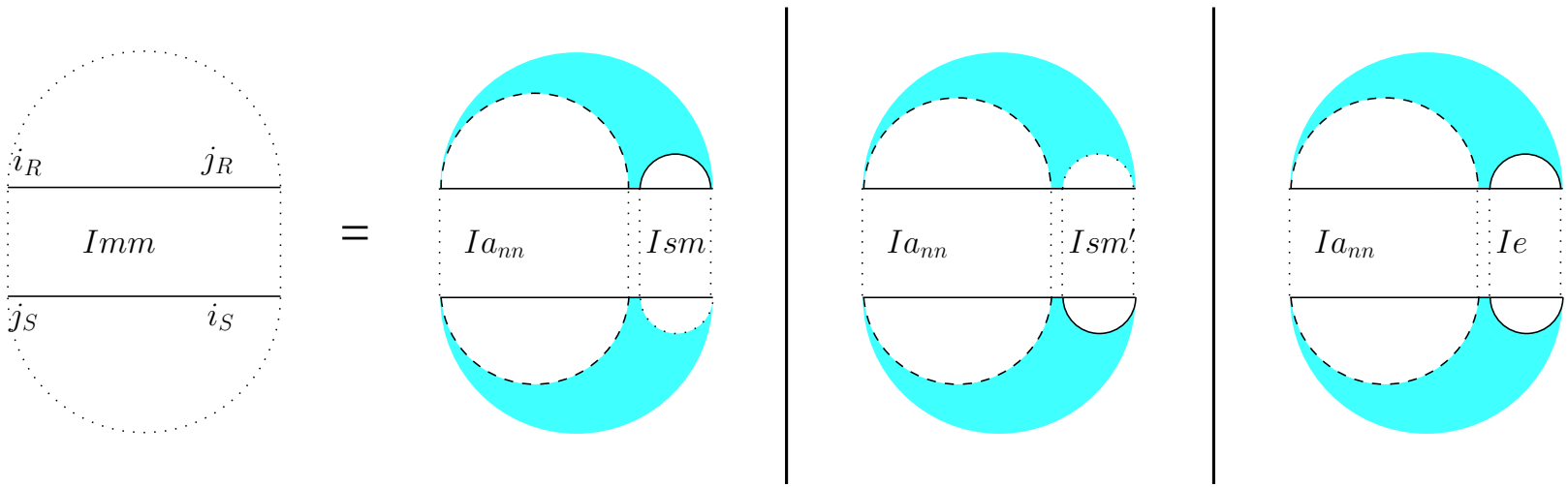
All tables



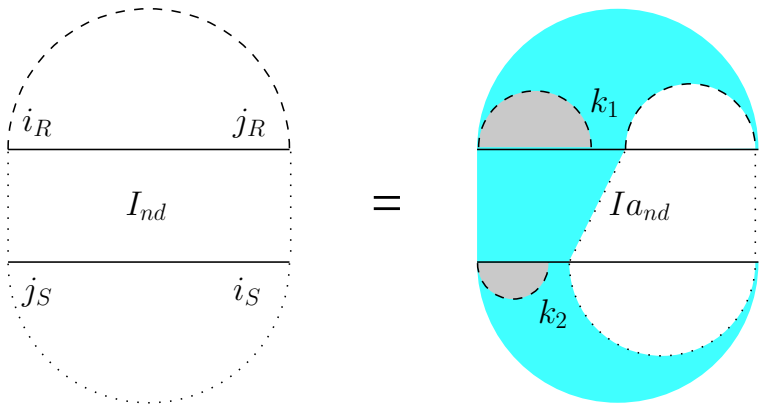
All tables



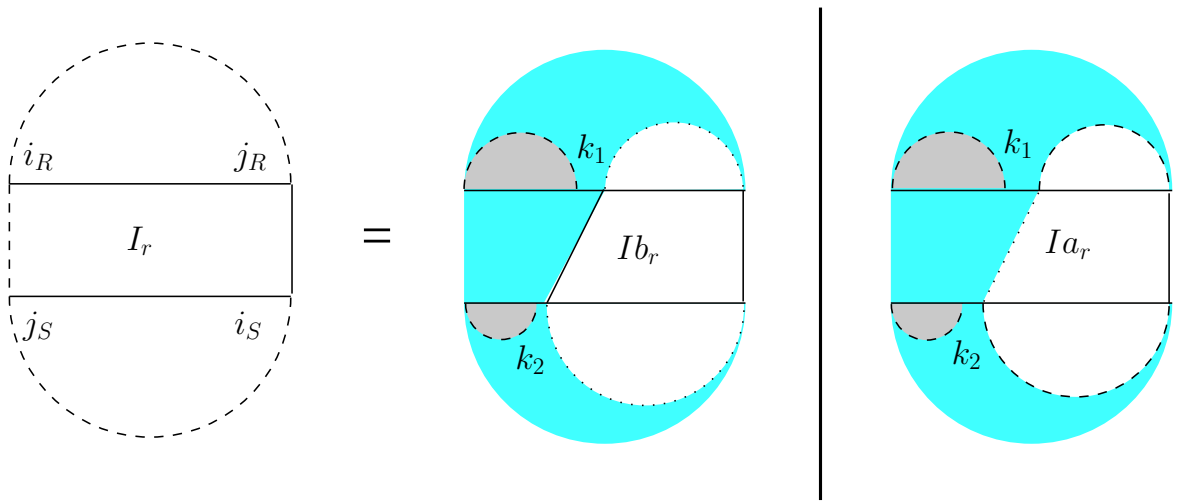
All tables



All tables



All tables



Equilibrium concentrations

For two RNAs **R** and **S**

Assume five types of chemical compounds: **R**, **S**, **RR**, **SS**, **RS**.
Solve

$$K_{\mathbf{R}} = \frac{Q^I_{\mathbf{RR}}}{Q^2_{\mathbf{R}}} = \frac{N_{\mathbf{RR}}}{N^2_{\mathbf{R}}},$$

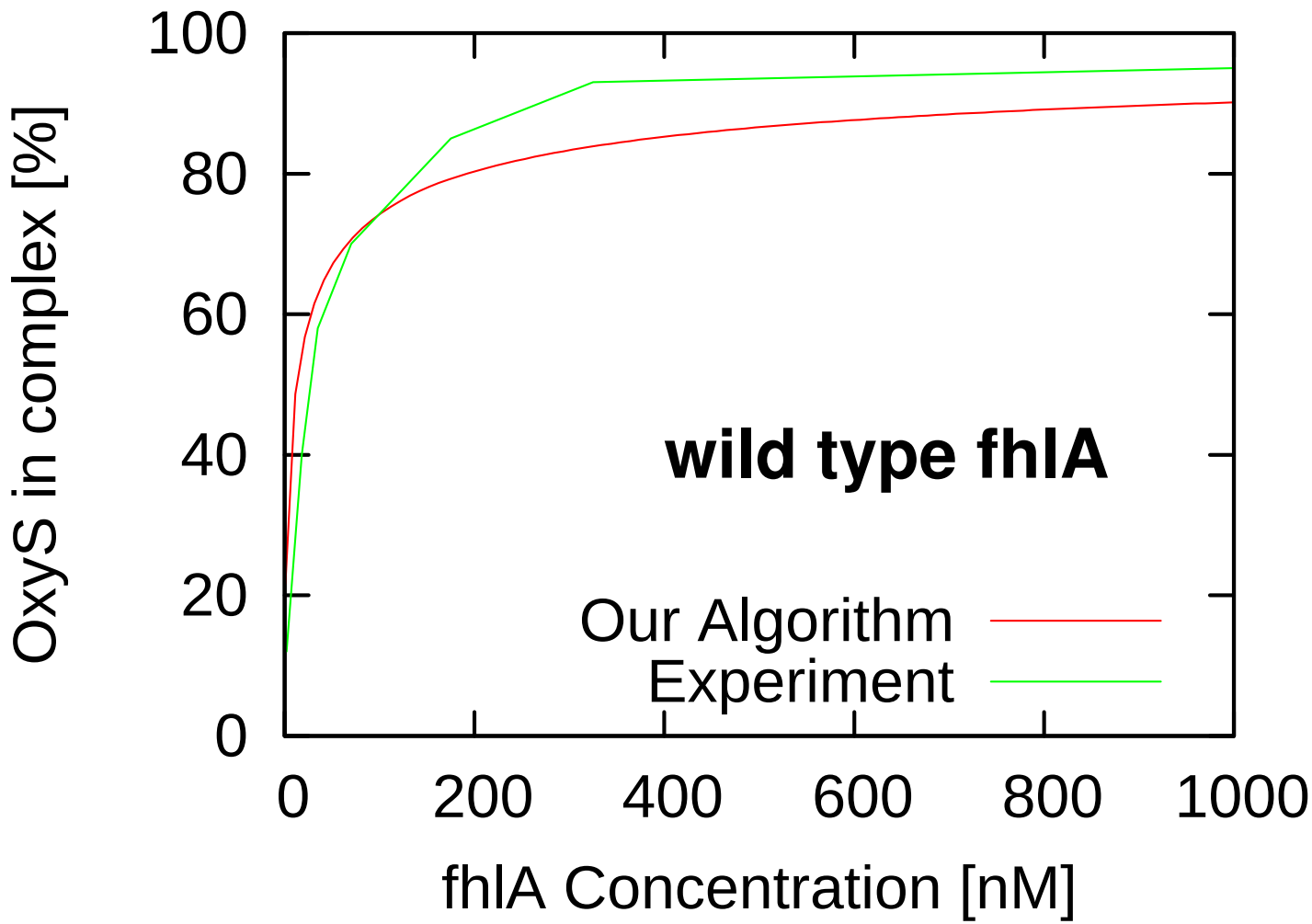
$$K_{\mathbf{S}} = \frac{Q^I_{\mathbf{SS}}}{Q^2_{\mathbf{S}}} = \frac{N_{\mathbf{SS}}}{N^2_{\mathbf{S}}},$$

$$K_{\mathbf{RS}} = \frac{Q^I_{\mathbf{RS}}}{Q_{\mathbf{R}} Q_{\mathbf{S}}} = \frac{N_{\mathbf{RS}}}{N_{\mathbf{R}} N_{\mathbf{S}}},$$

$$N_{\mathbf{RS}} = N^0_{\mathbf{R}} - 2N_{\mathbf{RR}} - N_{\mathbf{R}} = N^0_{\mathbf{S}} - 2N_{\mathbf{SS}} - N_{\mathbf{S}},$$

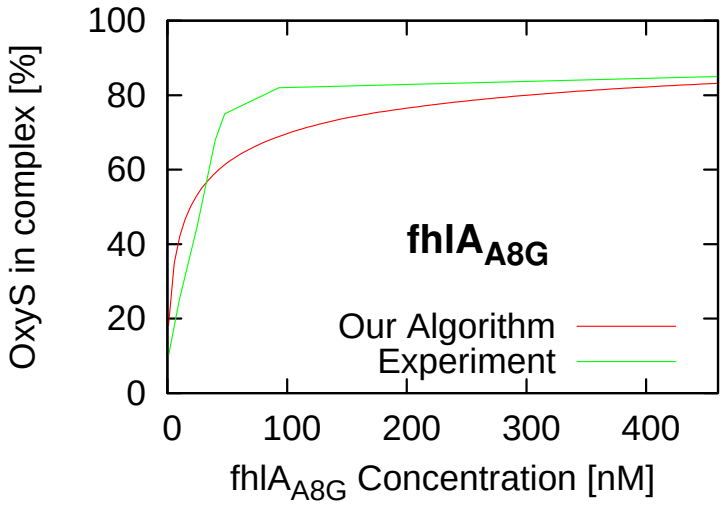
to obtain the equilibrium concentrations N. N^0 are the initial concentrations of single strands.

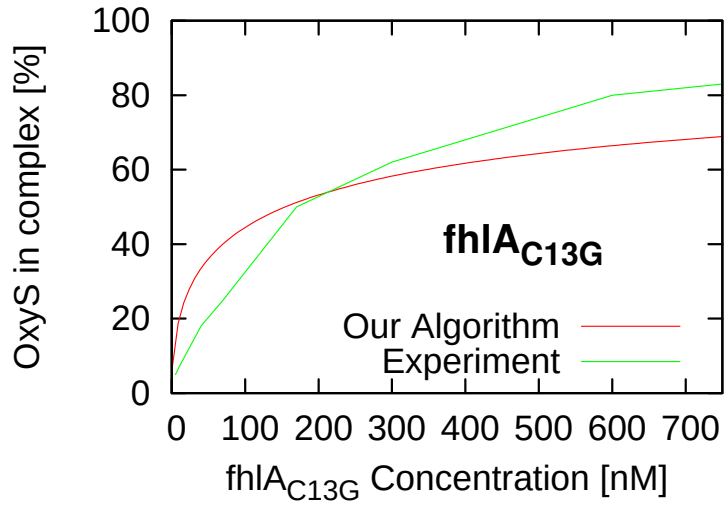
Equilibrium concentration of OxyS with wild type fhlA

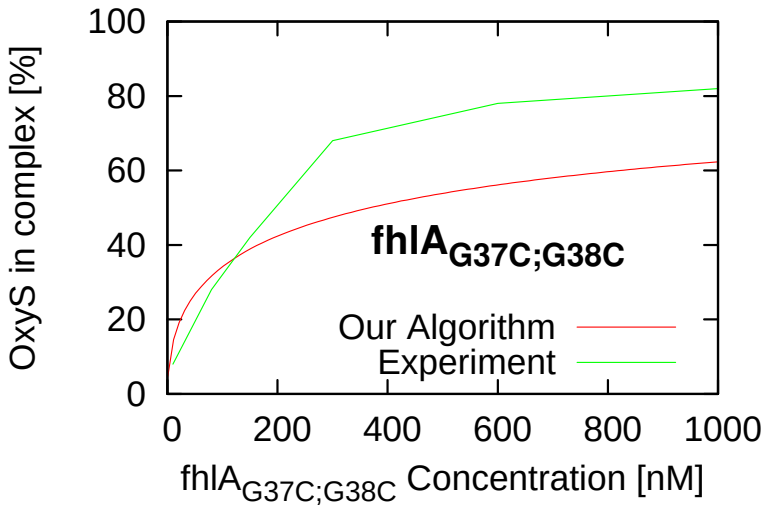


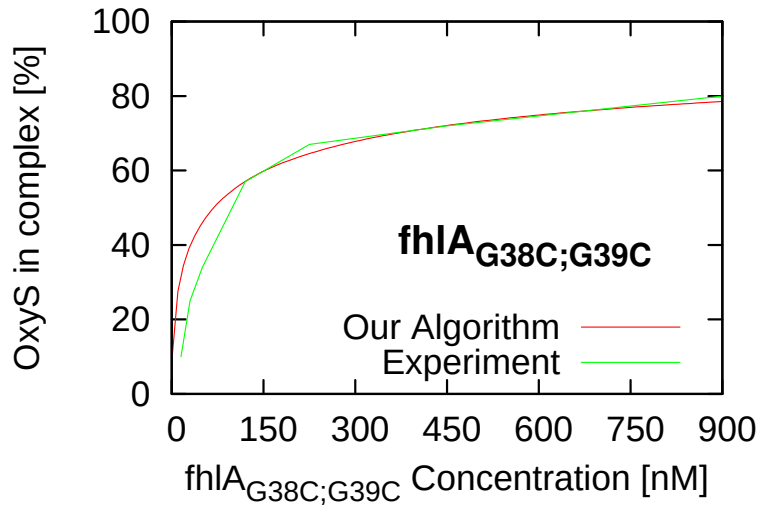
Init. [OxyS] = 2nM, [fhlA] = 0 to 1000nM

Equilibrium concentration of OxyS with fhlA mutants









Melting temperature prediction

Comparison of piRNA results over three data sets

Set	Size	Length	Avg error		
			piRNA	RNAcofold	UNAFold
I	9 short pairs	5-7nt	**1.48**°C	9.35°C	8.55°C
II	12 pairs	$\sim$ 20nt	**4.86**°C	22.97°C	9.12°C
III	62 pairs	$22-40$nt	**1.91**°C	14.34°C	26.53°C

Set	Size	Length	Spearman rank correlation		
			piRNA	RNAcofold	UNAFold
I	9 short pairs	5-7nt	**0.97**	**0.97**	0.57
II	12 pairs	$\sim$ 20nt	**0.41**	-0.03	0.1
III	62 pairs	$22-40$nt	**0.3**	-0.04	0.24

Promised base pairing probabilities

P^{l} and P^{la} examples

$$P^{l}_{i_R,j_R,i_S,j_S} = \sum_{\substack{1\le k_1<i_R \\ j_S<k_2\le L_S}} P^{la}_{k_1,j_R,i_S,k_2} \frac{(Q^{ls}_{k_1,i_R,j_S,k_2} + Q^{ls'}_{k_1,i_R,j_S,k_2} + Q^{le}_{k_1,i_R,j_S,k_2})Q^{l}_{i_R,j_R,i_S,j_S}}{Q^{la}_{k_1,j_R,i_S,k_2}},$$

$$P^{la}_{i_R,j_R,i_S,j_S} = \sum_{\substack{1\le k_1\le i_R \\ j_S\le k_2\le L_S}} P^{l}_{k_1,j_R,i_S,k_2} \frac{Q_{k_1,i_R-1}Q_{j_S+1,k_2}Q^{la}_{i_R,j_R,i_S,j_S}}{Q^{l}_{k_1,j_R,i_S,k_2}} + \sum_{\substack{1\le k_1<i_R \\ j_S\le k_2\le L_S}} P^{lb}_{k_1,j_R,i_S,k_2} \frac{Q^{lhh}_{k_1,i_R,j_S,k_2}Q^{la}_{i_R,j_R,i_S,j_S}}{Q^{lb}_{k_1,j_R,i_S,k_2}}.$$

More on this part will be presented by Peter Stadler.

Sampling from the Boltzmann ensemble

- Push $I(1,n,1,m)$ onto the stack.
- Iterate until the stack is empty, i.e. reaching a leaf (structure) in the recursions.
  - In each iteration, sample $0 \leq \alpha \leq 1$ uniformly at random.
  - Pop from the stack $top(i_R, j_R, i_S, j_S)$.
  - Pick a case of *top* according to α. For simplicity, we assume there is only one case here, i.e.

$$Q^{top} = \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q^{left}_{i_R,k_1,k_2,j_S} Q^{right}_{k_1+1,j_R,j_S,k_2+1}$$

  - Find k_1^*, k_2^* such that

$$\sum_{\substack{i_R \leq k_1 < k_1^* \\ i_S < k_2 \leq k_2^*}} \cdots \simeq \alpha \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} \cdots .$$

  - Push $left(i_R, k_1^*, k_2^*, j_S)$ and $right(k_1+1, j_R, i_S, k_2+1)$ onto the stack.

Sampling from the Boltzmann ensemble

- Push $I(1,n,1,m)$ onto the stack.
- Iterate until the stack is empty, i.e. reaching a leaf (structure) in the recursions.
  - In each iteration, sample $0 \leq \alpha \leq 1$ uniformly at random.
  - Pop from the stack $top(i_R, j_R, i_S, j_S)$.
  - Pick a case of *top* according to α. For simplicity, we assume there is only one case here, i.e.

$$Q^{top} = \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q^{left}_{i_R,k_1,k_2,j_S} Q^{right}_{k_1+1,j_R,i_S,k_2+1}$$

  - Find k_1^*, k_2^* such that

$$\sum_{\substack{i_R \leq k_1 < k_1^* \\ i_S < k_2 \leq k_2^*}} \cdots \simeq \alpha \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} \cdots .$$

  - Push $left(i_R, k_1^*, k_2^*, j_S)$ and $right(k_1+1, j_R, i_S, k_2+1)$ onto the stack.

Sampling from the Boltzmann ensemble

- Push $I(1,n,1,m)$ onto the stack.
- Iterate until the stack is empty, i.e. reaching a leaf (structure) in the recursions.
  - In each iteration, sample $0 \leq \alpha \leq 1$ uniformly at random.
  - Pop from the stack $top(i_R, j_R, i_S, j_S)$.
  - Pick a case of *top* according to α. For simplicity, we assume there is only one case here, i.e.

$$Q^{top} = \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q^{left}_{i_R,k_1,k_2,j_S} Q^{right}_{k_1+1,j_R,i_S,k_2+1}$$

  - Find k_1^*, k_2^* such that

$$\sum_{\substack{i_R \leq k_1 < k_1^* \\ i_S < k_2 \leq k_2^*}} \cdots \simeq \alpha \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} \cdots .$$

  - Push $left(i_R, k_1^*, k_2^*, j_S)$ and $right(k_1+1, j_R, i_S, k_2+1)$ onto the stack.

Sampling from the Boltzmann ensemble

- Push $I(1,n,1,m)$ onto the stack.
- Iterate until the stack is empty, i.e. reaching a leaf (structure) in the recursions.
  - In each iteration, sample $0 \leq \alpha \leq 1$ uniformly at random.
  - Pop from the stack $top(i_R, j_R, i_S, j_S)$.
  - Pick a case of *top* according to α. For simplicity, we assume there is only one case here, i.e.

$$Q^{top} = \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q^{left}_{i_R,k_1,k_2,j_S} Q^{right}_{k_1+1,j_R,i_S,k_2+1}$$

  - Find k_1^*, k_2^* such that

$$\sum_{\substack{i_R \leq k_1 < k_1^* \\ i_S < k_2 \leq k_2^*}} \cdots \simeq \alpha \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} \cdots .$$

  - Push $left(i_R, k_1^*, k_2^*, j_S)$ and $right(k_1+1, j_R, i_S, k_2+1)$ onto the stack.

Sampling from the Boltzmann ensemble

- Push $I(1,n,1,m)$ onto the stack.
- Iterate until the stack is empty, i.e. reaching a leaf (structure) in the recursions.
  - In each iteration, sample $0 \leq \alpha \leq 1$ uniformly at random.
  - Pop from the stack $top(i_R, j_R, i_S, j_S)$.
  - Pick a case of *top* according to α. For simplicity, we assume there is only one case here, i.e.

$$Q^{top} = \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q^{left}_{i_R,k_1,k_2,j_S} Q^{right}_{k_1+1,j_R,i_S,k_2+1}$$

  - Find k_1^*, k_2^* such that

$$\sum_{\substack{i_R \leq k_1 < k_1^* \\ i_S < k_2 \leq k_2^*}} \cdots \simeq \alpha \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} \cdots .$$

  - Push $left(i_R, k_1^*, k_2^*, j_S)$ and $right(k_1+1, j_R, i_S, k_2+1)$ onto the stack.

Sampling from the Boltzmann ensemble

- Push $I(1,n,1,m)$ onto the stack.
- Iterate until the stack is empty, i.e. reaching a leaf (structure) in the recursions.
  - In each iteration, sample $0 \leq \alpha \leq 1$ uniformly at random.
  - Pop from the stack $top(i_R, j_R, i_S, j_S)$.
  - Pick a case of *top* according to α. For simplicity, we assume there is only one case here, i.e.

$$Q^{top} = \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q^{left}_{i_R,k_1,k_2,j_S} Q^{right}_{k_1+1,j_R,i_S,k_2+1}$$

  - Find k_1^*, k_2^* such that

$$\sum_{\substack{i_R \leq k_1 < k_1^* \\ i_S < k_2 \leq k_2^*}} \cdots \simeq \alpha \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} \cdots .$$

  - Push $left(i_R, k_1^*, k_2^*, j_S)$ and $right(k_1+1, j_R, i_S, k_2+1)$ onto the stack.

Sampling from the Boltzmann ensemble

- Push $I(1,n,1,m)$ onto the stack.
- Iterate until the stack is empty, i.e. reaching a leaf (structure) in the recursions.
  - In each iteration, sample $0 \leq \alpha \leq 1$ uniformly at random.
  - Pop from the stack $top(i_R, j_R, i_S, j_S)$.
  - Pick a case of *top* according to α. For simplicity, we assume there is only one case here, i.e.

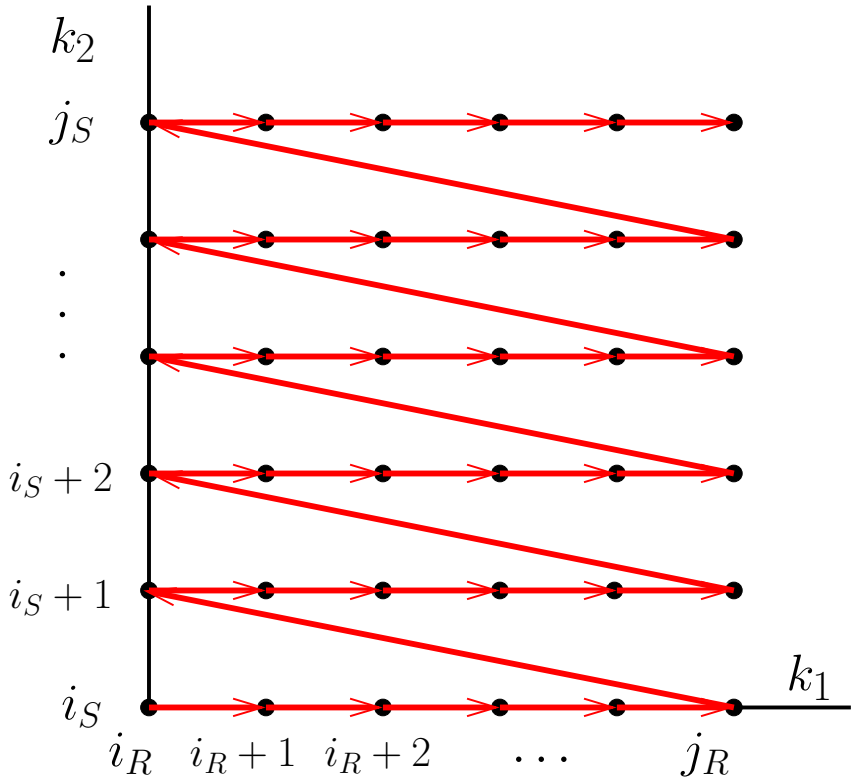
$$Q^{top} = \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q^{left}_{i_R,k_1,k_2,j_S} Q^{right}_{k_1+1,j_R,i_S,k_2+1}$$

  - Find k_1^*, k_2^* such that

$$\sum_{\substack{i_R \leq k_1 < k_1^* \\ i_S < k_2 \leq k_2^*}} \cdots \simeq \alpha \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} \cdots .$$

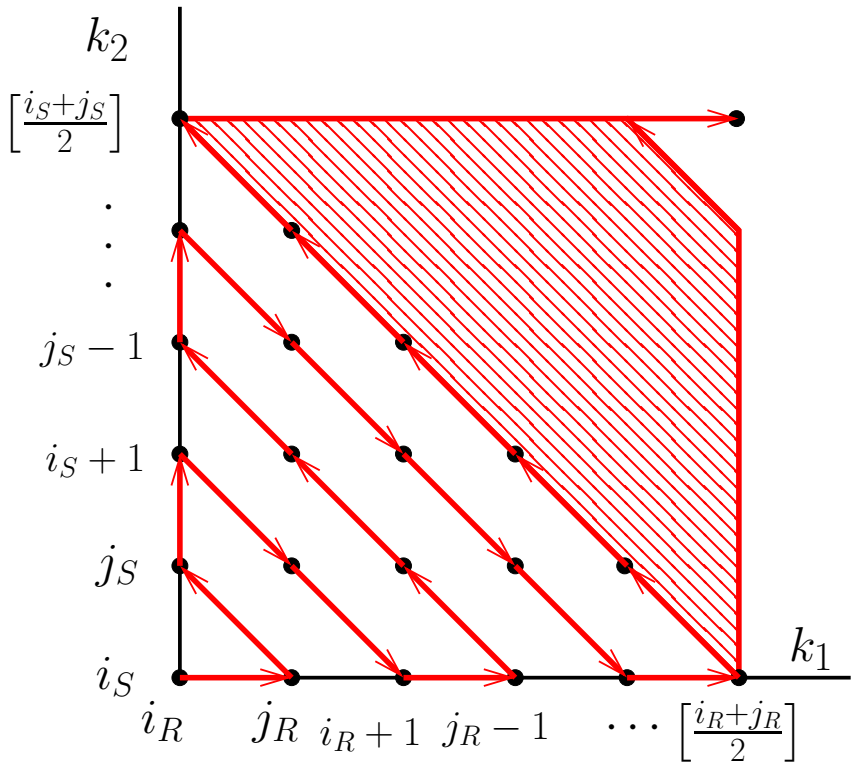
  - Push $left(i_R, k_1^*, k_2^*, j_S)$ and $right(k_1+1, j_R, i_S, k_2+1)$ onto the stack.

Fast Ponty-style sampling of the Boltzmann ensemble



Naïve traversal of indices

$O(n^4)$



Balanced traversal of indices

$O(n^2 \log n)$

Time and space complexity of piRNA

- $O(n^4m^2 + n^2m^4)$ time.
- $O(n^2m^2)$ space.
- about 100 tables in the dynamic programming.
- takes about 1 day on 64 CPUs with 150GB RAM for two 110nt RNAs (OxyS-fhlA).

Therefore, a fast heuristic is on demand for high-throughput applications, possibly as a filtering step.

Time and space complexity of piRNA

- $O(n^4m^2 + n^2m^4)$ time.
- $O(n^2m^2)$ space.
- about 100 tables in the dynamic programming.
- takes about 1 day on 64 CPUs with 150GB RAM for two 110nt RNAs (OxyS-fhlA).

Therefore, a fast heuristic is on demand for high-throughput applications, possibly as a filtering step.

Binding sites prediction

biRNA : a fast algorithm to predict simultaneous binding sites of two nucleic acids

Pros

- Predicts multiple simultaneous binding sites.
- Computes a more accurate local energy of binding.
- Considers zigzags and crossing interactions.
- Maintains tractability for existing cases in the literature.

Cons

- Approximates the intramolecular site accessibility energy.
- Its running time grows exponentially with the maximum number of simultaneous binding sites.

Binding sites prediction

biRNA : a fast algorithm to predict simultaneous binding sites of two nucleic acids

Pros

- Predicts multiple simultaneous binding sites.
- Computes a more accurate local energy of binding.
- Considers zigzags and crossing interactions.
- Maintains tractability for existing cases in the literature.

Cons

- Approximates the intramolecular site accessibility energy.
- Its running time grows exponentially with the maximum number of simultaneous binding sites.

Steps of the algorithm for R and S

1. For all short subsequences W, compute $P_u(W)$, the prob. of being unpaired (Mückstein *et al.* 2008).
2. Obtain $\mathcal{V}$, a short list of candidate sites.
3. For all pairs W_1, W_2, compute $P_u(W_1, W_2)$, the joint pairwise prob. of being simultaneously unpaired.
4. Build tree-structured Markov Random Fields (MRF) $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ to approximate the distribution of being simultaneously unpaired (Chow and Liu 1968).
5. Compute $Q^I_{W^R W^S}$, the interaction partition functions restricted to subsequences W^R and W^S using `piRNA`.
6. Compute a matching between $\mathcal{T}^R$ and $\mathcal{T}^S$ that minimizes the binding energy or equivalently maximizes the binding probability.

biRNA

Steps of the algorithm for R and S

1. For all short subsequences W, compute $P_u(W)$, the prob. of being unpaired (Mückstein *et al.* 2008).
2. Obtain $\mathcal{V}$, a short list of candidate sites.
3. For all pairs W_1, W_2, compute $P_u(W_1, W_2)$, the joint pairwise prob. of being simultaneously unpaired.
4. Build tree-structured Markov Random Fields (MRF) $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ to approximate the distribution of being simultaneously unpaired (Chow and Liu 1968).
5. Compute $Q^I_{W^R W^S}$, the interaction partition functions restricted to subsequences W^R and W^S using `piRNA`.
6. Compute a matching between $\mathcal{T}^R$ and $\mathcal{T}^S$ that minimizes the binding energy or equivalently maximizes the binding probability.

Steps of the algorithm for R and S

1. For all short subsequences W, compute $P_u(W)$, the prob. of being unpaired (Mückstein *et al.* 2008).
2. Obtain $\mathcal{V}$, a short list of candidate sites.
3. For all pairs W_1, W_2, compute $P_u(W_1, W_2)$, the joint pairwise prob. of being simultaneously unpaired.
4. Build tree-structured Markov Random Fields (MRF) $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ to approximate the distribution of being simultaneously unpaired (Chow and Liu 1968).
5. Compute $Q^I_{W^R W^S}$, the interaction partition functions restricted to subsequences W^R and W^S using `piRNA`.
6. Compute a matching between $\mathcal{T}^R$ and $\mathcal{T}^S$ that minimizes the binding energy or equivalently maximizes the binding probability.

Steps of the algorithm for R and S

1. For all short subsequences W, compute $P_u(W)$, the prob. of being unpaired (Mückstein *et al.* 2008).
2. Obtain $\mathcal{V}$, a short list of candidate sites.
3. For all pairs W_1, W_2, compute $P_u(W_1, W_2)$, the joint pairwise prob. of being simultaneously unpaired.
4. Build tree-structured Markov Random Fields (MRF) $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ to approximate the distribution of being simultaneously unpaired (Chow and Liu 1968).
5. Compute $Q^I_{W^R W^S}$, the interaction partition functions restricted to subsequences W^R and W^S using `piRNA`.
6. Compute a matching between $\mathcal{T}^R$ and $\mathcal{T}^S$ that minimizes the binding energy or equivalently maximizes the binding probability.

Steps of the algorithm for R and S

1. For all short subsequences W, compute $P_u(W)$, the prob. of being unpaired (Mückstein *et al.* 2008).
2. Obtain $\mathcal{V}$, a short list of candidate sites.
3. For all pairs W_1, W_2, compute $P_u(W_1, W_2)$, the joint pairwise prob. of being simultaneously unpaired.
4. Build tree-structured Markov Random Fields (MRF) $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ to approximate the distribution of being simultaneously unpaired (Chow and Liu 1968).
5. Compute $Q^I_{W^R W^S}$, the interaction partition functions restricted to subsequences W^R and W^S using `piRNA`.
6. Compute a matching between $\mathcal{T}^R$ and $\mathcal{T}^S$ that minimizes the binding energy or equivalently maximizes the binding probability.

Steps of the algorithm for R and S

1. For all short subsequences W, compute $P_u(W)$, the prob. of being unpaired (Mückstein *et al.* 2008).
2. Obtain $\mathcal{V}$, a short list of candidate sites.
3. For all pairs W_1, W_2, compute $P_u(W_1, W_2)$, the joint pairwise prob. of being simultaneously unpaired.
4. Build tree-structured Markov Random Fields (MRF) $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ to approximate the distribution of being simultaneously unpaired (Chow and Liu 1968).
5. Compute $Q^I_{W^R W^S}$, the interaction partition functions restricted to subsequences W^R and W^S using `piRNA`.
6. Compute a matching between $\mathcal{T}^R$ and $\mathcal{T}^S$ that minimizes the binding energy or equivalently maximizes the binding probability.

Exhaustive search to find matching
$M = \{(W_1^R, W_1^S), (W_2^R, W_2^S), \ldots, (W_k^R, W_k^S)\}$ that minimizes

$$\Delta G(M) = ED_u^R(M) + ED_u^S(M) + \Delta G_b^{RS}(M),$$

in which

$$ED_u^R(M) = -RT \log P_u^{R*}(W_1^R, W_2^R, \ldots, W_k^R)$$
$$ED_u^S(M) = -RT \log P_u^{S*}(W_1^S, W_2^S, \ldots, W_k^S)$$
$$\Delta G_b^{RS}(M) = -RT \sum_{1 \le i \le k} \log(Q^I_{W_i^R W_i^S} - Q_{W_i^R} Q_{W_i^S}).$$

R is the universal gas constant and T is temperature.

Experimental results

Multi-sites

Pair	Binding Site(s) Literature		biRNA Site(s)		RNAup Site	
OxyS-fhlA	[22,30] [98,104]	[95,87] [45,39]	(23,30) (96,104)	(94,87) (48,39)	- (96,104)	- (48,39)
CopA-CopT	[22,33] [48,56] [62,67]	[70,59] [44,36] [29,24]	(22,31) (49,57) (58,67)	(70,61) (43,35) (33,24)	- (49,67) -	- (43,24) -

Experimental results

Uni-sites

Pair	
GcvB	gltI
GcvB	argT
GcvB	dppA
GcvB	livJ
GcvB	livK
GcvB	oppA
GcvB	STM4351
MicA	lamB
MicA	ompA
DsrA	rpoS
RprA	rpoS
IstR	tisA
MicC	ompC
MicF	ompF
RyhB	sdhD
RyhB	sodB
SgrS	ptsG
$IncRNA_{54}$	repZ

Lengths: 71-253 nt

Running time: 10 min - 1 hour on 8 dual core CPUs and 20GB of RAM

Summary

- We presented piRNA an $O(n^4m^2 + n^2m^4)$-time $O(n^2m^2)$-space complexity algorithm for interaction partition function, base-pair probabilities, minimum free energy secondary structure, equilibrium concentrations, melting temperature, and some other derivatives of the partition function.
- piRNA outperforms all other alternatives and is available at http://compbio.cs.wayne.edu/chitsaz/.
- We presented biRNA , a fast RNA-RNA binding sites prediction algorithm.
- biRNA 's tree-structured MRF approximation is accurate enough for predicting binding sites and may be used in other applications.

Summary

- We presented `piRNA` an $O(n^4m^2 + n^2m^4)$-time $O(n^2m^2)$-space complexity algorithm for interaction partition function, base-pair probabilities, minimum free energy secondary structure, equilibrium concentrations, melting temperature, and some other derivatives of the partition function.
- `piRNA` outperforms all other alternatives and is available at `http://compbio.cs.wayne.edu/chitsaz/`.
- We presented `biRNA` , a fast RNA-RNA binding sites prediction algorithm.
- `biRNA` 's tree-structured MRF approximation is accurate enough for predicting binding sites and may be used in other applications.

Summary

- We presented `piRNA` an $O(n^4m^2 + n^2m^4)$-time $O(n^2m^2)$-space complexity algorithm for interaction partition function, base-pair probabilities, minimum free energy secondary structure, equilibrium concentrations, melting temperature, and some other derivatives of the partition function.
- `piRNA` outperforms all other alternatives and is available at `http://compbio.cs.wayne.edu/chitsaz/`.
- We presented `biRNA` , a fast RNA-RNA binding sites prediction algorithm.
- `biRNA` 's tree-structured MRF approximation is accurate enough for predicting binding sites and may be used in other applications.

Summary

- We presented `piRNA` an $O(n^4m^2 + n^2m^4)$-time $O(n^2m^2)$-space complexity algorithm for interaction partition function, base-pair probabilities, minimum free energy secondary structure, equilibrium concentrations, melting temperature, and some other derivatives of the partition function.
- `piRNA` outperforms all other alternatives and is available at `http://compbio.cs.wayne.edu/chitsaz/`.
- We presented `biRNA` , a fast RNA-RNA binding sites prediction algorithm.
- `biRNA` 's tree-structured MRF approximation is accurate enough for predicting binding sites and may be used in other applications.

Future work

- RNA design for positive and negative interactions.
- Better interaction energy model, which requires more data.
- Incorporation of non-canonical base pairs.

Future work

- RNA design for positive and negative interactions.
- Better interaction energy model, which requires more data.
- Incorporation of non-canonical base pairs.

Future work

- RNA design for positive and negative interactions.
- Better interaction energy model, which requires more data.
- Incorporation of non-canonical base pairs.

Acknowledgement

Collaborators

- Rolf Backofen, University of Freiburg, Germany
- Cenk Sahinalp, SFU, Canada
- Raheleh Salari

Funding

Michael Smith Foundation for
Health Research

MITACS

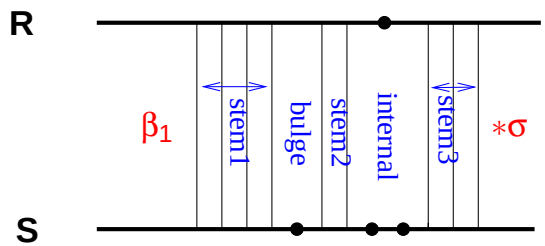
BCID
Bioinformatics for Combating Infectious Diseases

Deutsche
Forschungsgemeinschaft
DFG

Thanks for your attention!

Hybrid component

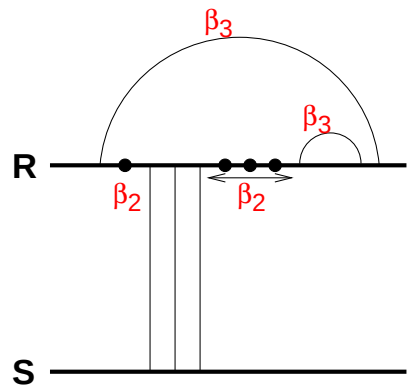
Example



$$G^{\text{hybrid}} = \beta_1 + \sigma(G^{stem_1} + G^{bulge} + G^{stem_2} + G^{internal} + G^{stem_3}).$$

Kissing loop

Example



$$G^{\text{kissing}} = 4\beta_2 + 2\beta_3.$$