INFO-RNA - A Fast Approach to Inverse RNA Folding

Anke Busch
Bioinformatics Group
Albert-Ludwigs-University Freiburg

Bled, February 2007

Outline

1. Introduction
2. The Algorithm
   - Overview
   - The Initializing Step
   - The Local Search Step
3. Results
   - Artificial Test Sets
   - Biological Test Sets
   - Discussion

The RNA Folding Problem

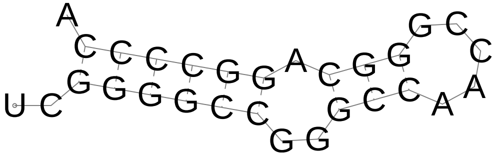
Aim: predicting the secondary structure of an RNA sequence

e.g.: 5'-UCGGGGCCGGGCCAACCGGGCAGGCCCCA-3'

The RNA Folding Problem

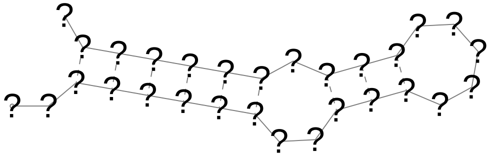
Aim: predicting the secondary structure of an RNA sequence

e.g.: 5'-UCGGGGCCGGGCCAACCGGGCAGGCCCCA-3'



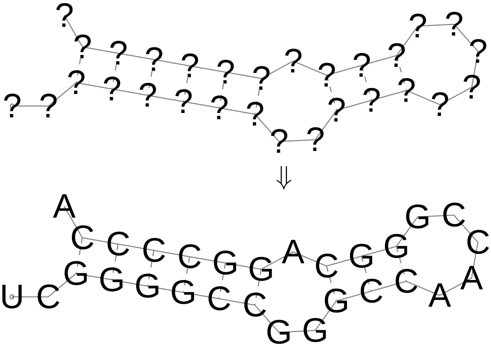
The Inverse RNA Folding Problem

Aim: designing an RNA sequence that folds into a target structure



The Inverse RNA Folding Problem

Aim: designing an RNA sequence that folds into a target structure



Overview and Definition

INFO-RNA

- new approach to INverse FOlding of RNA
- **Input**: RNA secondary structure T (pseudoknot-free) of length n (set of pairs (i_1, i_2), where $1 \leq i_1 < i_2 \leq n$)

Overview and Definition

INFO-RNA

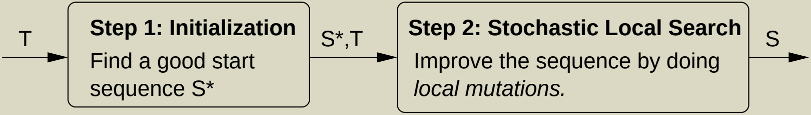
- new approach to INverse FOlding of RNA
- **Input**: RNA secondary structure T (pseudoknot-free) of length n (set of pairs (i_1, i_2), where $1 \leq i_1 < i_2 \leq n$)
- **Output**: RNA sequence $S = S_1 ... S_n$ that folds into T, where $S_i \in \{A, C, G, U\}$ for $1 \leq i \leq n$

Overview and Definition

INFO-RNA

- new approach to INverse FOlding of RNA
- **Input**: RNA secondary structure T (pseudoknot-free) of length n (set of pairs (i_1, i_2), where $1 \leq i_1 < i_2 \leq n$)
- **Output**: RNA sequence $S = S_1 ... S_n$ that folds into T, where $S_i \in \{A, C, G, U\}$ for $1 \leq i \leq n$

Algorithm



T

Step 1: Initialization

Find a good start sequence S*

S*,T

Step 2: Stochastic Local Search

Improve the sequence by doing *local mutations.*

S

The Initializing Step - 1

In and Out

Input: structure T

Output: sequence S^* (adopts T with the lowest possible energy)

$$S^* = \arg\min_{S'} e(S', T)$$

where $e(S', T)$ = free energy of S' folded into T

The Initializing Step - 1

In and Out

Input: structure T

Output: sequence S^* (adopts T with the lowest possible energy)

$$S^* = \arg\min_{S'} e(S', T)$$

where $e(S', T)$ = free energy of S' folded into T

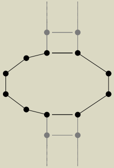
Remind!

$$e(structure) = \sum_{loop \in structure} e(loop)$$

The Initializing Step - 2

Energy Function

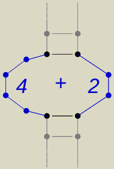
- free energies of structural elements (loops), depend on:



The Initializing Step - 2

Energy Function

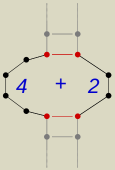
- free energies of structural elements (loops), depend on:
  - loop size



The Initializing Step - 2

Energy Function

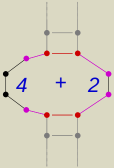
- free energies of structural elements (loops), depend on:
  - loop size
  - closing pairs



The Initializing Step - 2

Energy Function

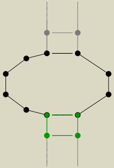
- free energies of structural elements (loops), depend on:
  - loop size
  - closing pairs + adjacent free bases



The Initializing Step - 2

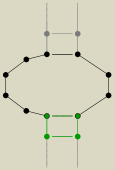
Energy Function

- free energies of structural elements (loops), depend on:
  - loop size
  - closing pairs + adjacent free bases
- each pair belongs to 2 elements → elements are linked



The Initializing Step - 2

Energy Function

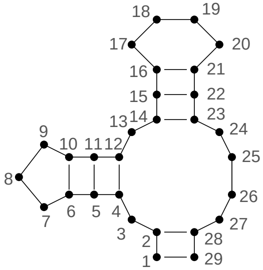


- free energies of structural elements (loops), depend on:
  - loop size
  - closing pairs + adjacent free bases
- each pair belongs to 2 elements → elements are linked

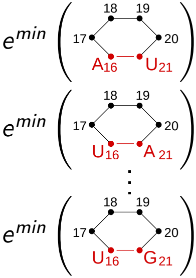
The Idea

- specify the minimum free energy of substructures depending on the closing base pair
- start with small substructures (hairpin loops)
- enlarge them gradually by 1 base pair

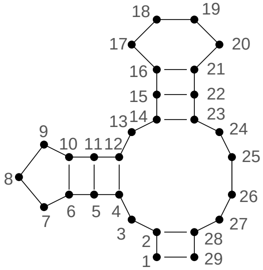
Example

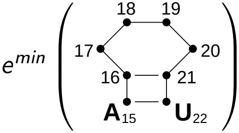


1. determine the mfe of a hairpin loop for all possible assignments of the closing pair (BP = {A-U, C-G, G-C, U-A, G-U, U-G})

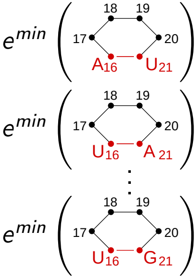


Example

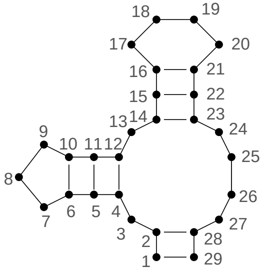




1. determine the mfe of a hairpin loop for all possible assignments of the closing pair (BP = {A-U, C-G, G-C, U-A, G-U, U-G})
2. fix the mfe of the substructure that is one base pair larger for all possible assignments of the last pair: (e.g. A-U)

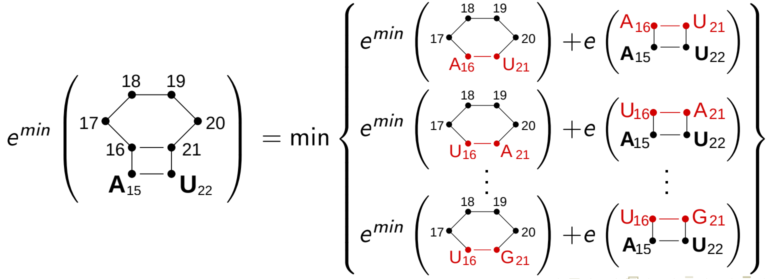


Example



1. determine the mfe of a hairpin loop for all possible assignments of the closing pair (BP $= \{$A-U, C-G, G-C, U-A, G-U, U-G$\}$)

2. fix the mfe of the substructure that is one base pair larger for all possible assignments of the last pair: (e.g. A-U)



The Order

Definition: (Base Pair Order)

The order in which base pairs of structure T are examined is defined as

$$(i_1, i_2) \prec (j_1, j_2) \quad \text{if and only if} \quad i_1 > j_1$$

where $(i_1, i_2), (j_1, j_2) \in T$ and $(i_1, i_2) \prec (j_1, j_2)$ means that base pair (i_1, i_2) is analyzed prior to base pair (j_1, j_2).

The Order

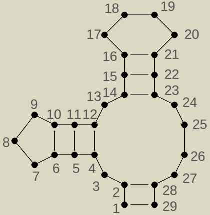
Definition: (Base Pair Order)

The order in which base pairs of structure T are examined is defined as

$$(i_1, i_2) \prec (j_1, j_2) \quad \text{if and only if} \quad i_1 > j_1$$

where $(i_1, i_2), (j_1, j_2) \in T$ and $(i_1, i_2) \prec (j_1, j_2)$ means that base pair (i_1, i_2) is analyzed prior to base pair (j_1, j_2).

Example



Order:

$(16, 21) \prec (15, 22) \prec (14, 23) \prec (6, 10) \prec (5, 11) \prec (4, 12) \prec (2, 28) \prec (1, 29)$

Predecessor Base Pairs

Definition: (Predecessor)

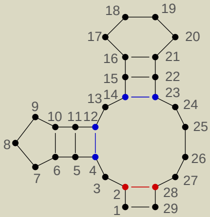
All pairs that are part of the structural element that is closed by the current base pair and that are smaller than the current one (conc. the order) are denoted as predecessors of the current pair.

Predecessor Base Pairs

Definition: (Predecessor)

All pairs that are part of the structural element that is closed by the current base pair and that are smaller than the current one (conc. the order) are denoted as predecessors of the current pair.

Example



Base pair	Predecessor(s)
$(16, 21)$	none
$(15, 22)$	$(16, 21)$
$(14, 23)$	$(15, 22)$
$(6, 10)$	none
$(5, 11)$	$(6, 10)$
$(4, 12)$	$(5, 11)$
$(2, 28)$	$(4, 12)$, $(14, 23)$
$(1, 29)$	$(2, 28)$

Dynamic Programming

D =

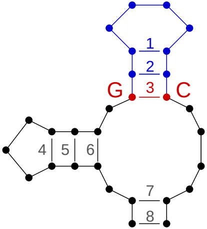
base pair no.	1 A−U	2 C−G	3 G−C	4 U−A	5 G−U	6 U−G
1						
2						
3						
4						
⋮						

- dynamic programming matrix filled with mfe's
- each row represents a base pair (numbered conc. the order)
- each column represents a possible pair assignment

Dynamic Programming

D =

base pair no.	1 A−U	2 C−G	3 G−C	4 U−A	5 G−U	6 U−G
1						
2						
3						
4						
⋮						



- dynamic programming matrix filled with mfe’s
- each row represents a base pair (numbered conc. the order)
- each column represents a possible pair assignment

Output of Step 1

S^*

1. smallest value of last row of D (mfe a sequence can have when folding into T)
2. traceback $\longrightarrow S^*$

$\Rightarrow$ no other sequence with lower energy when folding into T

Output of Step 1

S^*

1. smallest value of last row of D (mfe a sequence can have when folding into T)
2. traceback $\longrightarrow S^*$

$\Rightarrow$ no other sequence with lower energy when folding into T

But...

S^* can have less energy when folded into another structure

$\Rightarrow$ seconds step

The Local Search Step

Stochastic Local Search (SLS) - Overview

- finds local optima conc. an objective function

The Local Search Step

Stochastic Local Search (SLS) - Overview

- finds local optima conc. an objective function

Objective Function of INFO-RNA

- minimizing the structure distance between the mfe structure of the designed sequence and T

The Local Search Step

Stochastic Local Search (SLS) - Overview

- finds local optima conc. an objective function
- iterative mutations, allows moves to worse sequences

Objective Function of INFO-RNA

- minimizing the structure distance between the mfe structure of the designed sequence and T

The Local Search Step

Stochastic Local Search (SLS) - Overview

- finds local optima conc. an objective function
- iterative mutations, allows moves to worse sequences

Objective Function of INFO-RNA

- minimizing the structure distance between the mfe structure of the designed sequence and T

SLS - Search Criterion

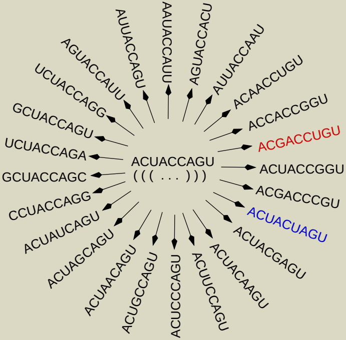
Retain a tested sequence if it has a better objective function than the current one. Otherwise, keep it with probability p.

Neighborhood

Sequence Neighbors

all sequences S' that differ from S either

- in one unbound position or
- in two paired positions



Test order of the neighbors

Test Order

Depends on a look-ahead of one mutation step:

1. calculate the energy of candidate sequences folded into T: $e(S', T)$
2. evaluate their energy difference to the energy of the current sequence S folded into T: $e(S, T) - e(S', T)$

Test order of the neighbors

Test Order

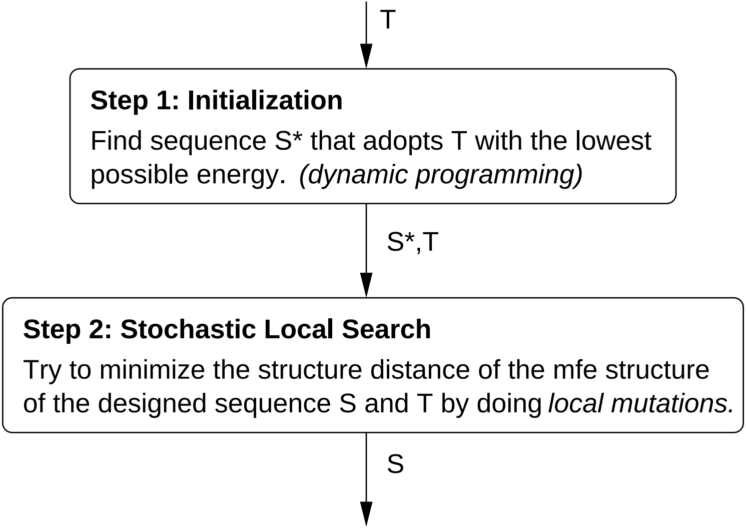
Depends on a look-ahead of one mutation step:

1. calculate the energy of candidate sequences folded into T: $e(S', T)$
2. evaluate their energy difference to the energy of the current sequence S folded into T: $e(S, T) - e(S', T)$

$$\Downarrow$$

The higher the difference, the earlier the candidate sequence is examined.

Summary



T

Step 1: Initialization

Find sequence S* that adopts T with the lowest possible energy. *(dynamic programming)*

S*,T

Step 2: Stochastic Local Search

Try to minimize the structure distance of the mfe structure of the designed sequence S and T by doing *local mutations.*

S

Results

Test sets

- artificial
- biological

Comparison with...

- RNAinverse (Hofacker *et al.*, 1994)
- RNA-SSD (Andronescu *et al.*, 2004)

Results

Test sets

- artificial
- biological

Comparison with...

- RNAinverse (Hofacker *et al.*, 1994)
- RNA-SSD (Andronescu *et al.*, 2004)

Successful run

mfe structure of final sequence $= T$ (otherwise: unsuccessful run)

Artificial Test Sets

Test sets Ia + Ib

- 300 artificially generated structures
- user-given features (size, loop sizes, stem lengths)

Artificial Test Sets

Test sets Ia + Ib

- 300 artificially generated structures
- user-given features (size, loop sizes, stem lengths)

Test set Ia

- size: ≤ 200

Test set Ib

- size: $300 - 700$

Artificial Test Sets

Test sets Ia + Ib

- 300 artificially generated structures
- user-given features (size, loop sizes, stem lengths)

Test set Ia

- size: ≤ 200

Test set Ib

- size: $300 - 700$

Results: Ia (100 runs per structure) + Ib (10 runs per structure)

	Ia (CSR)	**Ia** ($\bar{E}_T$)	**Ib** (CSR)	**Ib** ($\bar{E}_T$)
INFO-RNA	300/300	0.1	300/300	9.1
RNA-SSD	298/300	0.2	294/300	46.8
RNAinverse	294/300	41.9	1/300	-

CSR...fraction of struct. for which the algo. was successful in all runs
$\bar{E}_T$ ...average expected computation time

Biological Test Sets - 1

Test set II

- 308 computationally predicted structures of known RNA sequences (all annotated eukaryotic rRNA gene sequences of the Ribosomal Database Project)
- size: 220 – 1975, size-depending arrangement in classes

Biological Test Sets - 1

Test set II

- 308 computationally predicted structures of known RNA sequences (all annotated eukaryotic rRNA gene sequences of the Ribosomal Database Project)
- size: 220 – 1975, size-depending arrangement in classes

Results for test set II

Sizes in subset	220-400		400-900		900-1975	
	ASR	$\bar{E}_T$	ASR	$\bar{E}_T$	ASR	$\bar{E}_T$
INFO-RNA	100%	2.4	100%	93.3	100%	1447.4
RNA-SSD	93%	226.8	93%	285.3	81%	3043.9
RNAinverse	2.0%	-	0.3%	-	0.0%	-

ASR...average fraction of successful runs
$\bar{E}_T$ ...average expected computation time

Biological Test Sets - 2

Test set III

- structures from the biological literature
- pairs in pseudoknots are disregard

Biological Test Sets - 2

Test set III

- structures from the biological literature
- pairs in pseudoknots are disregard

Selected results for test set III (100 runs per structure)

		INFO-RNA		RNA-SSD	
	Size	SR	E_T	SR	E_T
VS Ribozyme from *Neurospora* mitochondria	167	100/100	0.1	100/100	0.3
R180 ribozyme	180	37/100 (63/100)(2)	194.0	58/100 (20/100)(2)	2267.8
Homo Sapiens RNase P RNA*	340	100/100	66.8	94/100	491.1
S20 mRNA from *E.coli*	372	100/100	110.8	87/100	728.2

SR...fraction of successful runs

E_T...expected computation time

*...originally pseudoknotted structure

Summary and Extensions

So far...

+ very fast and successful new approach to inverse RNA folding

+ outperforms other existing tools in most cases

Summary and Extensions

So far...

+ very fast and successful new approach to inverse RNA folding
+ outperforms other existing tools in most cases
- initializing sequence rather fixed, random sampling with random start sequence
- high GC content (G-C pairs are energetically most favorable)

Summary and Extensions

So far...

+ very fast and successful new approach to inverse RNA folding
+ outperforms other existing tools in most cases
- initializing sequence rather fixed, random sampling with random start sequence
- high GC content (G-C pairs are energetically most favorable)

Extentions

- sequence constraints
- allow some violations

Summary and Extensions

So far...

+ very fast and successful new approach to inverse RNA folding

+ outperforms other existing tools in most cases

- initializing sequence rather fixed, random sampling with random start sequence

- high GC content (G-C pairs are energetically most favorable)

Extentions

- sequence constraints
- allow some violations

Web server

http://www.bioinf.uni-freiburg.de/Software/INFO-RNA/

Finally

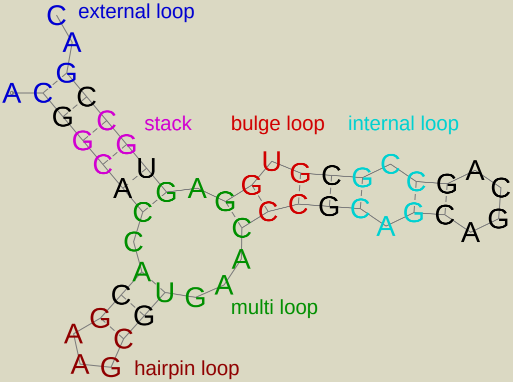
Thank you for your attention!

Finally

Thank you for your attention!

The Thermodynamic Model

Decomposition into Loops



energy: $e(structure) = \sum_{loop \in structure} e(loop)$

Recursion (simplified)

Pair *i* has no predecessor: (HL)

$$\forall\, a \in BP : \qquad D(i,a) = \min_{\text{free bases}} e(HL)$$

Recursion (simplified)

Pair i has no predecessor: (HL)

$$\forall\, a \in BP: \qquad D(i,a) = \min_{\text{free bases}} e(HL)$$

Pair i has exactly one predecessor:(stack, BL, IL)

$$\forall\, a \in BP: \;\; D(i,a) = \min_{b \in BP} \left\{ D(i-1,b) + \min_{\substack{\text{free bases}\\ \text{in } T_i^{i-1}}} e\left(T_i^{i-1} \middle| \begin{array}{l} i \rightarrow a \\ i-1 \rightarrow b \end{array} \right) \right\}$$

where T_i^{i-1} = structural element between base pairs $i-1$ and i

Recursion (simplified)

Pair i has no predecessor: (HL)

$$\forall\, a \in BP: \qquad D(i,a) = \min_{\text{free bases}} e(HL)$$

Pair i has exactly one predecessor:(stack, BL, IL)

$$\forall\, a \in BP: \; D(i,a) = \min_{b \in BP} \left\{ D(i-1,b) + \min_{\substack{\text{free bases}\\ \text{in } T_i^{i-1}}} e\left(T_i^{i-1} \,\middle|\, \begin{array}{l} i \rightarrow a \\ i-1 \rightarrow b \end{array} \right) \right\}$$

where T_i^{i-1} = structural element between base pairs $i-1$ and i

Pair i has more than one predecessors: (ML)

$$\forall\, a \in BP: \qquad D(i,a) = e(ML) + \min_{a_1,\ldots,a_s \in BP} \left\{ \sum_{k=1}^{s} D(p_k(i), a_k) \right\}$$

where $p_k(i)$ = k-th predecessor of base pair i

Complexity

- at most $3n$ values in D

no. of predecessor(s) of the base pair	max. no. of steps per base pair	complexity per entry
0	4^4	$O(1)$
1	$6 * 4^4$	$O(1)$
> 1	*straight forwardly:* exponential in the no. of predecessors *additional dynamic programming:* all closing pairs of all MLs $\rightarrow O(n)$	

$$\text{Complexity} = 3n * O(1) + O(n) = O(n)$$

Expected Time

Expected time for generating a solution E_T [in CPU seconds]

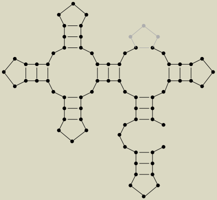
$$E_T = E_S + (\frac{1}{f_S} - 1)E_U$$

where	E_S	...	average time for a successful run
	E_U	...	average time for an unsuccessful run
	f_S	...	fraction of successful runs

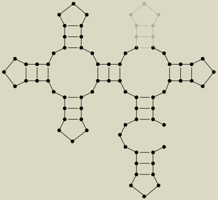
Artificial Test Set Ic

Test set Ic

Ic-1



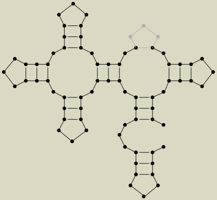
Ic-2



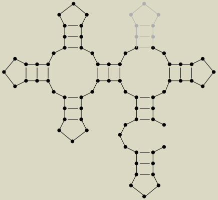
Artificial Test Set Ic

Test set Ic

Ic-1



Ic-2



Results for test set Ic (100 runs per structure)

	Ic-1(2) (SR)	**Ic-1**(2) (E_T)	**Ic-2** (SR)	**Ic-2** (E_T)
INFO-RNA	(100/100)	(6.1)	99/100	0.6
RNA-SSD	(87/100)	(2484)	62/100	1996.8
RNAinverse	(79/100)	(9.4)	44/100	21.3

SR...fraction of successful runs
E_T...expected computation time