Computational Design of New and Recombinant Selenoproteins

Rolf Backofen and Anke Busch

Friedrich-Schiller-University Jena

Institute of Computer Science

Chair for Bioinformatics

Outline

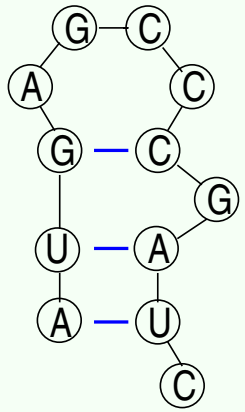
1. Biological Introduction
2. The Computational Problem
3. The Algorithm
4. Results
5. Conclusion and Future Work

Biological Introduction - 1

RNA Sequence: string over {A,C,G,U}, e.g. AUGAGCCCGAUC

Codon

RNA Secondary Structure: set of complementary and connected base pairs



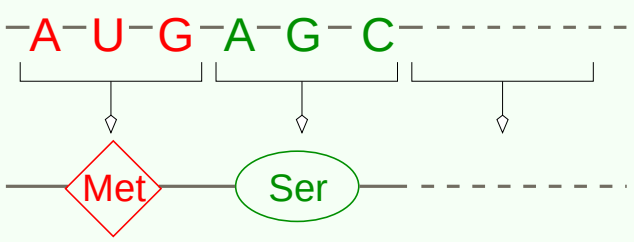
Biological Introduction - 1

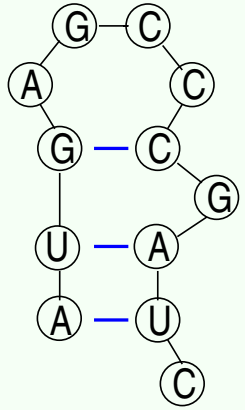
RNA Sequence: string over {A,C,G,U}, e.g. AUGAGCCCGAUC

Codon

RNA Secondary Structure: set of complementary and connected base pairs

mRNA → Protein:





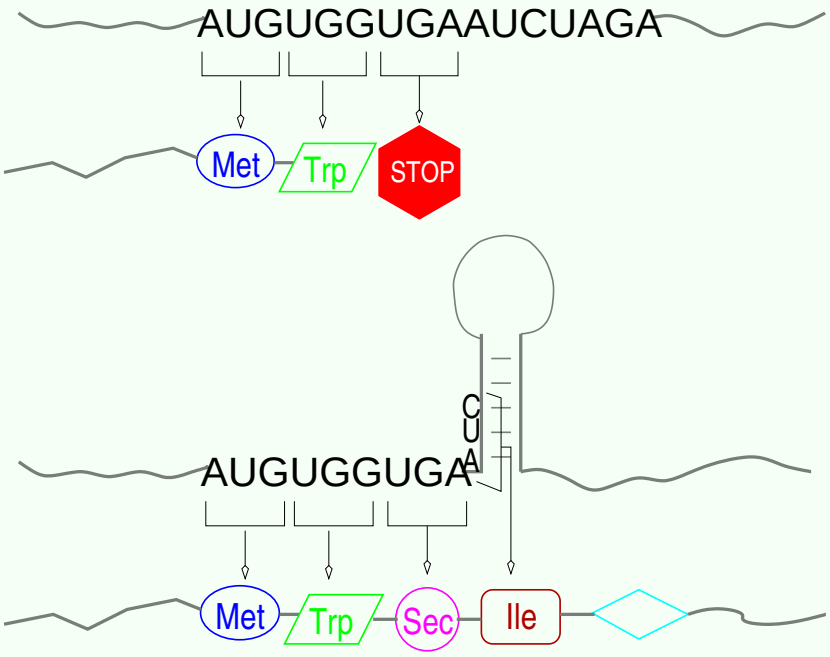
= linear translation

STOP-codon: UGA, UAA, UAG (symbolize end of translation)

Biological Introduction - 2

Selenocysteine:

- new amino acid, encoded by the STOP-codon UGA
- inserted, if UGA is followed by a *SECIS-element*



SECIS-element:

- specific mRNA sequence
- forms hairpin-like structure

Selenoproteins:

- contain *selenocysteine*
- important to human health, component of major metabolic pathways
- greatly enhanced enzymatic activities compared to the cysteine homologues

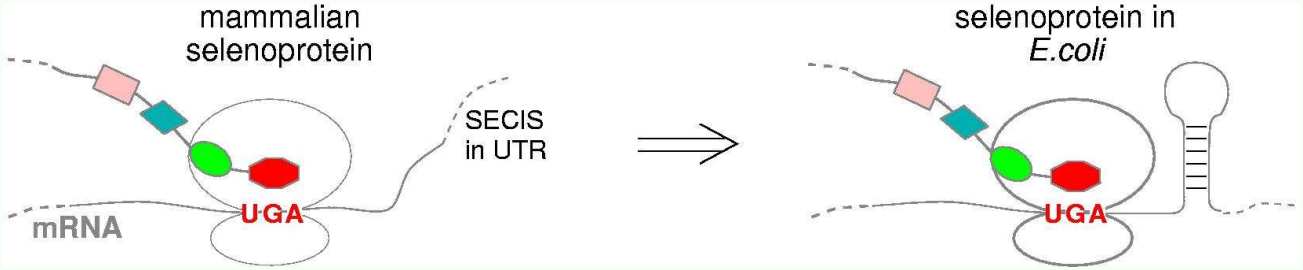
The Biological Problem

Protein expression system: *E.coli*

Eukaryotes: SECIS-element *outside* the coding sequence

E.coli (bacteria): SECIS-element immediately downstream the UGA-codon

→ located *inside* the coding part of the protein



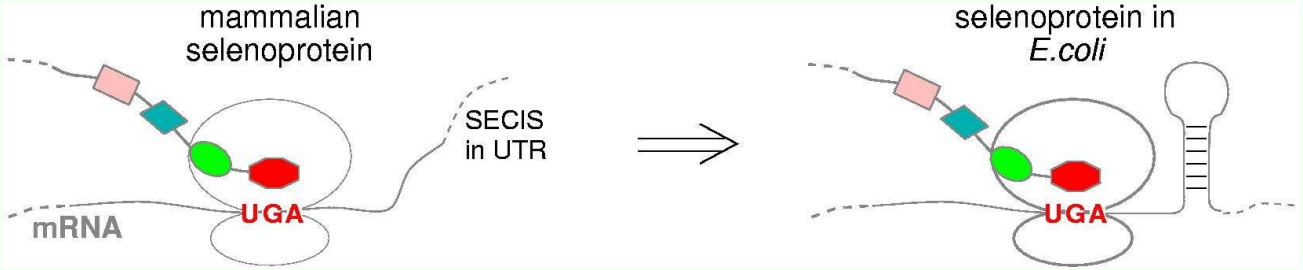
The Biological Problem

Protein expression system: *E.coli*

Eukaryotes: SECIS-element *outside* the coding sequence

E.coli (bacteria): SECIS-element immediately downstream the UGA-codon

→ located *inside* the coding part of the protein



⇓

Problem:

compromise between

- quality of the SECIS-element and
- changes in protein sequence

The Computational Problem

- **given:** G ... typical SECIS secondary structure

  $S = S_1 \ldots S_{3n}$ ... nucleotide sequence (SECIS-consensus)

  $A = A_1 \ldots A_n$ ... original amino acid sequence

The Computational Problem

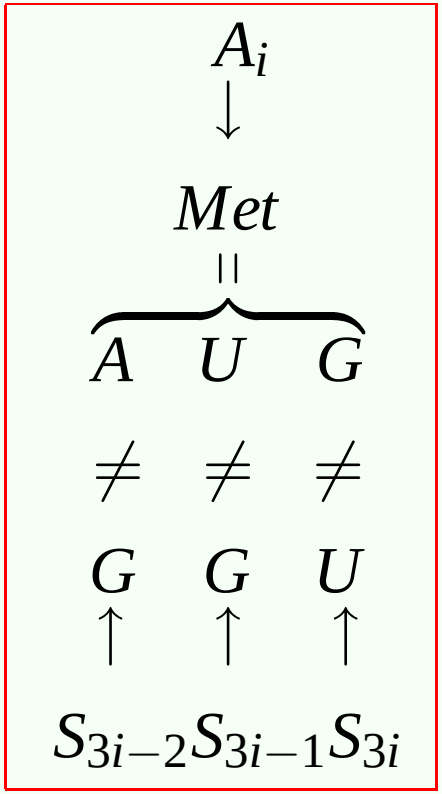
- **given:** G ... typical SECIS secondary structure
  $S = S_1 \ldots S_{3n}$ ... nucleotide sequence (SECIS-consensus)
  $A = A_1 \ldots A_n$ ... original amino acid sequence
- **wanted:** $N = N_1 \ldots N_{3n}$ ... mRNA sequence that
  - can adopt G
  - has maximum similarity with S
  - encodes amino acid sequence A' with maximal similarity to A

$$
\begin{array}{lccccccc}
A = & A_1 & \ldots & A_i & \ldots & A_n \\
 & \sim & & \sim & & \sim \\
A' = & A'_1 & \ldots & A'_i & \ldots & A'_n \\
N = & \overbrace{N_1 N_2 N_3} & \ldots & \overbrace{N_{3i-2} N_{3i-1} N_{3i}} & \ldots & \overbrace{N_{3n-2} N_{3n-1} N_{3n}} \\
 & \sim\ \sim\ \sim & & \sim\ \sim\ \sim & & \sim\ \sim\ \sim \\
S = & S_1 S_2 S_3 & \ldots & S_{3i-2} S_{3i-1} S_{3i} & \ldots & S_{3n-2} S_{3n-1} S_{3n}
\end{array}
$$

- **similarity function:**
  $F_{A_i}^{S_{3i-2} S_{3i-1} S_{3i}} (N_{3i-2} N_{3i-1} N_{3i})$
- **conditions** at conserved pos. (nucleotide + amino acid)

Contrary Conditions

Possible Problem: contrary conditions at nucleotide and amino acid level



$\Rightarrow$ insertions/deletions & *optional* bonds

The Computational Problem - Extentions - 1

(1) Insertions and Deletions (at amino acid level, represented by vector *t* with $t_i \in \{-1, 0, 1\}$)

A :	A_1	$-$	A_2
A' :	A'_1	A'_2	A'_3
N :	$\overbrace{N_1N_2N_3}$	$\overbrace{N_4N_5N_6}$	$\overbrace{N_7N_8N_9}$
S :	$\underbrace{S_1S_2S_3}$	$\underbrace{S_4S_5S_6}$	$\underbrace{S_7S_8S_9}$
t:	0	+1	0
SECpos	1	2	3

- insertion of A'_2, $t_2 = +1$
- A'_2 has no counterpart in A
- similarity: $IP + F^{S_4S_5S_6}(N_4N_5N_6)$

The Computational Problem - Extentions - 1

(1) Insertions and Deletions (at amino acid level, represented by vector t with $t_i \in \{-1, 0, 1\}$)

A :	A_1	$-$	A_2
A' :	A'_1	A'_2	A'_3
N :	$N_1N_2N_3$	$N_4N_5N_6$	$N_7N_8N_9$
S :	$S_1S_2S_3$	$S_4S_5S_6$	$S_7S_8S_9$
t:	0	+1	0
SECpos	1	2	3

· insertion of A'_2, $t_2 = +1$

· A'_2 has no counterpart in A

· similarity: $IP + F^{S_4S_5S_6}(N_4N_5N_6)$

A :	A_1	A_2	A_3	A_4
A' :	A'_1	$-$	A'_2	A'_3
N :	$N_1N_2N_3$	$---$	$N_4N_5N_6$	$N_7N_8N_9$
S :	$S_1S_2S_3$	$---$	$S_4S_5S_6$	$S_7S_8S_9$
t:	0	-1		0
SECpos	1	2		3

· deletion of A_2, $t_2 = -1$

· compare $N_4N_5N_6$ with A_3

· similarity: $\begin{cases} -\infty & \text{if amino acid cond. at 2} \\ DP + F_{A_3}^{S_4S_5S_6}(N_4N_5N_6) & \text{otherwise} \end{cases}$

The Computational Problem - Extentions - 2

Similarity Function: $f(L_i, a_i, t_i)$, where L_i ... *codon* corresponding to $N_{3i-2}N_{3i-1}N_{3i}$

$t_i \in \{-1 \text{ (deletion)}, 0 \text{ (subst.)}, +1 \text{ (insertion)}\}$

$a_i = \sum_{j=1}^{i} t_j$

$$f_i(L_i, a_i, 0) = F_{A_{i-a_i}}^{S_{3i-2}S_{3i-1}S_{3i}}(N_{3i-2}N_{3i-1}N_{3i})$$

$$f_i(L_i, a_i, +1) = IP + F^{S_{3i-2}S_{3i-1}S_{3i}}(N_{3i-2}N_{3i-1}N_{3i})$$

$$f_i(L_i, a_i, -1) = \begin{cases} -\infty & \text{if amino acid cond. at } i-(a_i - t_i) \\ DP + f_i(L_i, a_i, 0) & \text{otherwise} \end{cases}$$

The Computational Problem - Extentions - 2

Similarity Function: $f(L_i, a_i, t_i)$, where L_i ... *codon* corresponding to $N_{3i-2}N_{3i-1}N_{3i}$

$t_i \in \{-1 \text{ (deletion)}, 0 \text{ (subst.)}, +1 \text{ (insertion)}\}$

$a_i = \sum_{j=1}^{i} t_j$

$$f_i(L_i, a_i, 0) = F_{A_{i-a_i}}^{S_{3i-2}S_{3i-1}S_{3i}}(N_{3i-2}N_{3i-1}N_{3i})$$

$$f_i(L_i, a_i, +1) = IP + F^{S_{3i-2}S_{3i-1}S_{3i}}(N_{3i-2}N_{3i-1}N_{3i})$$

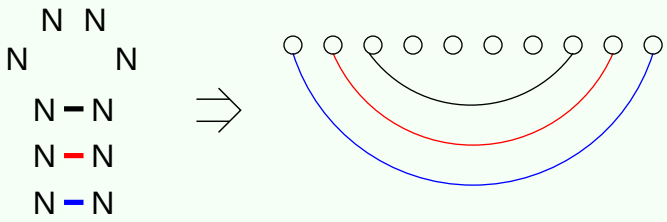
$$f_i(L_i, a_i, -1) = \begin{cases} -\infty & \text{if amino acid cond. at } i-(a_i - t_i) \\ DP + f_i(L_i, a_i, 0) & \text{otherwise} \end{cases}$$

(2) Optional Bonds

bond	...	mandatory bond
optional bond	...	not fixed, but of advantage if formed

The Algorithm - Input

- structure represented by a graph:



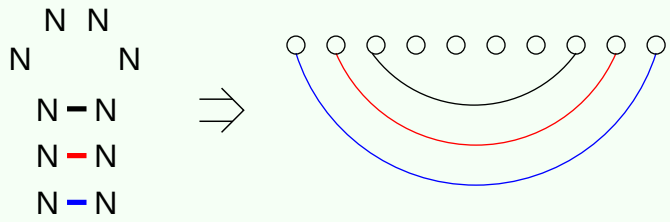
$\Rightarrow$ **outer-planar edgelabeled graph** $G = (V, E, Lab)$, with

- $V = \{v_1, \ldots, v_{3n}\}$
- $\forall \{v_k, v_l\} \in E(G)$:

$Lab(v_k, v_l)$	Condition	Meaning
+1	$N_k \in N_l^C$	bond
+2	$N_k \in \{A, C, G, U\}$	optional bond
-1	$N_k \notin N_l^C$	prohibited bond
-2	$N_k \in \{A, C, G, U\}$	opt. prohib. bond

The Algorithm - Input

- structure represented by a graph:



$\Rightarrow$ **outer-planar edgelabeled graph** $G = (V, E, Lab)$, with

- $V = \{v_1, \ldots, v_{3n}\}$
- $\forall \{v_k, v_l\} \in E(G)$:

$Lab(v_k, v_l)$	Condition	Meaning
+1	$N_k \in N_l^C$	bond
+2	$N_k \in \{A, C, G, U\}$	optional bond
-1	$N_k \notin N_l^C$	prohibited bond
-2	$N_k \in \{A, C, G, U\}$	opt. prohib. bond

- **amino acid sequence** $A = A_1 A_2 \ldots A_n$
- **amino acid + nucleotide conditions**
- n **similarity functions** $f_1, \ldots, f_n$, where f_i is associated with $\{v_{3i-2}, v_{3i-1}, v_{3i}\}$

The Algorithm - Output

- **Vectors** $N = (N_1, ..., N_{3n}) \in \{A, C, G, U\}^{3n}$ and
  $t = (t_1, ..., t_n) \in \{-1, 0, 1\}^n$,

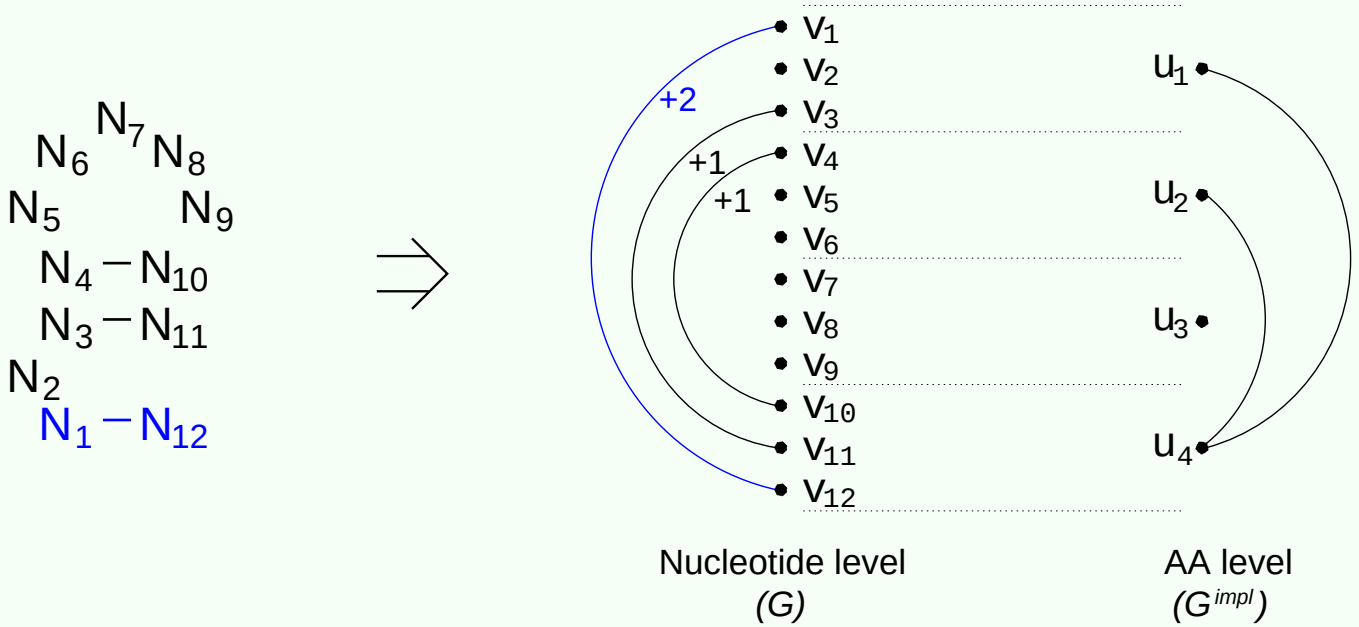
  with:

  - N_k assigned to v_k
  - t ... succession of insertions and deletions
  - all complementarity conditions hold (given by the labeling)
  - $\sum_{i=1}^{n} f_i(L_i, a_i, \boldsymbol{t}_i)$ **is maximized** (L_i correponds to $N_{3i-2}N_{3i-1}N_{3i}$)

The Algorithm - Preparing Definitions - 1

Definition 1 *Given a structure graph G, we define the* ***implied graph*** G^{impl} *as a graph on the vertices* $V(G^{impl}) = \{u_1, ..., u_n\}$ *with:*

$$E(G^{impl}) = \left\{ \{u_i, u_j\} \;\middle|\; \begin{array}{l} \exists r \in \{3i-2, 3i-1, 3i\} : \\ \exists s \in \{3j-2, 3j-1, 3j\} : (v_r, v_s) \in E(G) \end{array} \right\}$$



(independent of index: $u \in V(G^{impl})$, $v \in V(G)$)

The Algorithm - Preparing Definitions - 2

Compatibility: $\equiv_{E(G)}$

$$(u_i, L_i) \equiv_{E(G)} (u_j, L_j) = \begin{cases} \text{true}, & \text{if } \{u_i, u_j\} \notin E(G^{impl}) \\ \text{true}, & \text{if } \{u_i, u_j\} \in E(G^{impl}) \text{ and} \\ & \text{the corresp. nucleotides of } L_i \text{ and } L_j \\ & \text{satisfy the complementarity cond. of } G \\ \text{false}, & \text{otherwise} \end{cases}$$

The Algorithm - Preparing Definitions - 2

Compatibility: $\equiv_{E(G)}$

$$(u_i, L_i) \equiv_{E(G)} (u_j, L_j) = \begin{cases} \text{true}, & \text{if } \{u_i, u_j\} \notin E(G^{impl}) \\ \text{true}, & \text{if } \{u_i, u_j\} \in E(G^{impl}) \text{ and} \\ & \text{the corresp. nucleotides of } L_i \text{ and } L_j \\ & \text{satisfy the complementarity cond. of } G \\ \text{false}, & \text{otherwise} \end{cases}$$

Central Function:

$$\begin{bmatrix} l & \ldots & \text{insertions - deletions left} \\ s & \ldots & \text{insertions - deletions inside interval} \end{bmatrix}$$

$$w_n^1(L_1, L_n, m, l, s)$$

$$= \max_{\substack{L_2 \ldots L_{n-1} \\ t(s)}} \left\{ \sum_{1<j\leq n} f_j(L_j, l + \sum_{g=1}^{j-i} t_g, t_{j-i}) \;\middle|\; \begin{array}{l} \text{to } L_1 \ldots L_n \text{ corresp. nucl. satisfy } E(G), \\ m \text{ real. opt. bonds at } L_1 \ldots L_n \end{array} \right\}$$

The Algorithm - 1

- **Value of Interest:** $\max_{\substack{L_1, L_n, m \\ |l| \leq 1, |s| \leq n-1}} \{w_n^1(L_1, L_n, m, l, s) + f_1(L_1, l, l)\}$

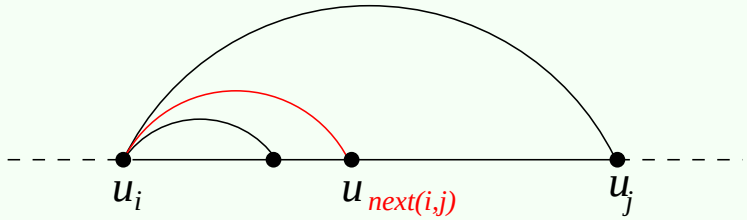
$\Rightarrow$ solved by dynamic programming

The Algorithm - 1

- **Value of Interest:** $\max_{\substack{L_1, L_n, m \\ |l| \leq 1, |s| \leq n-1}} \{w_n^1(L_1, L_n, m, l, s) + f_1(L_1, l, l)\}$

  $\Rightarrow$ solved by dynamic programming

- **Subproblems:**
  - split the problem on $\{1, \ldots, n\}$ into two parts, solve subproblems optimally
  - vertex to split: $next(i,j) = \begin{cases} r & \text{if } u_r \text{ is the farthest vertex in} \\ & \{u_i, \ldots, u_{j-1}\} \text{ adjacent to } u_i \\ i+1 & \text{otherwise} \end{cases}$



The Algorithm - 2

Recurrence Theorem: $\mathbf{w^i_{i+k}}(L_i, L_{i+k}, m, l, s)$ *equals*

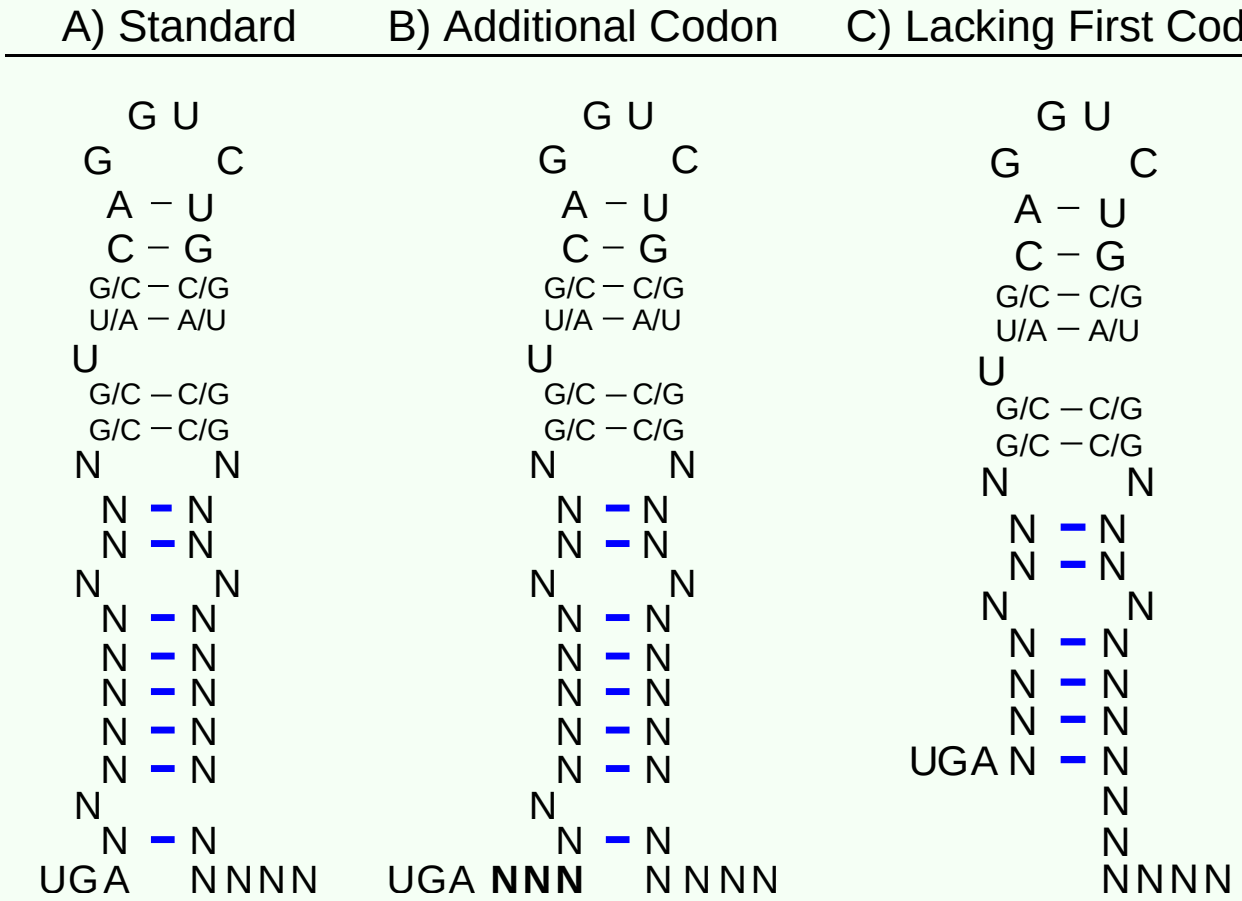
$$\begin{cases} -\infty & \text{if } (u_i, L_i) \not\equiv_{E(G)} (u_{i+k}, L_{i+k}) \\ \max\limits_{\substack{L_p \\ \text{splits } (m_1,m_2) \text{ of } m \\ \text{splits } (s_1,s_2) \text{ of } s}} \left(\mathbf{w}^i_p(L_i, L_p, m_1, l, s_1) + \mathbf{w}^p{}_{i+k}(L_p, L_{i+k}, m_2, l+s_1, s_2) \right) & \text{if } (u_i, L_i) \equiv_{E(G)} (u_{i+k}, L_{i+k}) \end{cases}$$

where $p = next(i, i+k)$ *and* $m = m_1 + m_2 + opt^{L_i}_{L_{i+k}}$ *with*

$opt^{L_i}_{L_{i+k}}$ = *number of realized optional bonds only between codons* L_i *and* L_{i+k}.

Results - 1

SECIS-element of *E.coli*:



from Liu et al., NAR 1998

Results - 2

Mammalian methionine sulfoxide reductase B (MsrB): (also analyzed by Bar-Noy et al., 2002)

A)	mouse MsrB	... **U I F S S S L K F V P K G K E** ...	Sim = 65
	new_MsrB	... **U I F S S L P G L V P K G K E** ...	Sim = **43**
	MsrB[Bar-Noy et al.]	... **U I F S T V A G L H P K G K E** ...	Sim = 35
	new_mRNA	... UGAAUUUUCUCUUCGCUACCAGGUCUGGUGCCAAAAGGAAAAGAA	
	new_struc	... .(((((.((.((.(((((....)))))).)).))))).....	opt = 8 (9)
B)	mouse MsrB	... **U I F S S S L K - F V P K G K E** ...	
	new_MsrB	... **U I F S S S L P G L V P K G K E** ...	Sim = $57 + IP$
	new_mRNA	... UGAAUAUUUUCCUCUUCGCUACCAGGUCUGGUGCCAAAAGGAAAAGAA	
	new_struc	... .(.(((((.((.((.(((((....)))))).)).))))))....	opt = 9 (9)
C)	mouse MsrB	... **U I F S S S L K F V P K G K E** ...	
	new_MsrB	... **U I F S L P - G L V P K G K E** ...	Sim = $41 + DP$
	new_mRNA	... UGAAUCUUUUCGCUACCAGGUCUGGUGCCAAAAGGUAAAGAA	
	new_struc	... ...(((((((.((.(((((....)))))).))))))).....	opt = 6 (7)

Results for Human Selenoproteins

A)

GPX2	... G T T T R D F T Q L N E L Q ...
GPX2'	... G T T T L A G L Q L N E L Q ...
SelN	... G S G R T L R E T V L E S S P ...
SelN'	... G - G R T L P G L V L E S S P ...
$SelP_{59}$	... S Y S L R Y I L L K K S L E ...
$SelP_{59}$'	... S Y S L L A G L L R K S L E ...

B)

SelN	... G S G R T L R E T V L E S S P ...
SelN'	... G S G R T L P G L V L E S S P ...
$SelP_{59}$	... Y L C I I E A S K L E D L R V ...
$SelP_{59}$'	... Y L C I Y V A G L L E D L R V ...
$SelP_{318}$	... Q C K E N L P S L C S C Q G L ...
$SelP_{318}$'	... Q C K E N L P G L V A C Q G L ...

C)

IOD2	... P P F T S Q L P A F R K L ...
IOD2'	... P P F L A G L Q A F R K L ...
IOD3	... P P F M A R M S A F Q R L ...
IOD3'	... P P F L A G L Q A F Q R L ...
GPX1	... G T T V R D Y T Q M N E L ...
GPX1'	... G T T V A G L L Q M N E L ...
GPX3	... G L T G Q Y I E L N A L Q ...
GPX3'	... G L T L P G L E L N A L Q ...
SelT	... G Y R R V - F E E Y M R V ...
SelT'	... G Y R L P G L E E Y M R V ...
SelV	... S Y S L R Y I L L K K S L ...
SelV'	... S Y S L A G L L L K K S L ...
SelW	... G Y K S K Y L Q L K K K L ...
SelW'	... G Y R L A G L Q L K K K L ...
SPS2	... G C K V P Q E A L L K L L ...
SPS2'	... G C R V P G L V L L K L L ...

Conclusion and Future Work

Conclusion:

- inserted bacterial SECIS-element in mRNA of eukaryotic selenoproteins
- only few amino acids changed

Conclusion and Future Work

Conclusion:

- inserted bacterial SECIS-element in mRNA of eukaryotic selenoproteins
- only few amino acids changed

Future Work:

- consider folding probability additionally (contrary)
- increase probability: local search methods
- keep a certain similarity

Thanks

Thanks to my supervisor Rolf Backofen.

Thanks

Thanks to my supervisor Rolf Backofen.

I thank you for your attention.