

Fixed-parameter algorithms for protein similarity search under mRNA structure constraints



UNIVERSITÀ DI PISA

Andrea Bozzi
Tiziano De Matteis

Università di Pisa

22 maggio 2009

Sommario

Il nostro seminario tratta di un problema di ottimizzazione su grafi (MRSO)

- 1 Motivazioni biologiche
- 2 Preliminari
- 3 A_{OP}
- 4 APX e Fixed-Parameters Algorithms
- 5 Due parametri naturali per MRSO
- 6 Cutwidth e A_{CUT}
- 7 Conclusioni e problemi aperti



Sommario

Il nostro seminario tratta di un problema di ottimizzazione su grafi (MRSO)

- 1 Motivazioni biologiche
- 2 Preliminari
- 3 A_{OP}
- 4 APX e Fixed-Parameters Algorithms
- 5 Due parametri naturali per MRSO
- 6 Cutwidth e A_{CUT}
- 7 Conclusioni e problemi aperti



Motivazioni biologiche mRNA e struttura secondaria

...ACGUGCCACG
AUUCAACGUGG
CACAG...



Figura: Struttura secondaria di un frammento di mRNA

Motivazioni biologiche Selenocisteina e SECIS

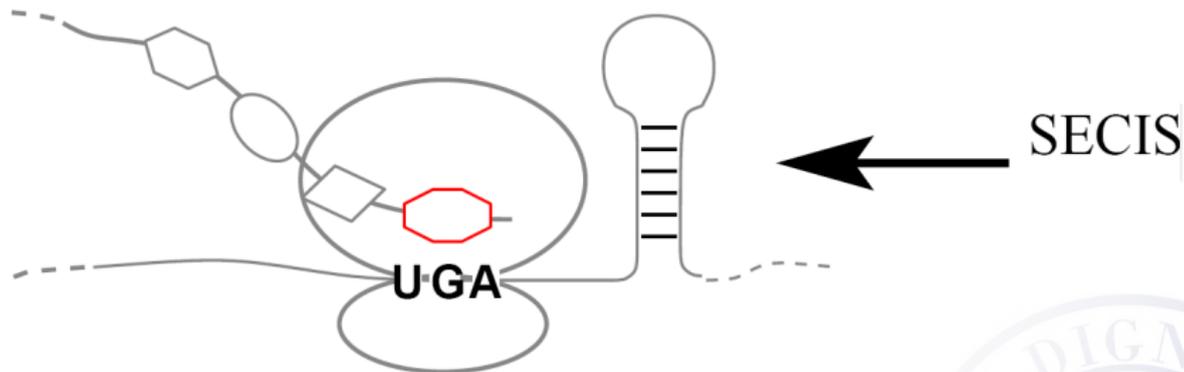


Figura: Inserzione dell'elemento SECIS



Sommario

- 1 Motivazioni biologiche
- 2 Preliminari
- 3 A_{OP}
- 4 APX e Fixed-Parameters Algorithms
- 5 Due parametri naturali per MRSO
- 6 Cutwidth e A_{CUT}
- 7 Conclusioni e problemi aperti



Preliminari

Formalizzazione

Sequenza mRNA: $S = s_1 \dots s_{3n}$

i-esimo codone: $s_{3i-2}s_{3i-1}s_{3i}$

Coppie complementari: $\{A, U\}\{G, C\}\{G, U\}$

Funzioni di similarit : $F = \{f_1, \dots, f_n\} \quad f_i : \Sigma^3 \rightarrow \mathbb{Q}$

Grado di similarit  tra S e T: $\sum_{i=1}^n f_i(t_{3i-2}t_{3i-1}t_{3i})$

Vincoli di struttura: $\Gamma \subseteq \{\{i, j\} \mid 1 \leq i < j \leq 3n\}$

Nucleotidi compatibili rispetto a $\Gamma \Leftrightarrow \{t_i, t_j\}$ complementari $\forall \{i, j\} \in \Gamma$



Preliminari

Formalizzazione

Sequenza mRNA: $S = s_1 \dots s_{3n}$

i-esimo codone: $s_{3i-2}s_{3i-1}s_{3i}$

Coppie complementari: $\{A, U\}\{G, C\}\{G, U\}$

Funzioni di similarit : $F = \{f_1, \dots, f_n\} \quad f_i : \Sigma^3 \rightarrow \mathbb{Q}$

Grado di similarit  tra S e T: $\sum_{i=1}^n f_i(t_{3i-2}t_{3i-1}t_{3i})$

Vincoli di struttura: $\Gamma \subseteq \{\{i, j\} \mid 1 \leq i < j \leq 3n\}$

Nucleotidi compatibili rispetto a $\Gamma \Leftrightarrow \{t_i, t_j\}$ complementari $\forall \{i, j\} \in \Gamma$



Preliminari

Formalizzazione

Sequenza mRNA: $S = s_1 \dots s_{3n}$

i-esimo codone: $s_{3i-2}s_{3i-1}s_{3i}$

Coppie complementari: $\{A, U\}\{G, C\}\{G, U\}$

Funzioni di similarit : $F = \{f_1, \dots, f_n\} \quad f_i : \Sigma^3 \rightarrow \mathbb{Q}$

Grado di similarit  tra S e T: $\sum_{i=1}^n f_i(t_{3i-2}t_{3i-1}t_{3i})$

Vincoli di struttura: $\Gamma \subseteq \{\{i, j\} \mid 1 \leq i < j \leq 3n\}$

Nucleotidi compatibili rispetto a $\Gamma \Leftrightarrow \{t_i, t_j\}$ complementari $\forall \{i, j\} \in \Gamma$



Preliminari

Formalizzazione

Sequenza mRNA: $S = s_1 \dots s_{3n}$

i-esimo codone: $s_{3i-2}s_{3i-1}s_{3i}$

Coppie complementari: $\{A, U\}\{G, C\}\{G, U\}$

Funzioni di similarit : $F = \{f_1, \dots, f_n\} \quad f_i : \Sigma^3 \rightarrow \mathbb{Q}$

Grado di similarit  tra S e T: $\sum_{i=1}^n f_i(t_{3i-2}t_{3i-1}t_{3i})$

Vincoli di struttura: $\Gamma \subseteq \{\{i, j\} \mid 1 \leq i < j \leq 3n\}$

Nucleotidi compatibili rispetto a $\Gamma \Leftrightarrow \{t_i, t_j\}$ complementari $\forall \{i, j\} \in \Gamma$



Preliminari

Formalizzazione

Sequenza mRNA: $S = s_1 \dots s_{3n}$

i-esimo codone: $s_{3i-2}s_{3i-1}s_{3i}$

Coppie complementari: $\{A, U\}\{G, C\}\{G, U\}$

Funzioni di similarit : $F = \{f_1, \dots, f_n\} \quad f_i : \Sigma^3 \rightarrow \mathbb{Q}$

Grado di similarit  tra S e T: $\sum_{i=1}^n f_i(t_{3i-2}t_{3i-1}t_{3i})$

Vincoli di struttura: $\Gamma \subseteq \{\{i, j\} \mid 1 \leq i < j \leq 3n\}$

Nucleotidi compatibili rispetto a $\Gamma \Leftrightarrow \{t_i, t_j\}$ complementari $\forall \{i, j\} \in \Gamma$



Preliminari

Formalizzazione

Sequenza mRNA: $S = s_1 \dots s_{3n}$

i-esimo codone: $s_{3i-2}s_{3i-1}s_{3i}$

Coppie complementari: $\{A, U\}\{G, C\}\{G, U\}$

Funzioni di similarit : $F = \{f_1, \dots, f_n\} \quad f_i : \Sigma^3 \rightarrow \mathbb{Q}$

Grado di similarit  tra S e T: $\sum_{i=1}^n f_i(t_{3i-2}t_{3i-1}t_{3i})$

Vincoli di struttura: $\Gamma \subseteq \{\{i, j\} \mid 1 \leq i < j \leq 3n\}$

Nucleotidi compatibili rispetto a $\Gamma \Leftrightarrow \{t_i, t_j\}$ complementari $\forall \{i, j\} \in \Gamma$



Preliminari

Formalizzazione

Sequenza mRNA: $S = s_1 \dots s_{3n}$

i-esimo codone: $s_{3i-2}s_{3i-1}s_{3i}$

Coppie complementari: $\{A, U\}\{G, C\}\{G, U\}$

Funzioni di similarit : $F = \{f_1, \dots, f_n\} \quad f_i : \Sigma^3 \rightarrow \mathbb{Q}$

Grado di similarit  tra S e T: $\sum_{i=1}^n f_i(t_{3i-2}t_{3i-1}t_{3i})$

Vincoli di struttura: $\Gamma \subseteq \{\{i, j\} \mid 1 \leq i < j \leq 3n\}$

Nucleotidi compatibili rispetto a $\Gamma \Leftrightarrow \{t_i, t_j\}$ complementari $\vee \{i, j\} \notin \Gamma$



Preliminari

MRSO

Definizione MRSO

Sia F un insieme di n funzioni di similarit  per una data sequenza di mRNA lunga $3n$ e sia Γ un insieme di vincoli strutturali.

MRSO richiede di trovare una sequenza di mRNA che sia compatibile con Γ e che raggiunga il pi  alto punteggio di similarit  possibile rispetto ad F



Preliminari

Teoria dei grafi

Grafo lineare $G \Leftrightarrow V(G) = \{1, 2, \dots, |V(G)|\}$

Definizione grafo di struttura implicata da Γ

G_Γ definito da

$V(G_\Gamma) = \{1, 2, \dots, n\} \rightarrow$ codoni

$E(G_\Gamma) = \{\{i, j\} \mid \exists \{x, y\} \in \Gamma : x \in \{3i-2, 3i-1, 3i\} \wedge y \in \{3j-2, 3j-1, 3j\}\}$

N.B. G_Γ é subcubico

Dato un sottoinsieme di vertici $V \subseteq V(G_\Gamma)$, $G_\Gamma[V]$ denota il sottografo di G_Γ indotto da V , ovvero con V come vertici e $E(G_\Gamma) \cap (V \times V)$ come archi. Similmente, $G_\Gamma[E]$ é il sottografo di G_Γ con vertici $\{i \mid \{i, j\} \in E\}$ e come archi gli elementi di E .



Preliminari

Teoria dei grafi

Grafo lineare $G \Leftrightarrow V(G) = \{1, 2, \dots, |V(G)|\}$

Definizione grafo di struttura implicata da Γ

G_Γ definito da

$V(G_\Gamma) = \{1, 2, \dots, n\} \rightarrow$ codoni

$E(G_\Gamma) = \{\{i, j\} \mid \exists \{x, y\} \in \Gamma : x \in \{3i-2, 3i-1, 3i\} \wedge y \in \{3j-2, 3j-1, 3j\}\}$

N.B. G_Γ é subcubico

Dato un sottoinsieme di vertici $V \subseteq V(G_\Gamma)$, $G_\Gamma[V]$ denota il sottografo di G_Γ indotto da V , ovvero con V come vertici e $E(G_\Gamma) \cap (V \times V)$ come archi. Similmente, $G_\Gamma[E]$ é il sottografo di G_Γ con vertici $\{i \mid \{i, j\} \in E\}$ e come archi gli elementi di E

Preliminari

Teoria dei grafi (2)

Due archi $\{i, j\}$ e $\{i', j'\}$ si **incrociano** in G_Γ se $i < i' < j < j'$ oppure $i' < i < j' < j$.

N.B. due archi incrociati possono non incrociarsi riordinando $V(G_\Gamma)$

Definizione grafo planare esterno

Se esiste un ordinamento di $V(G_\Gamma)$ che non introduce incroci di archi allora G_Γ è detto planare esterno

Se G_Γ planare esterno \Rightarrow MRSO è risolubile in $O(n)$



Preliminari

Teoria dei grafi (2)

Due archi $\{i, j\}$ e $\{i', j'\}$ si **incrociano** in G_Γ se $i < i' < j < j'$ oppure $i' < i < j' < j$.

N.B. due archi incrociati possono non incrociarsi riordinando $V(G_\Gamma)$

Definizione grafo planare esterno

Se esiste un ordinamento di $V(G_\Gamma)$ che non introduce incroci di archi allora G_Γ è detto planare esterno

Se G_Γ planare esterno \Rightarrow MRSO è risolubile in $O(n)$



Preliminari

Teoria dei grafi (2)

Due archi $\{i, j\}$ e $\{i', j'\}$ si **incrociano** in G_Γ se $i < i' < j < j'$ oppure $i' < i < j' < j$.

N.B. due archi incrociati possono non incrociarsi riordinando $V(G_\Gamma)$

Definizione grafo planare esterno

Se esiste un ordinamento di $V(G_\Gamma)$ che non introduce incroci di archi allora G_Γ é detto planare esterno

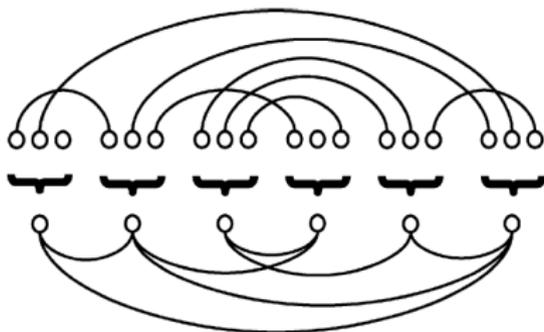
Se G_Γ planare esterno \Rightarrow MRSO è risolubile in $O(n)$



Preliminari

Esempio sui grafi

Vincoli strutturali dei nucleotidi: $\{1, 4\}$, $\{2, 17\}$, $\{5, 16\}$, $\{6, 10\}$, $\{7, 14\}$,
 $\{8, 13\}$, $\{9, 12\}$, $\{15, 18\}$



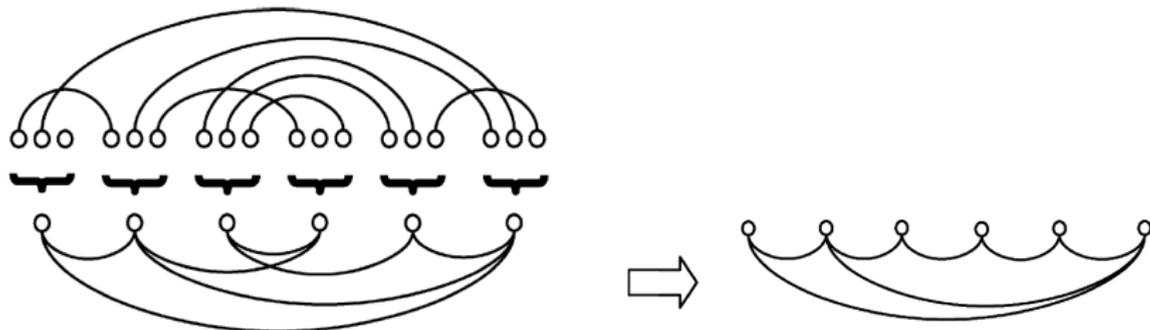
Archi del grafo di struttura implicata:

$\{1, 2\}$, $\{1, 6\}$, $\{2, 4\}$, $\{2, 6\}$, $\{3, 4\}$, $\{3, 5\}$, $\{3, 6\}$



Preliminari

Esempio sui grafi



(c) Il grafo di struttura implicata é planare esterno scambiando il terzo e il quarto nodo

Sommario

- 1 Motivazioni biologiche
- 2 Preliminari
- 3 A_{OP}
- 4 APX e Fixed-Parameters Algorithms
- 5 Due parametri naturali per MRSO
- 6 Cutwidth e A_{CUT}
- 7 Conclusioni e problemi aperti



A_{OP}

Prerequisiti

- G_{Γ} planare esterno (caso struttura SECIS)
- Ogni nodo in G ha al piú grado uno
- Assumiamo che ogni vertice di G_{Γ} sia uguale al suo indice $V(G_{\Gamma})[i] = i$
- Due codoni L_i e L_j **sono validi per** $\{u_i, u_j\}$ **rispetto a** $E(G)$ se i loro nucleotidi soddisfano le condizioni di complementariet  impo te da $E(G)$



A_{OP}

Compatibilità tra codoni

Compatibilità tra due codoni $\equiv_{E(G)}$

Dati due codoni L_i e L_j la definiamo per ogni $1 \leq i, j \leq n$ come:

$$(i, L_i) \equiv_{E(G)} (j, L_j) = \begin{cases} \text{vero} & \text{se } \{u_i, u_j\} \notin E(G_\Gamma) \\ \text{vero} & \text{se } \{u_i, u_j\} \in E(G_\Gamma) \text{ e } L_i \text{ e } L_j \text{ sono validi} \\ & \text{per } \{u_i, u_j\} \text{ rispetto a } E(G) \\ \text{false} & \text{altrimenti} \end{cases}$$



A_{OP}

Funzione best-score

Definiamo la funzione

$$w(i, i+k, L_i, L_{i+k}) =$$

$$\max_{L_{i+1}, \dots, L_{i+k-1}} \left\{ \sum_{i \leq j \leq i+k} f_j(L_j) \mid L_i, \dots, L_{i+k} \text{ soddisfa } E(G) \mid [i, \dots, i+k] \right\}$$

Se l'argomento del max risulta vuoto allora w vale $-\infty$

Caso base di w

$$w(i, i+1, L_i, L_{i+1}) = \begin{cases} f_i(L_i) + f_{i+1}(L_{i+1}) & \text{se } (i, L_i) \equiv_{E(G)} (i+1, L_{i+1}) \\ -\infty & \text{altrimenti} \end{cases}$$

A_{OP}

Funzione next

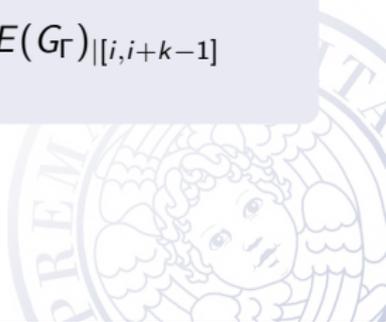
Applichiamo al problema la tecnica del *divide et impera*

Dato un intervallo $[i \dots i + k] \in V(G_\Gamma)$, abbiamo 2 possibili divisioni:

- $[i \dots i + 1]$ e $[i + 1 \dots i + k]$
- $[i \dots p]$ e $[p \dots i + k]$

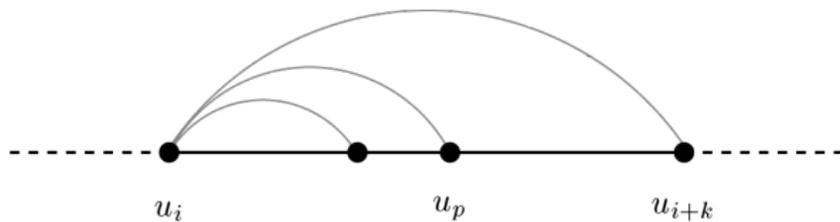
Dato un vertice i l'arco $\{i, p\}$ é detto massimale se p é il piú lontano vertice a cui i é connesso

$$\text{next}(i, i + k) = \begin{cases} i + r & \text{se } \{i, i + r\} \text{ é massimale in } E(G_\Gamma)_{|[i, i+k-1]} \\ i + 1 & \text{altrimenti} \end{cases}$$



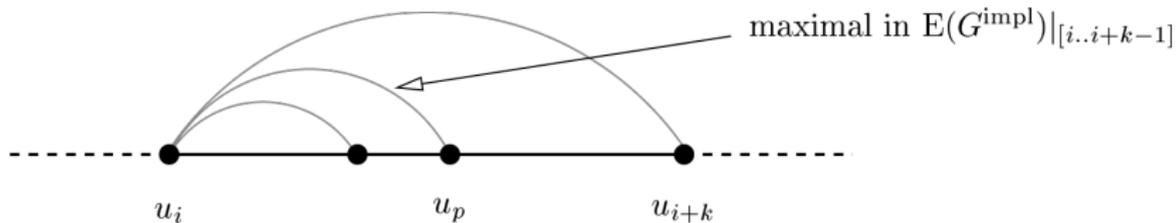
A_{OP}

Funzione next - Esempio



A_{OP}

Funzione next - Esempio



A_{OP} Correttezza A_{OP}

Teorema ricorsione

Sia (G, f_1, \dots, f_n) un'istanza di MRSO. Per $k \geq 2$, $1 \leq i \leq n - k$ sia $p = \text{next}(i, i + k)$ allora

$$w(i, i + k, L_i, L_{i+k}) = \begin{cases} -\infty & \text{se } (i, L_i) \not\equiv_{E(G)} (i + k, L_{i+k}) \\ \max_{L_p} \begin{pmatrix} w(i, p, L_i, L_p) \\ +w(p, i + k, L_p, L_{i+k}) \\ -f_p(L_p) \end{pmatrix} & \\ \text{se } (i, L_i) \equiv_{E(G)} (i + k, L_{i+k}) & \end{cases}$$

A_{OP}

Correttezza $A_{OP}(2)$

Dim:

- Primo caso \rightarrow trivial
- Secondo caso $\rightarrow G_{\Gamma}$ planare esterno $\Rightarrow \nexists$ archi da $[i + 1, \dots, p - 1]$ a $[p + 1, \dots, i + k]$
Per la correttezza rispetto $E(G)$ dei sottointervalli deriva dal caso base e dall'assunzione di partenza



A_{OP}

Proprietà di A_{OP} al massimo livello di ricorsione

- L'algoritmo coincide con l'implementazione di w per poi trovare $\max_{L_1, L_n} w(1, n, L_1, L_n)$
- G_Γ subcubico $\Rightarrow p = next(1, n)$ in al massimo tre passi
- Dati $w(1, p, L_1, L_p)$ e $w(p, n, L_p, L_n) \forall L_1, L_p, L_n$ calcolare $\max_{L_1, L_n} w(1, n, L_1, L_n)$ costa $O(64^3 + 3)$
- Il numero sottoproblemi é al massimo n

Complessità $A_{OP} = O((64^3 + 3)n)$



Sommario

- 1 Motivazioni biologiche
- 2 Preliminari
- 3 A_{OP}
- 4 APX e Fixed-Parameters Algorithms
- 5 Due parametri naturali per MRSO
- 6 Cutwidth e A_{CUT}
- 7 Conclusioni e problemi aperti



APX e Fixed-parameter algorithms

APX

$MRSO \in APX - H$

APX

Insieme dei problemi di ottimizzazione $\in NPO$ aventi algoritmi di risoluzione di complessità in tempo polinomiale, approssimati con grado di approssimazione limitato da una costante

Se $P \neq NP \Rightarrow PTAS \neq APX$

Se un problema è APX-Hard è una cattiva notizia!



APX e Fixed-parameter algorithms

APX

$MRSO \in APX - H$

APX

Insieme dei problemi di ottimizzazione $\in NPO$ aventi algoritmi di risoluzione di complessità in tempo polinomiale, approssimati con grado di approssimazione limitato da una costante

Se $P \neq NP \Rightarrow PTAS \neq APX$

Se un problema è APX-Hard è una cattiva notizia!



APX e Fixed-parameter algorithms

APX

$MRSO \in APX - H$

APX

Insieme dei problemi di ottimizzazione $\in NPO$ aventi algoritmi di risoluzione di complessità in tempo polinomiale, approssimati con grado di approssimazione limitato da una costante

Se $P \neq NP \Rightarrow PTAS \neq APX$

Se un problema é APX-Hard é una cattiva notizia!



APX e Fixed-parameter algorithms

Parametrized Complexity

Bongartz ha proposto di attaccare MRSO con l'approccio della complessità parametrizzata

Complessità parametrizzata

Classe di problemi per i quali si può restringere l'esplosione combinatoria ad una piccola parte dell'input: **il parametro**

Problemi fixed-parameter trattabili \rightarrow risolvibili in $f(k)n^{O(1)}$



APX e Fixed-parameter algorithms

Parametrized Complexity

Bongartz ha proposto di attaccare MRSO con l'approccio della complessità parametrizzata

Complessità parametrizzata

Classe di problemi per i quali si può restringere l'esplosione combinatoria ad una piccola parte dell'input: **il parametro**

Problemi fixed-parameter trattabili \rightarrow risolvibili in $f(k)n^{O(1)}$



Sommario

- 1 Motivazioni biologiche
- 2 Preliminari
- 3 A_{OP}
- 4 APX e Fixed-Parameters Algorithms
- 5 Due parametri naturali per MRSO
- 6 Cutwidth e A_{CUT}
- 7 Conclusioni e problemi aperti



Due parametri naturali per MRSO

Definizioni

Due parametri naturali che possono essere presi in considerazione per trattare MRSO con la complessità parametrizzata sono:

- χ il numero di incroci fra archi in G_T
- δ il numero di vertici con grado tre

Possono assumere valori piccoli in molte applicazioni!

Nel seguito formuleremo un primo algoritmo (A_{NEB}) per la risoluzione di MRSO e dimostreremo che ha complessità in tempo polinomiale se χ o δ sono fissati.



Due parametri naturali per MRSO

Definizioni

Due parametri naturali che possono essere presi in considerazione per trattare MRSO con la complessità parametrizzata sono:

- χ il numero di incroci fra archi in G_T
- δ il numero di vertici con grado tre

Possono assumere valori piccoli in molte applicazioni!

Nel seguito formuleremo un primo algoritmo (A_{NEB}) per la risoluzione di MRSO e dimostreremo che ha complessità in tempo polinomiale se χ o δ sono fissati.



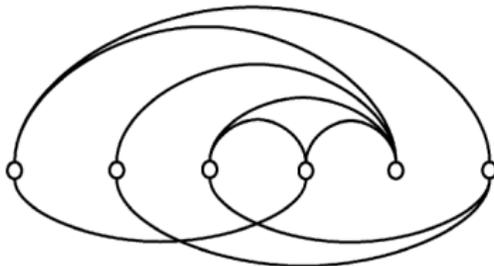
Due parametri naturali per MRSO

Nice edge bipartition

Definizione nice edge bipartition

Sia G_Γ un grafo di struttura implicata con n vertici. Una edge bipartition $P = (E_t, E_b)$ di G_Γ é una partizione degli archi di G_Γ in E_t e E_b (archi top e archi bottom), tale che $E_t \cup E_b = E(G_\Gamma)$ e $E_t \cap E_b = \emptyset$.

P é detta nice se $G_\Gamma[E_t]$ é planare esterno.



Due parametri naturali per MRSO

Codon assignment

Definizione codon assignment e compatibilit 

Un codon assignment per G_Γ   un mapping da $V \subseteq V(G_\Gamma)$ a Σ^3

Un c.a. per una coppia di vertici $\{i, j\}$

$$i \rightarrow t_{3i-2}t_{3i-1}t_{3i} \text{ e } j \rightarrow t_{3j-2}t_{3j-1}t_{3j}$$

  compatibile con G_Γ se $\{i, j\} \notin E(G_\Gamma)$ oppure $t_{i'}$ e $t_{j'}$ sono complementari
 $\forall \{i', j'\} \in \Gamma \cap \{3i-2, 3i-1, 3i\} \times \{3j-2, 3j-1, 3j\}$

In generale, un assegnamento $\phi : V \rightarrow \Sigma^3$ con $V \subseteq V(G_\Gamma)$   compatibile con G_Γ se $\forall i, j \in V, i \rightarrow \phi(i)$ e $j \rightarrow \phi(j)$ sono compatibili con G_Γ

Due parametri naturali per MRSO

Osservazione

Supponiamo di voler trovare la sequenza di mRNA di massima somiglianza che inizia con il codone AAA. Allora sostituiamo la funzione $f_1 \in F$ con f' t.c.:

$$f'(AAA) = f_1(AAA) \text{ e } f'(C) = -\infty \forall C \neq AAA$$

Andiamo quindi a risolvere MRSO per l'istanza (G_Γ, F') con $F' = f', f_2, \dots, f_n$



Due parametri naturali per MRSO

Funzioni corrispondenti di similaritá

Definizione funzioni corrispondenti di similaritá

Siano (G_T, F) istanza di MRSO e $\phi : V \rightarrow \Sigma^3$ codon assignment per $V \subseteq V(G_T)$. Definiamo $F^\phi = f_1^\phi, \dots, f_n^\phi$, insieme di funzioni corrispondenti di similaritá, come:

- $\forall i \in V : f_i^\phi(\phi(i)) = f_i(\phi(i))$ e $f_i^\phi(C) = -\infty \forall C \neq \phi(i)$
- $\forall j \in V(G_T) - V : f_j^\phi = f_j$



A_{NEB}

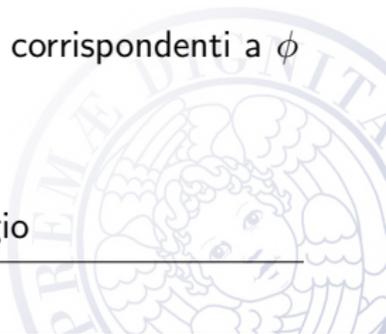
L'algoritmo

L'algoritmo A_{NEB} usa A_{OP} come sottoprocedura per la sua computazione:

Algoritmo 1 $A_{NEB}(G_\Gamma, F, P)$

▷ **INPUT:** G_Γ grafo di struttura implicata di ordine n , F insieme di funzioni di similarit  e P n.e.b

- 1: **for all** possibili codon assignment ϕ a vertici incidenti agli archi in E_b **do**
 - 2: **if** ϕ   compatibile rispetto $G_\Gamma[E_b]$ **then**
 - 3: Costruisci F_ϕ l'insieme di funzioni di similarit  corrispondenti a ϕ
 - 4: $A_{OP}(G_\Gamma[E_t], F_\phi)$
 - 5: **end if**
 - 6: **end for**
 - 7: **return** la sequenza di mRNA con il massimo punteggio
-



A_{NEB}

Ottimalit  e complessit 

Lemma

Data un'istanza (G_Γ, F) di MRSO e una n.e.b. $P = (E_t, E_b)$, A_{NEB} calcola la sequenza di mRNA ottima per l'istanza del problema in $O(2^{12\beta}n)$ con $n = |V(G_\Gamma)|$ e $\beta = |E_b|$

Dim:

Ottimalit : deriva da A_{OP} .

Complessit :

- numero di codon assignment considerati = $|\Sigma^3|^{V_b} \leq 64^{2\beta} = 2^{12\beta}$
- costruzione assegnamento e controllo compatibilit  $\rightarrow O(n)$
- chiamata ad $A_{OP} \rightarrow O(n)$

La complessit  dell' algoritmo   quindi $O(2^{12\beta}n)$



A_{NEB}

Ottimalit  e complessit 

Lemma

Data un'istanza (G_Γ, F) di MRSO e una n.e.b. $P = (E_t, E_b)$, A_{NEB} calcola la sequenza di mRNA ottima per l'istanza del problema in $O(2^{12\beta}n)$ con $n = |V(G_\Gamma)|$ e $\beta = |E_b|$

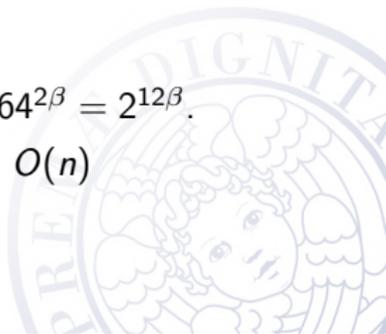
Dim:

Ottimalit : deriva da A_{OP} .

Complessit :

- numero di codon assignment considerati = $|\Sigma^3|^{V_b} \leq 64^{2\beta} = 2^{12\beta}$.
- costruzione assegnamento e controllo compatibilit  $\rightarrow O(n)$
- chiamata ad $A_{OP} \rightarrow O(n)$

La complessit  dell'algoritmo   quindi $O(2^{12\beta}n)$



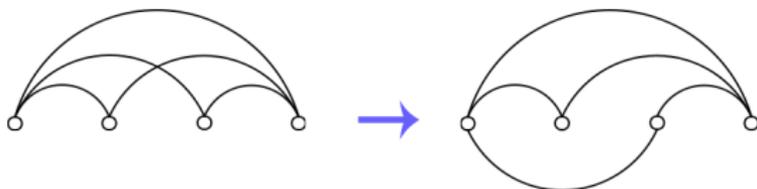
A_{NEB}

Parametrizzazione rispetto a χ

Se $\chi = 0 \Rightarrow G_F$ é planare esterno.

Altrimenti, avere un n.e.b. con χ archi bottom é sempre possibile.

Consideriamo una edge bipartition con un arco bottom per ogni coppia di archi incrociati in G_F :



Tale bipartition é nice, ha al piú χ archi bottom e puó essere costruita in tempo lineare.

Corollario

MRSO é risolvibile in $O(2^{12\chi} n)$

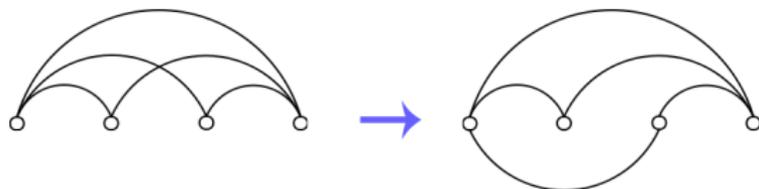
A_{NEB}

Parametrizzazione rispetto a χ

Se $\chi = 0 \Rightarrow G_F$ é planare esterno.

Altrimenti, avere un n.e.b. con χ archi bottom é sempre possibile.

Consideriamo una edge bipartition con un arco bottom per ogni coppia di archi incrociati in G_F :



Tale bipartition é nice, ha al piú χ archi bottom e puó essere costruita in tempo lineare.

Corollario

MRSO é risolvibile in $O(2^{12\chi} n)$

A_{NEB}

Parametrizzazione rispetto a δ

Costruire un n.e.b. con δ archi bottom é immediato dopo aver enunciato il seguente lemma:

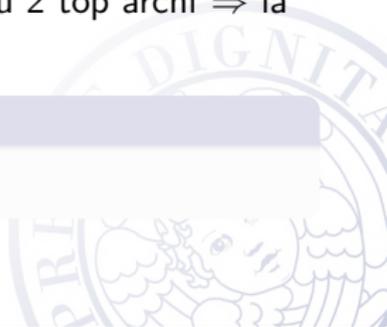
Lemma

Se G ha grado massimo 2 allora é planare esterno

Una bipartizione con δ archi bottom si puó costruire in tempo $O(n)$.
Siccome G_{Γ} é subcubico, ogni vertice é incidente ad al piú 2 top archi \Rightarrow la bipartizione é nice.

Corollario

MRSO é risolvibile in $O(2^{12\delta} n)$



A_{NEB}

Parametrizzazione rispetto a δ

Costruire un n.e.b. con δ archi bottom é immediato dopo aver enunciato il seguente lemma:

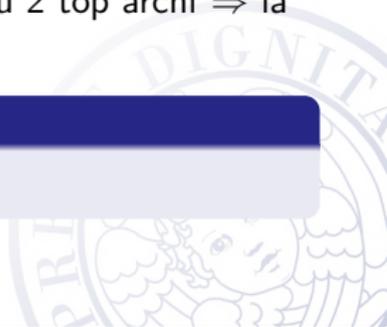
Lemma

Se G ha grado massimo 2 allora é planare esterno

Una bipartizione con δ archi bottom si puó costruire in tempo $O(n)$.
Siccome G_{Γ} é subcubico, ogni vertice é incidente ad al piú 2 top archi \Rightarrow la bipartizione é nice.

Corollario

MRSO é risolvibile in $O(2^{12\delta} n)$



A_{NEB}

Parametrizzazione rispetto al page number

MRSO é risolvibile in tempo polinomiale se entrambe le parti non hanno archi incrociati? (G_F ha page number 2)

Definizione page number

Il page number di un grafo G é il partizionamento di $E(G)$ nel piú piccolo numero possibile di sottoinsiemi, tali che ciascun sottoinsieme non abbia archi incrociati, sotto lo stesso ordinamento di vertici

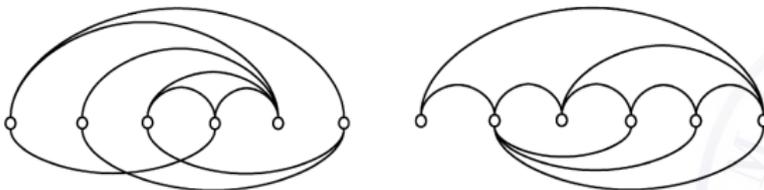


Figura: Un grafo con nice edge bipartition e un grafo con page number 2

A_{NEB}

Parametrizzazione rispetto al page number(2)

MRSO é parametrizzabile rispetto al page number?

La risposta é **no**.

MRSO é NP-Completo già per grafi con page number pari a 2



A_{NEB}

Parametrizzazione rispetto al page number(2)

MRSO é parametrizzabile rispetto al page number?

La risposta é **no**.

MRSO é NP-Completo già per grafi con page number pari a 2



Sommario

- 1 Motivazioni biologiche
- 2 Preliminari
- 3 A_{OP}
- 4 APX e Fixed-Parameters Algorithms
- 5 Due parametri naturali per MRSO
- 6 Cutwidth e A_{CUT}
- 7 Conclusioni e problemi aperti



Cutwidth e A_{CUT}

Definizioni

Definizioni

Sia (G_{Γ}, F) istanza di MRSO.

Per $p \in \{1, \dots, n-1\}$ definiamo la p -cutwidth di G_{Γ} come il numero di archi che connettono i vertici in $\{1, \dots, p\}$ ai vertici in $\{p+1, \dots, n\}$

La cutwidth di G_{Γ} é definita come

$$\max_{p \in \{1, \dots, n-1\}} p - \text{cutwidth}$$

Vogliamo parametrizzare MRSO rispetto questo nuovo parametro



Cutwidth e A_{CUT}

Definizioni

Definizioni

Sia (G_{Γ}, F) istanza di MRSO.

Per $p \in \{1, \dots, n-1\}$ definiamo la p -cutwidth di G_{Γ} come il numero di archi che connettono i vertici in $\{1, \dots, p\}$ ai vertici in $\{p+1, \dots, n\}$

La cutwidth di G_{Γ} é definita come

$$\max_{p \in \{1, \dots, n-1\}} p - \text{cutwidth}$$

Vogliamo parametrizzare MRSO rispetto questo nuovo parametro

Cutwidth e A_{CUT}

L'algoritmo

L'idea di base é di dividere ricorsivamente G_Γ in $G_\Gamma[1, \dots, p]$ e $G_\Gamma[p + 1, \dots, n]$ e concatenare le due sequenze di mRNA ottenute, T' e T'' , per ottenere la sequenza target T .

A_{CUT} ricerca esaustivamente tra tutti i possibili codon assignment compatibili con il sottoinsieme di archi che connettono i nodi dei due sottografi.

Dovranno valere le seguenti propriet :

- i vertici gi  assegnati rimangono compatibili per l'intera computazione
- una volta che vi   stato un assegnamento, tale assegnamento non sar  modificato



Cutwidth e A_{CUT}

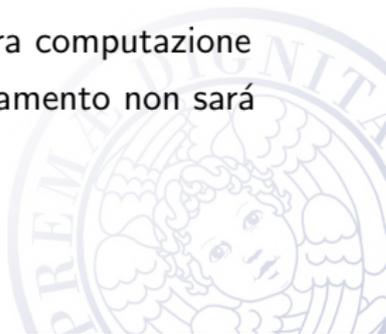
L'algoritmo

L'idea di base é di dividere ricorsivamente G_Γ in $G_\Gamma[1, \dots, p]$ e $G_\Gamma[p + 1, \dots, n]$ e concatenare le due sequenze di mRNA ottenute, T' e T'' , per ottenere la sequenza target T .

A_{CUT} ricerca esaustivamente tra tutti i possibili codon assignment compatibili con il sottoinsieme di archi che connettono i nodi dei due sottografi.

Dovranno valere le seguenti proprietà:

- i vertici già assegnati rimangono compatibili per l'intera computazione
- una volta che vi é stato un assegnamento, tale assegnamento non sarà modificato



Cutwidth e A_{CUT}

L'algoritmo (2)

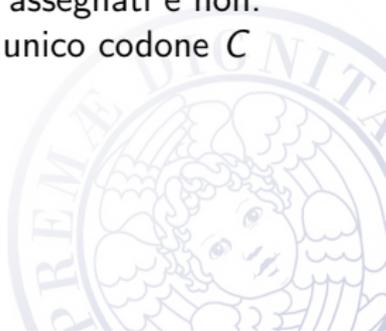
Per garantire le proprietà useremo le funzioni corrispondenti di similarità. In particolare introduciamo le funzioni degenerate di similarità:

Definizione

Chiamiamo f funzione degenerata di similarità se esiste un codone C t.c.

$$f(C) > -\infty \text{ e } f(C') = -\infty \forall C' \neq C$$

Useremo le funzioni degenerate per distinguere tra vertici assegnati e non: $i \in V(G_T)$ assegnato se f_i è degenerata e assegnata a un unico codone C



Cutwidth e A_{CUT}

Algoritmo 2 $A_{CUT}(G_{\Gamma}, F)$

▷ **INPUT:** G_{Γ} grafo di struttura implicata di ordine n , F insieme di funzioni di similarit 

- 1: **if** $E(G_{\Gamma}) = \emptyset$ **then**
 - 2: **return** sequenza T che massimizza F
 - 3: **end if**
 - 4: Seleziona il minimo $p \in \{1, \dots, n\}$ con p-cutwidth maggiore di 0
 - 5: Sia E_p l'insieme degli archi che collega i due sottografi cos  definiti
 - 6: Sia V_p l'insieme dei vertici ad essi incidenti
 - 7: Sia A_p l'insieme dei vertici con funzione degenerata di similarit  (gi  assegnati)
 - 8: Definiamo $\phi^{A_p} : A_p \leftarrow \Sigma^3$ t.c. $\phi^{A_p}(i) = C \Leftrightarrow f_i(C) > -\infty$
 - 9: **for all** possibili codon assignment $\phi^{V_p - A_p}$ **do**
 - 10: **if** $\phi = \phi^{A_p} \cup \phi^{V_p - A_p}$   compatibile rispetto $G_{\Gamma}[E_p]$ **then**
 - 11: $T' \rightarrow A_{CUT}(G_{\Gamma}[1, \dots, p], f_1^{\phi}, \dots, f_p^{\phi})$
 - 12: $T'' \rightarrow A_{CUT}(G_{\Gamma}[p + 1, \dots, n], f_{p+1}^{\phi}, \dots, f_n^{\phi})$
 - 13: **end if**
 - 14: **end for**
 - 15: **return** la sequenza di mRNA $T = T' T''$ con il massimo punteggio
-



Cutwidth e A_{CUT}

Complessità e ottimalità

Lemma

Dato (G_Γ, F) istanza di MRSO, A_{CUT} calcola una sequenza di mRNA ottima in $O(2^{12\psi} n)$ con $n = |V(G_\Gamma)|$ e $\psi = \text{cutwidth}(G_\Gamma)$

Dim:

Ottimalità per induzione sui passi ricorsivi

Complessità:

- il numero di sottografi generati è $O(n)$
- per ognuno vengono considerati al più $|\Sigma^3|^{|V_p|} \leq 2^{12\psi}$ codon assignment

La complessità dell'algoritmo è pari a $O(2^{12\psi} n)$



Cutwidth e A_{CUT}

Complessità e ottimalità

Lemma

Dato (G_Γ, F) istanza di MRSO, A_{CUT} calcola una sequenza di mRNA ottima in $O(2^{12\psi} n)$ con $n = |V(G_\Gamma)|$ e $\psi = \text{cutwidth}(G_\Gamma)$

Dim:

Ottimalità per induzione sui passi ricorsivi

Complessità:

- il numero di sottografi generati è $O(n)$
- per ognuno vengono considerati al più $|\Sigma^3|^{|\mathcal{V}_p|} \leq 2^{12\psi}$ codon assignment

La complessità dell'algoritmo è pari a $O(2^{12\psi} n)$



Cutwidth e A_{CUT}

Osservazioni

Corollario

MRSO é risolvibile in tempo polinomiale se $\psi = O(\log n)$

Treewidth misura il grado di somiglianza ad un albero di un dato grafo. Per grafi con numero di vertici, grado e treewidth costanti, si puó calcolare un ordinamento dei vertici t.c. il grafo ottenuto abbia $cutwidth \leq O(\log n)$

Corollario

MRSO é risolvibile in tempo polinomiale se G_Γ ha treewidth costante

NOTA: si noti che la treewidth di un grafo planare esterno é 2, quindi A_{CUT} generalizza A_{OP}

Cutwidth e A_{CUT}

Osservazioni

Corollario

MRSO é risolubile in tempo polinomiale se $\psi = O(\log n)$

Treewidth misura il grado di somiglianza ad un albero di un dato grafo. Per grafi con numero di vertici, grado e treewidth costanti, si puó calcolare un ordinamento dei vertici t.c. il grafo ottenuto abbia $cutwidth \leq O(\log n)$

Corollario

MRSO é risolubile in tempo polinomiale se G_Γ ha treewidth costante

NOTA: si noti che la treewidth di un grafo planare esterno é 2, quindi A_{CUT} generalizza A_{OP}



Cutwidth e A_{CUT}

Osservazioni

Corollario

MRSO é risolvibile in tempo polinomiale se $\psi = O(\log n)$

Treewidth misura il grado di somiglianza ad un albero di un dato grafo. Per grafi con numero di vertici, grado e treewidth costanti, si può calcolare un ordinamento dei vertici t.c. il grafo ottenuto abbia $cutwidth \leq O(\log n)$

Corollario

MRSO é risolvibile in tempo polinomiale se G_Γ ha treewidth costante

NOTA: si noti che la treewidth di un grafo planare esterno é 2, quindi A_{CUT} generalizza A_{OP}

Cutwidth e A_{CUT}

Osservazioni

Corollario

MRSO é risolubile in tempo polinomiale se $\psi = O(\log n)$

Treewidth misura il grado di somiglianza ad un albero di un dato grafo. Per grafi con numero di vertici, grado e treewidth costanti, si può calcolare un ordinamento dei vertici t.c. il grafo ottenuto abbia $cutwidth \leq O(\log n)$

Corollario

MRSO é risolubile in tempo polinomiale se G_Γ ha treewidth costante

NOTA: si noti che la treewidth di un grafo planare esterno é 2, quindi A_{CUT} generalizza A_{OP}

Sommario

- 1 Motivazioni biologiche
- 2 Preliminari
- 3 A_{OP}
- 4 APX e Fixed-Parameters Algorithms
- 5 Due parametri naturali per MRSO
- 6 Cutwidth e A_{CUT}
- 7 Conclusioni e problemi aperti



Conclusioni

Nel seminario è stato introdotto il problema MRSO e sono stati forniti degli algoritmi che possono risolverlo in maniera efficiente, anche grazie all'utilizzo della complessità parametrizzata.

Rimangono alcuni problemi aperti:

- Il numero di incroci tra archi (χ) e la cutwidth di G_Γ , sono parametri che dipendono dal particolare ordinamento dei nodi. Sarebbe interessante avere degli algoritmi che trovino degli ordinamenti per minimizzare tali parametri. Per il secondo caso, il problema è *NPC* ma esistono degli algoritmi parametrizzati. Il primo, non è invece stato trattato.
- Sulla base dell'algoritmo A_{NEB} sarebbe utile avere un algoritmo (fixed-parameter) che produca un ordinamento dei vertici e una n.e.b. con il minimo numero di archi bottom



Conclusioni

Nel seminario è stato introdotto il problema MRSO e sono stati forniti degli algoritmi che possono risolverlo in maniera efficiente, anche grazie all'utilizzo della complessità parametrizzata.

Rimangono alcuni problemi aperti:

- Il numero di incroci tra archi (χ) e la cutwidth di G_T , sono parametri che dipendono dal particolare ordinamento dei nodi. Sarebbe interessante avere degli algoritmi che trovino degli ordinamenti per minimizzare tali parametri. Per il secondo caso, il problema è *NPC* ma esistono degli algoritmi parametrizzati. Il primo, non è invece stato trattato.
- Sulla base dell'algoritmo A_{NEB} sarebbe utile avere un algoritmo (fixed-parameter) che produca un ordinamento dei vertici e una n.e.b. con il minimo numero di archi bottom

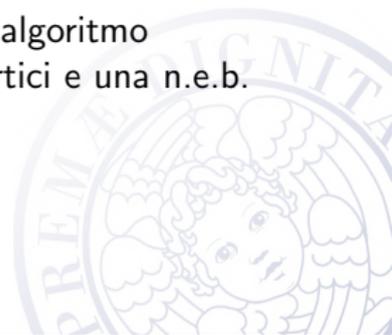


Conclusioni

Nel seminario è stato introdotto il problema MRSO e sono stati forniti degli algoritmi che possono risolverlo in maniera efficiente, anche grazie all'utilizzo della complessità parametrizzata.

Rimangono alcuni problemi aperti:

- Il numero di incroci tra archi (χ) e la cutwidth di G_T , sono parametri che dipendono dal particolare ordinamento dei nodi. Sarebbe interessante avere degli algoritmi che trovino degli ordinamenti per minimizzare tali parametri. Per il secondo caso, il problema è *NPC* ma esistono degli algoritmi parametrizzati. Il primo, non è invece stato trattato.
- Sulla base dell'algoritmo A_{NEB} sarebbe utile avere un algoritmo (fixed-parameter) che produca un ordinamento dei vertici e una n.e.b. con il minimo numero di archi bottom



Bibliografia

Blin, Fertin, Hermelin, Vialette. *Fixed-parameter algorithms for protein similarity search under mRNA structure constraints*. Journal of discrete algorithms (2008)

Backofen, Narayanaswamy, Swidan. *Protein similarity search under mRNA structure constraints: application to targeted selenocysteine insertion*. Silico Biology (2002).

Bongartz. *Some notes on the complexity of protein similarity search under mRNA structure constraints*. Sofsem (2004)

Wikipedia contributors. <http://en.wikipedia.org/wiki/APX> (2009)

