

1 Recursions

1.1 Definitions

S^1, S^2 target and query sequences

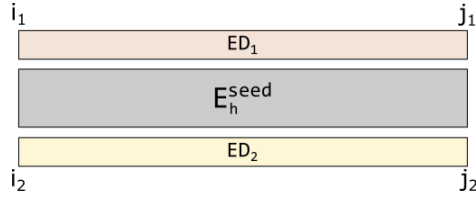
i_1, j_1, i_2, j_2 interaction boundaries

si_1, sj_1, si_2, sj_2 seed boundaries

N the maximum interaction length (~ 150)

M the enclosed unpaired positions in one loop (~ 15)

General energy computation:



$$E_{i_2, j_2}^{i_1, j_1} = E_h^{seed}(i_1, j_1) + ED_1(i_1) + ED_2(i_2)$$

Optimization task:

$$\min_{seed} \min_{\substack{j_1 - i_1 \leq N \\ j_2 - i_2 \leq N}} \left(E_h^{seed}(i_1, j_1) \right)$$

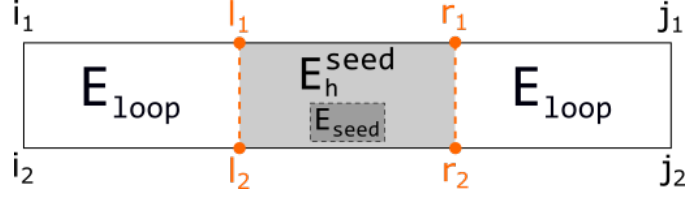
1.2 Initialization

$$\forall_{\substack{si_1 \leq i_1 \leq j_1 \leq sj_1 \\ si_2 \leq i_2 \leq j_2 \leq sj_2}} E_h^{seed}(i_1, j_1) = \infty$$

$$E_h^{seed}(si_1, sj_1) = E_{seed}$$

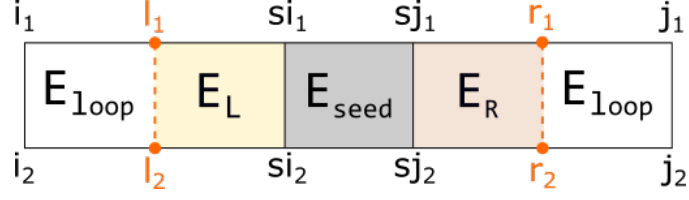
with E_{seed} including E_{init} .

1.3 Recursion 1 ($O(N^4)$ space + time)



$$E_h^{seed}(i_1, j_1) = \begin{cases} \infty & : \text{if no matching base pair} \\ \infty & : \text{if } j_1 - i_1 > N \text{ oder } j_2 - i_2 > N \\ \min_{\substack{i_1 < l_1 \leq r_1 < j_1 \\ i_2 < l_2 \leq r_2 < j_2 \\ l_1 - i_1 - 1 \leq M \\ j_1 - r_1 - 1 \leq M \\ l_2 - i_2 - 1 \leq M \\ j_2 - r_2 - 1 \leq M}} \left(E_{loop}(i_2, l_2) + E_h^{seed}(l_1, r_1) + E_{loop}(r_1, j_2) \right) & : \text{otherwise.} \end{cases}$$

1.4 Recursion 2 ($O(N^2)$ space + $O(N^4)$ time)



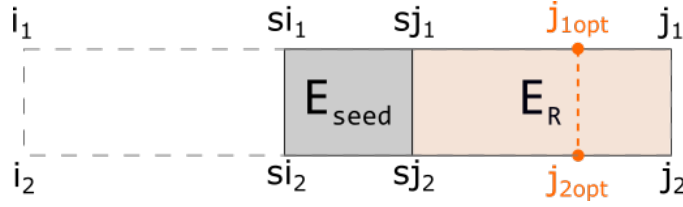
$$E_h^{seed}(i_1, j_1) = \begin{cases} \infty & : \text{if } j_1 - i_1 > N \text{ oder } j_2 - i_2 > N \\ \left(E_L(i_1) + E_{seed} + E_R(j_1) \right) & \\ \infty & : \text{otherwise.} \end{cases}$$

$$\forall_{\substack{si_1 - N \leq i_1 \leq si_1 \\ si_2 - N \leq i_2 \leq si_2}} E_L(i_1) = \begin{cases} \infty & : \text{if no matching base pair} \\ \min_{\substack{l_1 - i_1 - 1 \leq M \\ l_2 - i_2 - 1 \leq M}} \left(E_{loop}(i_1, l_1) + E_L(l_1) \right) & \\ \infty & : \text{otherwise.} \end{cases}$$

$$\forall_{\substack{sj_1 \leq j_1 \leq sj_1 + N \\ sj_2 \leq j_2 \leq sj_2 + N}} E_R(j_1) = \begin{cases} \infty & : \text{if no matching base pair} \\ \min_{\substack{j_1 - r_1 - 1 \leq M \\ j_2 - r_2 - 1 \leq M}} \left(E_R(r_1) + E_{loop}(r_1, j_1) \right) & \\ \infty & : \text{otherwise.} \end{cases}$$

1.5 Recursion 3 ($O(N^2)$ space + $O(N^2)$ time)

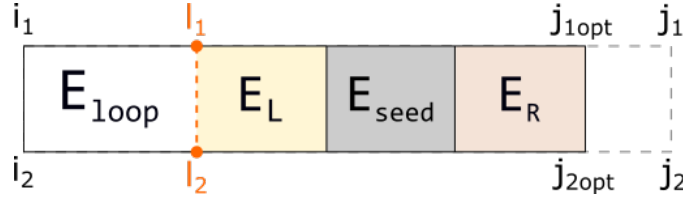
First find j_1 and j_2 that minimize right side. Call them j_{1opt} and j_{2opt} .



$$\arg \min_{j^1, j^2} \left(E_{seed} + E_R^{(sj_1, j_1)}(sj_2, j_2) \right)$$

with E_R defined as in Recursion 2.

Then minimize over entire interaction up to j_{1opt} and j_{2opt} .



$$E_h^{seed}(i_1, j_1) = \begin{cases} \infty & : \text{if no matching base pair or } j_1 \neq j_{1opt} \text{ or } j_2 \neq j_{2opt} \\ \min_{\substack{l_1 - i_1 - 1 \leq M \\ l_2 - i_2 - 1 \leq M}} \left(E_{loop}(i_1, l_1) + E_L(l_1) + E_{seed} + E_R^{(j_{1opt})}(sj_2, j_2) \right) & \\ \text{otherwise.} & \end{cases}$$

with E_L and E_R defined as in Recursion 2.

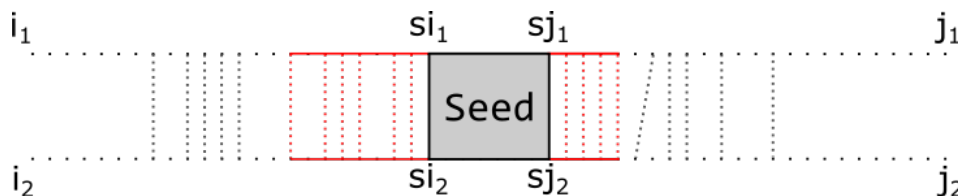
1.6 Recursion 4 (ideas from RiBlast2)

- * extend left + right without gaps
- * extend left + right with gaps
- * use approximated accessibility energies

1.6.1 Gapless extension

Given a seed, RiBlast first extends the interaction to the left and right of the seed without gaps. If target and query nucleotides can pair, the new interaction energy is calculated. If it is lower than the present minimum, then the minimum is updated and the extension continues. If the minimum does not change for *drop_out_length_wo_gap* steps, then the extension stops with *drop_out_length_wo_gap* being an input parameter.

Default *drop_out_length_wo_gap* = 5.



Example of gapless extension. The red lines indicate the range of extension. Here *drop_out_length_wo_gap* is set to 2, meaning that the extension is stopped if the minimal energy does not improve after 2 steps.

1.6.2 Gapped extension

After the gapless extension finished, RiBlast does a second extension. This time gaps are allowed. Again the minimum interaction energy is updated if a lower energy is found. If the minimum does not change for *drop_out_length_w_gap* steps, then the extension stops with *drop_out_length_w_gap* being an input parameter.

Default *drop_out_length_w_gap* = 16.

1.6.3 Accessibility energy

RiBlast uses approximated accessibility energies as proposed in RNAPlex-a.