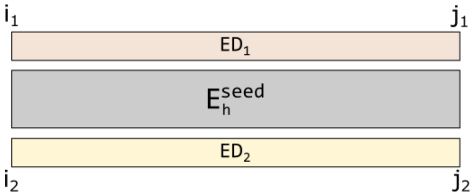
1 Recursions

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:



$$E\binom{i_1,j_1}{i_2,j_2} = E_h^{seed}\binom{i_1,j_1}{i_2,j_2} + ED_1\binom{i_1}{j_1} + ED_2\binom{i_2}{j_2}$$

Optimization task:

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

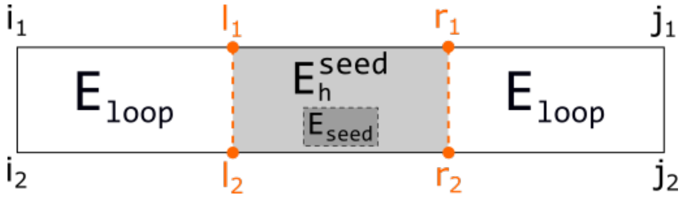
1.2 Initialization

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

$$E_h^{seed}\binom{si_1,sj_1}{si_2,sj_2} = E_{seed}$$

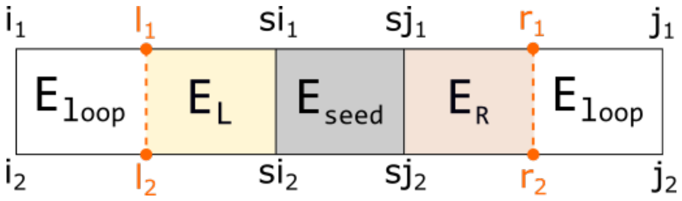
with E_{seed} including E_{init}.

1.3 Exact naive method ($O(N^4)$ space + time)



$$\underset{\substack{\forall \\ si_1-N\leq i_1\leq si_1 \\ si_2-N\leq i_2\leq si_2 \\ \forall \\ sj_1\leq j_1\leq sj_1+N \\ sj_2\leq j_2\leq sj_2+N}}{} E_h^{seed}\binom{i_1,j_1}{i_2,j_2} = \begin{cases} \infty & : \text{if } \text{ no matching base pair} \\ \infty & : \text{if } j_1 - i_1 > N \text{ oder } j_2 - i_2 > N \\ \min\limits_{\substack{i_1<l_1\leq r_1<j_1 \\ i_2<l_2\leq r_2<j_2 \\ l_1-i_1-1\leq M \\ j_1-r_1-1\leq M \\ l_2-i_2-1\leq M \\ j_2-r_2-1\leq M}} \left(E_{loop}\binom{i_1,l_1}{i_2,l_2} + E_h^{seed}\binom{l_1,r_1}{l_2,r_2} + E_{loop}\binom{r_1,j_1}{r_2,j_2}\right) & : \text{otherwise.} \end{cases}$$

1.4 Exact memory efficient method ($O(N^2)$ space + $O(N^4)$ time)



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

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

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

1.4.1 Exact method (memory efficient) results

Parameters:
master_20: –qIntLenMax 20 –tIntLenMax 20 –threads 8
memory_efficient_20: –pred X -m M –qIntLenMax 20 –tIntLenMax 20 –threads 8
memory_efficient_ensemble_20: –pred E -m M –qIntLenMax 20 –tIntLenMax 20 –threads 8

Figure 1: Performance

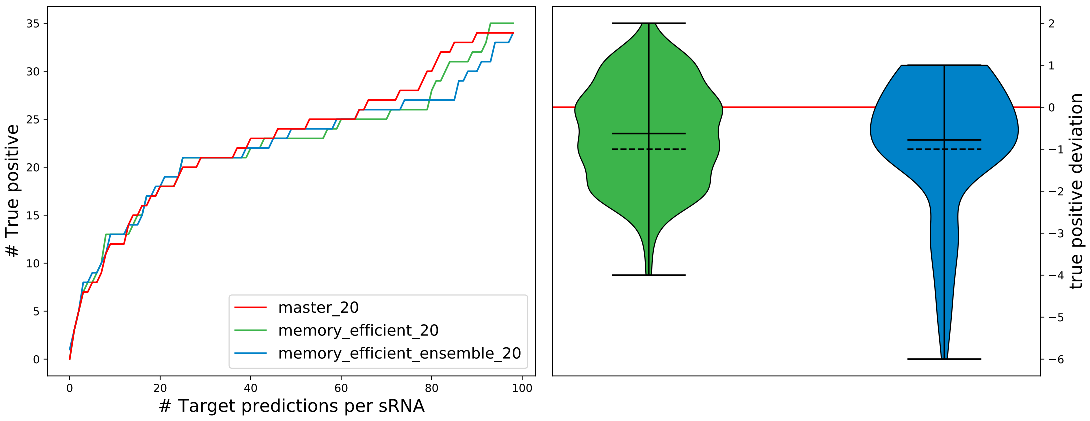
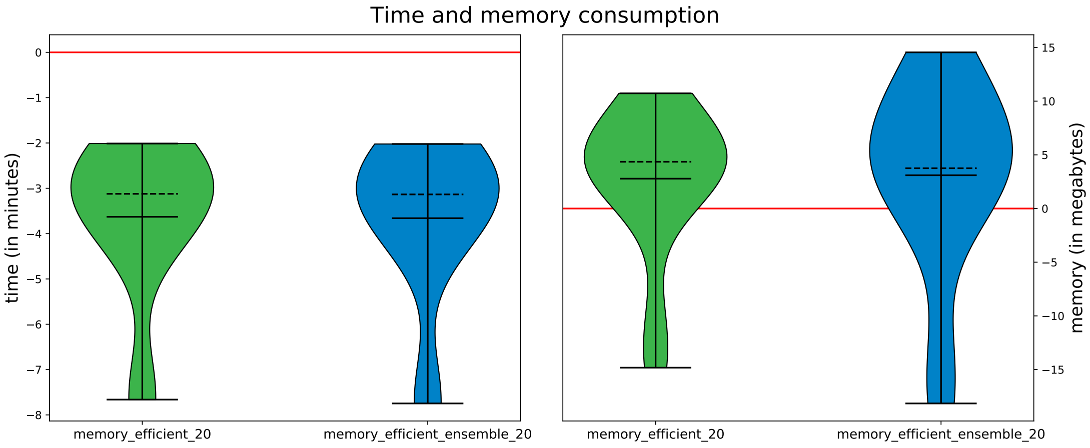
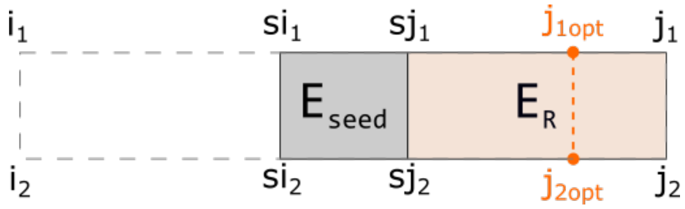


Figure 2: Time & memory



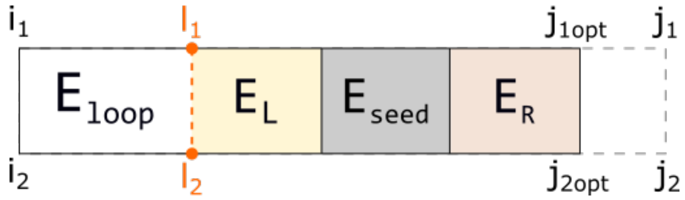
1.5 Heuristic method ($O(N^2)$ space + $O(N^2)$ time)

First find j1 and j2 that minimize right side. Call them j_{1opt} and j_{2opt}.



$$\arg\min_{j1,j2}\left(E_{seed}+E_R\binom{sj_1,j_1}{sj_2,j_2}\right)$$

with E_R defined as in Recursion 2.
Then minimize over entire interaction up to j_{1opt} and j_{2opt}.



$$\forall_{\substack{si_1-N\leq i_1\leq j_1\\ si_2-N\leq i_2\leq j_2}} E_h^{seed}\binom{i_1,j_1}{i_2,j_2}=\begin{cases}\infty \\ \quad : \text{ if } \text{ no matching base pair or } j_1\neq j_{1opt} \text{ or } j_2\neq j_{2opt} \\ \min\limits_{\substack{l_1-i_1-1\leq M\\ l_2-i_2-1\leq M}}\left(E_{loop}\binom{i_1,l_1}{i_2,l_2}+E_L\binom{l_1}{l_2}+E_{seed}+E_R\binom{j_{1opt}}{j_{2opt}}\right) \\ \quad : \text{ otherwise.}\end{cases}$$

with E_L and E_R defined as in Recursion 2.

1.5.1 Heuristic method results

Parameters:
master_20: –qIntLenMax 20 –tIntLenMax 20 –threads 8
heuristic_20: –pred X -m H –qIntLenMax 20 –tIntLenMax 20 –threads 8
heuristic_ensemble_20: –pred E -m H –qIntLenMax 20 –tIntLenMax 20 –threads 8

Figure 3: Performance

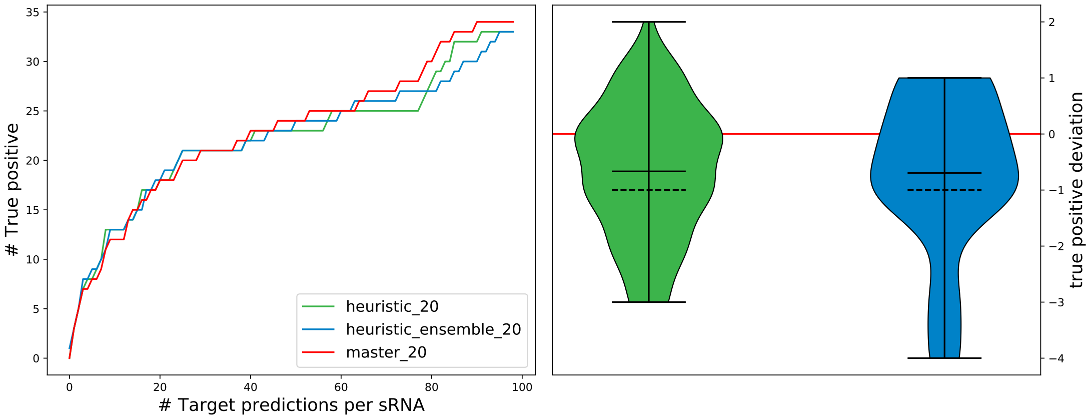
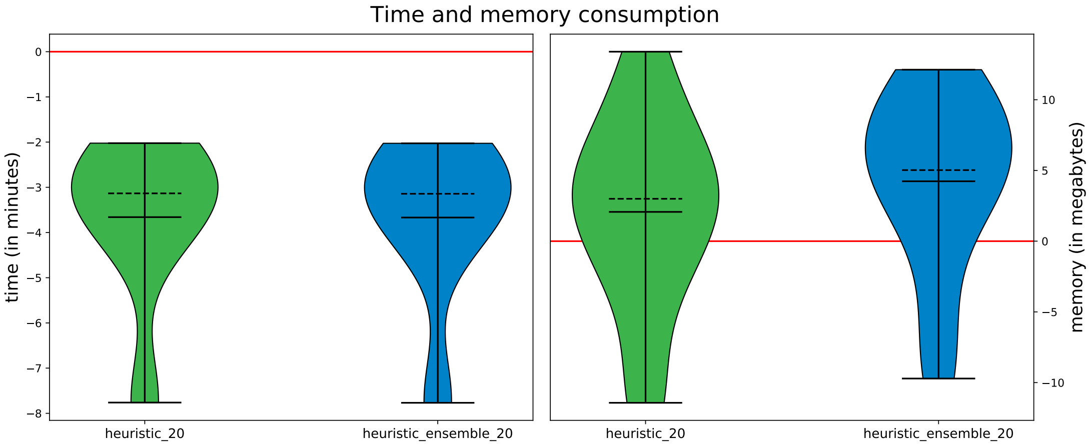


Figure 4: Time & memory



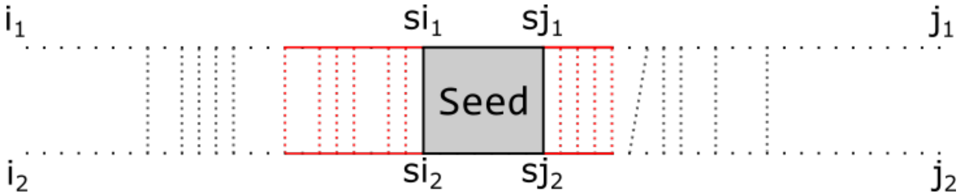
1.6 RiBlast method

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

1.6.1 Parallel extension

Given a seed, RiBlast first extends the interaction to the left and right of the seed without gaps. This means that we linearly loop over both sequences at the same time. 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 Thorough extension

After the parallel extension finished, RiBlast does a second extension. This time gaps are allowed. This means that we loop over both sequences independently, resulting in a quadratic complexity. 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

1.6.4 RiBlast method results

Parameters:
master_20: –qIntLenMax 20 –tIntLenMax 20 –threads 8
riblast_20: –pred X -m R –qIntLenMax 20 –tIntLenMax 20 –threads 8

Figure 5: Performance

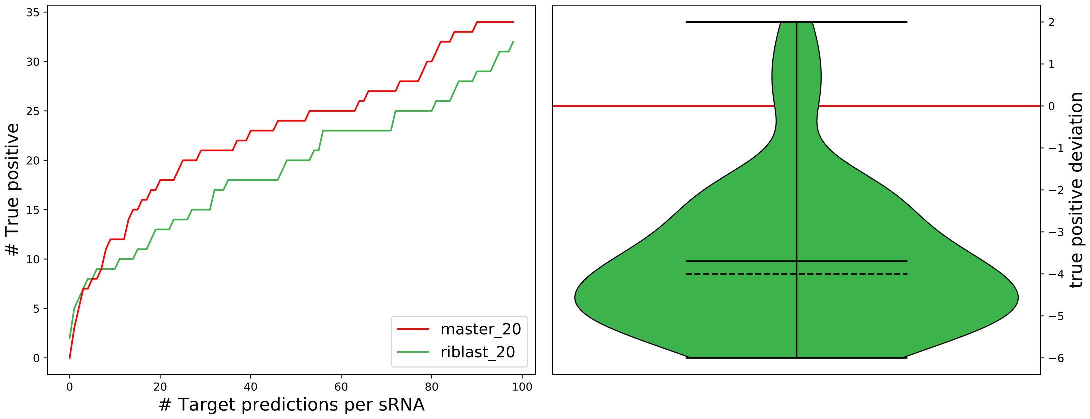
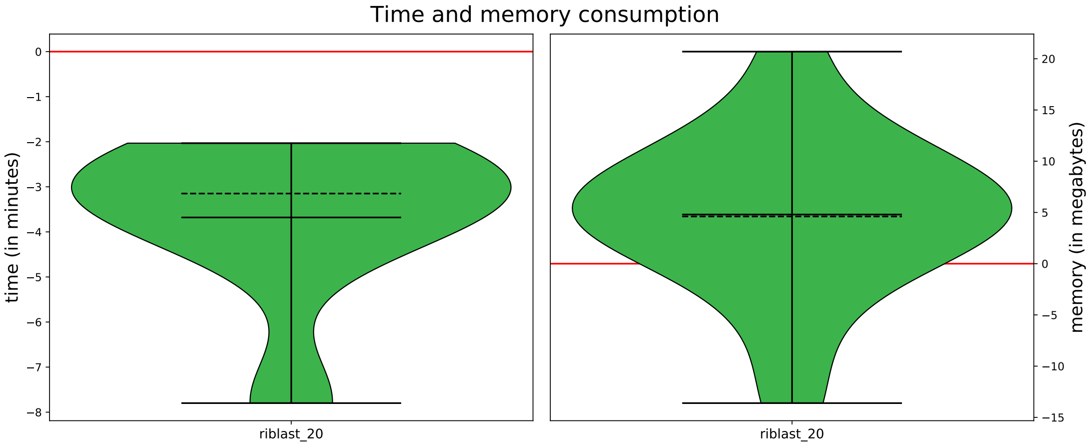


Figure 6: Time & memory



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