UNIVERSITY OF BRITISH COLUMBIA

Faster Elementary Steps in DNA Reaction Simulators

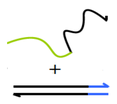
Boyan Beronov (*beronov@cs.ubc.ca*), Jordan Lovrod, Chenwei Zhang, Anne Condon

Motivation

Molecular programmers use elementary step simulators [1], e.g.,

Multistrand [2,3], to predict DNA reaction rates from secondary structure folding trajectories. Simulators should be fast!

Input

Sequence(s):	Initial state (secondary structure):	Final state (secondary structure):
…ACATTAG ACATTAG… …CTAATGT		

Multistrand trajectory generation:

repeat, starting from initial state:

Move generation:

sample base pair to form or break from applicable (e.g., pseudoknot free) moves

$$P(\text{move}) = \frac{\text{rate(move)}}{\sum \text{applicable move rates}}$$

$$\text{Holding time of state} \sim \text{Exp}\left(\frac{1}{\sum \text{applicable move rates}}\right)$$

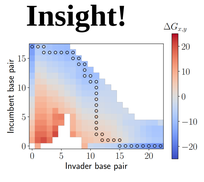
State update: compute rates for possible next moves

until reach final state

Output

Trajectories

Insight!



Contribution

Problem: ***State updates*** of current simulators [2,3,4] are slow in the worst case: $\Theta(N^2)$ time, where N is the total number of bases in the input strands. Can we do better?

Solution: By partitioning moves into ***types***, with moves of the same type having the same rate, we reduce the worst-case time for ***state updates*** to $O(N)$, keeping ***move generation*** at $O(\log N)$. $\Theta(N^2)$ pre-computation time is also required.

Methodology

Move generation $O(\log N)$ time via generative tree model:

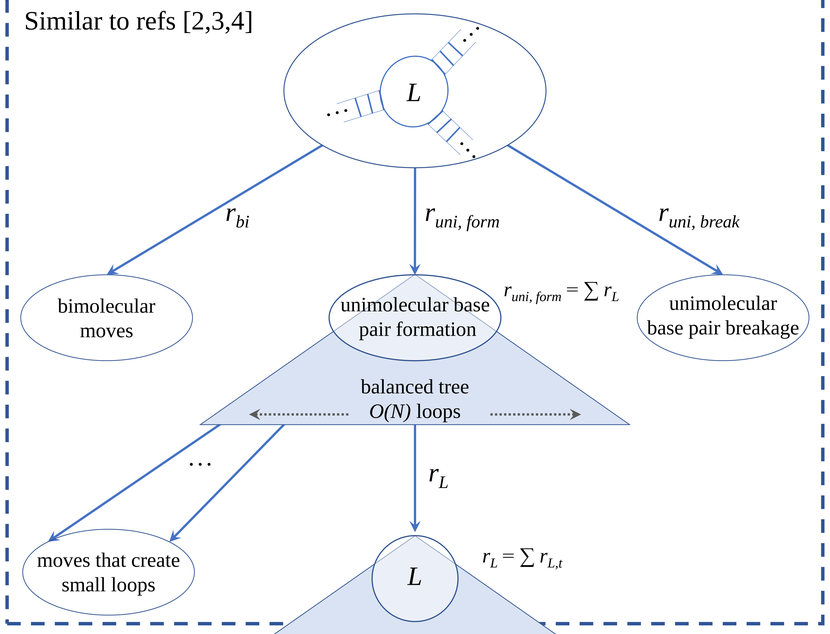
- Edge labels represent cumulative rates

$$\text{incoming edge rate} = \sum \text{outgoing edge rates}$$

- Binary search on a pre-computed table at the leaves

State update $O(\log N)$ time to update balanced tree of loops. $O(N)$ time to create balanced tree of ***move types*** for new loops

Similar to refs [2,3,4]



New: ***move types***

Implicit representation in **data structure**

Analysis

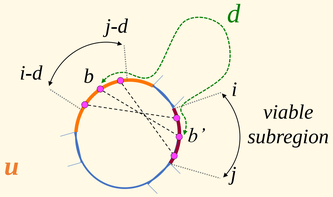
Assumptions:

- Pseudoknot free structures; $O(1)$ branches per loop
- Move rate computable in $O(1)$ time from :
  i. free energy change, plus
  ii. local context of base pair formed (consistent with Metropolis, Kawasaki, Arrhenius kinetic models)
- Free energy of "large" loops can be computed in $O(1)$ time from
  i. Local contexts around closing base pairs and
  ii. Number of unpaired bases

In loop *L* , *move type t* is specified as (*c*, *d*, *u*, *u'*) where

- ***c*** is local context of base pair (*b*,*b'*) formed
- ***d*** = *b'* – *b* (where *b* < *b'*) is sequence-level distance
- ***u*** and ***u'*** identify the unpaired region(s) in *L* that contain *b* and *b'*

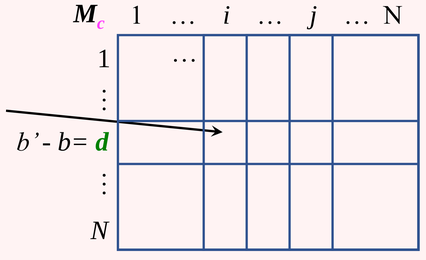
Distance ***d*** constrains *b*' to be within a unique (possibly empty) *viable subregion* [*i*,*j*] of ***u'***, where [*i*-*d*,*j*-*d*] is a subregion of ***u***



Data structure for loop *L*, ***move type t*** stores:

- *r*(*L*,*t*): ***all moves in L of type t have the same rate***
  - $O(1)$ time (see assumptions above)
- *M*(*L*,*t*): number of moves in *L* of ***type t***:
  - $O(1)$ time from pre-computed table:
  - $M(L,t) = M_c[j] - M_c[i]$, where [*i*,*j*] is the viable subregion of ***u'***

base pairs of distance ***d*** with local context ***c*** in prefix of DNA sequence up to *i*



References:

1. Flamm, C., et al., RNA 6, 2000.
2. Schaeffer, J.M., Ph.D. thesis, Caltech, 2013.
3. Schaeffer, J.M., et al. Proc. DNA21, 2015.
4. Dykeman, E.C., Nucleic Acids Res., 43, 2015.