BIOINFORMATICS

Vol. 00 no. 00
Pages 1–19

Basin Hopping Graph: A computational framework to characterize RNA folding landscapes SUPPLEMENTAL MATERIAL

Marcel Kucharík[1], Ivo L. Hofacker[1–3], Peter F. Stadler[1,4–7], and Jing Qin[3,8]

[1]Institute for Theoretical Chemistry, Univ. Vienna, Währingerstr. 17, 1090 Vienna, Austria
[2]Research group BCB, Faculty of Computer Science, Univ. Vienna, Austria
[3]RTH, University of Copenhagen, Grønnegårdsvej 3, Frederiksberg, Denmark
[4]Dept. of Computer Science & IZBI & iDiv & LIFE, Leipzig Univ., Härtelstr. 16-18, Leipzig, Germany
[5]Max Planck Institute for Mathematics in the Sciences, Inselstr. 22, Leipzig, Germany
[6]Fraunhofer Institute IZI, Perlickstr. 1, Leipzig, Germany
[7]Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM87501, USA
[8]IMADA, Univ. Southern Denmark, Campusvej 55, Odense, Denmark.

1 PART A: CONNECTIVITY OF THE BHG

First, for a given RNA sequence, its energy landscape is connected. This is because for any pair given secondary structures A and B, there exists a path between A and B by first removing all the base pairs in $A \setminus B$ and then adding the base pairs in $B \setminus A$.

Next, we prove that the "base hopping graph" is connected. We start from Theorem 1 of (Klemm *et al.*, 2014), which states that for any two local minima, there exists a zig-zag path between them defined as following: Given a path $P = (v_0, v_1, \ldots, v_\ell, v_{\ell+1}) \in X$, if $v_k > v_{k+1} = \cdots = v_{l-1} < v_l$, then all the structures v_j for $k+1 \le j \le l-1$ are called *valley points*. Analogously, *peak points* are the structures v_j with $k+1 \le j \le l-1$ if $v_k < v_{k+1} = \cdots = v_{l-1} > v_l$. A path $P = (x = w_0, w_1, \ldots, w_\ell, w_{\ell+1} = y)$ is a zig-zag path on (X, f) if the following three conditions are fulfilled: (a) $\max_i f(w_i) = \mathsf{S}(x, y)$; (b) if $w_k > w_{k+1} = \cdots = w_{l-1} < w_l$ then there is a minimal shelf L such that $w_j \in L$ for $k+1 \le j \le l-1$ and (c) if $w_k < w_{k+1} = \cdots = w_{l-1} > w_l$ then each w_j with $k+1 \le j \le l-1$ is a direct saddle separating the nearest valley points that the path P passed before and after w_j .

LEMMA 1.1. *(Klemm* et al.*, 2014) If x, y are two local minima, then there exists a zig-zag path connecting x and y.*

PROOF. The definition of the saddle height guarantees there is a path $\wp$ from x to y whose height does not exceed $\mathsf{S}(x, y)$. Denote by $X^f(y)$ the connected component of the induced subgraph with vertex set $\{z \in V | f(z) = f(y)\}$. In the local search literature, $X^f(x)$ is often called a *plateau* or a *neutral network* (Van Nimwegen & Crutchfield, 2000).

Consider the graph $X^* = X/\sim_f$ derived from the original landscape X by contracting any $X^f(y)$ into a vertex of X^*. This contracts a path $\wp$ in X to a path $\wp^*$ in X^*.

To prove the theorem, all we need is to first construct a zig-zag path $P^* \in X^*$ from $\wp^*$ and then prove the existence of a zig-zag path $P \in X$ such that P^* is the resulted graph of P after the contraction. The latter is trivial since by construction, $X^f(y)$ is connected for any $y \in X$. Therefore the proof reduces to the construction of $P^* \in X^*$ from $\wp^*$. This construction is described as follows and illustrated in Fig. 1.

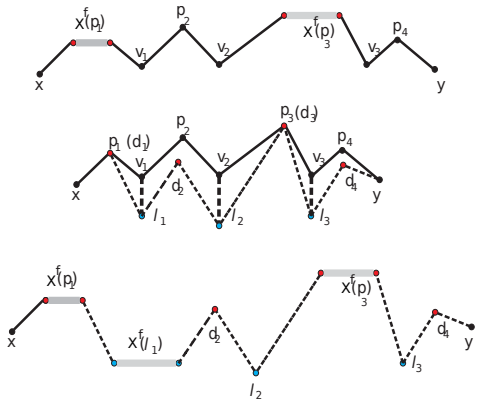

Fig. 1. The construction of the path $(\wp \to \wp^* \to P^* \to P)$ in the proof of Lemma 1.1. Bold lines in grey denote the path in the plateau $X^f(z)$ where z is inside, $z \in \{p_1, \ell_1, p_3\}$.

Let $\{v_i\}_{i=1}^t$ denote the valley points in $\wp^*$. From each valley point v_i, a gradient walk is simulated to reach some local minimum ℓ_i. Without loss of generality, we set $v_0 = \ell_0 = x$, $v_{t+1} = \ell_{t+1} = y$ and assume that all ℓ_i are different configurations. In this context, we observe that there exists a pair of hill-climbing walks from "adjacent" local minima ℓ_i and ℓ_{i+1} to some peak point of $\wp^*$, denoted by p_i. By definition, $f(p_i) \ge \mathsf{DS}[\ell_i, \ell_{i+1}]$. Depending on whether they are equivalent or not, there are two cases. In case of $f(p_i) = \mathsf{DS}[\ell_i, \ell_{i+1}]$, then we just substitute the pair of sections $([v_i, p_i], [p_i, v_{i+1}])$ in $\wp^*$ into the pair of hill-climbing walks from ℓ_i and ℓ_{i+1} to p_i, respectively. Otherwise, by definition, there must exist a configuration d_i such that $f(d_i) = \mathsf{DS}[\ell_i, \ell_{i+1}] < f(p_i)$. In this case, we substitute the pair of sections $([v_i, p_i], [p_i, v_{i+1}])$ in $\wp^*$ into the pair of hill-climbing walks from ℓ_i and ℓ_{i+1} to d_i, respectively. □

Thus the graph where LMs are adjacent only if there is a direct saddle between them, is connected. Since the BHG is obtained from this graph be removing only edges that can be replaced by path (with lower saddle height), it is also connected.

2 PART B

THEOREM 2.1. *For any two saddles s' and s'' either $B(s') \subseteq B(s'')$, $B(s'') \subseteq B(s')$ or $B(s'') \cap B(s') = \emptyset$ is satisfied, i.e., the basins below saddles of a landscape form a hierarchy with respect to the set inclusion order.*

PROOF. By definition, the basin $B(s) = B_{f(s)}(s)$ of s (Flamm *et al.*, 2002) is the set of all points in X that can be reached from s by a path whose elevation never exceeds $f(s)$.

Without loss of generality, we assume $f(s') \leq f(s'')$. Consider the saddle height between two saddles s' and s'', denoted by $\mathsf{S}(s', s'')$. There are three cases: (1) $\mathsf{S}(s', s'') = f(s') = f(s'')$; (2) $f(s') < \mathsf{S}(s', s'') = f(s'')$ and (3) $f(s') \leq f(s'') < \mathsf{S}(s', s'')$. Correspondingly, we have (1) $\mathcal{B}(s') = \mathcal{B}(s'')$; (2) $\mathcal{B}(s') \subset \mathcal{B}(s'')$ and (3) $\mathcal{B}(s'') \cap \mathcal{B}(s') = \emptyset$. Thus the theorem is true. □

3 PART C: BASIN HOPPING GRAPH AND BARRIER TREE

Let $G(V, E, \omega)$ be a finite, simple graph with the vertex set V, the edge set E and arbitrary edge weights $\omega : E \to \mathbb{R}$. We consider the following algorithm (Algorithm 3) to analyze the given graph G and obtain a binary, vertex-weighted tree $T_b(V_{T_b}, E_{T_b}, \omega_{T_b})$, accordingly. This algorithm is well-known as a naive version of the single linkage clustering with the time complexity $\mathcal{O}(|V|^3)$. In 1973, R. Sibson proposed an optimally efficient algorithm of only complexity $\mathcal{O}(|V|^2)$ known as SLINK (Sibson, 1973). Intuitively, we start with all vertices $x \in V$ in separate clusters (x). In each step, the pair of clusters connected by the smallest edge weight is merged. Edge weights to all other clusters are updated to the minimum of the edge weights of the merged clusters.

Require: $G(V, E, \omega)$
1: /*Initialize clusters, tree T_b and distance matrix*/
2: $\mathcal{L} \leftarrow \{(x)|x \in V\}$
3: $V_{T_b} \leftarrow \{(x)|(x) \in \mathcal{L}\}$ and $E_{T_b} \leftarrow \emptyset$
4: **for all** $(x) \in V_{T_b}$ **do**
5: $\quad \omega_{T_b}((x)) \leftarrow f(x)$
6: **end for**
7: **for all** $((x), (y)) \in \mathcal{L} \times \mathcal{L}$ **do**
8: $\quad W_{xy} \leftarrow \omega_{xy}$ if $\{x, y\} \in E$ and $W_{xy} \leftarrow \infty$ if $\{x, y\} \notin E$
9: **end for**
10: **while** $|\mathcal{L}| > 1$ **do**
11: $\quad$ Find a pair of clusters $\{(u), (v)\}$ such that $W_{uv} = \min_{(x,y)\in\binom{|\mathcal{L}|}{2}} W_{xy}$
12: $\quad$ /*update the distance matrix and the T_b-tree*/
13: $\quad$ **for all** $(x) \in \mathcal{L} \setminus \{(u), (v)\}$ **do**
14: $\quad\quad W_{ux} = \min\{W_{ux}, W_{vx}\}$
15: $\quad$ **end for**
16: $\quad$ create a new (internal) T_b-vertex $(uv) \leftarrow (u) \cup (v)$ with $\omega_{T_b}((uv)) \leftarrow W_{uv}$
17: $\quad V_{T_b} \leftarrow V_{T_b} \cup (uv)$
18: $\quad E_{T_b} \leftarrow E_{T_b} \cup \{(uv), (u)\} \cup \{(uv), (v)\}$
19: $\quad \mathcal{L} \leftarrow \mathcal{L} \setminus \{(v)\}$
20: **end while**

The single linkage clustering implicitly defines a binary tree T_b in which each internal node $(uv) = (u) \cup (v)$ corresponding to the merging of the clusters (u) and (v) has the minimum weight W_{uv}. Note that this algorithm is not deterministic if the pair with minimal weight (Line 3) is not unique, i.e., if

$$|\{(u, v)|W_{uv} = \min_{(x,y)\in\binom{|\mathcal{L}|}{2}} W_{xy}\}| > 1.$$

Clearly, ambiguities concern only pairs with the same weights. A unique tree T is obtained by contracting all edges in T_b for which the adjacent vertices are internal nodes with the same weight.

The barrier tree of a given landscape (X, f) can also be interpreted into a vertex weighted tree $T_b^*(V_{T_b}^*, E_{T_b}^*, \omega_{T_b}^*)$ with the local minima as its leaves. Internal nodes indicate the merging of basins surrounding two local minima at their saddle height.

THEOREM 3.1. *The barrier tree $T_b^*(V_{T_b}^*, E_{T_b}^*, \omega_{T_b}^*)$ of the landscape (X, f) is the tree $T_b(V_{T_b}, E_{T_b}, \omega_{T_b})$ computed by the single linkage clustering from the complete graph $K(V_K, E_K, w_K)$ whose vertex set V_K includes the local minima of the landscape and whose edges have weight $\omega_K(\{x, y\}) = \mathsf{S}(x, y)$ for all $\{x, y\} \in E_K$.*

PROOF. To prove this observation, we need to introduce a notion called the level number $LN : V \to \mathbb{Z}^*$ for each vertex in the tree. The level number is defined recursively: (1) the level number of each leaf is 0 and (2) the level number of each internal node v is defined as $\max_x\{LN(x)\} + 1$ where x runs over all the children of v. Thus the observation is reduced to:
For each level number $\ell \geq 0$, there exists an one-to-one mapping $Id : F_b^\ell \to F_b^{*,\ell}$ between the subgraph (forest) of F_b^ℓ and induced by vertices in T_b with level number $\leq \ell$ and the corresponding induced subgraph $F_b^{*,\ell}$ of T_b^*.

Firstly, when $\ell = 0$, the statement is trivial since the leaves for both forests are the set of local minima in the landscape. Now we assume the statement is true for all the vertices with level numbers less than or equal to k. Now consider an arbitrary vertex v in F_b^{k+1} with the level number $k+1$, we need to prove: (1) $Id(v) \in F_b^{*,k+1}$; (2) if w is a child of v in F_b^{k+1} then $Id(w)$ is a child of $Id(v)$ in $F_b^{*,k+1}$ and (3) $\omega_{T_b}(v) = \omega_{T_b}^*(Id(v))$.

Clearly, since the level number of v is $k + 1$, then there exists at least one of its children, say w whose level number is k. Now consider the parent node $v^* \in T_b^*$ of $Id(w)$ and its children set $\{w_0 = Id(w), w_1, \dots w_t\}$. Let m_i and m_j be the leaves of w_i and w_j $(0 \leq i < j \leq t)$, respectively. Then by definition of the barrier tree, the saddle height between (m_i, m_j) is exactly $w_{T_b}^*(v^*)$. Now consider the subtrees rooted in $w_i^{-1} =: Id^{-1}(w_i)$ and $w_j^{-1} =: Id^{-1}(w_j)$ in T_b. By construction, the level numbers of w_i^{-1} and w_j^{-1} are no more than k. Therefore by assumption, they both exist. Furthermore, there exist $m_i \in w_i^{-1}$ and $m_j \in w_j^{-1}$. Now we claim $\omega_{T_b}(v) = \omega_{T_b}(v^*)$. On one hand side, the single clustering algorithm gives rise to $\omega_{T_b}(v) = \min_{(x,y)} \mathsf{S}(x, y)$, where x and y run over all the leaves of the subtrees rooted in w_i^{-1} and w_j^{-1}, respectively. Therefore, $\omega_{T_b}(v) \leq \mathsf{S}(m_i, m_j) = w_{T_b}(v^*)$. On the other hand, if $\omega_{T_b}(v) < \omega_{T_b}(v^*)$, then there exists a pair of local minima $m_p \in T_b(w_i^{-1})$ and $m_q \in T_b(w_j^{-1})$ with $\omega_{T_b}(v) = \mathsf{S}(m_p, m_q) > \max\{\omega_{T_b}(w_i^{-1}), \omega_{T_b}(w_j^{-1})\}$. In which, $T_b(w)$ denotes the subtree of T_b rooted at w. In this case, we can construct a path $m_i \to m_p \to m_q \to m_j$ with cost $\mathsf{S}(m_p, m_q)$ - strictly less than $\mathsf{S}(m_i, m_j)$, which is a contradiction to the definition of saddle height. Therefore, the claim is true. Thus, we set $Id(v) = v^*$ and the proof is complete. □

THEOREM 3.2. *Let $B(V_B, E_B, \omega_B)$ be the basin hopping graph of the landscape (X, f) with V_B denoting the set of local minima in (X, f). Then, for all $\{x, y\} \in \binom{V_B}{2}$,*

$$\mathsf{S}(x, y) = \min_{\wp\in path(x,y)} \max_{\{u,v\}\in\wp} \omega_B(\{u, v\}) \tag{1}$$

where path(x, y) denotes the set of the paths between x and y in $B(V_B, E_B, \omega_B)$ and each path $\wp \in path(x, y)$ is represented by the sequence of edges it passes through.

PROOF. This theorem is indicated in the proof the Lemma 1.1 since the crucial observation in Lemma 1.1 is that every path connecting two local minima can be replaced by a sequence of local minima that are connected by directed saddles. □

THEOREM 3.3. *The barrier tree $T_b^*(V_{T_b}^*, E_{T_b}^*, \omega_{T_b}^*)$ of the landscape (X, f) is the tree $T_B(V_{T_B}, E_{T_B}, \omega_{T_B})$ computed by single linkage clustering from the BHG.*

PROOF. According to Theorem 3.1, we only need to prove that there exists an identity map $I : V_{T_B} \to V_{T_b}$ between the trees $T_B(V_{T_B}, E_{T_B}, \omega_{T_B})$ and $T_b(V_{T_b}, E_{T_b}, \omega_{T_b})$ constructed in Theorem 3.1 with the following properties: (1) for two vertices $\{x, y\} \subset V_{T_B}$ if x is a child of y, then $I(x)$ is a child of $I(y)$ and (2) for any $x \in V_{T_B}$, there is $\omega_{T_B}(x) = \omega_{T_b}(I(x))$.

To prove this, we will use induction on the level number ℓ of these two trees again. When $\ell = 0$, the proof is trivial. Assume that the two forests F_B^k and F_b^k are induced by vertices of level numbers $\ell \leq k$ in T_B and T_b, respectively. Consider an arbitrary vertex $v \in V_B$ with the level number $\ell = k + 1$, by definition, it has at least one child z of the level k. Consider the T_b-subtree rooted in the parent node v^* of the vertex $I(z)$. Let $\{w_0 = I(z), w_1, \ldots, w_t\}$ denote the set of children of v^*. Consider an arbitrary pair of children w_i and w_j, where $0 \leq i < j \leq t$. Furthermore, let m_i and m_j be the leaves of the T_b-subtrees rooted in w_i and w_j, respectively. Clearly, there exists $\omega_{T_b}(v^*) = \mathsf{S}(m_i, m_j)$. According to the assumption that the statement is true for $\ell \leq k$, m_i and m_j, the leaves of the T_B-subtrees are rooted in w_i^{-1} and w_j^{-1} as well. According to the construction of the single linkage clustering and Theorem 3.2, we have $\omega_{T_B}(v) \leq W((w_i^{-1}), (w_j^{-1})) \leq \mathsf{S}(m_i, m_j)$, where $W((w_i^{-1}), (w_j^{-1}))$ denotes the distance between (w_i^{-1}) and (w_j^{-1}). Clearly $W((w_i^{-1}), (w_j^{-1})) < \mathsf{S}(m_i, m_j)$ indicates the existence of a zig-zag path between m_i and m_j with a cost strictly less than its saddle height, which contradicts to the Lemma 1.1. Therefore, we obtain $\omega_{T_B}(v) = \mathsf{S}(m_i, m_j) = \omega_{T_b}(v^*)$. □

PART D: THE NUMBER OF DETECTED LOCAL MINIMA GROWS LINEARLY WITH RESPECT TO THE RUNNING TIME

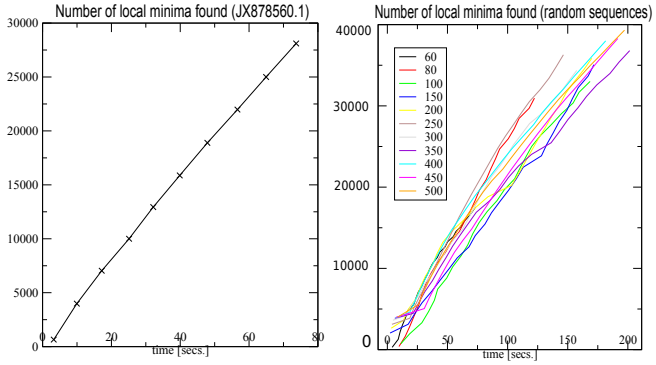

Fig. 2. The number of detected local minima grows linearly with respect to running time: for both natural sequences and random sequences. (Left) The performance of `RNAlocmin` for *Melitaea cinxia* U6 snRNA JX878560.1 (107nt) (Right) The average performance of `RNAlocmin` for random generated RNA sequences of length 60–500. The adaptive ξ-schedule is effective since for different RNA lengths, the speed of finding new LMs keeps stable, i.e. the number of detected local minima grows linearly with respect to the running time.

PART E: ADDITIONAL EXAMPLES OF `RNALOCMIN`

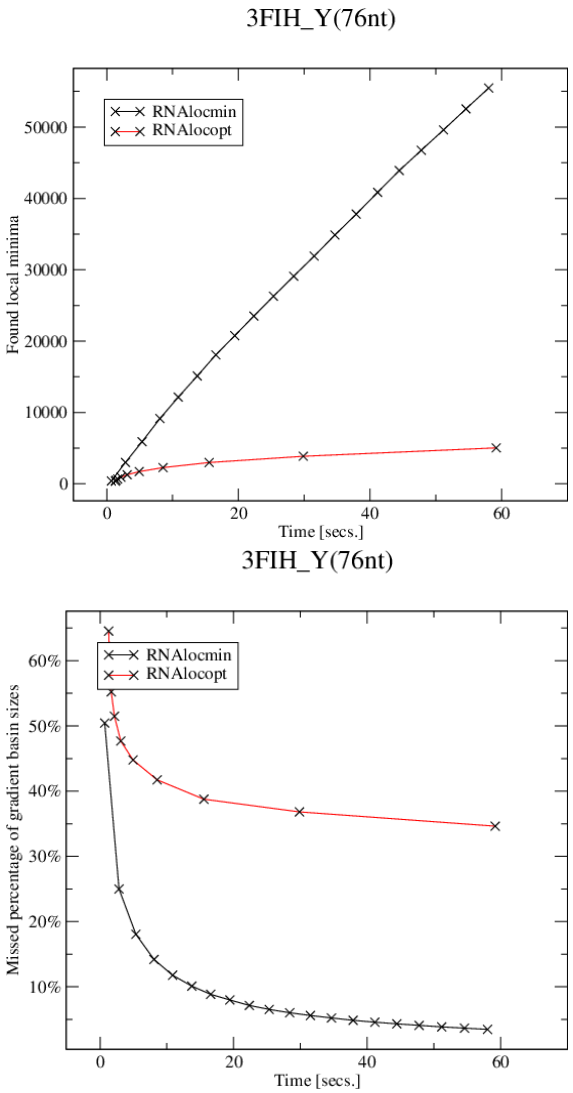

Fig. 3. The comparison between `RNAlocmin` and `RNAlocopt` for the A/T-site tRNA Phe 3FIH_Y (76nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

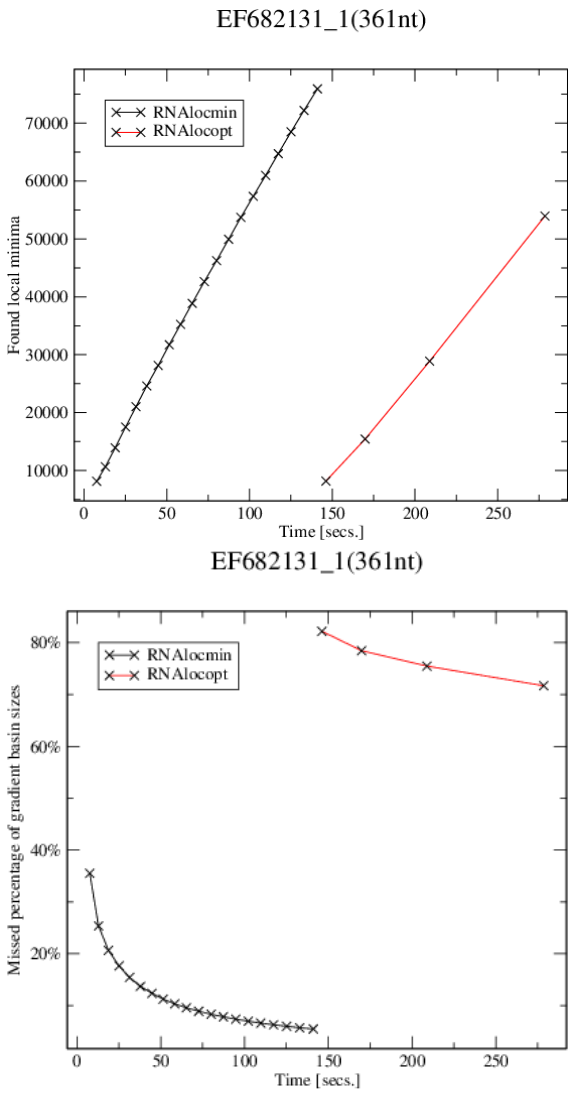

Fig. 4. The comparison between `RNAlocmin` and `RNAlocopt` for the snRNA EF682131.1 (361nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

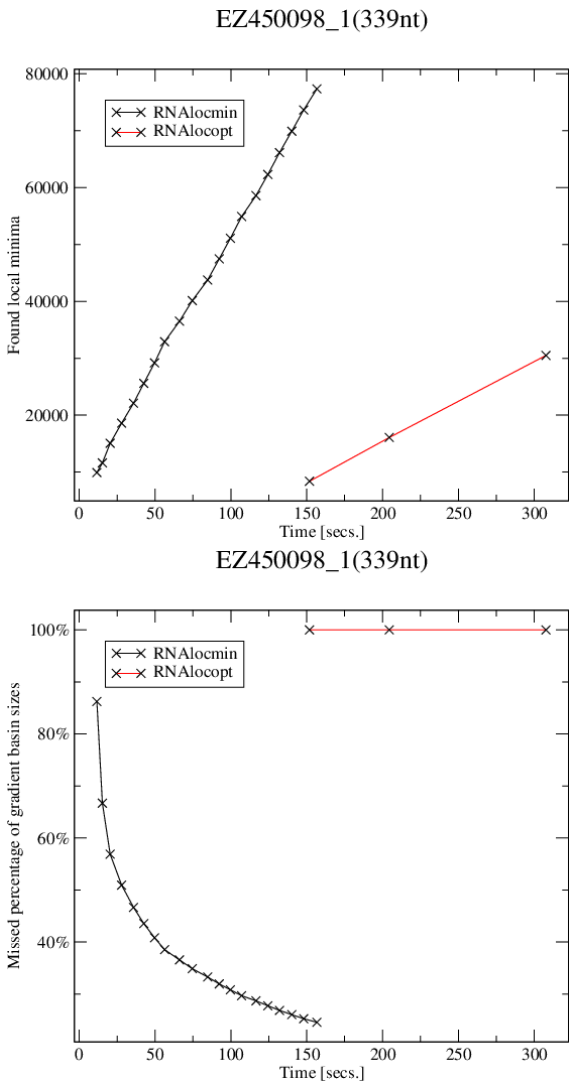

Fig. 5. The comparison between `RNAlocmin` and `RNAlocopt` for the mRNA EZ450098.1 (339nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

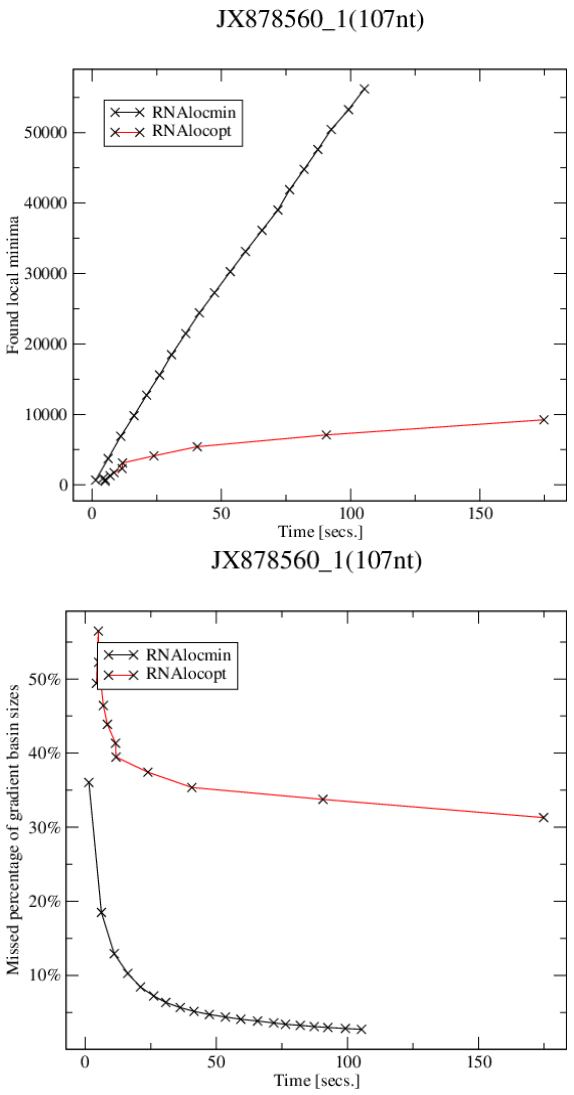

Fig. 6. The comparison between `RNAlocmin` and `RNAlocopt` for the U6 snRNA JX878560 (107nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

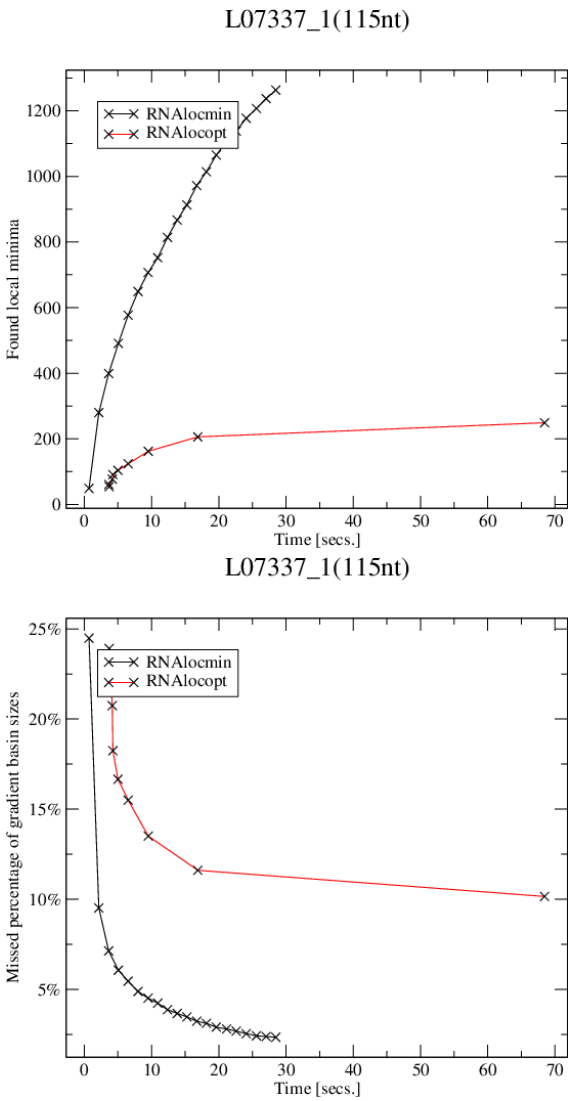

Fig. 7. The comparison between `RNAlocmin` and `RNAlocopt` for the SV11 RNA L07337.1 (115nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

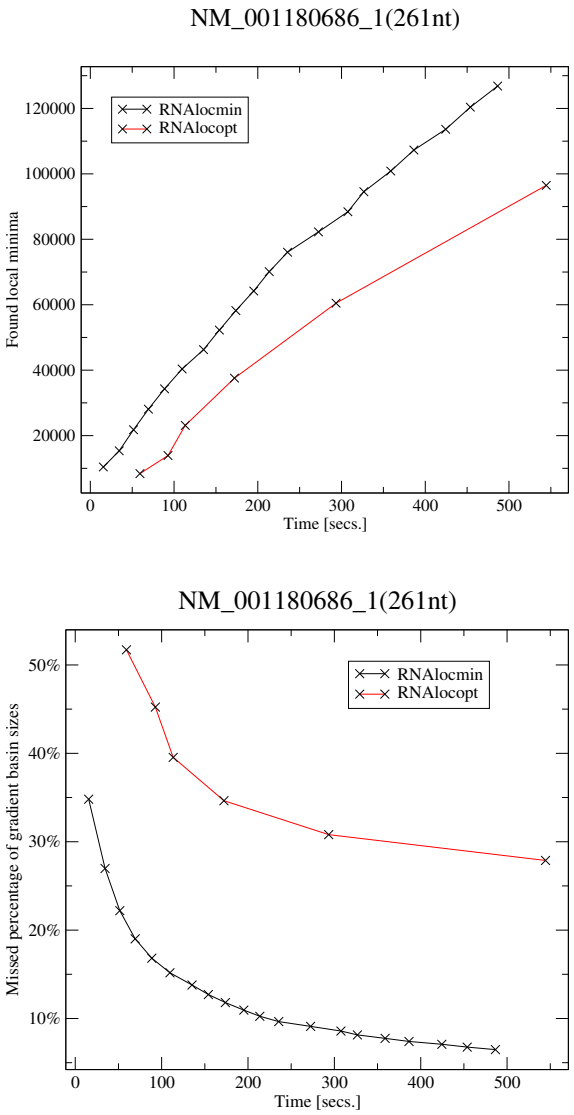

Fig. 8. The comparison between `RNAlocmin` and `RNAlocopt` for the mRNA NM_001180686.1 (261nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

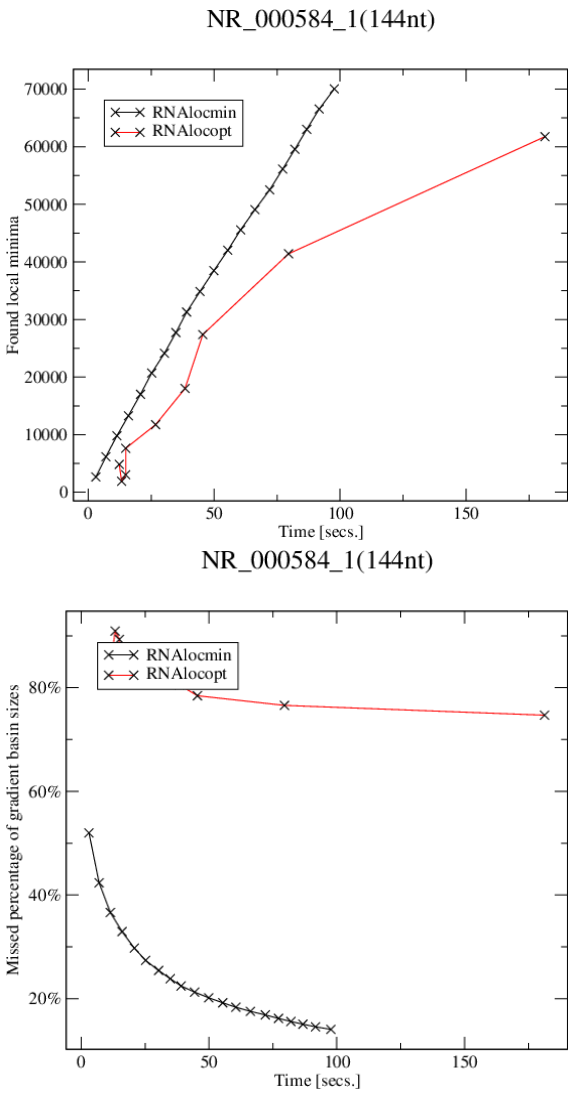

Fig. 9. The comparison between `RNAlocmin` and `RNAlocopt` for the ncRNA NR_000584.1 (144nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

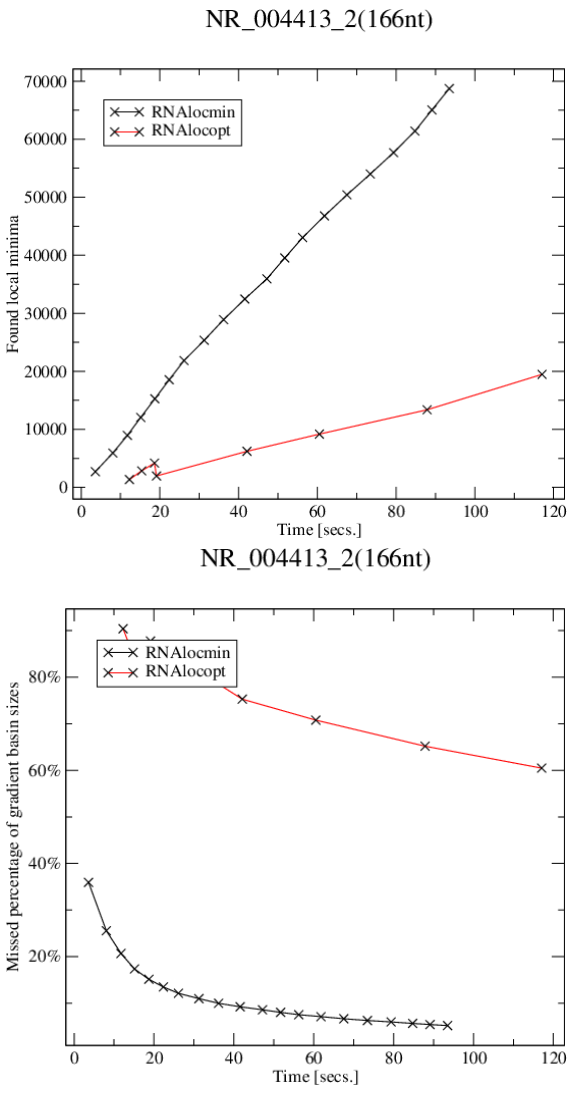

Fig. 10. The comparison between `RNAlocmin` and `RNAlocopt` for the small nuclear RNA NR_004413.2 (166nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

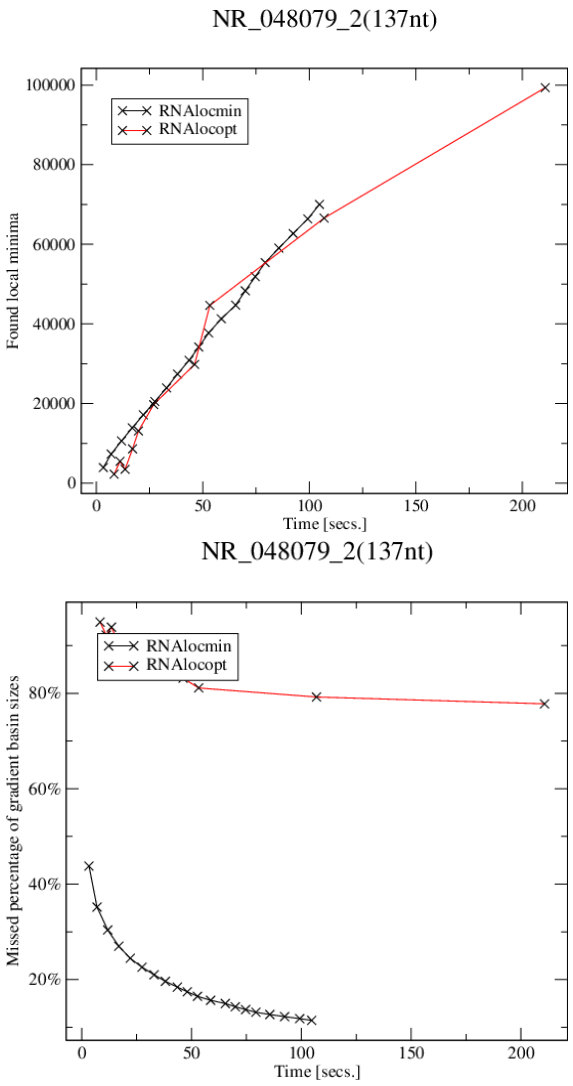

Fig. 11. The comparison between `RNAlocmin` and `RNAlocopt` for the ncRNA NR_048079.2 (137nt). The sample size for `RNAlocmin` was limited to $N = 400,000$ structures. The fraction of undetected basins was estimated by an enumeration of $10 \cdot N$ suboptimal structures with `RNAsubopt -e` and the subsequent evaluation of gradient basins with `barriers`.

4 PART F: RESULTS OF BHGBUILDER

As expected, all examples show that BHGbuilder outperforms or at least performs equally well as findpath. To be precise, the results of 10 examples can be divided into 2 categories based on their result comparisons: (1) Three RNAs (NR_000584.1, NR_004413.2, and NR_048079.2) with significant improvements between 0.2–2.0 kcal/mol in about 30% of all the LM pairs. We present also difference plots for these examples. (2) Seven RNAs with minor improvements. For two relatively short RNAs (NR_073613.1 and 3FIH_Y), both two algorithms obtained almost the exact results derived from barriers, therefore no improvement was possible.

For these three RNAs in Category (1), a better estimation of saddle heights between LM pairs (in particular these ones with 1–2.5 kcal/mol improvements) can help to derive more exact RNA kinetic information, since kinetic properties are exponentially dependent on the energy barriers. This is because the saddle height between two BHG-adjacent LMs are closely related with the transition rate between their corresponding basins.

For the seven RNAs in Category (2), we point out here, estimating the saddle heights is just one of the two important functions of BHGbuilder. The other one is to detect adjacency of basins represented by their LMs via the filtration procedure. This function, however, can not be achieved via simply utilizing some established procedure such as findpath for all possible pairs of LMs.

Examples are compared to findpath and also to barriers where applicable (only the smallest three examples). The x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights estimations derived from different methods.

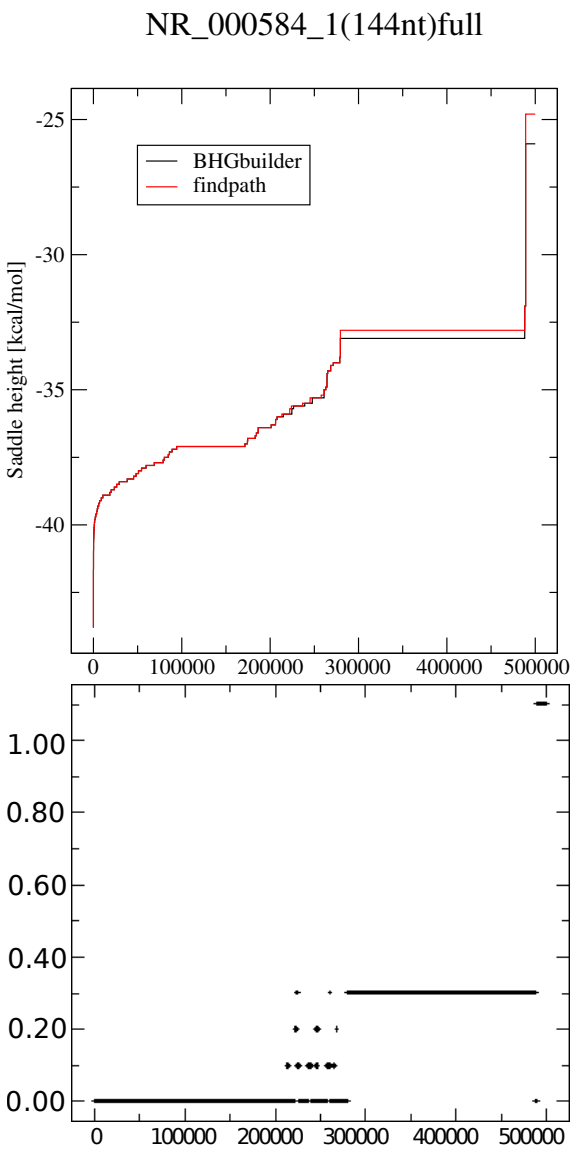

Fig. 12. The comparison of the saddle height estimates of BHGbuilder and findpath for ncRNA NR_000584.1 (144nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. The 2nd panel shows the difference in the saddle height prediction between BHGbuilder and findpath algorithm.

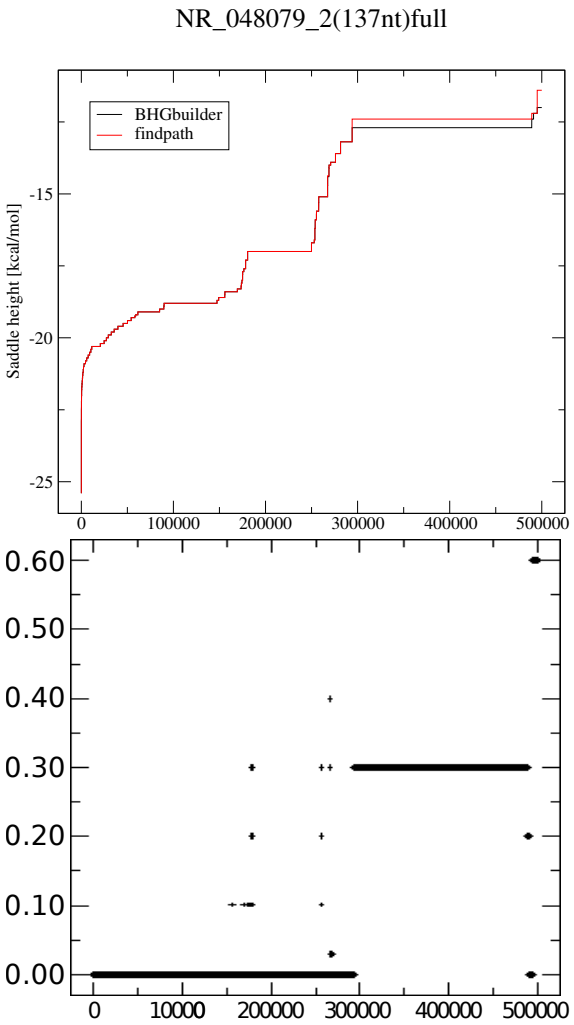

Fig. 13. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for ncRNA NR_048079.2 (137nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. The 2nd panel shows the difference in the saddle height prediction between `BHGbuilder` and `findpath` algorithm.

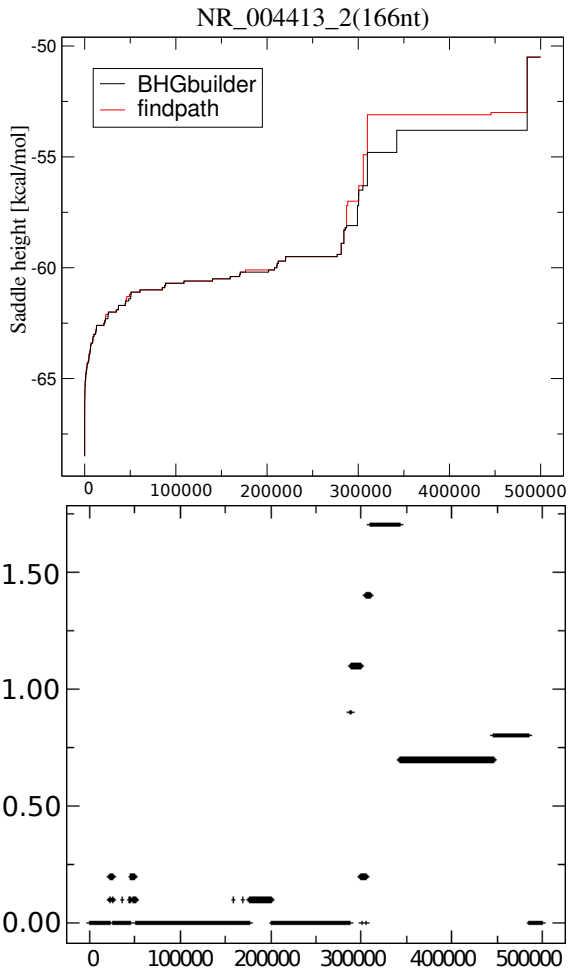

Fig. 14. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for small nuclear RNA NR_004413.2 (166nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. The 2nd panel shows the difference in the saddle height prediction between `BHGbuilder` and `findpath` algorithm.

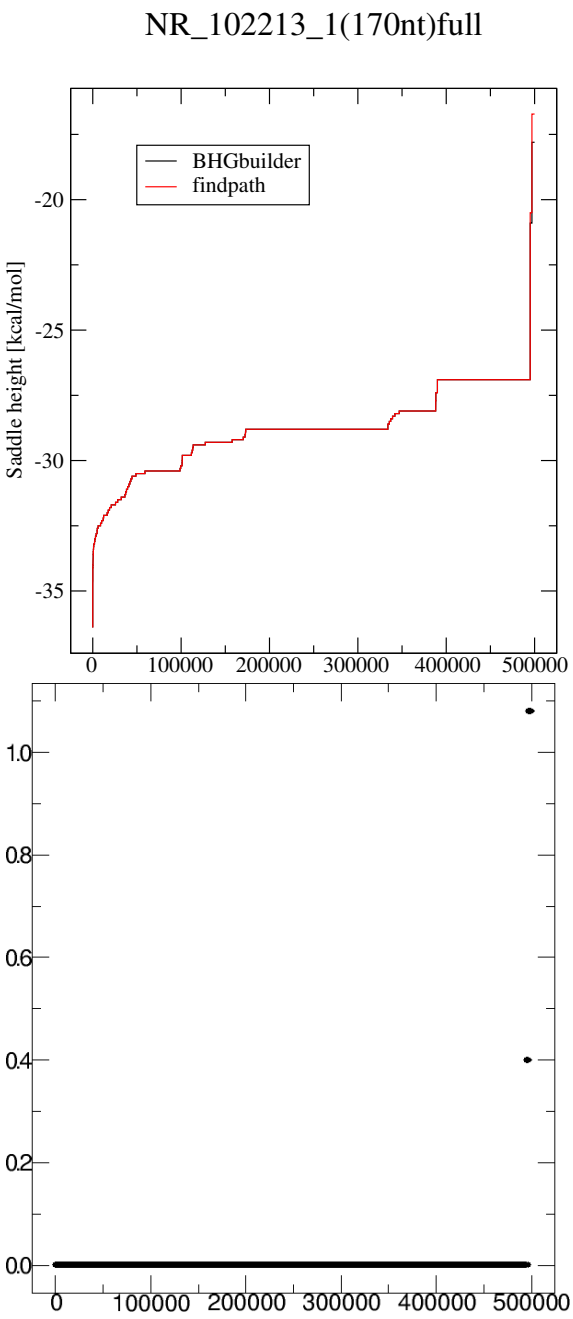

Fig. 15. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for ncRNA NR_102213.1 (170nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. The 2nd panel shows the difference in the saddle height prediction between `BHGbuilder` and `findpath` algorithm.

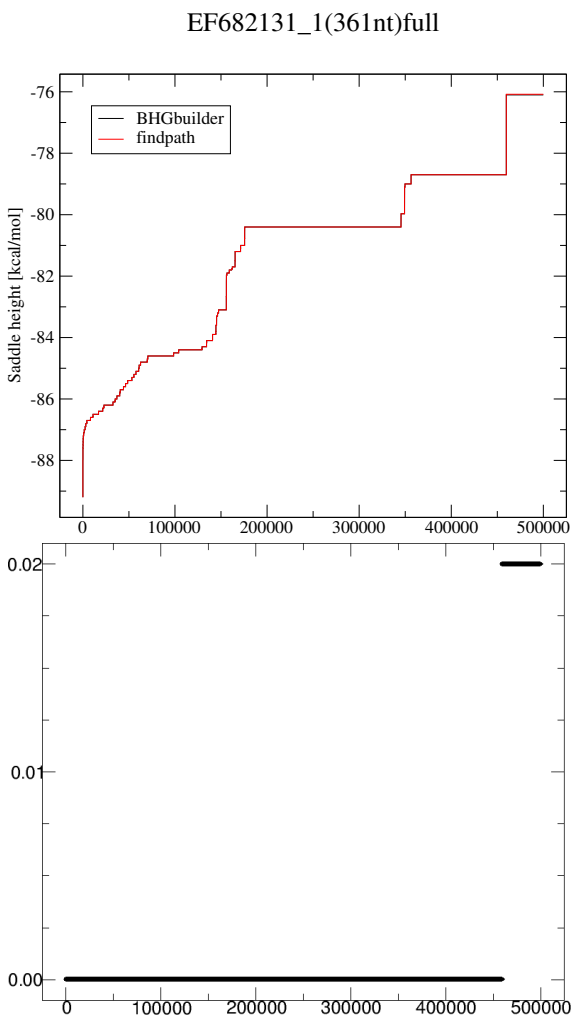

Fig. 16. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for snRNA EF682131_1 (361nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. The 2nd panel shows the difference in the saddle height prediction between `BHGbuilder` and `findpath` algorithm.

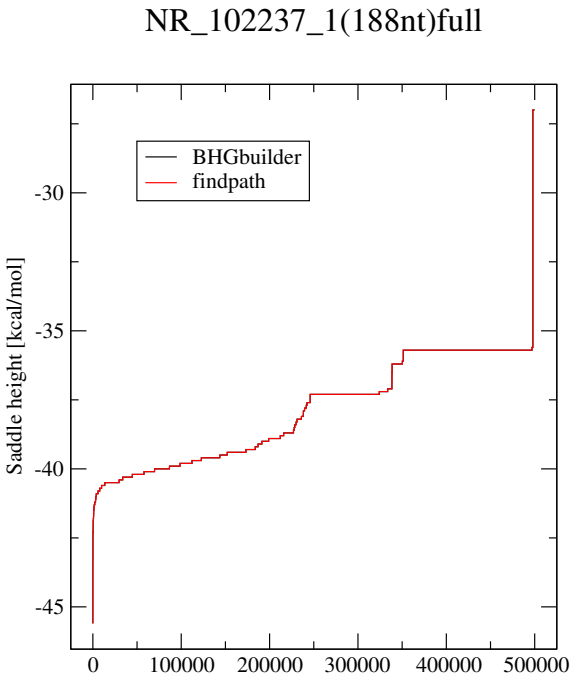

Fig. 17. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for ncRNA NR_102237.1 (188nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. Both programs performed equally in this example.

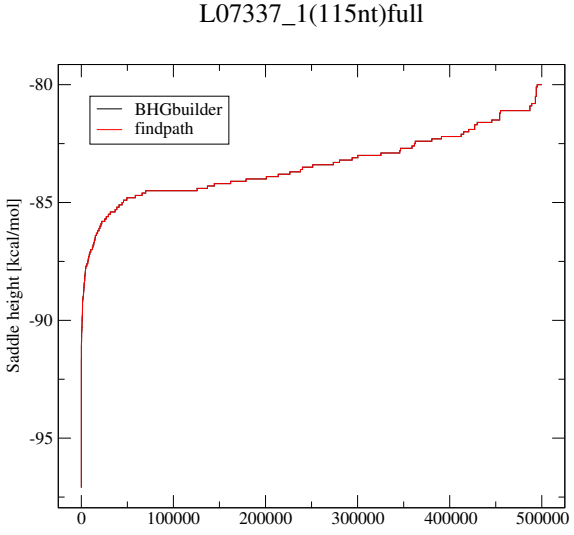

Fig. 18. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for SV11 RNA L07337_1 (115nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. Both programs performed equally in this example.

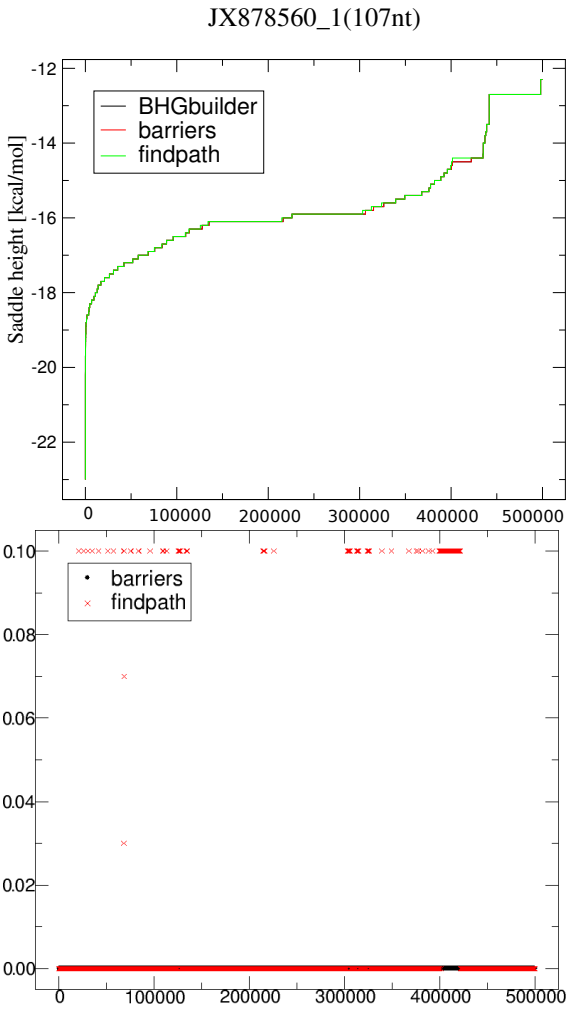

Fig. 19. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for U6 snRNA JX878560.1 (107nt) . Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. The 2nd panel shows the difference in the saddle height prediction between `BHGbuilder` and competing algorithms.

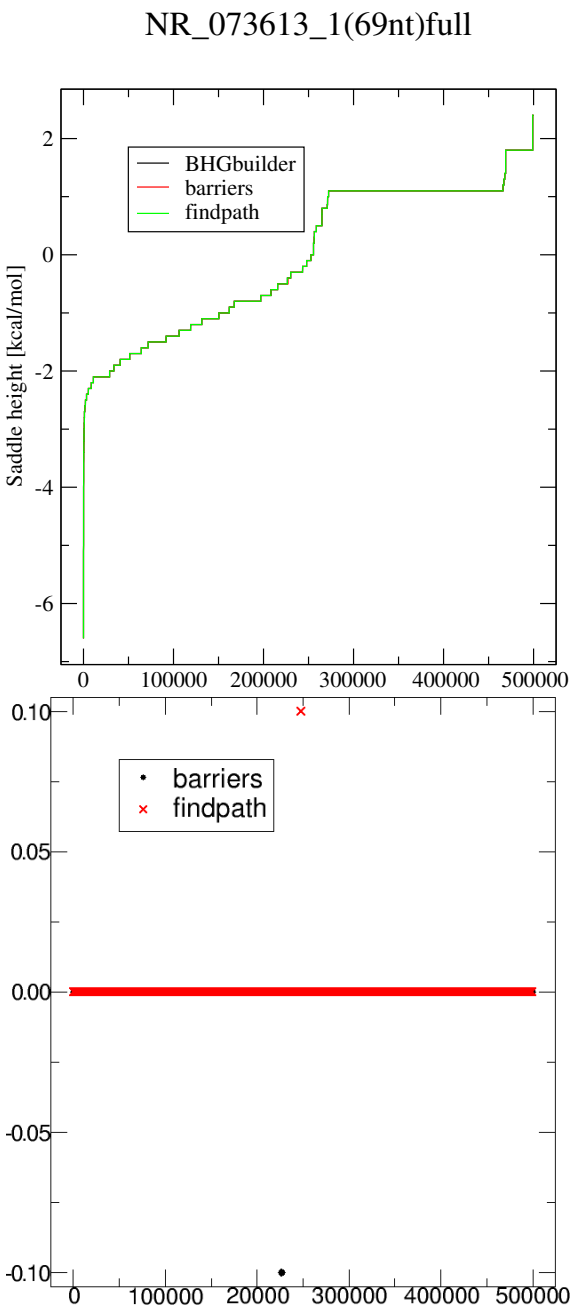

Fig. 20. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for ncRNA NR_073613.1 (69nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. The 2nd panel shows the difference in the saddle height prediction between `BHGbuilder` and competing algorithms.

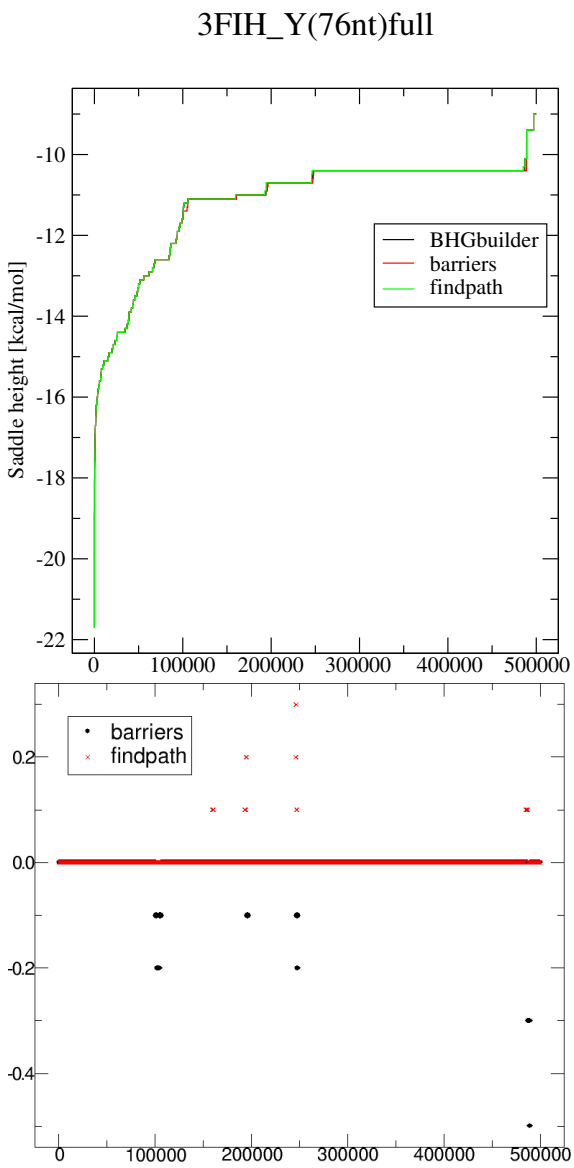

Fig. 21. The comparison of the saddle height estimates of `BHGbuilder` and `findpath` for tRNA phe 3FIH_Y (76nt). Here, the x-axes denote the indices of LM-pairs which are sorted according to their saddle heights in an increasing order and the y-axes are the corresponding saddle heights [kcal/mol] estimations derived from different methods. The 2nd panel shows the difference in the saddle height prediction between `BHGbuilder` and competing algorithms.

PART G: FOLDING KINETICS USING THE BHG

Folding kinetics on a toy example
`GUGUCGCUUUCGAUUAAGGACCUACAACAGGCU`
for different approaches are shown here. A hundred lowest local minima for this sequence were computed, the local minima with 50th lowest energy have been assigned probability of 1 at the start of the experiment. Figures capture the population probabilities of different local minima (only those with population density $> 5\%$ shown) of the approach by Wolfinger *et al.* (2004), the approach using Arrhenius rates on the barrier tree (without using the topology), and the approach using Arrhenius rate on the BHG (the topology is taken into account). The exhaustive enumeration approach depicted in the first plot closely approximates the real kinetics as is discussed in Wolfinger *et al.* (2004). Therefore, it can be used as a ground truth for comparison of the latter two. However, this approach consumes a lot of system resources, making it available only for sequences below 100nt. The second approach require only a precise saddle height approximation to be done, but it misses a lot of kinetic properties (for example the local minima no.13 is completely missing from the picture). Finally, our approach using both the precise saddle height estimation and the topology reconstructed by `BHGbuilder` performs very closely to the exhaustive enumeration approach.

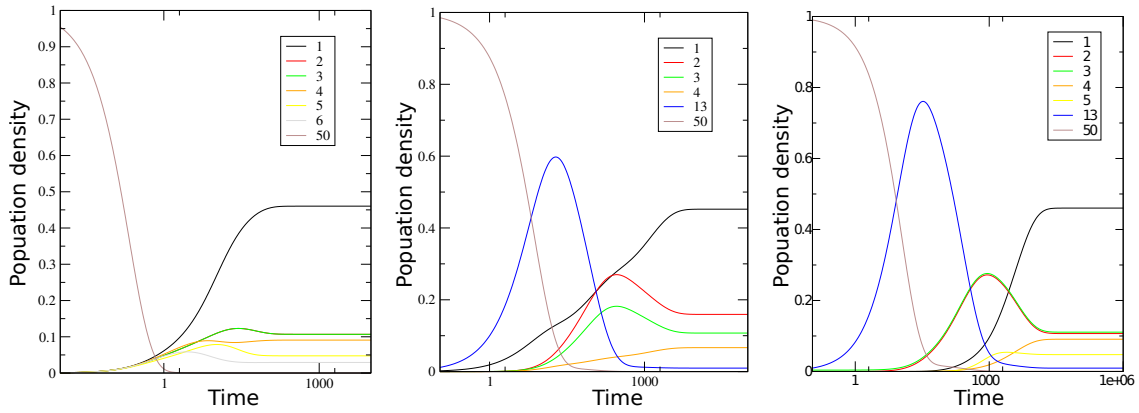

Fig. 22. The comparison of folding kinetics computed by different methods. Transition rates are computed using the Arrhenius kinetic rule based on different underlying transition graphs: (Left) barrier tree (Middle) BHG. In (Right), transition rates were computed using the exhaustive enumeration approach by Wolfinger *et al.* (2004). The time axis is given in arbitrary units which would need to be scaled with the help of an experimental data. The Arrhenius approximation (Left and Middle) also requires an entropic pre-factor that cannot be determined from the graphs, hence the time units in the different approximations are shifted by an unknown constant factor relative to each other.

PART H: PRELIMINARY RESULTS OF BHG TAKING PSEUDOKNOTS INTO CONSIDERATION

Both the BHG and the sampling strategy for local minima generalizes to structures with pseudoknots in a straightforward manner. To this end, three issues need to be addressed: (a1) the extension of the search space to a suitable class of pseudoknots, (a2) the energy function necessary to score these pseudoknots, and (a3) the definition of an appropriate move set.

There is no generally accepted or universally used class of pseudoknots. Instead, a wide variety of more or less restrictive sub-classes has been explored in the literature. These choices are driven less by biophysical considerations but rather by algorithmic practicability, see e.g. (Condon *et al.*, 2004). A major practical concern is that the energy models for pseudoknots are simple, heuristic extensions of the standard energy model (Mathews *et al.*, 1999) that use "developer-defined" energy penalties to score pseudoknots. These parameters are grounded in very sparse experimental data. An alternative, rather general energy function for pseudoknotted structures has been derived from the "cross-linked gel model" (Isambert & Siggia, 2000), it however suffers from the same lack of experimental data. Furthermore, no open source implementation of this energy function is available.

A key constraint for our approach is that we require a reasonably efficient way to generate structures with prescribed expected energies in order to construct a generalization of `RNAlocmin`. In practice, this restricts us to dynamic programming approaches. To our knowledge, the only software that implements Boltzmann sampling is `gfold` (Reidys *et al.*, 2011). It computes the so-called γ_1-structures, see Fig. 23, which comprise 4 basic types of pseudoknots characterized by the topological genus $g = 1$ of their "elementary" components, see (Bon *et al.*, 2008; Reidys *et al.*, 2011) for details.

With small modifications to the implementation, `gfold` can be used as a replacement for `RNAsubopt -p` that considers a larger class of RNA structures. It is computationally much more demanding: the folding step takes $O(n^6)$ time and $O(n^4)$ space. Sampling a single structure requires $O(n^5)$ time compared to $O(n^2)$ for sampling pseudoknot-free secondary structures. In practice, this restricts the method to moderate sequence lengths.

Much more efficient sampling algorithms can be devised e.g. using the boustrophedon method (Ponty, 2008), to make the `RNAlocmin` approach feasible also for much larger pseudoknotted RNAs.

In order to implement the gradient walk required in `RNAlocmin` we need a move set within the class of γ_1-structures. Opening and closing of individual base pairs is of course sufficient. The difficulty is to efficiently determine which base pairs can be added without leaving the class of γ_1-structure and to compute the resulting change in energy without re-evaluating the entire structure. An example of an invalid move is shown in Fig. 24.

Because of these difficulties, we restrict ourselves in this paper to the subset of γ_1-structures with at most one H-type pseudoknot. Fig. 25 shows how to add base pairs in order to obtain a valid pseudoknot structure in this restricted class. Removing base pairs is relatively simple since they will never result in an invalid structure. The general case involving four types of pseudoknots is rather involved, even with the restriction to structures with at most one pseudoknot, see Tab. 4. We therefore defer a complete treatment

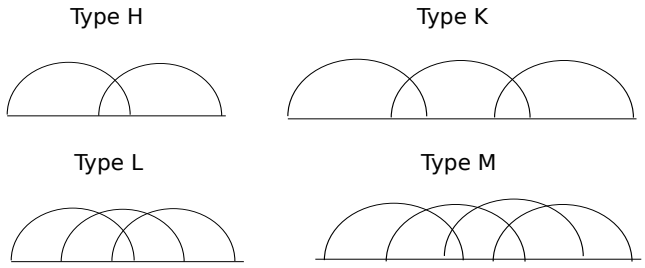

Fig. 23. Four types of pseudoknots considered in `gfold`.

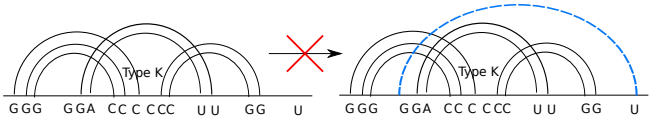

Fig. 24. An invalid move for an RNA structure with pseudoknot of type K. The blue GC pair can not be added since the resulted structure is not a γ_1 structure. Therefore it is invalid in the structure ensemble in `gfold`.

Table 1. Possible transitions between types of pseudoknots upon adding (+) or removing (-) a single base pair. S refers to structures without pseudoknots. 1 and 0 indicates whether a transition is possible or not.

+	S	H	K	L	M
S	1	1	1	0	0
H	0	1	1	1	0
K	0	0	1	0	1
L	0	0	0	1	1
M	0	0	0	0	1

-	S	H	K	L	M
S	1	0	0	0	0
H	1	1	0	0	0
K	1	1	1	0	0
L	0	1	0	1	0
M	0	0	1	1	1

of the general cases to future work, and restrict ourselves here to structures with a single H-type pseudoknot as a proof of concept.

As an example, we investigate the 27 nt pseudoknot PK1 of the upstream pseudoknot domain of the 3'-UTR of tobacco mild green mosaic virus, `pseudobase` ID *PKB92* (Leathers *et al.*, 1993). Its ground state structure

`.(((((.[[[[[)))))....]]]]].`

is correctly predicted by `gfold` with an energy of -4.3 kcal/mol. The competing pseudoknot-free minimum free energy secondary is

`.(((((......)))))..........`

with an energy of -3.9 kcal/mol as predicted by `RNAfold`, see Fig. 26.

`BHGbuilder` requires a path-searching algorithm to connect the LM obtained by the modified version of `RNAlocmin`, but is otherwise independent of the specification of the search space. We therefore extend `findpath` (Flamm *et al.*, 2000) to accommodate the class of pseudoknots under consideration by incorporating the expanded move set and energy function. Otherwise the algorithm remains unchanged.

We limited the sample size for the modified `RNAlocmin` program to $N = 10,000$ structures and obtained 69 LMs within the energy interval $[-4.3, 11.5]$ kcal/mol. Among these 12 structures contain a pseudoknot. Fig. 27 shows the low energy part of the

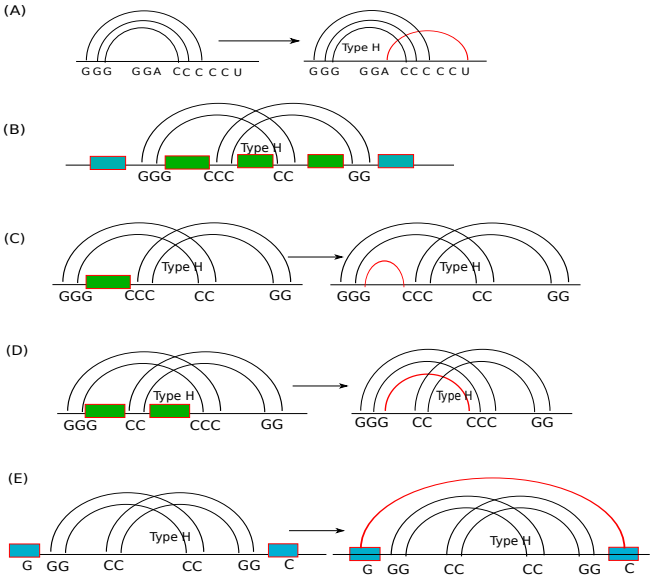

Fig. 25. The insertion of base pairs to derive a valid pseudoknot structure: (A) Adding a base pair crossing a stack results in an H-type pseudoknot. (B) An H-type pseudoknot naturally divides the RNA into five regions: two external regions (blue) and three internal regions (green); There are three basic ways to add base pairs: (C) add a base pair which involves nucleotides exactly in one green region without crossing with other existing base pairs; (D) add a base pair which involves two green regions to thinner the existing stacks, and (E) add a base pair which involves two blue regions without crossing other existing base pairs.

resulting BHG. We find that *PKB92* is more likely to fold into its most stable secondary structure first and only then refolds to form the pseudoknot. The optimal folding pathway is detailed in Tab. 2. The second, suboptimal pathway forming the second stem of the pseudoknot at first can be observed in the BHG as $L6 \rightarrow S7 \rightarrow L3 \rightarrow S4 \rightarrow L1$ in the Fig. 27.

Table 2. Optimal folding pathway of *PKB92* from the open structure to its MFE. Local minima and saddle points in the second column refer to Fig. 27. Structures and energies [kcal/mol] are given in the third and columns, respectively.

	Index	Structure	E
0	L6	`.....`	0.00
1	S5	`.....(......).....`	3.20
2		`....((......)).....`	1.30
3		`...(((......))).....`	-1.40
4		`..((((......)))).....`	-3.00
5	L2	`.(((((......))))).....`	-3.90
6	S6	`.(((((...[..)))))......]...`	3.80
7		`.(((((...[[.)))))......]]...`	1.20
8		`.(((((..[[[.)))))......]]]..`	-0.50
9		`.(((((.[[[[.)))))......]]]].`	-3.20
10	L1	`.(((((.[[[[[)))))....]]]]].`	-4.30

Fig. 26. The minimum free energy structure with pseudoknot predicted by `gfold` and without pseudoknot predicted by `RNAfold`. Secondary structure with and without pseudoknot drawings were produced with `PseudoViewer` (Han *et al.*, 2002) and `VARNA` (Darty *et al.*, 2009), respectively.

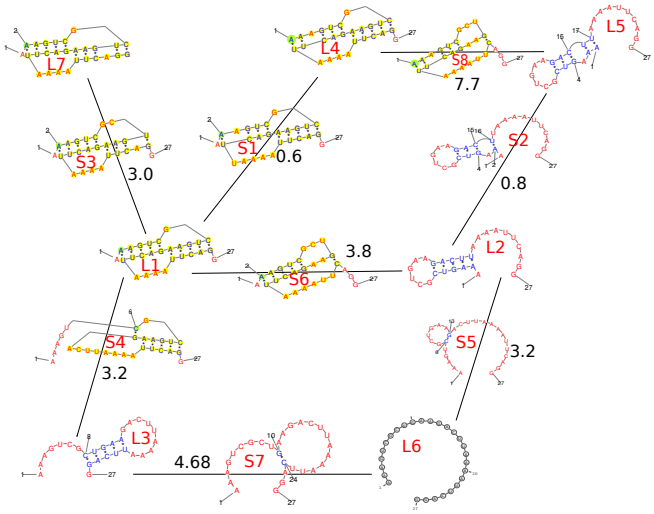

Fig. 27. Low energy part of the BHG for *PKB92*. Vertices are labeled for later use. In which, LMs and (direct) saddles are labeled by "Lx" and "Sx", respectively. Edges are labeled by their energy barriers [kcal/mol] and the corresponding saddle structures. Secondary structure with and without pseudoknot drawings were produced with `PseudoViewer` (Han *et al.*, 2002) and `VARNA` (Darty *et al.*, 2009), respectively.

REFERENCES

Bon, M., Vernizzi, G., Orland, H. & Zee, A. (2008) Topological classification of rna structures. *J. Mol. Biol.,* **379** (4), 900–911.

Condon, A., Davy, B., Rastegari, B., Zhao, S. & Tarrant, F. (2004) Classifying RNA pseudoknotted structures. *Theor. Comp. Sci.,* **320**, 35–50.

Darty, K., Denise, A. & Ponty, Y. (2009) VARNA: interactive drawing and editing of the RNA secondary structure. *Bioinformatics,* **25**, 1974–1975.

Flamm, C., Hofacker, I., Stadler, P. & Wolfinger, W. (2002) Barrier trees of degenerate landscapes. *Z. Phys. Chem.,* **216**, 155–173.

Flamm, C., Hofacker, I. L., Maurer-Stroh, S., Stadler, P. F. & Zehl, M. (2000) Design of multi-stable RNA molecules. *RNA,* **7**, 254–265.

Han, K., Lee, Y. & W., K. (2002) PseudoViewer: automatic visualization of RNA pseudoknots. *Bioinformatics,* **18** (Suppl 1), 321–328.

Isambert, H. & Siggia, E. D. (2000) Modeling RNA folding paths with pseudoknots: Application to hepatitis delta virus ribozyme. *Proc. Natl. Acad. Sci. USA,* **97**, 6515–6520.

Klemm, K., Qin, J. & Stadler, P. (2014) *Recent Advances in the Theory and Application of Fitness Landscapes* vol. 6,. Berlin: Springer-Verlag pp. 153–176.

Leathers, V., Tanguay, R., Kobayashi, M. & Gallie, D. R. (1993) A phylogenetically conserved sequence within viral 3' untranslated RNA pseudoknots regulates translation. *Mol Cell Biol,* **13**, 5331–5347.

Mathews, D., Sabina, J., Zuker, M. & Turner, D. H. (1999) Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J. Mol. Biol.,* **288**, 911–940.

Ponty, Y. (2008) Efficient sampling of RNA secondary structures from the Boltzmann ensemble of low-energy: The boustrophedon method. *J. Math. Biol.,* **56**, 107–127.

Reidys, C., Huang, F., Andersen, J., Penner, R., Stadler, P. & Nebel, M. (2011) Topology and prediction of rna pseudoknots. *Bioinformatics,* **27** (8), 1076–1085.

Sibson, R. (1973) SLINK: an optimally efficient algorithm for the single-link cluster method. *The Computer Journal (British Computer Society),* **16**, 30–34.

Van Nimwegen, E. & Crutchfield, J. P. (2000) Metastable evolutionary dynamics: crossing fitness barriers or escaping via neutral paths? *Bull. Math. Biol.,* **62**, 799–848.

Wolfinger, M. T., Svrcek-Seiler, W. A., Flamm, C., Hofacker, I. L. & Stadler, P. F. (2004) Exact folding dynamics of RNA secondary structures. *J. Phys. A: Math. Gen.,* **37**, 4731–4741.