Extending the Hardness of RNA Secondary Structure Comparison

Guillaume Blin[1], Guillaume Fertin[2], Irena Rusu[2], and Christine Sinoquet[2]

[1] IGM-LabInfo - UMR CNRS 8049 - Université de Marne-la-Vallée - France
gblin@univ-mlv.fr
[2] LINA - FRE CNRS 2729 - Université de Nantes - France
{fertin,rusu,sinoquet}@lina.univ-nantes.fr

Abstract. In molecular biology, RNA structure comparison is of great interest to help solving problems as different as phylogeny reconstruction, prediction of molecule folding and identification of a function common to a set of molecules. Lin *et al.* [6] proposed to define a similarity criterion between RNA structures using a concept of edit distance ; they named the corresponding problem Edit. Recently, Blin *et al.* [3] showed that another problem, the Longest Arc-Preserving Common Subsequence problem (or Lapcs), is in fact a subproblem of Edit. This relationship between those two problems induces the hardness of what was the last open case for the Edit problem, Edit(Nested,Nested), which corresponds to computing the edit distance between two secondary structures without pseudoknots. Nevertheless, Lapcs is a very restricted subproblem of Edit: in particular, it corresponds to a given system of editing costs, whose biological relevance can be discussed ; hence, giving a more precise categorization of the computational complexity of the Edit problem remains of interest. In this paper, we answer this question by showing that Edit(Nested,Nested) is **NP**-complete for a large class of instances, not overlapping with the ones used in the proof for Lapcs, and which represent more biologically relevant cost systems.

Keywords: computational biology, RNA structures, arc-annotated sequences, edit distance, NP-hardness.

1 Introduction

The understanding of biological mechanisms, at a molecular scale, is induced by the discovery and the study of various RNA functions. It is established that the conformation of an RNA molecule (a single strand composed of bases A, U, C and G also called primary structure) partially determines the function of the molecule. This conformation results from the molecule folding due to local pairing between complementary bases ($A-U$ and $C-G$, connected by a hydrogen bond). Thus, such a molecule has both double-stranded areas (stems) and various types of loops or areas with unpaired bases. A model underlying a given RNA conformation is the secondary structure, with its stems, bulges, and various loops.

RNA secondary structure comparison is essential for (*i*) identification of highly conserved structures during evolution (which cannot always be detected in the primary sequence, since it is often unpreserved) which suggest a significant common function for the studied RNA molecules, (*ii*) RNA classification of various species (phylogeny), (*iii*) RNA folding prediction by considering a set of already known secondary structures and (*iv*) identification of a consensus structure and consequently of a common role for molecules.

At a theoretical level, the RNA structure comparison problem can be modeled by the class of problems Edit(T_1,T_2) which consist in computing the minimum number of edit operations needed to transform a structure of type T_1 into a structure of type T_2, where T_1, T_2 take values in {plain, nested, crossing, unlimited}(cf. Section 2 for more details). Lin *et al.* [6] proposed to take simultaneously into account primary and secondary structures in RNA comparison by jointly considering a base and its potential hydrogen bond with another base in the similarity computation. They proposed in [6] exact and approximate polynomial algorithms for some classes of problems Edit(T_1,T_2). They also gave some complexity proofs for some other classes. The complexity of the Edit(Nested, Nested) problem was left as an open problem.

Recently, Blin *et al.* [3] showed that the complexity of this last problem is actually closed since it simply follows from the complexity of the Longest Arc-Preserving Common Subsequence problem [4] (Lapcs for short). However, a sharp analysis of the equivalence between the Lapcs and the Edit problems shows in fact that only a very restricted number of instances of Edit(Nested, Nested) are shown to be **NP**-complete. Moreover, the cost system should satisfy restrictions which can be biologically discussed. Therefore, as another step towards establishing the precise complexity landscape of the Edit problem, it is of interest to consider a more precise class of instances – but not overlapping with the one used in the proof from Lapcs –, for determining more precisely what makes the problem hard. For that purpose, we propose after defining some notations (Section 2) a non-trivial reduction via a 2-page book embedding with some special requirements on the costs for the edit operations (Section 3).

2 Notations and problem description

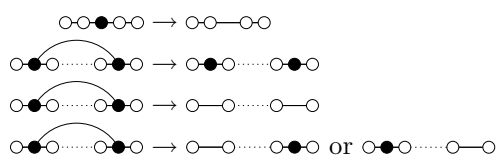
An RNA structure can be described by the sequence of its bases together with the set of hydrogen bonds possibly connecting bases A to bases U or bases C to bases G. This structure is commonly represented by an arc-annotated sequence. Given a finite alphabet Σ, an arc-annotated sequence is defined by a pair (S, P), where S is a string on Σ^* and P is a set of arcs connecting pairs of characters of S. In the following, we will refer to the characters as *bases* in reference to RNA structures. Any base with no arc incident to it is called *free*. As usually considered in arc-annotated sequences comparison, we distinguish four levels of arc structure [4]:

– Plain: there is no arc,

– Nested: no base is incident to more than one arc and no arcs are crossing,
– Crossing: no base is incident to more than one arc,
– Unlimited: no restriction at all.

Those four levels respect an obvious inclusion relation denoted by the $\subset$ operator: Plain $\subset$ Nested $\subset$ Crossing $\subset$ Unlimited. In order to compare two arc-annotated sequences, we consider the set of edit operations (and their associated costs) introduced in [6]. There are four substitution operations which induce renaming of bases in the arc-annotated sequence. They are listed together with their associated cost: *base-match* ($w_m : \Sigma^2 \to \mathbb{R}$), *base-mismatch* ($w_m : \Sigma^2 \to \mathbb{R}$), *arc-match* ($w_{am} : \Sigma^4 \to \mathbb{R}$), *arc-mismatch* ($w_{am} : \Sigma^4 \to \mathbb{R}$). Moreover, there are four deletion operations which induce deletion of bases and/or of arcs, which we list together with their associated cost:

base-deletion ($w_d : \Sigma \to \mathbb{R}$)
arc-breaking ($w_b : \Sigma^4 \to \mathbb{R}$)
arc-removing ($w_r : \Sigma^2 \to \mathbb{R}$)
arc-altering ($w_a : \Sigma^3 \to \mathbb{R}$)



In the following, given two arc-annotated sequences (S, P) and (T, Q), an *edit script* from (S, P) to (T, Q) will refer to a series of non-oriented edit operations transforming (S, P) into (T, Q). The *cost of an edit script* from (S, P) to (T, Q) is the sum of the costs of each operation involved in the edit script. We define the *edit distance* between (S, P) and (T, Q) as the minimum cost of an edit script from (S, P) to (T, Q). Finding this edit distance is called the Edit problem. To any edit script from (S, P) to (T, Q) corresponds an *alignment* of the characters of S and T such that (i) if a base is inserted or deleted in a sequence, it is aligned with a *gap* (indicated by $-$) and (ii) if a base of one sequence is (mis)matched with a base of the other sequence, there are aligned together. In the following, we will call *cost* of an alignment A, denoted by $cost(A)$, the cost of the edit script from which the alignment A is obtained. An *optimal alignment* A is an alignment of minimum cost, that is an alignment whose cost is equal to the edit distance.

Lin *et al.* proved in [6] that the problem Edit(Crossing, Plain) is *MAX-SNP hard*. Thus, any harder problem (in terms of restriction levels) is also *MAX-SNP hard*. Moreover, they gave a polynomial dynamic programming algorithm for the problem Edit(Nested, Plain), while Sankoff [7] had previously solved the problem Edit(Plain, Plain). The complexity of the Edit(Nested, Nested) problem was left as an open problem (see Table 1).

Recently, Blin *et al.* [3] showed that the complexity of this last problem was in fact closed, since it directly follows from the complexity of a different problem, called Longest Arc-Preserving Common Subsequence. As introduced by Evans in [4], the Lapcs problem is defined as follows: given two arc-annotated sequences (S, P) and (T, Q), find the longest – in terms of sequence length – common arc-annotated subsequence (R, U) of (S, P) and (T, Q) such that any arc in U corresponds to an existing arc both in P and in Q. In [3], Blin *et al.*

	UNLIMITED	CROSSING	NESTED	PLAIN
UNLIMITED	Max-SNP hard			
CROSSING		Max-SNP hard		
NESTED			NP-Complete $\star$	$O(nm^3)$ $\bullet$
PLAIN				$O(nm)$

Table 1. EDIT problem complexity (n and m are the number of bases of each sequence). $\bullet$ m is the size of the PLAIN sequence. $\star$ when $w_d = w_a = 2w_r$, the hardness follows from [5]; when $w_a > w_d$ (with some additional restrictions), our contribution.

proved that the LAPCS problem is a specific case of the EDIT problem provided that the cost system for edit operations is correctly chosen. The cost system is the following: $w_r = 2w_d = 2w_a$, and all substitutions operations as well as arc-breakings are prohibited (that is, they have arbitrary high costs). The main idea is to penalize the deletion operations proportionally to the number of bases that are deleted.

Since the LAPCS problem is **NP**-complete for arc-annotated sequences of NESTED types, so does the last open case for the EDIT problem, EDIT(NESTED, NESTED). Nevertheless, LAPCS is a very specific subproblem of EDIT: it corresponds to instances of EDIT for which $w_r = 2w_d = 2w_a$. **In particular, this means that the cost for deleting an unpaired base or a base linked to an hydrogen bound is similar. This is not realistic.** Indeed, this model would be more realistic if we had $w_a > w_d$, as breaking an hydrogen bond requires energy. More generally, considering a larger class of instances (not overlapping with the one used in the proof from LAPCS), would help us determine more precisely what makes the problem hard. Hence, we suggest a more general categorization of EDIT problem complexity by defining a non-trivial reduction which provides a larger and non-overlapping class of instances leading to the hardness.

3 Hardness of RNA secondary structure comparison

As mentioned before, the main contribution of this paper is the proof of the hardness of the RNA secondary structure comparison for a large class of instances not considered previously. The proof relying on the **NP**-completeness of LAPCS requires that $w_r = 2w_a = 2w_d$. In this article, we investigate a more precise and non-intersecting class of instances. More precisely, we will prove that the problem is also **NP**-complete when the cost system respects the following requirements:

$$w_a > w_b > w_d > 0 \tag{1}$$

$$w_r > w_a + w_d \tag{2}$$

$$w_b + \frac{w_d}{3} > w_a \tag{3}$$

$$w_m > 2w_r \tag{4}$$

The hardness result thus holds no matter how the costs are chosen so as to satisfy the constraints given above. The decision problem is defined formally as follows.

INPUT*: Two arc-annotated sequences* (S, P) *and* (T, Q) *of* NESTED *type, a set of costs for the edit operations and an integer* ℓ*.*
QUESTION*: Is there an alignment of the two sequences* (S, P) *and* (T, Q) *whose cost is less than or equal to* ℓ*?*

We initially notice that this problem is in **NP** since given an alignment we can check polynomially if its cost is less than or equal to ℓ. In order to prove that it is **NP**-complete, we propose a polynomial reduction from the **NP**-complete problem MIS-3P [2].

MIS-3P
INPUT*: A cubic planar bridgeless connected graph* $G = (V, E)$ *and an integer* k*.*
QUESTION*: Is there an independent set of cardinality greater than or equal to* k *in* G *?*

A graph $G = (V, E)$ is said to be a *cubic planar bridgeless connected* graph if any vertex of V is of degree three (cubic), G can be drawn in the plane in such a way that no two edges of E cross (planar), and there are at least two edge-disjoint paths connecting any pair of vertices of V (bridgeless connected). The idea of the proof is to encode any cubic planar bridgeless connected graph by two arc-annotated sequences. The construction first uses the notion of 2-page book embedding: a *2-page book embedding* of a graph G is a linear ordering of the vertices of G along a line and an assignment of the edges of G to the two half-planes delimited by the line – called the *pages* – such that no two edges assigned to the same page cross. For convenience, we will refer to the page above (resp. below) the line as the *top-page* (resp. *bottom-page*). In the following, a *2-page s-embedding* will denote a 2-page book embedding with the additional property that in each page, every vertex has degree at least one.

Theorem 1 (Bernhart *et al.* [1]). *Given any cubic planar bridgeless connected graph* G*, it is possible to find, in polynomial time, a 2-page s-embedding of* G*.*

Given a 2-page s-embedding of a cubic planar bridgeless connected graph $G = (V, E)$, we construct two arc-annotated sequences of NESTED type (S, P) and (T, Q). The underlying raw sequences S and T are defined as follows:

$$S = S_c\ S_1\ S_c\ S_2\ \ldots\ S_c\ S_n\ S_c$$
$$T = T_c\ T_1\ T_c\ T_2\ \ldots\ T_c\ T_n\ T_c$$

where (i) $n = |V|$, (ii) for each $1 \leq i \leq n$, S_i (resp. T_i) is a segment UAUAGG if the degree of the vertex $v_i \in V$ in the top-page (resp. bottom-page) is equal to two, a segment GGUAUA otherwise, and (iii) S_c and T_c are segments made of a given number q of bases C, where $q > \frac{3nw_r}{w_d}$ (the value of q will be justified in the proof of Lemma 2).

Now that the sequences S and T are defined, we have to copy the arc configuration of the top-page (resp. bottom-page) on S (resp. T). Each edge (v_i, v_j), $i < j$, of the top-page is represented by two arcs in P. More precisely, one arc a_1 links a base U of S_i and a base A of S_j. The second arc a_2 is nested in the first one : it links the base A directly to the right of the base U of a_1 to the base U directly to the left of the base A of a_1. We proceed in a similar way for the bottom-page by adding, for each edge in that page, two arcs in Q. Moreover, we impose that when a vertex v_i is of degree one on the top-page (resp. bottom-page), the two corresponding arcs in P (resp. Q) are incident to the rightmost bases A and U of the segment S_i (resp. T_i). It is easy to check that it is always possible to reproduce on (S, P) and (T, Q) the non-crossing edge configuration of each page. An example of such a construction is given in Figure 1. The size of the sequences is clearly polynomial in n: the length of S and T is $6n + (n + 1)q$ and the total number of arcs is $3n$. In the following, we will refer to any such construction as an *UA-construction*.

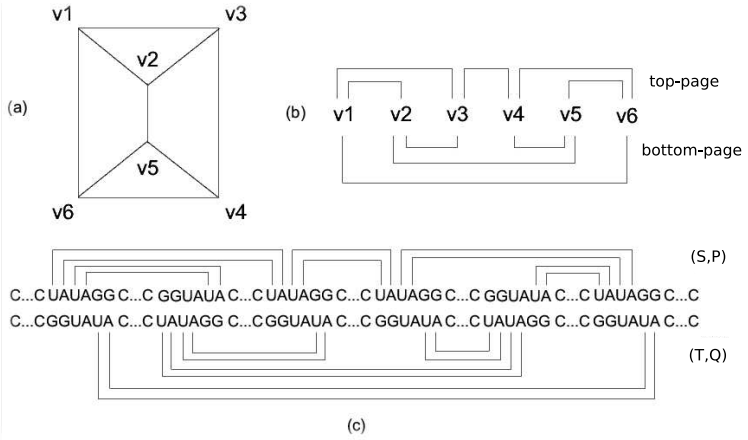


Fig. 1. Example of an UA-construction. Graph (a) is a cubic planar bridgeless connected graph having 6 vertices. Graph (b) is a 2-page s-embedding of graph (a). (c) The two arc-annotated sequences of NESTED type obtained from graph (a) by an UA-construction.

In order to complete the instance of the EDIT(NESTED,NESTED) problem, we define formally the parameter $\ell = 3n(w_b + \frac{4w_d}{3}) - p(6w_b + 2w_d - 6w_a)$ (p will be formally defined later on). We consider, further, that every instance of the EDIT(NESTED,NESTED) problem that we construct respects the cost system defined by equations (1) to (4).

We start the proof that the reduction from MIS-3P to EDIT(NESTED, NESTED) is correct by giving some properties (Lemmas 1 to 5) about optimal alignments of the sequences (S, P) and (T, Q). Then, these results will be used in Lemma 6 to conclude. We consider in all these lemmas that the conditions imposed by the equations (1) to (4) are verified.

Lemma 1. *In any optimal alignment of* (S, P) *and* (T, Q)*, there is no base substitution.*

Proof. Note that base substitution is an operation on bases which occurs either independently (no arc operation is involved) or following an arc-breaking/arc-altering. As the cost of non-pairing base alignment is included in the cost of an arc-mismatch, a base involved in a base substitution cannot be incident to an arc inducing an arc-mismatch. The principle of this proof is to show that, under the conditions imposed by equations (1) to (4), starting from an alignment A containing a base substitution, we can build an alignment A' which does not contain this substitution satisfying $cost(A') < cost(A)$. Base substitution can occur in three different configurations :

- substitution between two bases non incident to an arc : then A' is obtained from A by changing the base substitution into a base insertion and a base deletion. Thus, we have $cost(A') - cost(A) = 2w_d - w_m$.
- substitution between a base non incident to an arc and a base incident to an arc a. There are two subcases : a induces an (i) arc-breaking or (ii) an arc-altering in A. Let A' be an alignment obtained from A by aligning each base concerned by the substitution with a gap. Then any arc-breaking (resp. arc-altering) is transformed into an arc-altering (resp. arc-removing). Therefore, in case (i) we have $cost(A') - cost(A) = w_a + w_d - (w_b + w_m)$, while in case (ii) we have $cost(A') - cost(A) = w_r + w_d - (w_a + w_m)$.
- substitution between a base incident to an arc a_1 and a base incident to an arc a_2. There are three subcases : a_1 and a_2 induce (i) two arc-breaking, (ii) two arc-altering, (iii) an arc-altering and an arc-breaking in A. Let A' be an alignment obtained from A by aligning each base concerned by the substitution with a gap. In case (i), we have $cost(A') - cost(A) = 2w_a - (2w_b + w_m)$. In case (ii) we have $cost(A') - cost(A) = 2w_r - (2w_a + w_m)$. Finally, in case (iii) we have $cost(A') - cost(A) = w_r + w_a - (w_b + w_m + w_a) = w_r - (w_b + w_m)$.

Since $w_m > 2w_r$ and $w_r > w_a > w_b > w_d$ (see equations (1), (2) and (4)), we deduce that $cost(A') - cost(A) < 0$ in every case. Thus, for any given alignment A with at least one substitution, it is possible to find an alignment A' without this substitution such that $cost(A') < cost(A)$. This proves the lemma. □

Definition 1. *A canonical alignment of two sequences* (S, P) *and* (T, Q) *obtained from an UA-construction is an alignment where, for each* $1 \leq i \leq n + 1$*, the* i^{th} *segment* S_c *in* (S, P) *is aligned base to base to the* i^{th} *segment* T_c *in* (T, Q)*.*

Note that by construction no arc-match or arc-mismatch can be present in a canonical alignment of (S, P) and (T, Q).

Lemma 2. *Any optimal alignment of* (S, P) *and* (T, Q) *is canonical.*

Proof. Let A be a non canonical alignment. We will show that this is not an optimal alignment. By Lemma 1, we assume that A does not contain any substitution. In that case, non canonicity can arise for two reasons:

`Case 1.` There exists a *crossing alignment Up-Down* or *Down-Up* in the alignment A. We denote by crossing alignment Up-Down (resp. Down-Up), an alignment where at least one base of S_k (resp. T_k) is aligned with a base of T_m (resp. S_m) or with a gap situated between two bases of T_m (resp. S_m) and such that $k < m$.

Let A' be a canonical alignment without substitution. According to the conditions imposed by equations (1) to (4), the cost associated with any operation on a base not incident to an arc can be upper bounded by $\frac{w_r}{2}$ (since $w_r > w_a + w_d > 2w_d$) and the cost associated with an operation on any base incident to an arc can be upper bounded by $\frac{w_r}{2}$ as well (by equitably distributing the cost of the arc on its two incident bases : the cost w_b of an arc can be seen as composed of the cost $\frac{w_b}{2}$ on each of its incident bases ; since $w_b < w_r$, then $\frac{w_b}{2} < \frac{w_r}{2}$). Since any vertex of G is represented by two segments (one in (S, P) and one in (T, Q)) containing six bases each, the cost of the alignment of S_i and T_i for any vertex v_i is strictly less than $6w_r$, thus $cost(A') < 6nw_r$.

Let A be a crossing non canonical alignment, and let us suppose first that this crossing is Up-Down. In such an alignment the crossing imposes that at least q ($q = |S_c|$) bases C of (T, Q) on the left of T_m must be inserted and that at least as many bases C of (S, P) on the right of S_k must be deleted. Thus we have $cost(A) \geq 2qw_d$. Therefore $cost(A') < 6nw_r < 2qw_d \leq cost(A)$ since $q > \frac{3nw_r}{w_d}$. The alignment A is thus non optimal in case 1. In the case where the crossing is Down-Up, the proof is similar and the same result follows.

`Case 2.` There is no crossing alignment in A. In this case, for any k, any base of S_k is aligned either with a base of T_k or with a gap situated between two bases of T_k. Let us denote by ξ the sum of the alignment costs of the segments S_k and T_k for $k = 1, \ldots, n$ representing the n vertices of G. Thus, we have $cost(A) = \xi + R$ where R is the total cost of base to base alignments of segments S_c. The initial assumption (*i.e.* A is a non canonical alignment) imposes that at least one base C is deleted and one base C is inserted in A. Thus, we have $R \geq 2w_d$. Consequently, $cost(A) \geq \xi + 2w_d$. Now, let A' be a canonical alignment in which, for any k, any base of S_k is aligned exactly as in A, *i.e.* with the same base of T_k or with a gap. We have $cost(A') = \xi < cost(A)$, therefore A is not optimal. □

By Lemma 2, the cost of an optimal alignment depends on the local alignments of the segments S_k and T_k representing the vertices of G. By Lemma 1, a case analysis leads to a set of exactly eighteen types of local alignments, as illustrated in Figure 2. It is easy to see that any other alignment of the segments representing a vertex is equivalent, in terms of cost, to one of the above mentioned eighteen types.

Definition 2. *We call symmetric of a type of alignment t_i, denoted by t_{iSym}, the type of alignment obtained from t_i by inverting the two segments (i.e. such that the segment on (S, P) is now on (T, Q) and vice versa).*

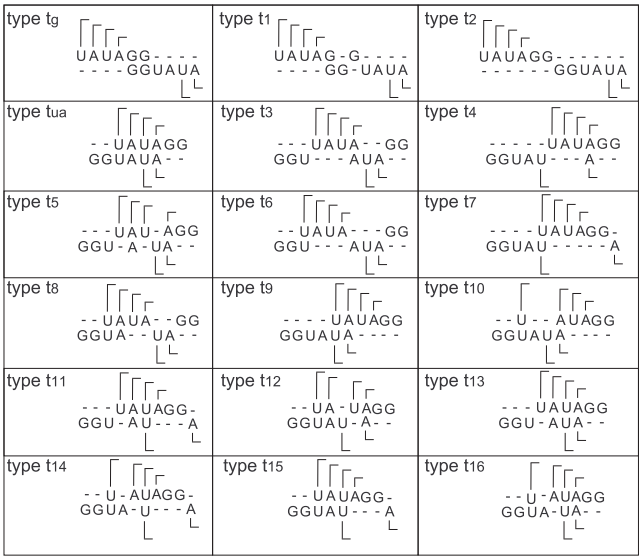


Fig. 2. The eighteen types of local alignments for the segments S_k and T_k.

Lemma 3. *An optimal alignment A' of (S, P) and (T, Q) contains only local alignments of types t_g, t_{ua}, t_{gSym} and t_{uaSym}.*

Proof. First, we notice that by definition of the operations on bases and arcs, two symmetric local alignments have the same cost. Thus, to prove Lemma 3, we will only show that any canonical alignment containing a local alignment of S_i, T_i of type t_1 or t_2 (resp. t_3, t_4, ...or t_{16}) has a cost higher than the cost of the same alignment where this local alignment is of type t_g (resp. t_{ua}). The similar conclusion for symmetric local alignments will then follow.

Let A and A' be two canonical alignments that differ only on the local alignment of S_i and T_i for a given $1 \leq i \leq n$. More precisely, let this local alignment be of type t_{ua} or t_g in A' and of any different type in A. The cost difference between A and A' can only be due to the local alignment of S_i and T_i. Let us notice that this difference is due locally to the alignment of a subset of bases of S_i and T_i. The alignments of bases of S_i and T_i common to A and A' will thus not be taken into account in the computation of the cost difference between A and A'. Moreover, if a change affects a base incident to an arc (say an arc between a base of S_i and a base of S_j), it is necessary to consider not only the base affected (say the base of S_i), but both bases incident to this arc.

The principle of the following proof is to show that from the conditions imposed by equations (1) to (4) and for the alignments A, A' defined below, we always have $cost(A') - cost(A) < 0$, meaning that the alignment A is not optimal.

`Case 1.` Let A be a canonical alignment containing a local alignment of S_i and T_i of type t_j for some $3 \leq j \leq 16$. Let A' be the alignment obtained from A by replacing the local alignment of S_i and T_i by a local alignment of type t_{ua}. Let

us denote by k_1 (resp. k_2) the number of bases in S_i and T_i in A which induce an arc-removing (resp. arc-altering) and which do not induce this arc-removing (resp. arc-altering) but an arc-altering (resp. arc-breaking) in A'.
Let us denote by k_0 the number of bases deleted or inserted in S_i and T_i in A and which are not deleted or inserted anymore in A'.

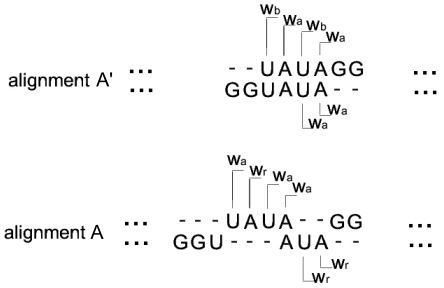


Fig. 3. Example of a canonical alignment A containing a local alignment of type t_3 and its corresponding alignment A' where a local alignment of type t_{ua} substitutes for the former local alignment. Here, $k_0 = 1, k_1 = 3, k_2 = 2$.

We obtain $k_0 + k_1 + k_2 > 0$, $k_0, k_1, k_2 \geq 0$ and $cost(A') - cost(A) = k_1(w_a - w_r) + k_2(w_b - w_a) + k_0(-w_d)$. According to the conditions imposed by equations (1) and (2), $0 < w_d$ and $w_b < w_a < w_r$, we deduce that $cost(A') - cost(A) < 0$.

`Case 2.` Let A be an alignment containing a local alignment of S_i and T_i of type t_j for $j \in \{1, 2\}$. Let A' be the alignment obtained from A by replacing the local alignment of S_i and T_i by a local alignment of type t_g. Let us denote by k_3 the number of bases deleted or inserted in S_i and T_i in A and which are not deleted or inserted any more in A'. We obtain $k_3 > 0$ and $cost(A') - cost(A) = k_3(-w_d)$. According to the conditions imposed by equation (1), $0 < w_d$ therefore we deduce that $cost(A') - cost(A) < 0$. Hence, any canonical alignment containing a local alignment of S_i and T_i of type t_1 or t_2 (resp. t_3, t_4, ... or t_{16}) has a cost strictly greater than the cost of the same alignment where this local alignment is of type t_g (resp. t_{ua}). More generally, since symmetric types have the same cost, we conclude that an optimal alignment of (S, P) and (T, Q) is canonical (from Lemma 2) and contains only local alignments of types t_g, t_{ua}, t_{gSym} and t_{uaSym}. □

Lemma 4. *In any optimal alignment, no two segments S_i and S_j (resp. T_i and T_j) having local alignments of type t_g or t_{gSym} can be connected by an arc.*

Proof. Let A be an alignment containing an arc connecting a base of S_i to a base of S_j, and whose local alignments are both of type t_g or t_{gSym}. Let A' be an alignment obtained from A where one of these segments, say S_i, has a local alignment of type t_{ua} or respectively t_{uaSym}. We will show that $cost(A') - cost(A) < 0$. Let k_1 (resp. k_2) denote the number of bases of S_i in A which induce an arc-removing (resp. arc-altering) and which in A' do not induce this arc-removing (resp. arc-altering) but an arc-altering (resp. arc-breaking). Thus, we have $cost(A') - cost(A) = 4w_d - 2w_d + k_1(w_b - w_a) + k_2(w_a - w_r) = 2w_d +$

$2w_a - 2w_r + k_1(w_b - w_a) + (k_2 - 2)(w_a - w_r)$ where $k_1 + k_2 = 6$, $k_1 \geq 0$ and $k_2 \geq 2$. According to the conditions imposed by equations (1) and (2), $0 < w_d$ and $w_b < w_a < w_r$, therefore we have $cost(A') - cost(A) < 0$. The proof is similar considering T_i and T_j. The lemma is thus proved. □

Definition 3. *A canonical alignment A' of (S, P) and (T, Q) containing only local alignments of types t_g, t_{ua}, t_{gSym} and t_{uaSym} and in which no arc connects two segments whose local alignments are of type t_g or t_{gSym} (i.e. respecting the conditions of Lemmas 3 and 4), is called t_g-stable.*

Lemma 5. *The cost of a t_g-stable canonical alignment A' is $cost(A') = 3n(w_b + \frac{4w_d}{3}) - p(6w_b + 2w_d - 6w_a)$ where p is the number of segments whose alignment is of type t_g or t_{gSym}.*

Proof. As mentioned previously, on the whole (S, P) and (T, Q) contains $3n$ arcs. If p is the number of local alignments of type t_g or t_{gSym} in A', then there exists $6p$ arcs connecting a base belonging to a local alignment of type t_g or t_{gSym} to a base belonging to a local alignment of type t_{ua} or t_{uaSym}, and thus $3n - 6p$ arcs between pairs of bases belonging to local alignments of type t_{ua} or t_{uaSym}. We compute the cost of any arc joining two local alignments of types t_{ua} and t_{ua} (resp. t_{ua} and t_g) or symmetric by adding to w_b (resp. w_a) a supplementary cost computed for each incident base and resulting from the equitable distribution of costs w_d between the six arcs involved in each concerned local alignment. These costs w_d deal with the free bases, inside each local alignment.

The cost of an arc between two local alignments of type t_{ua} or t_{uaSym} is computed as follows : $4w_d$ must be distributed on the six arcs involved. Thus for each base incident to such an arc, a supplementary cost of $\frac{4w_d}{6} = \frac{2w_d}{3}$ must be taken into account. The cost of any arc involved in a t_{ua}-t_{ua} junction (or symmetric) is then $w_b + \frac{2w_d}{3} + \frac{2w_d}{3} = w_b + \frac{4w_d}{3}$. For a local alignment of type t_g (or symmetric), we must distribute $2w_d$ on the six arcs involved, which leads to a supplementary cost of $\frac{2w_d}{6} = \frac{w_d}{3}$ for any base incident to such an arc. Thus the cost of any arc involved in a t_{ua}-t_g junction (or symmetric) is $w_a + \frac{w_d}{3} + \frac{2w_d}{3} = w_a + w_d$. We obtain $cost(A') = (3n - 6p)(w_b + \frac{4w_d}{3}) + 6p(w_d + w_a) = 3n(w_b + \frac{4w_d}{3}) - p(6w_b + \frac{24w_d}{3} - 6w_d - 6w_a) = 3n(w_b + \frac{4w_d}{3}) - p(6w_b + 2w_d - 6w_a)$. □

Lemmas 1 to 5 provide us with all the necessary intermediate results to show that the reduction from MIS-3P to EDIT(NESTED, NESTED) is valid.

Lemma 6. *A cubic planar bridgeless connected graph G has an independent set V' such that $|V'| \geq k$ if and only if the edit distance between the sequences (S, P), (T, Q) obtained from G by an UA-construction is at most $\ell = 3n(w_b + \frac{4w_d}{3}) - k(6w_b + 2w_d - 6w_a)$.*

Proof. ($\Rightarrow$) Let $V' \subseteq V$ be an independent set of G such that $|V'| \geq k$. Let A be the canonical alignment of (S, P) and (T, Q) such that (i) $\forall v_i \in V'$, the local alignment of S_i and T_i is of type t_g or t_{gSym} and (ii) $\forall v_j \in V - V'$, the local alignment of S_j and T_j is of type t_{ua} or t_{uaSym}. Thus, by definition, the alignment

is t_g-*stable*. By Lemma 5, $cost(A) = 3n(w_b + \frac{4w_d}{3}) - |V'|(6w_b + 2w_d - 6w_a)$. Since $|V'| \geq k$ by hypothesis, we have $cost(A) \leq 3n(w_b + \frac{4w_d}{3}) - k(6w_b + 2w_d - 6w_a) = \ell$.

($\Leftarrow$) Suppose there exists an edit script between the sequences (S, P), (T, Q), for which the corresponding alignment A' satisfies $cost(A') \leq \ell$. Now let A_{OPT} be an optimal alignment of (S, P) and (T, Q). Let V' be the set of vertices v of G for which, in A_{OPT}, local alignments of the corresponding segments are of type t_g or t_{gSym}. Since we know by Lemma 4 that no arc connects segments of type t_g or t_{gSym} in A_{OPT}, we conclude that V' is an independent set of G. Moreover, by Lemma 5, we have $cost(A_{OPT}) = 3n(w_b + \frac{4w_d}{3}) - |V'|(6w_b + 2w_d - 6w_a)$ and since $cost(A_{OPT}) \leq cost(A') \leq \ell$ with $\ell = 3n(w_b + \frac{4w_d}{3}) - k(6w_b + 2w_d - 6w_a)$, we conclude that $k \leq |V'|$. Lemma 6 is proved. □

4 Conclusion

In this paper, we have proved that the problem EDIT(NESTED,NESTED) defined in [6] is **NP**-complete. This is done using a non trivial reduction from MIS-3P, via a 2-page s-embedding. Though the **NP**-completeness of the problem was already known due to the fact that the LAPCS problem for nested structures was proved to be **NP**-complete [5, 3], we have extended this result to a larger and non-intersecting class of instances, for which the set of cost is biologically more relevant. Though the result we give in this paper is in some sense negative, we point out that EDIT(NESTED,NESTED) has a polynomial approximation algorithm of ratio $\beta = \max\{\frac{2w_a}{w_b+w_r}, \frac{w_b+w_r}{2w_a}\}$ [6]. However, this approximation ratio depends on the respective values of the parameters w_a, w_b and w_r. An interesting question would be to know whether there exists a polynomial algorithm for EDIT(NESTED,NESTED) with *constant* approximation ratio.

References

1. F. Bernhart and P.C. Kainen. The book thickness of a graph. *Journal of Combinatorial Theory, Series B*, 27(3):320–331, 1979.
2. T. Biedl, G. Kant, and M. Kaufmann. On triangulating planar graphs under the four-connectivity constraint. *Algorithmica*, 19:427–446, 1997.
3. G. Blin and H. Touzet. How to compare arc-annotated sequences : the alignment hierarchy. In *13th Symposium on String Processing and Information Retrieval (SPIRE'06)*, volume 4209 of *LNCS*, pages 291–303, 2006.
4. P.A. Evans. *Algorithms and Complexity for Annotated Sequence Analysis*. PhD thesis, University of Victoria, 1999.
5. G.H. Lin, Z.Z. Chen, T. Jiang, and J. Wen. The longest common subsequence problem for sequences with nested arc annotations. In *Proc. of the 28th Int. Coll. on Automata, Languages and Programming (ICALP'01)*, volume 2076 of *LNCS*, pages 444–455, 2001.
6. G.H. Lin, B. Ma, and K. Zhang. Edit distance between two RNA structures. In *Proceedings of the 5th International Conference on Computational Biology (RECOMB'01)*, pages 211–220. ACM Press, 2001.
7. D. Sankoff and B. Kruskal. *Time Warps, String Edits and Macromolecules: the Theory and Practice of Sequence Comparison*. Addison-Wesley, 1983.