Polynomial algorithms for protein similarity search for restricted mRNA structures

Frank Gurski*

February 1, 2008

Abstract

In this paper we consider the problem of computing an mRNA sequence of maximal similarity for a given mRNA of secondary structure constraints, introduced by Backofen et al. in [BNS02a] denoted as the MRSO problem. The problem is known to be NP-complete for planar associated implied structure graphs of vertex degree at most 3. In [BFHV05] a first polynomial dynamic programming algorithms for MRSO on implied structure graphs with maximum vertex degree 3 of bounded cut-width is shown. We give a simple but more general polynomial dynamic programming solution for the MRSO problem for associated implied structure graphs of bounded clique-width. Our result implies that MRSO is polynomial for graphs of bounded tree-width, co-graphs, $P_4$-sparse graphs, and distance hereditary graphs. Further we conclude that the problem of comparing two solutions for MRSO is hard for the class $P_{||NP}$, which is defined as the set of problems which can be solved in polynomial time with a number of parallel queries to an oracle in NP.

Keywords: graph algorithms, protein similarity search, mRNA structure, computational biology, $P_{||NP}$-completeness

1 Introduction

One of the main processes in biology is the transformation of DNA into proteins. This process is divided into two steps. The first step is the transcription, which copies the DNA into a certain RNA molecule called messenger RNA (mRNA). Each mRNA is a string of four types of nucleotides, i.e. elements of $\{A, C, G, U\}$. ($A, U$) and ($C, G$) are known as the complementary nucleotide pairs. Every string of three nucleotides is called a codon. The second step is the translation, which converts block wise a codon in the mRNA into an amino acid. Every protein is the result of a translation of some mRNA.

We can represent every mRNA as a graph by considering its nucleotides as vertices and possible edges (so called bonds) between vertices representing complementary nucleotides. The resulting graph is also denoted as (secondary) structure graph of the mRNA. If we consider the codons as vertices we obtain the associated implied structure graph.

In this paper consider the MRna Structure Optimization (MRSO) problem, introduced by Backofen et al. [BNS02a][BNS02b]. The problem is to compute an mRNA sequence of maximal similarity for a given mRNA that additionally satisfies some secondary structure

--

*Heinrich-Heine Universität Düsseldorf, Department of Computer Science, D-40225 Düsseldorf, Germany, E-Mail: gurski-corr@acs.uni-duesseldorf.de,
constraints. If the input structure graph of the problem has vertex degree at most one, we will denote the restriction of the problem by MRSO-d1. The MRSO-d1 problem (and thus also the MRSO problem) has shown to be NP-complete for planar implied structure graphs \cite{BFHV05}. In \cite{BNS02a}, a linear time algorithm for the MRSO-d1 problem has been shown for outer-planar implied structure graphs.

A very useful tool to solve hard problems on restricted inputs is parameterized complexity \cite{DF99}. The main idea is that only input graphs of a bounded graph parameter \( k \) are considered. The running time of the algorithms is exponential in \( k \), but for fixed \( k \) polynomial. Fixed parameter algorithms are frequently used in computational biology, see e.g. works of Bodlaender et al. \cite{BDF95,BDF+95}. In \cite{BFHV05} first fixed parameter algorithms for the MRSO-d1 problem for implied structure graphs of bounded cut-width are given. There are several known secondary (tertiary and quaternary) structures which are not simple, recursive, or do not correspond to outer-planar implied structure graphs \cite{Ak00}. Further it seems to be likely that amino acids of more complicated secondary structures will be discovered in the future \cite{BFHV05}. Therefore, we give a more general fixed parameter solution for MRSO on graphs of bounded clique-width which form a very large class of implied structure graphs.

The clique-width of a graph is defined by a composition mechanism for vertex-labeled graphs \cite{CO00}. The operations are the vertex disjoint union, the addition of edges between vertices controlled by a label pair, and the relabeling of vertices. The clique-width of a graph \( G \) is the minimum number of labels needed to define it. Each such composition leads a tree structure. Using this tree structure a lot of NP-complete graph problems can be solved by dynamic programming in polynomial time for graphs of bounded clique-width, see e.g. \cite{CMR00,EGW01,GW06,KR03}.

This paper is organized as follows. In Section 2, we recall the definition of clique-width and a general method how to solve graph problems on graph of bounded clique-width. In Section 3, we recall the definition of MRSO from Backofen et al. \cite{BNS02a,BNS02b}. In Section 4, we show a simple but very general polynomial time solution of the problem for implied structure graphs of bounded clique-width. Our result implies that MRSO is polynomial for graphs of bounded tree-width, co-graphs, \( P_4 \)-sparse graphs, and distance hereditary graphs and re-proofs the existence of polynomial time algorithms for MRSO-d1 of \cite{BFHV05} for graphs of bounded tree-width and graphs of bounded cut-width. In Section 5, we briefly conclude that the problem of comparing to solutions for MRSO-d1 is hard for the class \( P^{\|} \), which is defined as the set of problems which can be solved in polynomial time with a number of parallel queries to an oracle in NP.

\section{Clique-width and polynomial time algorithms}

Let \([k] := \{1, \ldots, k\}\) be the set of all integers between 1 and \( k \). We work with finite undirected labeled graphs \( G = (V_G, E_G, \text{lab}_G) \), where \( V_G \) is a finite set of vertices labeled by some mapping \( \text{lab}_G : V_G \to [k] \) and \( E_G \subseteq \{ \{u, v\} \mid u, v \in V_G, u \neq v \} \) is a finite set of edges. The labeled graph consisting of a single vertex labeled by \( a \in [k] \) is denoted by \( \bullet_a \). For the definition of special graph classes we refer to the survey of Brandstädt et al. \cite{BLS99}.

The notion of clique-width\footnote{This complexity measure was first considered by Courcelle, Engelfriet, and Rozenberg \cite{CER91,CER93}, the notion of clique-width was introduced by Courcelle and Olariu in \cite{CO00}.} for labeled graphs is defined by Courcelle and Olariu in \cite{CO00} as follows.
Definition 2.1 (Clique-width, [CO00]) Let \( k \) be some positive integer. The class \( CW_k \) of labeled graphs is recursively defined as follows.

1. The single vertex graph \( \bullet_a \) for some \( a \in [k] \) is in \( CW_k \).

2. Let \( G, J \in CW_k \) be two vertex disjoint labeled graphs, then \( G \oplus J := (V', E', \text{lab}') \) defined by \( V' := V_G \cup V_J \), \( E' := E_G \cup E_J \), and

\[
\text{lab}'(u) := \begin{cases} 
\text{lab}_G(u) & \text{if } u \in V_G \\
\text{lab}_J(u) & \text{if } u \in V_J
\end{cases}, \quad \forall u \in V'
\]

is in \( CW_k \).

3. Let \( a, b \in [k] \) be two distinct integers and \( G \in CW_k \) be a labeled graph, then

(a) \( \rho_{a \rightarrow b}(G) := (V_G, E_G, \text{lab}') \) defined by

\[
\text{lab}'(u) := \begin{cases} 
\text{lab}_G(u) & \text{if } \text{lab}_G(u) \neq a \\
b & \text{if } \text{lab}_G(u) = a
\end{cases}, \quad \forall u \in V_G
\]

is in \( CW_k \) and

(b) \( \eta_{a,b}(G) := (V_G, E', \text{lab}_G) \) defined by \( E' := E_G \cup \left\{ \{u,v\} \mid u,v \in V_G, u \neq v, \text{lab}(u) = a, \text{lab}(v) = b \right\} \) is in \( CW_k \).

The clique-width of a labeled graph \( G \) is the least integer \( k \) such that \( G \in CW_k \). The clique-width of an unlabeled graph \( G = (V_G, E_G) \) is the smallest integer \( k \), such that there is some mapping \( \text{lab}_G : V_G \to [k] \) such that the labeled graph \( (V_G, E_G, \text{lab}_G) \) has clique-width at most \( k \).

A class of graphs \( \mathcal{L} \) has bounded clique-width if there is some integer \( k \) such that any graph in \( \mathcal{L} \) has clique-width at most \( k \), i.e. there is some \( k \) such that \( \mathcal{L} \subseteq CW_k \). The minimal \( k \), if exists, is defined as clique-width of class \( \mathcal{L} \).

An expression built with the operations \( \bullet_a, \oplus, \rho_{a \rightarrow b}, \eta_{a,b} \) for integers \( a, b \in [k] \) is called a clique-width \( k \)-expression. The graph defined by expression \( X \) is denoted by \( \text{val}(X) \). The following two clique-width expressions \( X_1 \) and \( X_2 \) define the labeled graphs \( G_1 \) and \( G_2 \) in Fig. 1:

\[
X_1 = \eta_{1,2}(\rho_{2 \rightarrow 1}(\eta_{1,2}(\bullet_1 \oplus \bullet_2)) \oplus \bullet_2)
\]

\[
X_2 = \rho_{1 \rightarrow 2}(\eta_{2,3}(\eta_{1,2}(\bullet_1 \oplus \bullet_2) \oplus (\eta_{1,2}(\bullet_1 \oplus \bullet_2)) \oplus \bullet_3))
\]

If a graph \( G \) has clique-width at most \( k \) then the edge complement \( \overline{G} \) has clique-width at most \( 2k \) [CO00]. Distance hereditary graphs have clique-width at most 3 [GR00]. Co-graphs, i.e. \( P_4 \)-free graphs have clique-width at most 2 [CO00]. Further, many graph classes defined by a limited number of \( P_4 \) have bounded clique-width, e.g. \( P_4 \)-sparse graphs, \( P_4 \)-tidy, and \((q,t)\)-graphs [CMR00, MR99]. The clique-width of permutation graphs, interval graphs, grids and planar graphs is not bounded [GR00]. An arbitrary graph with \( n \) vertices has clique-width at most \( n - r \), if \( 2^r < n - r \). Every graph of tree-width at most \( k \) has clique-width at most \( 3 \cdot 2^{k-1} \) [CR05]. The recognition problem for graphs of clique-width at most \( k \) is still open for \( k \geq 4 \). Clique-width of at most 3 is decidable in polynomial time [CHL+00].
Clique-width of at most 2 is decidable in linear time [CPS85]. Both algorithms also give a clique-width expression if the input graph has clique-width at most 3 or clique-width at most 2, respectively. The clique-width of tree-width bounded graphs is also computable in linear time [EGW03]. Minimizing clique-width is NP-complete [FRRS06].

Courcelle et al. have shown in [CMR00] that all graph properties which are expressible in monadic second order logic with quantifications over vertices and vertex sets (MSO$_1$-logic) are decidable in linear time on clique-width bounded graphs. Furthermore, there are many NP-complete graph problems which are not expressible in extended MSO$_1$-logic like Hamiltonicity, chromatic number, partition problems, and bounded degree subgraph problems but which can also be solved in polynomial time on clique-width bounded graphs. The algorithms can be found in [EGW01, GW06, KR03]. The proofs are based on the following general dynamic programming scheme.

**Theorem 2.2** ([EGW01]) Let $\Pi$ be a graph problem and $k$ be a positive integer. If there is a mapping $F$ that maps each clique-width $k$-expression $X$ onto some structure $F(X)$, such that for all clique-width $k$-expressions $X, Y$ and all $a, b \in [k]$

1. the size of $F(X)$ is polynomially bounded in the size of $X$,
2. the answer to $\Pi$ for $val(X)$ is computable in polynomial time from $F(X)$,
3. $F(\bullet_a)$, is computable in time $O(1)$,
4. $F(X \oplus Y)$ is computable in polynomial time from $F(X)$ and $F(Y)$, and
5. $F(\eta_{a,b}(X))$ and $F(\rho_{a\rightarrow b}(X))$, are computable in polynomial time from $F(X)$.

Then for every clique-width $k$-expression $X$, the answer to $\Pi$ for graph $val(X)$ is computable in polynomial time from expression $X$.

One of the main important questions is how to find clique-width expressions. For graphs of clique-width at most 3 an expression can be found in polynomial time, as stated above. For graphs of larger clique-width, approximations of rank-width [OS06, Oum05, Oum06] lead approximations of clique-width and a corresponding expression. The best known result is the following.

**Theorem 2.3** ([Oum06]) For every fixed integer $k$ there is a $O(|V_G|^3)$ algorithm that either outputs a clique-width $(8^k - 1)$-expression of an input graph $G$, or confirms that the clique-width of $G$ is larger than $k$. 
3 MRna Structure Optimization (MRSO)

In this section we recall the MRna Structure Optimization (MRSO) problem as introduced by Backofen et al. in [BNS02a].

A codon is a sequence of three nucleotides, i.e. a string of \{A,C,G,U\}³. UAA,UAG and UGA are called stop codons, the remaining codons represent 20 amino acids. An mRNA is a sequence of \(n\) consecutive codons \(S = s_1 \ldots s_{3n}\) over \{A,C,G,U\}, i.e. each codon of \(S\) is of the form \(s_{3i-2}s_{3i-1}s_{3i}\) for some \(1 \leq i \leq n\).

The MRna Structure Optimization (MRSO) problem is defined in [BNS02a] as follows. Let \(S = S_1 \ldots S_{3n}\) be the nucleotide sequence of an mRNA and let \(A = A_1 \ldots A_n\) be a given amino acid sequence. The problem is to find an approximative mRNA sequence \(N = N_1 \ldots N_{3n}\) with amino acid sequence \(A' = A'_1 \ldots A'_n\), such that \(N\) and \(S\) have the same secondary structure and \(A\) and \(A'\) are of maximum similarity. The similarity between amino acid sequences is measured by PAM matrices introduced by Dayhoff et al. [DSO78]. We will use \(n\) functions \(f_i, 1 \leq i \leq n\) measuring the similarity between \(A_i\) and \(A'_i\).

In order to define the MRSO as a general graph problem we use the following notions. Let \(\Sigma\) be a finite alphabet (in biological application \(\Sigma = \{A,C,G,U\}\) corresponds to the set of nucleotides) and \(\Gamma \subseteq \Sigma \times \Sigma\) be a set of complementary pairs over \(\Sigma\) (in biological application \(\Gamma = \{(C,G), (A,U)\}\) corresponds to the set of complementary nucleotide pairs). We denote the complement of some \(X \subseteq \Sigma\) by \(\overline{X}\). For some mRNA with nucleotide sequence \(S\), we define the structure graph of \(S\) by taking the nucleotides as vertices and edges between any two vertices representing complementary nucleotides.

That is, to solve the MRSO problem we have to compute an admissible labeling over \(\Sigma\) (i.e. a labeling that satisfies the complementary conditions) for the vertices of the given structure graph of highest possible value with respect to functions \(f_i, 1 \leq i \leq n\).

**Problem 3.1 (MRSO)**

**INSTANCE:** A structure graph \(G = (\{v_1, \ldots, v_{3n}\}, E_G)\), and \(n\) functions \(f_1, \ldots, f_n, f_i : \Sigma^3 \rightarrow \mathbb{Q}\) is associated with \(\{v_{3i-2}, v_{3i-1}, v_{3i}\}, 1 \leq i \leq n\).

**OUTPUT:** A function \(L : V_G \rightarrow \Sigma\), such that \(\{v_k, v_l\} \in E_G\) implies that \((L(v_k), L(v_l)) \in \Gamma\) and the cost

\[
MRSO(G, f_1, \ldots, f_n) := \sum_{i=1}^{n} f_i(L(v_{3i-2}), L(v_{3i-1}), L(v_{3i}))
\]

is maximized.

In several motivations from biology, the structure graph of problem MRSO has vertex degree at most one. Following the notions of [Bon04], we denote the corresponding problem by MRSO-\(d1\).

Since functions \(f_i, i = 1, \ldots, n\) correspond to \(n\) amino acids, we next describe the MRSO problem on the amino acid level instead of the given nucleotide level definition. For some structure graph \(G = (\{v_1, \ldots, v_{3n}\}, E_G)\) we define the implied structure graph \(G_{\text{impl}} = (V_{\text{impl}}, E_{\text{impl}})\) by \(V_{\text{impl}} = \{u_1, \ldots, u_n\}\) and \(E_{\text{impl}} = \{\{u_i, u_j\} | \exists r \in \{3i - 2, 3i - 1, 3i\} : \exists s \in \{3j - 2, 3j - 1, 3j\} : \{v_r, v_s\} \in E_G\}\). Fig. 2 shows an example for a structure graph and the corresponding implied structure graph.

Next we generalize the complementary conditions given by \(\Gamma\) for amino acids, i.e. strings of \(\Sigma^3\). Let \((l_{3i-2}l_{3i-1}l_{3i}, l_{3j-2}l_{3j-1}l_{3j}) \in \Sigma^3 \times \Sigma^3\) be a pair and \(v_{3i-2}, v_{3i-1}, v_{3i}, v_{3j-2}, v_{3j-1}, v_{3j}\) be
the six corresponding vertices of structure graph \( G \). We define pair \((l_{3i-2}l_{3i-1}l_{3i}, l_{3j-2}l_{3j-1}l_{3j})\) satisfies \( \Gamma \), if for every edge \( \{v_i, v_j\} \in E_G, 3i - 2 \leq i' \leq 3i, 3j - 2 \leq j' \leq 3j:\ (l_i, l_j) \in \Gamma \).

Obviously, every solution for the MRSO problem on a structure graph \( G \) can be transformed into a solution for the corresponding implied structure graph \( G_{\text{impl}} \), and vice versa. Further for every structure graph \( G \) which is an instance of problem MRSO-d1, every vertex in \( G_{\text{impl}} \) has at most 3 adjacent edges as shown in the example of Fig 2.

The following results for the MRSO-d1 problem have been shown. Problem MRSO-d1 is known to be NP-complete for implied structure graphs with page number at most 2, see [BFHV05], and thus for planar implied structure graphs, further in [Bon04] it is shown that MRSO-d1 generalizes the Maximum independent set problem for graphs of vertex degree at most 3, which is also known to be NP-complete [GJ79]. Even the decision problem, where an input graph \( G \) and \( n \) functions \( f_1, \ldots, f_n \) are accepted if some assignment of the vertices reach costs of \( c \) is NP-complete for implied structure graphs of vertex degree at most 3 [BNS02a].

If the implied structure graph is outer-planar, MRSO-d1 is solvable in linear time [BNS02a]. Further in [BFHV05] polynomial fixed parameter algorithms for MRSO-d1 for implied structure graphs of a bounded number of edge crossings, implied structure graphs of a bounded number of degree 3 vertices, and implied structure graphs of a bounded cut-width (which also implies a polynomial solution for MRSO-d1 on tree-width bounded graphs) are given.

Since there also exist mRNA structures with bonds between more than two nucleotides [Aku00] and amino acids of more complicated secondary structures [BFHV05], we next give a more general solution for problem MRSO for implied structure graphs of bounded clique-width.

## 4 MRSO on implied structure graphs of bounded clique-width

We next will use the scheme of Theorem 2.2 to obtain a polynomial time solution for the MRSO problem for associated implied structure graphs of bounded clique-width.

**Theorem 4.1** For every positive integer \( k \), problem MRSO can be solved in polynomial time for every structure graph that defines an implied structure graph which is given by some clique-width \( k \)-expression.

**Proof** Let \( G \) be a structure graph for the implied structure graph \( G_{\text{impl}} = (\{u_1, \ldots, u_n\}, E_G, \text{lab}_G) \), which is defined by some clique-width \( k \)-expression \( X \). For every admissible labeling \( \text{lab}^X : V_G \rightarrow \Sigma^3 \) of \( \text{val}(X) \) we define a pair \((L, f)\), where \( L = \{\text{lab}_{\text{val}(X)}(u), \text{lab}^X(u) \mid u \in V_G\} \)
V_{val(X)} \subseteq [k] \times \Sigma^3 and \( f = \sum_{u_i \in val(X)} f_i(lab(u_i)) \). Let \( F(X) \) be the set of all mutually different pairs \((L, f)\) for all admissible labelings of the vertices of graph \( val(X) \) with labels of \( \Sigma \). Then \( F(X) \) is polynomially bounded in the size of \( X \), because \( F(X) \) has at most \((|V| - 1)|\Sigma|^3 \cdot k \cdot |V| |\Sigma|^3 \) mutually different pairs. Each pair contains a label set with at most \(|\Sigma|^3 \cdot k \) different pairs of \([k] \times \Sigma^3 \) and a sum of at most \(|\Sigma|^3 \) different addends.

The following observations show that for every fixed integer \( k \), \( F(\bullet_a) \), \( a \in [k] \), is computable in time \( O(1) \), \( F(X \oplus Y) \) is computable in polynomial time from \( F(X) \) and \( F(Y) \), and \( F(\eta_{a,b}(X)) \) and \( F(\rho_{a-b}(X)) \), \( a,b \in [k] \), are computable in polynomial time from \( F(X) \).

1. If \( val(X) \) consists of a single vertex \( u_i \), then
\[
F(\bullet_a) = \{((a, l), f_i(l)) | l \subseteq \Sigma^3\}
\]
2. \( F(X \oplus Y) \) is the set of all pairs \((L \cup L', f + f')\) which can be obtained by a pair \((L, f) \in F(X)\) and a pair \((L', f') \in F(Y)\).
3. \( F(\eta_{a,b}(X)) = \{(L, f) \in F(X) | (a, l_1), (b, l_2) \in L \Rightarrow (l_1, l_2) \) satisfies \( \Gamma \}\}
4. \( F(\rho_{a-b}(X)) = \{(\{\rho_{a-b}(a_1), l_1\}, \ldots, (\rho_{a-b}(a_m), l_m\}), f) | \{(a_1, l_1), \ldots, (a_m, l_m\}, f) \in F(X)\}

There is an admissible labeling of the vertices of \( val(X) \) with cost \( f \) if and only if there is some pair \((L, f) \in F(X)\). The corresponding labeling of the vertices of \( val(X) \) from \( \Sigma^3 \) can be recomputed from expression \( X \). By Theorem 2.2 the results follows. □ □

By Theorem 2.3 we conclude our main result of this section.

**Theorem 4.2** MRSO is computable in polynomial time for every class of structure graphs that define implied structure graphs of bounded clique-width.

Since every class of graphs of bounded tree-width has bounded clique-width [CR05, our result implies that even the MRSO problem can be solved in polynomial time for structure graphs which define implied structure graphs of bounded tree-width which has been shown in [BFHV05] for the MRSO-d1 problem.

Note that our solution is independent of alphabet \( \Sigma \) and set of complementary pairs \( \Gamma \), it is only important that \( \Sigma \) has a bounded size.

## 5 Comparing two solutions of MRSO-d1

In this section we consider for two given implied structure graphs \( G_1, G_2 \), and two sequences of similarly functions \( f_i, g_i, 1 \leq i \leq n \), the complexity of comparing the costs of the corresponding two solutions of problem MRSO-d1. We will show that these compare problems are even complete for the complexity class \( P_{NP} \), which is assumed to be a strong super set of \( NP \). Class \( P_{NP} \) is defined as the set of problems which can be solved in polynomial time with a number of parallel queries to an oracle in \( NP \). For more results concerning \( P_{NP} \)-hardness see [Wag87, SV00].

We next assume the restricted case that \( \Sigma = \{a, b, \pi, \overline{\pi}\} \) and \( \Gamma = \{(a, \pi), (b, \overline{\pi})\} \), see [Bon04].
Problem 5.1 (Comparing (Equality) MRSO-d1)

INSTANCE: Two structure graphs $G_1 = (\{v_1, \ldots, v_{3n}\}, E_1)$ and $G_2 = (\{u_1, \ldots, u_{3m}\}, E_2)$, $n$ functions $f_1, \ldots, f_n : \Sigma^3 \to Q$, and $m$ functions $g_1, \ldots, g_m : \Sigma^3 \to Q$.

QUESTION: Is $\text{MRSO-d1}(G_1, f_1, \ldots, f_n) \leq \text{MRSO-d1}(G_2, g_1, \ldots, g_m)$?

(QUESTION: Is $\text{MRSO-d1}(G_1, f_1, \ldots, f_n) = \text{MRSO-d1}(G_2, g_1, \ldots, g_m)$?)

Theorem 5.2 Comparing MRSO-d1 and Equality MRSO-d1 is $P^{\text{NP}}\|\text{-complete for planar graphs.}$

Proof First we have to show that Comparing MRSO-d1 and Equality MRSO-d1 is contained in $P^{\text{NP}}\|$. Therefor, we define a polynomial time algorithm solving the problem Comparing MRSO-d1 with a number of parallel queries to an oracle in NP. We take the MRSO-d1 problem as our oracle, which is in NP. Given two graphs $G_1$ and $G_2$, we ask the oracle the following two queries: $\text{MRSO-d1}(G_1, f_1, \ldots, f_n)$ and $\text{MRSO-d1}(G_2, g_1, \ldots, g_n)$. We accept for the problem Comparing MRSO-d1 if both values are equal. Analogously we can define parallel queries to an oracle in NP for the problem Equality MRSO-d1.

In [SV00] the problem of comparing the maximum vertex cover of two graphs has been shown to be $P^{\text{NP}}\|$-complete. Using the reductions of [GJS76] Theorem 2.7 and [GJ77] Lemma 1 we conclude that comparing the maximum vertex cover of two graphs remains $P^{\text{NP}}\|$-complete for planar graphs of vertex degree at most 3.

Since every vertex cover $C \subseteq V$ of a graph $G = (V, E)$, obviously corresponds to an independent set $V - C$ in graph $G$, we conclude that comparing the maximum independent set of two graphs remains $P^{\text{NP}}\|$-complete for planar graphs of vertex degree at most 3.

This allows us to show the $P^{\text{NP}}\|$-hardness of Comparing MRSO-d1 and Equality MRSO-d1 by a reduction from comparing independent set for planar graphs of vertex degree at most 3 by the idea of the proof of Theorem 3 in [Bon04]. Given an instance $G$ of maximum independent set of vertex degree at most 3, the proof constructs an instance $(G_{\text{impl}}, f_1, \ldots, f_n)$ for problem MRSO-d1, such that $\text{MRSO-d1}(G_{\text{impl}}, f_1, \ldots, f_n)$ is equal to the maximum independent set of $G$. \(\square\)

References

[Aku00] T. Akutsu. Dynamic programming algorithms for RNA secondary structure prediction with pseudoknots. Discrete Applied Mathematics, 104:45–62, 2000.

[BDF+95] H.L. Bodlaender, R.G. Downey, M.R. Fellows, M.T. Hallett, and H.T. Wareham. Parameterized complexity analysis in computational biology. Computer Applications in the Biosciences, 11(1):49–57, 1995.

[BDFW95] H.L. Bodlaender, R.G. Downey, M.R. Fellows, and H.T. Wareham. The parameterized complexity of sequence alignment and consensus. Theoretical Computer Science, 147(1-2):31–54, 1995.

[BFHV05] G. Blin, G. Fertin, D. Hermelin, and S. Vialette. Fixed-parameter algorithms for protein similarity search under mRNA structure constraints. In Proceedings of Graph-Theoretical Concepts in Computer Science, volume 3787 of LNCS, pages 271–282. Springer, 2005.
[BLS99] A. Brandstädt, V.B. Le, and J.P. Spinrad. Graph Classes: A Survey. SIAM Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, 1999.

[BNS02a] R. Backofen, N.S. Narayanaswamy, and F. Swidan. On the Complexity of Protein Similarity Search under mRNA Structure Constraints. In Proceedings of the Annual Symposium of Theoretical Aspects of Computer Science, volume 2285 of LNCS, pages 274–286. Springer, 2002.

[BNS02b] R. Backofen, N.S. Narayanaswamy, and F. Swidan. Protein similarity search under mrna structural constraints: application to selenocysteine incorporation. In Silico Biology 2, 26:275–290, 2002.

[Bon04] D. Bongartz. Some notes on the complexity of protein similarity search under mRNA structure constraints. In Proceedings of the Conference on Current Trends in Theory and Practice of Computer Science, volume 2932 of LNCS, pages 174–183. Springer, 2004.

[CER91] B. Courcelle, J. Engelfriet, and G. Rozenberg. Context-free handle-rewriting hypergraph grammars. In Graph-Grammars and Their Application to Computer Science, volume 532 of LNCS, pages 253–268. Springer, 1991.

[CER93] B. Courcelle, J. Engelfriet, and G. Rozenberg. Handle-rewriting hypergraph grammars. Journal of Computer and System Sciences, 46:218–270, 1993.

[CHL+00] D.G. Corneil, M. Habib, J.M. Lanlignel, B. Reed, and U. Rotics. Polynomial time recognition of clique-width at most three graphs. In Proceedings of Latin American Symposium on Theoretical Informatics, volume 1776 of LNCS, pages 126–134. Springer, 2000.

[CMR00] B. Courcelle, J.A. Makowsky, and U. Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. Theory of Computing Systems, 33(2):125–150, 2000.

[CO00] B. Courcelle and S. Olariu. Upper bounds to the clique width of graphs. Discrete Applied Mathematics, 101:77–114, 2000.

[CPS85] D.G. Corneil, Y. Perl, and L.K. Stewart. A linear recognition algorithm for cographs. SIAM Journal on Computing, 14(4):926–934, 1985.

[CR05] D.G. Corneil and U. Rotics. On the relationship between clique-width and treewidth. SIAM Journal on Computing, 4:825–847, 2005.

[DF99] R.G. Downey and M.R. Fellows. Parameterized Complexity. Springer, New York, 1999.

[DSO78] M. Dayhoff, R. Schwartz, and B. Orcutt. A model of evolutionary change in proteins. In Atlas of protein sequence and structure, volume 5, pages 345–352. National Biomedical Research Foundation, Washington, D.C., 1978.

[EGW01] W. Espelage, F. Gurski, and E. Wanke. How to solve NP-hard graph problems on clique-width bounded graphs in polynomial time. In Proceedings of Graph-Theoretical Concepts in Computer Science, volume 2204 of LNCS, pages 117–128. Springer, 2001.
[EGW03] W. Espelage, F. Gurski, and E. Wanke. Deciding clique-width for graphs of bounded tree-width. *Journal of Graph Algorithms and Applications - Special Issue of JGAA on WADS 2001*, 7(2):141–180, 2003.

[FRRS06] M.R. Fellows, F.A. Rosamund, U. Rotics, and S. Szeider. Clique-width minimization is NP-hard. In *Proceedings of the Annual ACM Symposium on Theory of Computing*, pages 354–362. ACM, 2006.

[GJ77] M.R. Garey and D.S. Johnson. The rectilinear steiner tree problem is NP-complete. *SIAM J. Appl. Math.*, 32(4):826–834, 1977.

[GJ79] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, San Francisco, 1979.

[GJS76] M.R. Garey, D.S. Johnson, and L. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1(3):237–267, 1976.

[GR00] M.C. Golumbic and U. Rotics. On the clique-width of some perfect graph classes. *International Journal of Foundations of Computer Science*, 11(3):423–443, 2000.

[GW06] F. Gurski and E. Wanke. Vertex disjoint paths on clique-width bounded graphs. *Theoretical Computer Science*, 359(1-3):188–199, 2006.

[KR03] D. Kobler and U. Rotics. Edge dominating set and colorings on graphs with fixed clique-width. *Discrete Applied Mathematics*, 126(2-3):197–221, 2003.

[MR99] J.A. Makowsky and U. Rotics. On the clique-width of graphs with few $P_4$. *International Journal of Foundations of Computer Science*, 10:329–348, 1999.

[OS06] S.-I. Oum and P.D. Seymour. Approximating clique-width and branch-width. *Journal of Combinatorial Theory, Series B*, 96(4):514–528, 2006.

[Oum05] S.-I. Oum. Approximating rank-width and clique-width quickly. In *Proceedings of Graph-Theoretical Concepts in Computer Science*, volume 3787 of *LNCS*, pages 49–58. Springer, 2005.

[Oum06] S.-I. Oum. Approximating rank-width and clique-width quickly. Manuscript, 2006.

[SV00] H. Spakowski and J. Vogel. $\theta_2^p$-completeness: A classical approach for new results. In *Proceedings of the twentieth conference on the foundations of software technology and theoretical computer science*, LNCS, pages 348–360. Springer, 2000.

[Wag87] K.W. Wagner. More complicated questions about maxima and minima, and some closures of NP. *Theoretical Computer Science*, 51(1-2):53–80, 1987.