

On finding the longest antisymmetric path in directed acyclic graphs



Yinglei Song^{a,*}, Menghong Yu^b

^a School of Computer Science and Engineering, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China

^b School of Electronics and Information Sciences, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China

ARTICLE INFO

Article history:

Received 4 April 2014

Received in revised form 12 October 2014

Accepted 13 October 2014

Available online 22 October 2014

Communicated by Jinhui Xu

Keywords:

Computational complexity

Lower bound

Inapproximability

Longest antisymmetric path

Directed acyclic graphs

ABSTRACT

Given a directed acyclic graph and a set of vertex pairs, an antisymmetric path is a directed path that does not contain vertices from the same P in the set. The goal of the LONGEST ANTISYMMETRIC PATH problem is to determine the longest antisymmetric path that connects two given vertices in the graph. The problem has important applications in software testing and bioinformatics. The problem is known to be NP-hard. In this paper, we study the computational lower bound of the problem, we show that the problem cannot be solved in time $2^{o(n^{1/3})}$ unless 3SAT can be solved in subexponential time. In addition, we study the inapproximability of the problem and show that the problem cannot be approximated within a ratio of $\frac{(n-2)}{2}$ in polynomial time unless $P = NP$. Our technique also suggests that both the lower bound and inapproximability also hold for directed acyclic graphs of degree at most 6.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Given a directed acyclic graph $G = (V, E)$, and a set S of vertex pairs, an *antisymmetric path* is a directed path that does not contain two vertices that are both included in a pair in S . A pair in S is a *forbidden pair*. Given two vertices s and t in G , the goal of the LONGEST ANTISYMMETRIC PATH problem is to find the antisymmetric path that connects s and t in G and is the longest one of all such paths.

The problem was originally proposed in [10] and [17], where the automatic software testing and validation is considered. In [6], it is shown that the problem is NP-hard. Later, the problem is found to be associated with the *De Novo* sequencing problem in bioinformatics [4]. The

goal of the *de novo* sequencing problem is to determine the amino acids sequence of a peptide from its experimental tandem mass spectrometry (MS/MS) spectrum. In [4], it is proposed that a spectrum graph can be used to model the mass peaks in a spectrum and the vertices that represent complementary mass peaks form a set of vertex pairs. The amino acids sequence of the peptide can be obtained by computing a longest antisymmetric path in the spectrum graph. A linear time dynamic programming algorithm is developed in [2] to compute the path. However, the algorithm needs linear time only in the ideal case, where no additional noisy mass peaks appear in the spectrum. The computation time of the algorithm becomes exponential when the spectrum contains a large number of noisy mass peaks.

In practice, the experimental spectrum often contains noisy mass peaks due to multiple fragmentations of a single peptide. A large number of vertices in the spectrum

* Corresponding author.

E-mail addresses: yingleisong@gmail.com (Y. Song), ymhjz2691@163.com (M. Yu).

graph thus do not appear in the longest antisymmetric path that need to be computed. An algorithm that can efficiently compute a longest antisymmetric path in any given directed acyclic graph is thus highly desirable for determining the amino acids sequence of a peptide solely from its experimental MS/MS spectrum.

The problem has been studied intensively. For example, in our previous work, we introduce the notion of extended spectrum graph to model the vertex pairs in a spectrum graph. Specifically, we join each pair of vertices that form a pair with an undirected edge and the resulting graph is an extended spectrum graph. We develop an algorithm that can solve the problem in time $O(6^t n)$, where t is the tree width of the extended spectrum graph and n is the number of mass peaks in the spectrum [11, 12]. Recently, we showed that the problem can be solved in time $O(p^2 2^p n)$ given a path decomposition of the extended spectrum graph, where p is the path width of the path decomposition [16]. In [9], it is shown that the problem remains NP-hard even if the forbidden pairs have a halving structure. However, it can be solved in polynomial time if the forbidden pairs have a hierarchical structure. Recently, an efficient algorithm [1] based on linear integer programming techniques was developed to solve the problem. Although the problem is known to be NP-hard, to our best knowledge, a computational lower bound for the problem is still not available. In addition, it remains unknown whether the problem can be approximated within a good approximation ratio in polynomial time.

In this paper, we study the computational lower bound and inapproximability of the problem. We show that the problem cannot be solved in time $2^{o(n^{1/3})}$, where n is the number of vertices in the graph. For inapproximability, we show that the problem cannot be approximated within a ratio of $\frac{(n-2)}{2}$ in polynomial time unless $P = NP$.

2. Computational complexity

A directed graph $G = (V, E)$ is *acyclic* if G does not contain a directed cycle. Given a directed acyclic graph $G = (V, E)$ and a set $P = \{(v_1, u_1), (v_2, u_2), \dots, (v_n, u_n)\}$ of vertex pairs, two vertices u and v are *symmetric* if $(u, v) \in P$. A directed path P in G is *antisymmetric* if it does not contain two vertices u and v that are symmetric.

Given a directed acyclic graph $G = (V, E)$, two vertices $s \in V$, $t \in V$, and a set P of vertex pairs, the LONGEST ANTISYMMETRIC PATH problem is to find the longest antisymmetric path that connects s and t in G . We show that this problem cannot be solved in time $2^{o(n^{1/3})}$ unless 3SAT can be solved in subexponential time.

Theorem 2.1. *Given a directed acyclic graph $G = (V, E)$, the LONGEST ANTISYMMETRIC PATH cannot be solved in time $2^{o(|V|^{1/3})}$ unless 3SAT can be solved in subexponential time.*

Proof. We construct a polynomial time reduction from the 3SAT problem to the LONGEST ANTISYMMETRIC PATH problem. Given an instance of the 3SAT problem, we assume its sentence is the conjunction of m clauses $c_1, c_2, c_3, \dots, c_m$

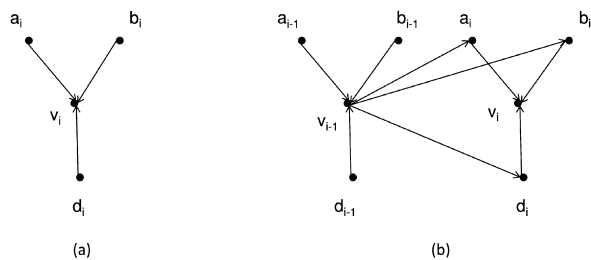


Fig. 1. (a) A graph gadget that represents a clause; (b) the directed edges that join vertices in the gadget of c_{i-1} to those in that of c_i .

and each clause c_i ($1 \leq i \leq m$) contains three literals l_{i1} , l_{i2} , l_{i3} . For c_i , we construct a graph gadget as follows.

1. Create four vertices v_i , a_i , b_i , d_i , where v_i represents the clause c_i and a_i , b_i , d_i represent the three literals l_{i1} , l_{i2} , and l_{i3} in c_i .
2. Join a_i , b_i and d_i to v_i with directed edges that go from the three vertices to v_i .

Fig. 1(a) shows an example of a gadget where v_i is the vertex that represents clause c_i and a_i , b_i and d_i represent the three literals in the clause. In the rest of the paper, a vertex that represents a literal is referred to as a *literal vertex* and a vertex that represents a clause is referred to as a *clause vertex*. As the next step of the construction, we construct a directed graph G based on the graph gadgets of all clauses.

To construct G , all clauses are placed into an order of c_1, c_2, \dots, c_m , where c_{i-1} is placed immediately before c_i . ($2 \leq i \leq m$) We then join v_{i-1} to each of the three vertices a_i , b_i , and d_i with a directed edge. All these three edges are from v_{i-1} to a_i , b_i , and d_i respectively. Fig. 1(b) shows the directed edges that join the vertices in the gadgets of clauses c_{i-1} and c_i .

As the last step of the construction, we create two additional vertices s and t . We join s to each of the three vertices a_1 , b_1 , and d_1 with a directed edge. All these three edges go from s to a_1 , b_1 , and d_1 respectively. In addition, we join each of the three vertices a_m , b_m , and d_m to t with a directed edge. All these three edges go from a_m , b_m , and d_m to t .

We then construct a set P of vertex pairs. Any pair of literal vertices that represent different literals of the same variable are included as a pair in P . In other words, they cannot both be true under an assignment of the truth value of the variable.

We show that G is acyclic. Since if G contains a cycle, the cycle does not contain two clause vertices v_i and v_j ($1 \leq i < j \leq m$), since otherwise there exists a directed path from v_j to v_i in G , which is contradictory to the construction of G . The cycle thus must contain at least one directed edge from a literal vertex to another literal one, which is a contradiction since such an edge does not exist in G . Therefore, G must be acyclic.

We now show that the sentence is satisfiable if and only if s and t are connected with an antisymmetric path of length $2m + 1$ in G . Firstly, we assume the sentence is satisfiable and there exists a truth assignment A to the variables in the sentence such that

Download English Version:

<https://daneshyari.com/en/article/10331921>

Download Persian Version:

<https://daneshyari.com/article/10331921>

[Daneshyari.com](https://daneshyari.com)