ELSEVIER

Information Processing Letters 65 (1998) 119-123

Information Processing Letters

The NPO-completeness of the longest Hamiltonian cycle problem

Q.S. Wu [a,1], K.M. Chao [b,2], R.C.T. Lee [b,*]

[a] Department of Computer Science, National Tsing Hua University, Hsinchu 30043, Taiwan, ROC

[b] Department of Computer Science, Providence University, Shalu, Taichung Hsien 43309, Taiwan, ROC

Received 30 June 1997; accepted 6 December 1997
Communicated by T. Asano

Abstract

In this paper, the longest Hamiltonian cycle problem and the longest Hamiltonian path problem are proved to be NPO-complete. © 1998 Published by Elsevier Science B.V.

Keywords: Computational complexity; NPO-complete; Strict reduction; Longest Hamiltonian cycle problem; Longest traveling salesperson problem

1. Introduction

The concept of NP-completeness was proposed by Cook [2] to show the hardness of a problem. Many problems have been shown to be NP-complete [3]. This is equivalent to saying that those problems can hardly have any efficient algorithm.

Since it is believed that an NP-complete problem will unlikely have any efficient algorithm, people turn to searching for good approximation algorithms. For some problems, we have good approximation algorithms; for some others, none has been found. It is interesting to study whether there can be a good approximation algorithm for an NP-complete problem. Papadimitriou proposed the concept of MAX SNP to show that some problems cannot have polynomial time approximation schemes [8]. However, the class MAX SNP only contains problems which can be expressed in a special logical form. Therefore it is a very restricted class. Recently, Ausiello, Crescenzi, and Protasi suggested the concept of NPO-completeness [1]. They first defined the class of NPO problems, which contains optimization problems and is a very large class. Formally, an *NPO problem* is defined as follows [1].

Definition 1. An *NP optimization* (NPO) problem A is a four-tuple $(I, sol, m, goal)$ such that

(1) I is the set of the *instances* of A and it is recognizable in polynomial time.

(2) Given an instance x of I, $sol(x)$ denotes the set of *feasible solutions* of x. A polynomial function p exists such that, for any x and for any $y \in sol(x)$, $|y| \leqslant p(|x|)$. Moreover, for any x and for any y such that $|y| \leqslant p(|x|)$, it is decidable in polynomial time whether $y \in sol(x)$.

* Corresponding author. Email: rctlee@simon.pu.edu.tw.

[1] Email: solomon@cs.nthu.edu.tw. Research supported in part by a grant of National Science Council of Republic of China (NSC87-2213-E-126-001).

[2] Email: kmchao@csim.pu.edu.tw.

PII S0020-0190(98)00003-9

(3) Given an instance x and a feasible solution y of x, $m(x,y)$ denotes the positive integer measure of y (also called the value of y). The function m is computable in polynomial time and is also called the *objective function*.
(4) *goal* $\in \{\max, \min\}$.

Intuitively, according to the above definition, a nondeterministic algorithm can be associated with any NPO problem A that, for any instance x of A, performs the following algorithm.

begin
  guess $y \in \{0,1\}^{p(|x|)}$;
  if y is a feasible solution of x
  then output $m(x,y)$
  else abort
end.

To construct the completeness in the NPO class, we need a reduction. We describe the definition in [6] as follows:

Definition 2. Given NPO problems A and B, (f,g) is a *strict-reduction* from A to B if
(1) For every instance x in A, $f(x)$ is an instance in B.
(2) For every feasible solution y to $f(x)$ in B, $g(y)$ is a feasible solution in A.
(3) The absolute error of $g(y)$ to the optimal of x is less than or equal to the absolute error of y to the optimal of $f(x)$ in B. That is, $|g(y) - OPT_A(x)| \leqslant |y - OPT_B(f(x))|$.

Fig. 1 illustrates the concept of "strict reduction".

Definition 3. An NPO problem is *NPO-complete* if all NPO problems strictly reduce to it.

It can be seen that, if problem A strictly reduces to problem B, and B has an approximation algorithm whose error relative to the optimal is smaller than ε, then we can use it to construct an approximation algorithm of A whose error is guaranteed to be smaller than ε. Therefore, if an optimization problem A strictly reduces to an optimization problem B, then the fact that problem B has an approximation algorithm with a constant performance ratio will imply that problem A also has an approximation algorithm with a constant performance ratio. Hence, if an NPO-complete problem has any constant-ratio approximation algorithm, then all NPO problems have constant-ratio approximation algorithm. Thus the hardness of problem A is established.

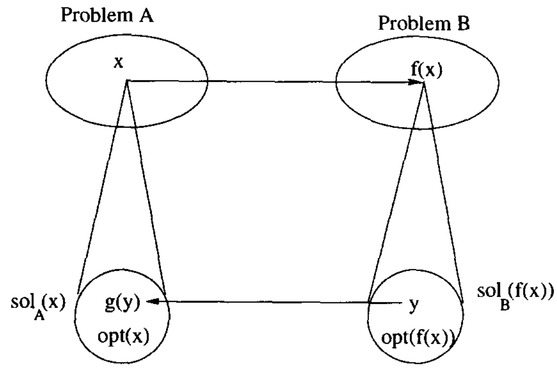

Fig. 1. Reduction from A to B.

2. Problem definition

The Euclidean traveling salesperson problem is defined as follows: *Given the distances between each pair of n cities, the salesperson is required to find the shortest tour that traverses all the cities exactly once.* However, on the contrary, it might also be possible for people to consider the *longest* tour. In [5] it was proposed that for a salesman whose expenses will be provided by the company, he may prefer to maximize the length of the tour so that his flying mileage can be maximized. If we generalize this problem from Euclidean planes in [5] to general graphs, it leads to the longest Hamiltonian cycle problem and the longest Hamiltonian path problem which we study in this paper.

Problem 1. *The longest Hamiltonian cycle problem.*
Instance: A weighted graph: $G = (V, E)$, where each $e \in E$ has a weight $w(e)$. Let $n = |V|$.
Solution: A simple cycle in G, i.e. a sequence of vertices $v_1, v_2, \ldots, v_n, v_1$ such that, for any $1 \leqslant i \leqslant n-1$, $(v_i, v_{i+1}) \in E$ and $(v_n, v_1) \in E$.
Measure: The length of the cycle, i.e., $\sum_{i=1}^{n-1} w(v_i v_{i+1}) + w(v_n v_1)$.
Goal: max.

Problem 2. *The longest Hamiltonian path problem.*
Instance: A weighted graph: $G = (V, E)$, where each $e \in E$ has a weight $w(e)$, and two given vertices u, v. Let $n = |V|$.
Solution: A simple path in G, i.e. a sequence of vertices $u = v_1, v_2, \ldots, v_n = v$ such that, for any $1 \leqslant i \leqslant n-1$, $(v_i, v_{i+1}) \in E$.
Measure: The length of the path, i.e., $\sum_{i=1}^{n-1} w(v_i v_{i+1})$.
Goal: max.

We are interested in whether it is possible for the longest Hamiltonian cycle problem and the longest Hamiltonian path problem to have any approximation algorithm with constant performance ratio. In this paper, we shall show the NPO-completeness of the longest Hamiltonian cycle problem and the longest Hamiltonian path problem. Thus it is unlikely for them to have any approximation algorithm with constant performance ratio. It is worth noting that a related problem, namely the longest path problem, has been proved to have no approximation algorithm with constant performance ratio unless NP = P [4].

3. NPO-completeness of the longest Hamiltonian cycle problem

Theorem 4. *The longest Hamiltonian cycle problem is* NPO-*complete.*

Essentially, we shall show that the maximum weighted 3-satisfiability problem (MAX-W3SAT [6]) strictly reduces to the *longest Hamiltonian cycle* problem. The definition of MAX-W3SAT is as follows:

Problem 3. *The maximum weighted 3-satisfiability problem.*
Instance: A boolean formula ϕ which is a conjunction of 3-clauses (clauses with exactly 3 literals), with positive integer weights $w(x_i)$ on the variables appearing in ϕ.
Solution: A truth assignment $\tau(x_i)$ assigned to the variables, where $\tau(x_i)$ satisfies the formula ϕ.
Measure: $\sum_{\tau(x_i)=true} w(x_i)$.
Goal: max.

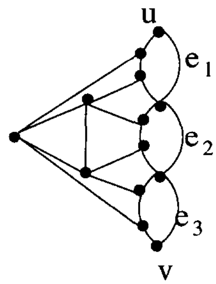

Fig. 2. The component representing a clause.

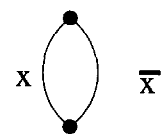

Fig. 3. The component representing a variable.

Proof of Theorem 4. The construction of our strict-reduction consists of 2 stages:

(1) We use the reduction from 3SAT to the Hamiltonian cycle problem as in [7]. Fig. 2 shows a basic graph corresponding to a 3-clause and Fig. 3 corresponds to a Boolean variable.

Given a set of 3-clauses, we shall construct a graph. There will be a component, such as that shown in Fig. 2, for each clause. All such components are identical. For each variable which appears in the clauses, there will be a component also. Again, all such components are identical. The component for a clause C will be connected to the component for variable x if and only if x appears in C. The connection depends upon how x appears, positively or negatively, in C. (This is called *A-connector* in [7].) The connection scheme is shown in Fig. 4. As illustrated in Fig. 4, if x_1 appears positively in clause 1, we combine the left edge of the component representing x_1 to the edge which represents the literal in the clause. If x_1 appears negatively in clause 2, we combine its right edge to the edge which represents the corresponding literal in clause 2.

In Fig. 2, for every Hamiltonian path from u to v, at least one of the edges e_1, e_2, e_3 will not be traversed (this corresponds to the true literal in a satisfied clause). Similarly, for the graph of Fig. 3, either x or $\bar{x}$ will be traversed.

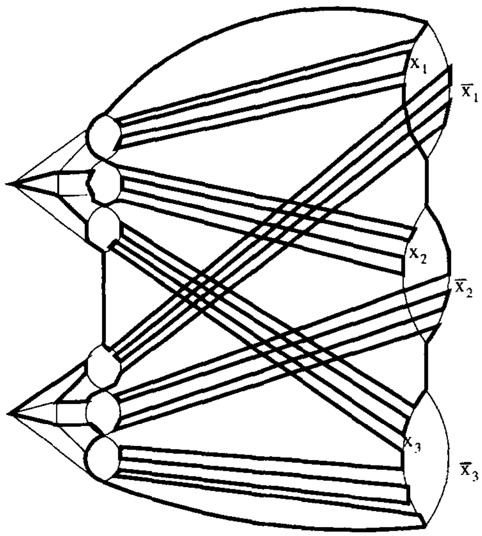

Fig. 4. An example: $(x_1 \vee x_2 \vee x_3) \wedge (\bar{x_1} \vee \bar{x_2} \vee x_3)$ with the assignment $(x_1, \bar{x_2}, x_3)$.

It was shown in [7] that this graph has a Hamiltonian cycle if and only if there is a truth assignment of the formula in 3-SAT. In our example, the bold line shown in Fig. 4, which is a tour, corresponds to a truth assignment $x_1 = T$, $x_2 = F$, $x_3 = T$.

(2) The weight of each edge is assigned as follows. The left edge corresponding to x_i is assigned weight $w(x_i)$, and all the other edges assigned weights 0. It can be seen easily that G has a Hamiltonian Cycle with weight W if and only if ϕ has a truth assignment with weight W.

Thus, by (1) and (2), we obtain a strict-reduction from the MAX-W3SAT problem to the longest Hamiltonian cycle problem. In [6] it was already shown that MAX-W3SAT is NPO-complete. Therefore, by strictly reducing MAX-W3SAT to the longest Hamiltonian cycle problem, we have successfully proved that the longest Hamiltonian cycle problem is NPO-complete. □

4. NPO-completeness of the longest Hamiltonian path problem

Theorem 5. *The longest Hamiltonian path problem is* NPO-*complete.*

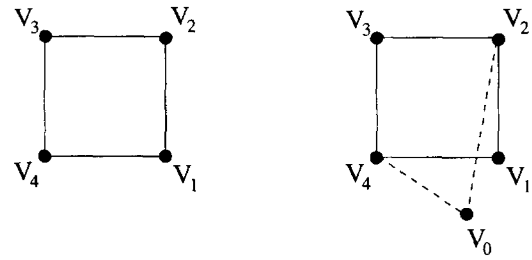

Fig. 5. Reduction from the longest Hamiltonian cycle problem to the longest Hamiltonian path problem.

Proof. We strictly reduce the longest Hamiltonian cycle problem to the longest Hamiltonian path problem.

Suppose $G = (V, E)$ is the given graph in the longest Hamiltonian cycle problem, and the weight of each edge e is $w(e)$. Let us construct G' in the longest Hamiltonian path problem such that, $G' = (V', E')$, where $V' = V \cup \{v_0\}$ and $E' = E \cup \{(v_0, u) \mid \forall (v_1, u) \in E\}$. We set the weights of edges in E' to be

$$w'(uv) = w(uv) \quad \forall uv \in E,$$

and

$$w'(uv_0) = w(uv_1) \quad \forall uv_0 \in E'.$$

It can be easily seen that, G has a Hamiltonian cycle if and only if G' has a Hamiltonian path from v_1 to v_0. See Fig. 5.

Because we assign the weight of (v_0, u) to be the same as the weight of (v_1, u), G has a Hamiltonian cycle with length W if and only if G' has a Hamiltonian path with length W. Hence we have strictly reduced the Longest Hamiltonian Cycle problem to the longest Hamiltonian path problem. □

5. Conclusion

In this paper, we have proved the NPO-completeness of the longest Hamiltonian cycle problem and the longest Hamiltonian path problem. Hence, it is unlikely that these two problems have approximation algorithm with constant performance ratios. The maximum traveling salesperson problem on general graphs [5], is thus hard to approximate since it is equivalent to the longest Hamiltonian cycle problem.

References

[1] G. Ausiello, P. Crescenzi, M. Protasi, Approximate solution of NP optimization problems, Theoret. Comput. Sci. 150 (1995) 1–55.

[2] S.A. Cook, The complexity of theorem-proving procedures, in: Proc. 3rd Ann. ACM Symp. on Theory of Computing, ACM, New York, 1971, pp. 151–158.

[3] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-completeness, Freeman, San Francisco, CA, 1979, pp. 56–60.

[4] D. Karger et al., On approximating the longest path in a graph, in: Proc. 3rd Workshop on Algorithms and Data Structures, 1993, pp. 421–432.

[5] U. Manber, Introduction to Algorithms: A Creative Approach, 1989, p. 374.

[6] P. Orponen, H. Mannila, On approximation preserving reductions: Complete problems and robust measures, Tech. Rept. C-1987-28, Department of Computer Science, University of Helsinki, 1987.

[7] C.H. Papadimitriou, K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity 1982, pp. 366–371.

[8] C.H. Papadimitriou, M. Yannakakis, Optimization approximation, and complexity classes, J. Comput. Systems Sci. 43 (1991) 425–440.

[9] S. Sahni, T. Gonzalez, P-complete approximation problems, J. ACM 23 (1976) 555–565.