ISSN 0918-2802

Technical Report

NP-completeness Results for NONOGRAM via Parsimonious Reductions

Nobuhisa Ueda and Tadaaki Nagao

TR96-0008 may

Department of Computer Science
Tokyo Institute of Technology
Ôokayama 2-12-1 Meguro Tokyo 152, Japan
http://www.cs.titech.ac.jp/

NP-completeness Results for NONOGRAM via Parsimonious Reductions

Nobuhisa UEDA
(ueda@cs.titech.ac.jp)
c/o Prof. T. SATO
Department of Electrical and Electronic Engineering,

Tadaaki NAGAO
(nagao@cs.titech.ac.jp)
c/o Prof. O. WATANABE
Department of Computer Science,

Tokyo Institute of Technology Meguro-ku, Tokyo 152, Japan

Abstract

We introduce a new class of NP problems called ANOTHER SOLUTION PROBLEMs. For a given NP problem X, ANOTHER SOLUTION PROBLEM for X (ASP for X) is to ask, for a given instance I for X and its solution, whether there is another solution for I. The difficulty of ASP for X may not be immediate from the difficulty of X. For example, for some NP-complete problems such as 3SAT or 3DM, it is easy to show that their ASPs are NP-complete; on the other hand, ASP for, e.g., VERTEX COLORING is trivially in P. In this paper, we propose one approach for showing the NP-completeness of ASP for a given NP problem X. That is, for some NP problem Y for which it is provable that ASP is NP-complete, we show a parsimonious reduction from Y to X that has the following additional property: given a solution for an instance for Y, a solution for the corresponding instance for X is computable in polynomial time. Then the NP-completeness of ASP for Y clearly implies that of ASP for X. As an example of this approach, we demostrate the NP-completeness of ASP for one interesting puzzle "NONOGRAM".

1 Introduction

There are cases where we would like to know whether there is *any other* solution for a given problem and its solution. For example, consider TRAVELING SALESMAN PROBLEM (TSP). Suppose that, for some given TSP instance, we have already obtained one solution with the desired cost. But time to time, we might want to ask whether there is any other solution (with the same cost) for this instance. In this paper, we name this type of problem as "ANOTHER SOLUTION PROBLEM (ASP)". We propose one approach for studying the complexity of ASP for NP problems, and following this approach, we demonstrate the NP-completeness of ASP for some problem.

ANOTHER SOLUTION PROBLEM (ASP) occur frequently in puzzles. When designing a puzzle, we often determine one solution first and make a puzzle consistent with this solution. Then it is important to check whether the obtained puzzle has no other solution besides the desired one. In this paper, we consider the puzzle called "NONOGRAM" and investigate ASP for NONOGRAM.

NONOGRAM is a puzzle which seeks a pattern that matches some given integer sequences [2]. An example of an instance of NONOGRAM is shown in Figure 1 (a), and its solution is in Figure 1 (b). More precisely, an instance of NONOGRAM is a matrix and a sequence of non-negative integers on each row and each column. A solution is an assignment to each entry of the matrix with either a ■ (block) or a $\boxed{\bullet}$ (point) such that the number of contiguous blocks matches the given integer sequences on each row and each column.

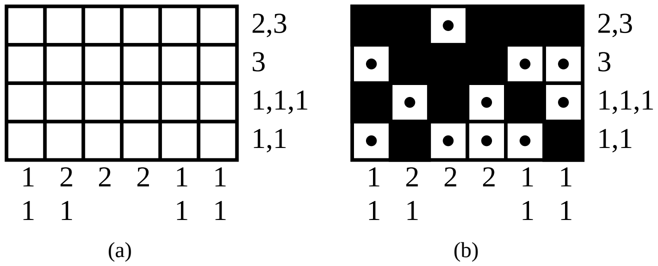

Figure 1: An instance of NONOGRAM and its solution

It is not so hard to show that NONOGRAM itself is NP-complete. Notice that even if a problem X is NP-complete, it does not always imply that ANOTHER SOLUTION PROBLEM (ASP) for X is NP-complete. In fact, for some NP-complete problem, such as Vertex Coloring, ASP is trivial. Thus, we need to find some way to determine whether ASP for a given X is NP-complete. Here we propose one approach: First for some problem Y, show that ASP for Y is NP-complete. Then show a *parsimonious reduction* from Y to X that has the following additional property: $(*)$ given a solution for an instance for Y, a solution for the corresponding instance for X is computable in polynomial time. (Recall that parsimonious reductions are reductions that preserve the number of solutions [3]. Hence if ASP for Y is NP-complete, and Y is reducible to X via such parsimonious reduction, then ASP for X is NP-complete.) In this paper, we show that ASP for NONOGRAM is NP-complete by showing (i) that ASP for 3-DIMENSIONAL MATCHING (3DM) is NP-complete, and (ii) a parsimonious reduction from 3DM to NONOGRAM with the above property $(*)$.

2 Preliminaries

We explain our notations. Let h and w denote the height and the width of a NONOGRAM instance. We use a (h, w)-matrix P to denote the assingment to the NONOGRAM instance, and let $p(s, t)$ be the (s, t)th entry of P. Let 1,0 and * represent the values ■ , $\boxdot$ and □ respectively. For example, P_0 and P_1 represent Figure 1(a) and 1(b),

$$P_0 = \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \end{pmatrix}, \qquad P_1 = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Let $r_s = \langle r_{s,1}, \ldots, r_{s,i_s} \rangle$ denote a sequence of the integers on the sth row, and $c_t = \langle c_{t,1}, \ldots, c_{t,j_t} \rangle$ on the tth column. NONOGRAM is defined formally as follows:

NONOGRAM

INPUT:	$h, w \in \mathbf{N}$ and sequences of non-negative integers $R = \langle r_1, \ldots r_h \rangle$ and $C = \langle c_1, \ldots, c_w \rangle$.
QUESTION:	Is there an assignment P of a $h \times w$ matrix to $\{0, 1\}$ such that NONOGRAM property is satisfied?

Since an instance of NONOGRAM is represented by a (h, w)-matrix and the integers sequences, we can assume the size of an instance of NONOGRAM to be $h \times w$. Notice that for a given instance I and an assignment P, one can check in polynomial time whether P is a solution of I. Thus, NONOGRAM is in NP.

Given an instance of NONOGRAM and one of its solution, ANOTHER SOLUTION PROBLEM (ASP) for NONOGRAM is to ask whether there is another solution. That is, the following problem.

ASP for NONOGRAM

INPUT:	An instance of NONOGRAM and a solution P.
QUESTION:	Is there another solution $P' \neq P$?

To achive our goal, we introduce a known NP-complete problem 3DM:

3DM

INPUT:	Set $M \subseteq X \times Y \times Z$, where X, Y and Z are disjoint sets having the same number q of elements.
QUESTION:	Does M contain a matching, i.e., a subset $M' \subseteq M$ such that $\|M'\| = q$ and no two elements of M' agree in any coordinate?

We also define ASP for 3DM as follows:

ASP for 3-DIMENSIONAL MATCHING (ASP-3DM)

INPUT:	Set $M \subseteq X \times Y \times Z$ and $M' \subseteq M$, where X, Y and Z are disjoint sets having the same number q of elements, and M' is a matching for M.
QUESTION:	Does M contain another matching?

3 NP-Completeness of NONOGRAM

We show our main result, i.e., a parsimonious reduction from 3DM to NONOGRAM.

Theorem 3.1. There is a parsimonious reduction from 3DM to NONOGRAM.

Proof. Consider any instance of 3DM; that is, for any $X = \{x_1, \ldots, x_q\}$, $Y = \{y_1, \ldots, y_q\}$, and $Z = \{z_1, \ldots, z_q\}$, let M be any subset $\{m_1, \ldots, m_n\}$ of $X \times Y \times Z$. (Here we use $n = |M|$ to denote the size of the instance.) We may assume without loss of generality that $n \geq q$. Here we will show the way to define a NONOGRAM instance I corresponding to M.

But first, we explain the idea of our reduction by using an example. Consider the following instance of 3DM:

$$M_e = \{\langle x_1, y_1, z_1\rangle, \langle x_1, y_2, z_3\rangle, \langle x_2, y_2, z_3\rangle, \langle x_2, y_3, z_2\rangle, \langle x_3, y_2, z_3\rangle, \langle x_3, y_3, z_2\rangle\},$$

where we use $q_e = 3$. Then M_e has a matching $M'_e = \{\langle x_1, y_1, z_1\rangle, \langle x_2, y_2, z_3\rangle, \langle x_3, y_3, z_2\rangle\}$.

We reduce this instance M_e to the instance I_e illustrated in Figure 2. Notice that Figure 2 also illustrates a solution corresponding to the matching M'_e. That is, the NONOGRAM instance I_e corresponding to M_e is a blank matrix with the numbers as in Figure 2, and the assignment to each entry of the matrix corresponds to M'_e.

As explained in Figure 2, each $(2l-1)$th row $(1 \leq l \leq 6)$ corresponds to the lth element $m_l \in M_e$. On the other hand, each pair of two columns except the 1st column and the wth column corresponds to a particular element $v \in X \cup Y \cup Z$ in order of X, Y and Z.

In this solution, the assignment to I_e's $(2l-1)$th row is determined depending on whether $m_l \in M_e$ is selected as matching M'_e. The difference is whether two contiguous points $\boxed{\bullet}$ appear in the x's part (in which case m_l is not in M'_e), or in the z's part (in which case m_l is in M'_e). More precisely, we design NONOGRAM instances so that no other assignment is valid w.r.t. the instance.

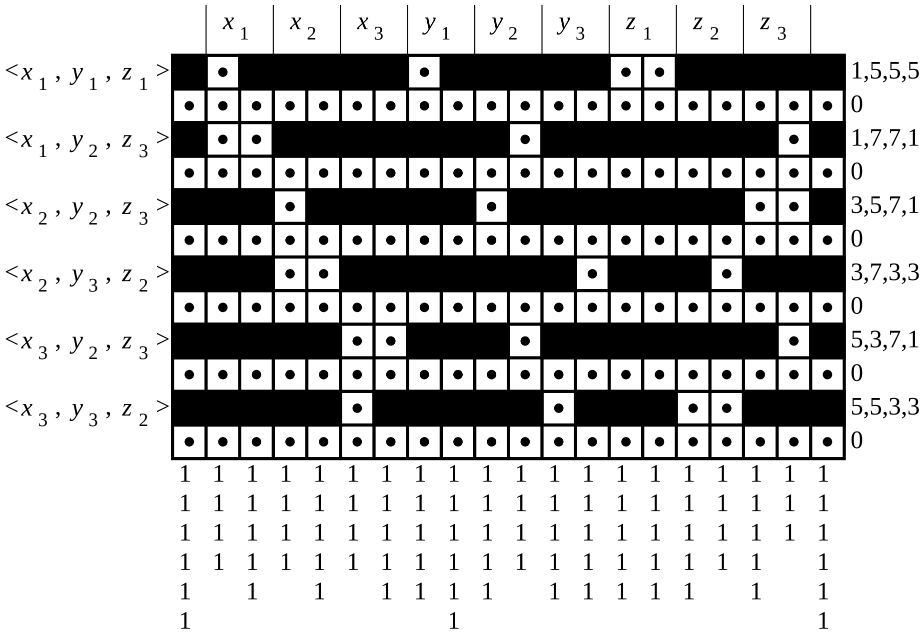

Figure 2: A NONOGRAM instance and its solution corresponding to M_e and M'_e.

Now we explain in general requirements for "desired" solutions. Recall M is any instance of 3DM, and I is the NONOGRAM instance that our reduction yields from M. Here I is designed so that it has (if it indeed has) only solutions that satisfy the following requirements:

(R1) Every even row of I is assigned $\boxed{\bullet}$; that is, $p(2l, -)$ is assinged 0. (Recall that $p(s,t)$ is the (s,t) entry of the assignment matrix, and we use $p(s,-)$ and $p(-,t)$ to denote the sth row and the tth column respectively.)

(R2) Every $(2l-1)$th row of I is assinged one of the two patterns. More specifically, letting $m_l = \langle x_i, y_j, z_k \rangle$, the $(2l-1)$th row of I (i.e., $p(2l-1, -)$) is set either pattern (A) or (B) of Figure 3. (The choice of (A) or (B) will be determined depending on whether m_l is selected for the corresponding matching M'.)

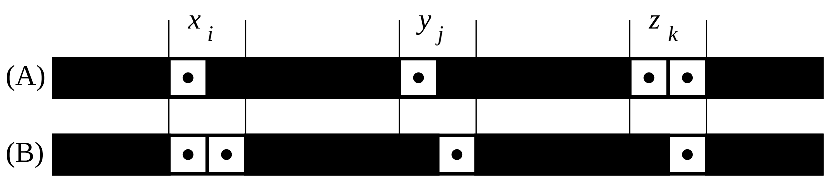

Figure 3: (A) $m_l \in M'$, and (B) $m_l \in M - M'$.

Now we explain precisely how I is constructed from M. First we prepare some functions. Define f_1, f_2, f_3 and f_4 to be functions mapping each $m_l \in M$ to some integer in the following way.

$$
\begin{aligned}
f_1(\langle x_i, y_j, z_k\rangle) &= 2i-1,\\
f_2(\langle x_i, y_j, z_k\rangle) &= 2(q+j-i)-1,\\
f_3(\langle x_i, y_j, z_k\rangle) &= 2(q+k-j)-1,\\
f_4(\langle x_i, y_j, z_k\rangle) &= 2(q-k)+1.
\end{aligned}
$$

Also define g_1 and g_2 as follows:

$$
g_1(v) = \begin{cases} n-N(v), & v\in X,\\ n-1, & v\in Y, v\in Z,\end{cases} \qquad g_2(v) = \begin{cases} n-N(v)+1, & v\in X, v\in Y,\\ n-N(v), & v\in Z.\end{cases}
$$

where $N(v)$ denotes the number of occurrence of $v \in X\cup Y\cup Z$ in M.

For notational convenience, we introduce the following g;

$$
g(t) = \begin{cases}
n, & t = 1, 6q+2\\
g_1(x_i), & t = 2i \text{ for some } i,\ 1\le i\le q,\\
g_2(x_i), & t = 2i+1 \text{ for some } i,\ 1\le i\le q,\\
g_1(y_i), & t = 2q+2i \text{ for some } i,\ 1\le i\le q,\\
g_2(y_i), & t = 2q+(2i+1) \text{ for some } i,\ 1\le i\le q,\\
g_1(z_i), & t = 4q+2i \text{ for some } i,\ 1\le i\le q,\\
g_2(z_i), & t = 4q+(2i+1) \text{ for some } i,\ 1\le i\le q.
\end{cases}
$$

Then the instance I of NONOGRAM obtained by our reduction from M is defined as follows:

- $h = 2n$, $w = 6q+2$. (Recall that h and w are height and width of I.)
- For each l, $1 \le l \le n$ (letting m_l be the lth element of M), define $r_{2l-1} = \langle f_1(m_l), f_2(m_l), f_3(m_l), f_4(m_l)\rangle$, and $r_{2l} = 0$.
- For each t, $1\le t\le w$, define $c_t = \langle\overbrace{1,\ldots,1}^{g(t)}\rangle$.

Since $h \times w = 2n\times(6q+2)$ is bounded by some polynomial in n, we can construct I in polynomial time.

Now we show every solution of I satisfies the requirements (R1) and (R2). First notice that for every even s, $r_s = 0$; hence, $p(s,-)$ must be 0. That is, every solution must satisfy (R1).

Next notice that $c_1 = c_w = \langle\overbrace{1,\ldots,1}^{n}\rangle$; hence, both $p(s,1)$ and $p(s,w)$ must be 1 for every odd s. Then for every l, $1\le l\le n$, $p(2l-1,1)=1$, $p(2l-1,w)=1$, and moreover, the difference between w and $r_{2l-1,1}+\cdots+r_{2l-1,4}$ is merely 4; thus, the possible solutions for the $(2l-1)$th row are (a), (b) and (c) of Figure 4. However, we will show below that (c) does not appear in any solution of I.

Let u be the number of occurences of (c) appeared in the solution. For each odd s, $\sum_{t=2q+2}^{4q+1} p(s,t)$ equals to $2q-1$ if the sth row in the solution is either (a) or (b), and it equals to $2q-2$ otherwise. Thus we have

$$
\sum_{t=2q+2}^{4q+1}\sum_{s=1}^{h} p(s,t) = 2nq - n - u.
$$

On the other hand,

$$
\sum_{t=2q+2}^{4q+1} g(t) = \sum_{j=1}^{q}(2n - N(y_j)) = 2nq - n.
$$

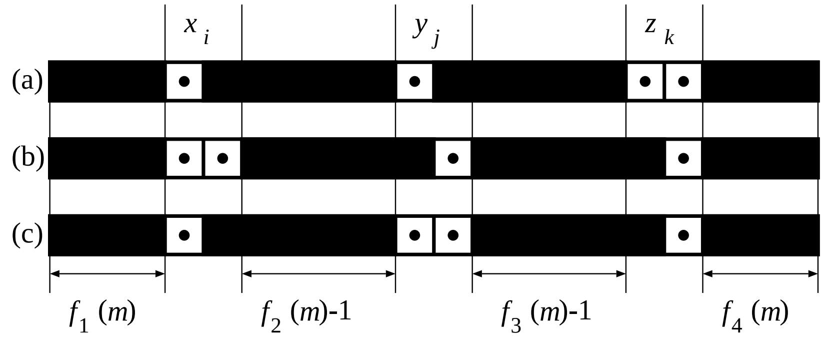

Figure 4: Possible solutions for each odd row.

These two equalities imply that $u = 0$ because every solution must satisfy $\sum_{t=2q+2}^{4q+1} \sum_{s=1}^{h} p(s,t) = \sum_{t=2q+2}^{4q+1} g(t)$. Therefore, each odd row of every solution is either (a) or (b). That is, every solution must satisfy (R2).

With this in mind, we show that M has a matching if and only if I has a solution. First, we show the way to construct a matching from a given solution of I. Suppose that we are given some solution of I, and consider any j, $1 \le j \le q$. Recall that we set c_{2j+2q} so that the solution contains $n-1$ 1's at the $(2j+2q)$th column; thus, in the solution, there must be unique l_j such that $p(2l_j - 1, 2j+2q) = 0$. Then we have $p(2l_j - 1, 2j+2q+1) = 1$. Hence, the solution has part (i) of Figure 5.

Let $m_{l_j} = \langle x_i, y_j, z_k \rangle$. Then for these i and k, we have

$$\begin{array}{ll} p(2l_j - 1, 2i) = 0, & p(2l_j - 1, 2i+1) = 1, \\ p(2l_j - 1, 2k+4q) = 0, & p(2l_j - 1, 2k+4q+1) = 0. \end{array}$$

Hence, the $(2l_j - 1)$th row contains parts (ii) and (iii) of Figure 5. That is, the $(2l_j - 1)$th row has the pattern (a) of Figure 4. Now define M' be the set $\{m_{l_1}, ..., m_{l_q}\}$.

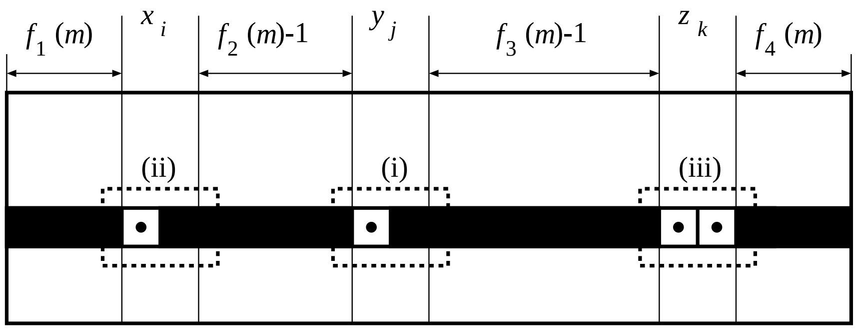

Figure 5: A part of the solution of I

We show that M' is a matching. Clearly, each y_j appears exactly once in m_{l_j} of M'. First note that we set c_{2i+1} so that the solution contains $g_2(x_i)$ $(= n - N(x_i) + 1)$ 1's at the $(2i+1)$th column; hence, the number of l such that $p(2l-1, 2i+1) = 1$ is $n - N(x_i) + 1$. On the other hand, note that for any l' such that $m_{l'} = \langle x, y, z \rangle$ and $x \ne x_i$, $p(2l'-1, 2i+1) = 1$, and that the number of such $m_{l'}$'s is $n - N(x_i)$. Thus, for each i, there must be unique l such that $p(2l-1, 2i+1) = 1$, which is l_j for some j; that is, each x_i appears exactly once in some m_{l_j} of

M'. Similarly, for each k, there must be unique l such that $p(2l-1, 2k+4q) = 0$, which is l_j for some j; that is, each z_k appears exactly once in some m_{l_j} of M'. Therefore, M' is a matching of M.

Conversely, for any matching M' of M, we can construct a solution of I in the following way. First, we assign 0 (or, $\boxed{\bullet}$) to the every even row. Then for every $(2l-1)$th row, $1 \leq l \leq n$, we assign the pattern (a) of Figure 4 if $m_l \in M'$, and the pattern (b) if $m_l \in M - M'$. It is easy to see that this assignment satisfies the constraints of I.

Finally, we need to show one-to-one correspondence between M's matchings and I's solutions. For this, it suffices to show that the above explained procedure yields unique maching from each solution of I. Let P_1 and P_2 be two different solutions of I. Possible difference between P_1 and P_2 is that P_1 and P_2 have different patterns at some odd row. That is, without losing generality, we may assume that P_1 has the pattern (a), and P_2 has the pattern (b) at some $(2l-1)$th row. Notice that the above procedure selects m_l for a maching if and only if the pattern (a) appears at the $(2l-1)$th row of a given solution. Thus, matching M_1' obtained from P_1 contains m_l while M_2' obtained from P_2 does not. Thus, M_1' differs from M_2'. ∎

Consider the procedure for obtaining a solution of I from a matching of M described in the above proof. Clearly it is polynomial time computable. Thus, we have the following corollary.

Collorary 3.2. There is a parsimonious reduction f from 3DM to NONOGRAM that has the following property: $(*)$ For any instance M for 3DM and any solution M' for M, some solution for $f(M)$ is polynomial time computable from M'.

4 NP-Completeness of ASP for NONOGRAM

From Corollary 3.2, ASP for NONOGRAM is NP-complete if we can show that ASP-3DM is NP-complete.

Theorem 4.1. ASP-3DM is NP-complete.

Proof. ASP-3DM is in NP, since we can check in polynomial time that no two elements of M'' agree in any coordinate when $M''(\neq M')$ is given.

Thus, our main task is to show that ASP-3DM is NP-hard. For this, we reduce 3DM to ASP-3DM. For any $\tilde{X} = \{x_1, x_2, \ldots, x_q\}$, $\tilde{Y} = \{y_1, y_2, \ldots, y_q\}$, and $\tilde{Z} = \{z_1, z_2, \ldots, z_q\}$, let $\tilde{M} \subseteq \tilde{X} \times \tilde{Y} \times \tilde{Z}$ be any instance of 3DM. We may assume without loss of generality that $\left|\tilde{M}\right| \geq q$. Here we construct disjoint sets X, Y and Z, with $|X| = |Y| = |Z|$, and a set $M \subseteq X \times Y \times Z$ such that M contains two matchings if and only if $\tilde{M}$ contains a matching.

First we set

$$X = \tilde{X} \cup \{x_1^+, \ldots, x_q^+\}, \quad Y = \tilde{Y} \cup \{y_1^+, \ldots, y_q^+\}, \quad Z = \tilde{Z} \cup \{z_1^+, \ldots, z_q^+\}$$

Note that M has to contain some matching M' for any $\tilde{X}, \tilde{Y}, \tilde{Z}$, and $\tilde{M}$. For this, we will define M as a superset of the following M':

$$M' = \{\langle x_i, y_i, z_i^+ \rangle : 1 \leq i \leq q\} \cup \{\langle x_i^+, y_i^+, z_i \rangle : 1 \leq i \leq q\}$$

Then for any $\tilde{X}, \tilde{Y}, \tilde{Z}$, and $\tilde{M}$, M contains a matching, namely M'.

Next, assume that $\tilde{M}$ contains a matching $\tilde{M}'$. In this case, M must include another matching M''. For this, we will define M as a superset of $\tilde{M}'$ and the following M_1:

$$M_1 = \{\langle x_i^+, y_{i+1}^+, z_i^+ \rangle : 1 \leq i < q\} \cup \{\langle x_q^+, y_q^+, z_1^+ \rangle\}$$

Then M contains a maching $M'' = \tilde{M}' \cup M_1$.

To summarize, we define $M = \tilde{M} \cup M' \cup M_1$. Clearly, M can be constructed in polynomial time from $\tilde{M}$, because M contains no more than $4|\tilde{M}|$ triples. Also as explained above, M always has a matching M', and it has a matching $\tilde{M}' \cup M_1$ if $\tilde{M}$ has a matching $\tilde{M}'$.

Now consider the case where $\tilde{M}$ does not contain any matching. From the structure of M' and M_1, any matching M^* for M cannot contain elements in both M' and M_1. Thus, M^* must contain either all elements of M' or all elements of M_1. M^* must be M' if $M' \subseteq M^*$. However, if $M_1 \subseteq M^*$, then M^* cannot be a matching because $\tilde{M}$ does not contain any matching. Thus, M^* must be M'. That is, if $\tilde{M}$ has no matching, then M has only one maching M'.

Therefore, we can show that M has a matching other than M' if and only if $\tilde{M}$ has a matching. This proves that 3DM is reducible to ASP-3DM. ∎

Finally, from Corollary 3.2 and Theorem 4.1, we conclude as follows.

Theorem 4.2. ASP for NONOGRAM is NP-complete.

5 Conclusion

In this paper, we show that ASP-3DM is NP-complete and there is a parsimonious reduction from 3DM to NONOGRAM with some additional property. This allows us to conclude that ASP for NONOGRAM is also NP-complete.

This is one approach for showing the NP-completeness of ASP. One interesting open question is to find some other way to show the NP-completeness of ASP, or show that this is only the way to show the NP-completeness of ASP. In more general, it is interesting if we can give some classification of NP-complete problems based on the difficulty of its ANOTHER SOLUTION PROBLEM.

Acknowledgement

This works has been done as a part of the first author's Bachelor thesis project. The authors would like to thank Professor Osamu Watanabe for supervising this work, and people in the Watanabe Kenkyushitsu and Professor Akihisa Tamura for helpful discussions. The authors also thank Mr. Hoong-Chuin Lau for helping us for preparing this paper.

References

[1] M.R.Garey and D.S.Johnson, *COMPUTER AND INTRACTABILITY : A Guide to the Theory of NP-completeness*, W.H.Freeman and Company (1983).

[2] N.Ishida, Game “NONOGRAM”(in Japanese), *Mathematical Seminar*, 10 (1994), pp.21-22.

[3] C.H.Papadimitriou, *Computational Complexity*, Addison-Wesley, (1994).