Theoretical Computer Science 530 (2014) 1–11

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

# Solving the maximum duo-preservation string mapping problem with linear programming

Wenbin Chen [a,b,c,1,*], Zhengzhang Chen [d,e], Nagiza F. Samatova [d,e], Lingxi Peng [a], Jianxiong Wang [a], Maobin Tang [a]

[a] Department of Computer Science, Guangzhou University, PR China
[b] Shanghai Key Laboratory of Intelligent Information Processing, Fudan University, PR China
[c] State Key Laboratory for Novel Software Technology, Nanjing University, PR China
[d] Computer Science Department, North Carolina State University, Raleigh, NC 27695, United States
[e] Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, United States

A R T I C L E I N F O

*Article history:*
Received 27 November 2011
Received in revised form 12 February 2014
Accepted 16 February 2014
Communicated by V.Th. Paschos

*Keywords:*
Approximation algorithm
Constrained maximum induced subgraph problem
Duo-preservation string mapping
Linear programming
Integer programming
Randomized rounding

A B S T R A C T

In this paper, we introduce the maximum duo-preservation string mapping problem (MPSM), which is complementary to the minimum common string partition problem (MCSP). When each letter occurs at most $k$ times in any input string, the version of MPSM is called $k$-MPSM. In order to design approximation algorithms for MPSM, we also introduce the constrained maximum induced subgraph problem (CMIS) and the constrained minimum induced subgraph (CNIS) problem.
We show that both CMIS and CNIS are NP-complete. We also study the approximation algorithms for the restricted version of CMIS, which is called $k$-CMIS ($k \geqslant 2$). Using Linear Programming method, we propose an approximation algorithm for 2-CMIS with approximation ratio 2 and an approximation algorithm for $k$-CMIS ($k \geqslant 3$) with approximation ratio $k^2$. Based on approximation algorithms for $k$-CMIS, we get approximation algorithms for $k$-MPSM with the same approximation ratio.


## 1. Introduction

The minimum common string partition problem (MCSP) has been well-investigated as a fundamental problem in computer science [8,12]. Given two finite length strings over the finite letter alphabet, MCSP is to partition strings into identical substrings with the minimum number of partitions. MCSP is also viewed as the problem of finding a letter-preserving bijective mapping $\pi$ from letters in one string $A$ to letters in the other string $B$ with the minimum number of breaks, where a *letter-preserving bijective mapping* $\pi$ means that each letter in $A$ is mapped into the same letter in $B$ and the mapping is bijective, and a break is a pair of consecutive letters in $A$ that are mapped by $\pi$ to non-consecutive letters in $B$ [12]. In a string, a pair of consecutive letters is called a *duo* [12].

As an example, let us assume that there is a letter-preserving bijective mapping $\pi$ (see Fig. 1.1) between two strings $A = abcab$ and $B = ababc$. From Fig. 1.1, we can see that $\pi$ has only one break: $ca$ is a duo of $A$, but $\pi(c)\pi(a)$ is not a duo

* Corresponding author at: Department of Computer Science, Guangzhou University, PR China.
*E-mail address:* cwb802@aliyun.com (W. Chen).

[1] The material in this paper was presented in part at the 2010 International Conference on Future Information Technology [9], Changsha, PR China, December 2010.

http://dx.doi.org/10.1016/j.tcs.2014.02.017
0304-3975/

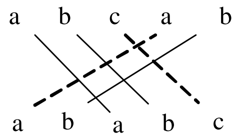


**Fig. 1.1.** A letter-preserving bijective mapping $\pi$ for two strings: $A = abcab$, $B = ababc$.

**Table 1**
The approximation ratio summary for $k$-MCSP.

| Paper | 2-MCSP | 3-MCSP | 4-MCSP | $k$-MCSP |
|---|---|---|---|---|
| [8] | 1.5 | | | |
| [12] | 1.1037 | 4 | | |
| [6] | 3 | | $\Omega(\log n)$[a] | $O(n^{0.69})$ |
| [18] | | | | $O(k^2)$ |
| [19] | | | | $4k$ |

[a] It is a lower bound.

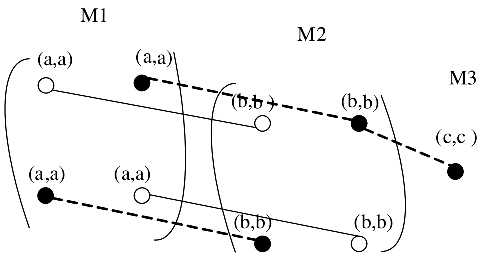


**Fig. 1.2.** Two strings $A = abcab$ and $B = ababc$ are transformed into a graph $G_{AB}$.

of $B$. However, the other three duos in $A$ ($ab$, $bc$, $ab$) are kept by $\pi$, each of which is called duo-preservation. So, the sum of the number of breaks and the number of duo-preservations is four, which is the length of any input string minus 1.

For a letter-preserving bijective mapping between two strings, on the one hand, the optimization goal can be to minimize the number of breaks that is known as the MCSP problem. On the other hand, the optimization goal can be to maximize the number of duo-preservations. We define the maximization version of the problem as the maximum duo-preservation string mapping problem (MPSM), i.e. the problem of finding a letter-preservation bijective mapping $\pi$ from one string to the other string with the maximum number of duo-preservations. When each letter occurs at most $k$ times in any input string, the version of MPSM is called $k$-MPSM. The MPSM problem is complementary to the MCSP problem as shown in Section 2. From this complementary relationship, it follows that MPSM is also NP-hard since the MCSP problem is NP-hard [12].

While the MCSP problem has been widely studied, to the best of our knowledge, the MPSM problem has not been addressed before. Specifically, various approximation algorithms have been proposed to solve the $k$-MCSP problem, a version of MCSP, where each letter appears at most $k$ times in any input string. These results are surveyed in Table 1.

Although there are approximation algorithms for MCSP, it is still required to design approximation algorithms for MPSM, because a pair of complementary NP-hard problems may have different approximation cases, i.e. an approximation algorithm for one problem sometimes cannot be used to approximate its complementary problem. For example, the minimum vertex cover problem and the maximum independent set problem are two well-known complementary problems in computer science [11]. For a given graph with $n$ vertices, the minimum vertex cover problem can be approximated within a ratio of 2, but the maximum independent set is NP-hard to approximate within a factor $n^{\delta}$, for some $\delta > 0$ [10,4,3]. Another pair of complementary problems is the Max-Satisfy problem and the Min-Unsatisfy problem [1,2]. Both the Max-Satisfy problem and the Min-Unsatisfy problem are NP-hard, but their approximation cases are also different. For a system of $m$ linear equations with $n$ variables over fractional numbers $\mathbf{Q}$, the Min-Unsatisfy problem can be approximated within a factor of $m + 1$, but the Max-Satisfy problem is NP-hard to approximate within a factor of $n^{\delta}$, for some $\delta > 0$ [1,2].

We notice that the MPSM problem can be transformed to a graph optimization problem. We use the example in Fig. 1.1 to explain it. For two strings $A = abcab$ and $B = ababc$ in Fig. 1.1, we can construct a graph $G_{AB}$ as follows (see Fig. 1.2). $G_{AB}$ has three parts $M_1$, $M_2$, and $M_3$. Part $M_1$ contains four $(a, a)$ nodes. Part $M_2$ contains four $(b, b)$ nodes, and part $M_3$ contains one $(c, c)$ node. In $G_{AB}$, there is an edge between $(a, a)$ node at the position $(1, 1)$ of $M_1$ and $(b, b)$ node at the position $(1, 1)$ of $M_2$, because the first $a$ and the first $b$ in $A$ form a duo $ab$ and the first $a$ and the first $b$ in $B$ also form a duo $ab$. Other edges are similarly constructed.

In the graph $G_{AB}$, the five black nodes are chosen from different rows and different columns in each $M_i$ part, respectively, because the subgraph induced by these five nodes has the maximum edge number of 3 (dashed lines in Fig. 1.2).