An efficient algorithm for the Closest String Problem

ICOMP/UFAM - Universidade Federal do Amazonas

Omar Latorre Vilca and Rosiane de Freitas

{omarlatorre,rosiane}@icomp.ufam.edu.br

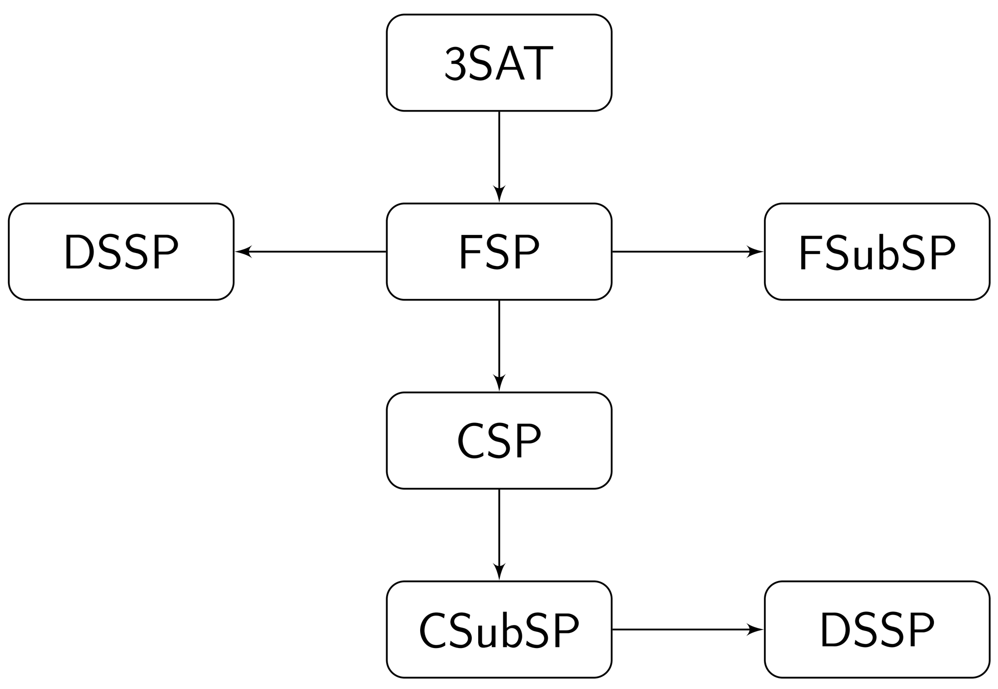
São Paulo School of Advanced Science on Algorithms, Combinatorics and Optimization

July 18 to 29, 2016 IME-USP.

Introduction

The Closest String problem (CSP) is a combinatorial optimization problem that receives as input a set of strings of the same length and seeks a string whose distance from the given strings is minimal. The distance from a solution to a given string is measured by the Hamming distance. The distance from the solution to the farthest input string is considered as the objective value of the solution, which is to be minimized. The CSP is NP-hard [Frances and Litman, 97].

NP-hard problems hierarchy for the CSP [Lanctot et al., 2003]



Theorem [O.L.Vilca]

Let a CSP instance with 3-sequences, which denotes 3-CSP, $S = \{s^i \in \Sigma^m, 1 \leq i \leq 3\}$ with alphabet $|\Sigma| > 2$, so the 3-CSP-A algorithm always finds an exact solution to 3-CSP [Liu et al., 2001] for Binary case.

Proof

This proof is by construction method, based on normalized instance [Gramm et al., 2001]

$$\phi(c_j)_{1\leq i\leq 3, 1\leq j\leq m} = \begin{cases} \alpha & \text{if } \sigma_1 \in c_j : max_{\sigma_i}|c_j^i = \sigma_1| \\ \beta & \text{else if } \sigma_2 \in c_j : max_{\sigma_i}|c_j^i \setminus \sigma_1 = \sigma_2| \\ \gamma & \text{else if } \sigma_3 \in c_j : max_{\sigma_i}|c_j^i \setminus \{\sigma_1, \sigma_2\} = \sigma_3| \end{cases}$$

After that, we reduce into five different cases

$$\begin{array}{r|l} \alpha\alpha\alpha & v_1^T \\ \beta\alpha\alpha & v_2^T \\ \alpha\beta\alpha & v_3^T \\ \alpha\alpha\beta & v_4^T \\ \alpha\beta\gamma & v_5^T \end{array}$$

For blocks of 2 and 3-length, with $\{i, j, k\} \in \{2, 3, 4\}$, we have

x	y	$d_H(x^T, y^T)$
$v_i v_j$	$\alpha\alpha$	1
$v_i v_i$	$\alpha\beta$	1
$v_2 v_5$	$\alpha\alpha$	1
$v_3 v_5$	$\alpha\beta$	1
$v_4 v_5$	$\alpha\gamma$	1
$v_5 v_5$	$\alpha\alpha$	2

x		y	$d_H(x^T, y^T)$
$v_i v_j v_k$	r	$\beta\alpha\alpha$	2
$v_i v_j v_k$	$-$	$\alpha\alpha\alpha$	1
$v_5 v_i v_j$	r	$\alpha\alpha\alpha$	2
$v_i v_5 v_j$	r	$\alpha\alpha\alpha$	2
$v_i v_j v_5$	r	$\alpha\alpha\alpha$	2
$v_5 v_5 v_5$	$-$	$\alpha\beta\gamma$	2

So in truth we are interested in the cases when the Hamming distance is equals to 1. Let l_i the number of times that v_i is repeted, $l_2 \leq l_3 \leq l_4$, then $l_3 = l_3 - l_2$, $l_4 = l_4 - (l_2 + l_3)$, after these calculations, one of them is greater or equal to zero. let $\rho_{i\alpha\beta} = \lfloor \frac{1}{2} l_i \rfloor$ for $2 \leq i \leq 4$

- If l_5 mod $3 > 0$ and $\{l_2, l_3, l_4\}$ mod $2 > 0$ then $\{\rho_{5\alpha\gamma}, \rho_{\alpha\beta}, \rho_{\alpha\alpha}\} = 1$
- If l_5 mod $2 > 0$ then $\rho_{5\alpha\beta} = 1$

Let t a string that represents an optimal solution of 3-CSP-A, with $1 \leq i \leq m$, we have:

$$\begin{array}{lll} s_i^1 = s_i^2 & & t_i = s_i^1 = s_i^2 \\ s_i^1 = s_i^2 \; s^2 \neq s^3 & \rho_{2\alpha\beta} > 0 & t_i = s_i^3 \\ s_i^1 \neq s_i^2 \; s_i^1 = s_i^3 & \rho_{3\alpha\beta} > 0 & t_i = s_i^2 \\ s_i^1 \neq s_i^2 \; s_i^1 \neq s_i^3 \; s_i^2 = s_i^3 & \rho_{4\alpha\beta} > 0 & t_i = s_i^1 \\ s_i^1 \neq s_i^2 \; s_i^2 \neq s_i^3 \; s_i^2 \neq s_i^3 & \rho_{5\alpha\beta} > 0 & t_i = s_i^2 \\ s_i^1 \neq s_i^2 \; s_i^2 \neq s_i^3 \; s_i^2 \neq s_i^3 & \rho_{5\alpha\gamma} > 0 & t_i = s_i^3 \\ s_i^1 \neq s_i^2 \; s_i^2 \neq s_i^3 \; s_i^2 \neq s_i^3 & \rho_{\alpha\beta} > 0 & t_i = s_i^2 \\ s_i^1 \neq s_i^2 \; s_i^2 \neq s_i^3 \; s_i^2 \neq s_i^3 & \rho_{\alpha\alpha} > 0 & t_i = s_i^1 \end{array}$$

Thus the theorem holds.

Algorithm

```
// Let v_i: number of times that {aaa, aab, aba, baa, abc} appears in the jth column
with 1 ≤ i ≤ 5 and 1 ≤ j ≤ m
if |S| = 3 then
    smallest ← getSmallest(v_2, v_3, v_4);
    v_k ← v_k − smallest; for k ∈ {2, 3, 4}
    for i=1 to m do
        if s_i^1 = s_i^2 then
            t_i ← s_i^1;
        else if s_i^2 = s_i^3 then
            t_i ← s_i^2;
            else if s_i^1 = s_i^3 then
                t_i ← s_i^1;
                // when all the characters are different, case: abc
                else if v_4 > 0 then
                    t_i ← s_i^1; v_4 ← v_4 − 1;
                    else if v_3 > 0 then
                        t_i ← s_i^2; v_3 ← v_3 − 1;
                    else if v_2 > 0 then
                        t_i ← s_i^3; v_2 ← v_2 − 1;
                    else
                        t_i ← s_i^j; //where j ∈ {1, 2, 3} one of them each time
    end
```

Computational results

Linear time algorithm 3-CSP-A, for 3-sequences.

Instance	2 Characters			4 Characters			20 Characters		
m	Seed	Val	Time	Seed	Val	Time	Seed	Val	Time
100	542319	27	0.01	791034	48	0.05	691425	63	0.14
200	7121	52	0.02	52151	91	0.14	351554	124	0.49
300	64874	77	0.01	68724	132	0.17	98121	183	0.36
400	6487	112	0.08	193185	180	0.47	246754	244	0.78
500	94115	124	0.12	15364	215	0.96	658745	311	0.55
600	5419	153	0.10	5419	260	0.41	154525	373	0.42
700	43212	180	0.13	524514	306	0.85	487754	437	0.74
800	2454	215	0.21	55364	354	0.84	754812	494	0.63
900	645387	234	0.35	6487	389	0.29	722451	557	0.85
1000	94315	260	0.27	153364	444	0.24	34567	616	0.90
2000	264554	508	0.78	6487	881	1.12	65743	1219	1.13
3000	68174	765	2.01	2454	1305	2.52	4432	1857	1.52
4000	4212	1003	3.19	53214	1760	3.70	543	2484	3.77
5000	722312	1278	5.41	934145	2188	6.12	344	3099	5.96

Concluding remark

We proposed an exact algorithm for the special case of CSP with 3-sequences and alphabet size $|\Sigma| > 2$ and gave the corresponding theoretical analysis.

Others names for the CSP.

Year	Name	References
1997	Minimum Radius Problem	[Frances and Litman, 97]
1999	Hamming Center Problem	[Gasieniec et al. 99]
2001	Center String Problem	[Gramm et al., 2001]
2003	Consensus String Problem	[Lanctot et al., 2003]
2008	Motif Finding Problem	[Gomes et al., 2008]

References

M. Frances and A. Litman.
On covering problems of codes.
Theory of Computing Systems, 30(2):113–119, 1997.

L. Gasieniec, J. Jansson, and A. Lingas.
Efficient approximation algorithms for the hamming center problem.
pages 905–906, 1999.

J. Gramm, R. Niedermeier, and P. Rossmanith.
Exact solutions for closest string and related problems.
In *Proceedings of the 12th International Symposium on Algorithms and Computation*, ISAAC '01, pages 441–453, 2001.

K. Lanctot, M. Li, B. Ma, S. Wang, and L. Zhang.
Distinguishing string selection problems.
Information and Computation, 185(1):41–55, 2003.

X. Liu, S. Liu, Z. Hao, and H. Mauch.
Exact algorithm and heuristic for the closest string problem.
Computers & Operations Research, 38(11):1513–1520, 2011.

C. N. Meneses, Z. Lu, C. A. S. Oliveira, and P. M. Pardalos.
Optimal solutions for the closest-string problem via integer programming.
INFORMS Journal on Computing, 16:2004, 2004.