Local Search for String Problems: Brute Force is Essentially Optimal

Jiong Guo, Danny Hermelin, Christian Komusiewicz

Universität des Saarlandes, Ben-Gurion University,
TU Berlin & Université de Nantes

CPM 2013

A local search problem Π consists of

- a set $\mathcal{I}$ of **instances**,
- for each instance $I \in \mathcal{I}$ a set $F(I)$ of **feasible solutions**,
- an **objective function** $f : F(I) \to \mathbb{Z}$, and
- for each solution $s \in F(I)$ a **neighborhood** $N(s, I) \subseteq F(I)$.

Goal: Find **locally optimal** solution $s \in F(I)$ such that for all $s' \in N(s, I)$, $f(s) \leq f(s')$.

Generic local search algorithm:

```
s := "some element of F(I)"
while s is not locally optimal :
    find s' ∈ N(s) with f(s) < f(s')
    s := s'
return s
```

Examples: k-Means, Simplex

Advantages of local search:

- fast in practice
- local optima often close to global optimum
- generic algorithm scheme $\rightsquigarrow$ easy to implement

Tuning local search algorithms:

- faster convergence to local optimum
- changing type/size of neighborhood N

Theoretical analysis:

- number of iterations $\rightsquigarrow$
  PLS-completeness [Johnson, Papadimitriou & Yannakakis, JCSS 1988],
  Smoothed Analysis [Spielman & Teng, JACM 2004]
- efficiently searching the neighborhood N $\rightsquigarrow$
  Parameterized Local Search

Example:

LS-TSP
Input: A set V of n cities, a distance matrix $d : V \times V \to \mathbb{Z}$, a tour T, and an integer k.
Question: Can we obtain a **shorter** tour T' by replacing $\leq k$ arcs of T?

Obvious: Solvable in $n^{O(k)}$ time

Aim: Improve running time to $f(k) \cdot \text{poly}(n)$

But: LS-TSP is W[1]-hard parameterized by k [Marx, ORL 2008]

$\leadsto$ probably no $f(k) \cdot \text{poly}(n)$-time algorithm

Can we avoid brute-force $n^{O(k)}$ algorithms for searching the neighborhood?

General: Parameterized local search problems usually W[1]-hard with respect to the k-change neighborhood
[Fellows et al., JCSS, 2012]

Positive results for...

... restricted inputs:

- **LS-Feedback Arc Set** in tournaments [Fomin et al., AAAI 2010]
- **LS-TSP** on planar graphs [Guo et al., Algorithmica, to appear],

... other neighborhood types:

- **LS-TSP** with $\leq k$ swaps in an m-bounded range
  [Guo et al., Algorithmica, to appear]
- **LS-TSP** with maximum shift k [Balas, AOR 1999]

Can we avoid brute-force $n^{O(k)}$ algorithms for searching the neighborhood?

General: Parameterized local search problems usually W[1]-hard with respect to the k-change neighborhood
[Fellows et al., JCSS, 2012]

Positive results for...

... restricted inputs:

- **LS-Feedback Arc Set** in tournaments [Fomin et al., AAAI 2010]
- **LS-TSP** on planar graphs [Guo et al., Algorithmica, to appear],

... other neighborhood types:

- **LS-TSP** with $\leq k$ swaps in an m-bounded range [Guo et al., Algorithmica, to appear]
- **LS-TSP** with maximum shift k [Balas, AOR 1999]

How easy are **string** problems with respect to parameterized local search?

Can we avoid brute-force $n^{O(k)}$ algorithms for searching the neighborhood?

General: Parameterized local search problems usually W[1]-hard with respect to the k-change neighborhood
[Fellows et al., JCSS, 2012]

Positive results for...

... restricted inputs:

- **LS-Feedback Arc Set** in tournaments [Fomin et al., AAAI 2010]
- **LS-TSP** on planar graphs [Guo et al., Algorithmica, to appear],

... other neighborhood types:

- **LS-TSP** with $\leq k$ swaps in an m-bounded range [Guo et al., Algorithmica, to appear]
- **LS-TSP** with maximum shift k [Balas, AOR 1999]

How easy are **string** problems with respect to parameterized local search for the **Hamming distance neighborhood**?

Closest String
Input: Strings $S_1, \ldots, S_m$ of length n over alphabet Σ, an integer d.
Question: Is there a string S of length n with Hamming distance $\leq d$ to each $S_i \in \mathcal{S}$?

Known:

- NP-hard [Frances & Litmann, TOCS 1999]
- $O(d^{d+1} \cdot m + nm)$-time algorithm
  [Gramm, Niedermeier & Rossmanith, Algorithmica 2003]
- $f(m) \cdot \text{poly}(n)$-time algorithm
  [Gramm, Niedermeier & Rossmanith, Algorithmica 2003]

LS-Closest String
Input: Strings $S_1, \ldots, S_m$ of length n over alphabet Σ, a string S with Hamming distance $\leq d$ to each $S_i \in \mathcal{S}$, and $k \geq 0$.
Question: Is there a string S' of length n with Hamming distance

- $\leq k$ to S, and
- $< d$ to each $S_i \in \mathcal{S}$?

$$S_1 := \text{BAAAA}$$
$$S_2 := \text{CCACB}$$
$$S_3 := \text{ABBAC}$$

LS-Closest String

Input: Strings $S_1, \ldots, S_m$ of length n over alphabet Σ, a string S with Hamming distance $\leq d$ to each $S_i \in \mathcal{S}$, and $k \geq 0$.

Question: Is there a string S' of length n with Hamming distance

- $\leq k$ to S, and
- $< d$ to each $S_i \in \mathcal{S}$?

$S_1 :=$ BAAAA	$d_H(S, S_1) = 4$
$S_2 :=$ CCACB	$d_H(S, S_2) = 4$
$S_3 :=$ ABBAC	$d_H(S, S_3) = 3$
$S :=$ BBBBB	$k = 2$

LS-Closest String

Input: Strings $S_1, \ldots, S_m$ of length n over alphabet Σ, a string S with Hamming distance $\leq d$ to each $S_i \in \mathcal{S}$, and $k \geq 0$.

Question: Is there a string S' of length n with Hamming distance

- $\leq k$ to S, and
- $< d$ to each $S_i \in \mathcal{S}$?

$$
\begin{array}{lll}
S_1 := \text{BAAAA} & d_H(S, S_1) = 4 & d_H(S', S_1) = 2 \\
S_2 := \text{CCACB} & d_H(S, S_2) = 4 & d_H(S', S_2) = 3 \\
S_3 := \text{ABBAC} & d_H(S, S_3) = 3 & d_H(S', S_3) = 3 \\
\hline
S := \text{BBBBB} & k = 2 & d_H(S', S) = 2 \\
S' := \text{BBAAB} & &
\end{array}
$$

LS-Closest String
Input: Strings $S_1, \ldots, S_m$ of length n over alphabet Σ, a string S with Hamming distance $\leq d$ to each $S_i \in \mathcal{S}$, and $k \geq 0$.
Question: Is there a string S' of length n with Hamming distance

- $\leq k$ to S, and
- $< d$ to each $S_i \in \mathcal{S}$?

$S_1 :=$ BAAAA	$d_H(S, S_1) = 4$	$d_H(S', S_1) = 2$
$S_2 :=$ CCACB	$d_H(S, S_2) = 4$	$d_H(S', S_2) = 3$
$S_3 :=$ ABBAC	$d_H(S, S_3) = 3$	$d_H(S', S_3) = 3$
$S :=$ BBBBB	$k = 2$	$d_H(S', S) = 2$
$S' :=$ BBAAB		

Proposition: **LS-Closest String** can be solved in $d^k \cdot \text{poly}(n, m)$ time.

Theorem: **LS-Closest String** is NP-hard and W[2]-hard for the radius k of the Hamming neighborhood.

Proof by **parameterized reduction** from

Multicolored Hitting Set
Input: A hypergraph $G = (V, \mathcal{E})$ and vertex-coloring $c : V \to \{1, \ldots, k\}$.
Question: Is there a size-k set $V' \subseteq V$ such that

- $V' \cap E \neq \emptyset$ for all $E \in \mathcal{E}$, and
- V' is colorful?

Known: Multicolored Hitting Set parameterized by k is W[2]-hard

Parameterized Reduction:

$$(I, k) \quad \xrightarrow{\mathcal{A}} \quad (I', k')$$

(I, k) is a yes-instance $\quad \Leftrightarrow \quad$ (I', k') is a yes-instance

- $\mathcal{A}$ runs in $f(k) \cdot \text{poly}(|I|)$ time
- $k' \leq g(k)$

A parameterized reduction from a W[t]-hard problem L to a problem L' shows $W[t]$-hardness of L'

$V = \{1, 2, 3, 4, 5, 6\}$

$\mathcal{E} = \{E_1, E_2, E_3, E_4\}$

$E_1 = \{1, 2, 3\}$ $E_2 = \{4, 6\}$ $E_3 = \{1, 6\}$ $E_4 = \{3, 5\}$

	1	2	3	4	5	6
S_{E_1}	R	B	R	X	X	X
S_{E_2}	X	X	X	O	X	B
S_{E_3}	R	X	X	X	X	B
S_{E_4}	X	X	R	X	O	X

$V = \{1, 2, 3, 4, 5, 6\}$

$\mathcal{E} = \{E_1, E_2, E_3, E_4\}$

$E_1 = \{1, 2, 3\}$ $E_2 = \{4, 6\}$ $E_3 = \{1, 6\}$ $E_4 = \{3, 5\}$

	1	2	3	4	5	6
S_{E_1}	R	B	R	X	X	X
S_{E_2}	X	X	X	O	X	B
S_{E_3}	R	X	X	X	X	B
S_{E_4}	X	X	R	X	O	X
S_R	R	R	R	R	R	R
S_B	B	B	B	B	B	B
S_O	O	O	O	O	O	O

$V = \{1, 2, 3, 4, 5, 6\}$

$\mathcal{E} = \{E_1, E_2, E_3, E_4\}$

$E_1 = \{1, 2, 3\}$ $E_2 = \{4, 6\}$ $E_3 = \{1, 6\}$ $E_4 = \{3, 5\}$

	1	2	3	4	5	6	
S_{E_1}	R	B	R	X	X	X	$\|V\|$
S_{E_2}	X	X	X	O	X	B	$\|V\|$
S_{E_3}	R	X	X	X	X	B	$\|V\|$
S_{E_4}	X	X	R	X	O	X	$\|V\|$
S_R	R	R	R	R	R	R	$\|V\|$
S_B	B	B	B	B	B	B	$\|V\|$
S_O	O	O	O	O	O	O	$\|V\|$
S	*	*	*	*	*	*	

$V = \{1, 2, 3, 4, 5, 6\}$

$\mathcal{E} = \{E_1, E_2, E_3, E_4\}$

$E_1 = \{1, 2, 3\}$ $E_2 = \{4, 6\}$ $E_3 = \{1, 6\}$ $E_4 = \{3, 5\}$

	1	2	3	4	5	6	
S_{E_1}	R	B	R	X	X	X	$\|V\|$
S_{E_2}	X	X	X	O	X	B	$\|V\|$
S_{E_3}	R	X	X	X	X	B	$\|V\|$
S_{E_4}	X	X	R	X	O	X	$\|V\|$
S_R	R	R	R	R	R	R	$\|V\|$
S_B	B	B	B	B	B	B	$\|V\|$
S_O	O	O	O	O	O	O	$\|V\|$
S	*	*	*	*	*	*	
S'	R	*	*	*	O	B	

For binary alphabet:

S_{E_1}	1	1	1	0	0	0	1				$\lvert V\rvert - k$
S_{E_2}	0	0	1	0	1	0					$\lvert V\rvert - k$
S_{E_3}	1	0	0	1	0	0					$\lvert V\rvert - k$
S_{E_4}	0	1	0	0	0	1					$\lvert V\rvert - k$
$S_{\{R,B,O\}}$	1	1	1	1	1	1					$\lvert V\rvert$
$S_{\{B,O\}}$	0	0	1	1	1	1					$\lvert V\rvert - 1$
$S_{\{R,O\}}$	1	1	0	0	1	1					$\lvert V\rvert - 1$
$S_{\{R,B\}}$	1	1	1	1	0	0					$\lvert V\rvert - 1$
$S_{\{R\}}$	1	1	0	0	0	0					$\lvert V\rvert - 2$
$S_{\{B\}}$	0	0	1	1	0	0					$\lvert V\rvert - 2$
$S_{\{O\}}$	0	0	0	0	1	1				1	$\lvert V\rvert - 2$
S	0	0	0	0	0	0	0	$\cdots$	$\cdots$	0	
S'	1	0	1	0	0	1	0	$\cdots$	$\cdots$	0	

LS-Closest String:

- NP-hard
- W[2]-hard for radius k of the Hamming neighborhood even in case $|\Sigma| = 2$
- no $n^{o(k)}$-time algorithm
  (assuming the Exponential Time Hypothesis)
- solvable in $d^k \cdot \text{poly}(n, m)$ time

Further results:

W[1]-hardness for radius k of the Hamming neighborhood for local search variants of

- **Longest Common Subsequence**
- **Shortest Common Supersequence**
- **Shortest Common Superstring**

LS-Longest Common Subsequence
Input: Strings $T_1, \ldots, T_m$ over alphabet Σ, a string S that is a subsequence of each T_i, and integer k.
Question: Is there a letter $\sigma \in \Sigma$ and a string $\widetilde{S}$ of length $|S|$ such that $\widetilde{S}\sigma$ is a subsequence of each string in τ and $d_H(\widetilde{S}, S) \leq k$?

$S_1 :=$ BCBAAACBA
$S_2 :=$ BCCACBAC
$S_3 :=$ CABBAC

LS-Longest Common Subsequence

Input: Strings $T_1, \ldots, T_m$ over alphabet Σ, a string S that is a subsequence of each T_i, and integer k.

Question: Is there a letter $\sigma \in \Sigma$ and a string $\widetilde{S}$ of length $|S|$ such that $\widetilde{S}\sigma$ is a subsequence of each string in τ and $d_H(\widetilde{S}, S) \leq k$?

$S_1 :=$ BCBAAACBA

$S_2 :=$ BCCACBAC

$S_3 :=$ CABBAC

$S :=$ ABA, $k = 1$

LS-Longest Common Subsequence

Input: Strings $T_1, \ldots, T_m$ over alphabet Σ, a string S that is a subsequence of each T_i, and integer k.

Question: Is there a letter $\sigma \in \Sigma$ and a string $\widetilde{S}$ of length $|S|$ such that $\widetilde{S}\sigma$ is a subsequence of each string in τ and $d_H(\widetilde{S}, S) \leq k$?

$S_1 :=$ BCBAAACBA
$S_2 :=$ BCCACBAC
$S_3 :=$ CABBAC

$S :=$ ABA, $k = 1$
$\widetilde{S}\sigma :=$ BBAC

LS-Longest Common Subsequence
Input: Strings $T_1, \ldots, T_m$ over alphabet Σ, a string S that is a subsequence of each T_i, and integer k.
Question: Is there a letter $\sigma \in \Sigma$ and a string $\widetilde{S}$ of length $|S|$ such that $\widetilde{S}\sigma$ is a subsequence of each string in τ and $d_H(\widetilde{S}, S) \leq k$?

$S_1 :=$ BCBAAACBA
$S_2 :=$ BCCACBAC
$S_3 :=$ CABBAC

$S :=$ ABA, $k = 1$
$\widetilde{S}_\sigma :=$ BBAC

Theorem: **LS-Longest Common Subsequence** with $|\Sigma| = O(1)$ is W[1]-hard for the radius k of the Hamming neighborhood.

So far: only negative results

Possible directions:

- other problems: **Multiple Sequence Alignment**, ...
- other neighborhood types: "shift"-based distances, insertions/deletions at arbitrary solution positions