Finding Consensus Strings With Small Length Difference Between Input and Solution Strings

Markus L. Schmid

Trier University, Germany

MFCS 2015

The Closest String Problem

Closest String (CloseStr)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size m, $d \in \mathbb{N}$.
Question: $\exists$ string s of size m with $\mathsf{d_H}(s, s_i) \leq d$?

The Closest String Problem

Closest String (CloseStr)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size m, $d \in \mathbb{N}$.
Question: $\exists$ string s of size m with $\mathsf{d_H}(s, s_i) \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$d = 2$.

s_1	c	b	c	a	b	a	a
s_2	c	b	c	a	b	c	b
s_3	a	b	c	c	c	c	a
s_4	c	c	c	a	b	c	a
s_5	c	b	c	a	a	c	a
s_6	c	b	c	a	a	c	a
s_7	a	b	b	a	b	a	a
s_8	b	b	c	a	a	c	a

The Closest String Problem

CLOSEST STRING (CLOSESTR)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size m, $d \in \mathbb{N}$.
Question: $\exists$ string s of size m with $\mathsf{d_H}(s, s_i) \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$d = 2$.
Solution string: $s =$ abcabca.

s_1	c	b	c	a	b	a	a
s_2	c	b	c	a	b	c	b
s_3	a	b	c	c	c	c	a
s_4	c	c	c	a	b	c	a
s_5	c	b	c	a	a	c	a
s_6	c	b	c	a	a	c	a
s_7	a	b	b	a	b	a	a
s_8	b	b	c	a	a	c	a
s	a	b	c	a	b	c	a

The Closest String Problem

Closest String (CloseStr)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size m, $d \in \mathbb{N}$.
Question: $\exists$ string s of size m with $\mathsf{d_H}(s, s_i) \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$d = 2$.
Solution string: $s = \texttt{abcabca}$.

s_1	c	b	c	a	b	a	a
s_2	c	b	c	a	b	c	b
s_3	a	b	c	c	c	c	a
s_4	c	c	c	a	b	c	a
s_5	c	b	c	a	a	c	a
s_6	c	b	c	a	a	c	a
s_7	a	b	b	a	b	a	a
s_8	b	b	c	a	a	c	a
s	a	b	c	a	b	c	a

The Closest Substring Problem

Closest Substring (CloseSubstr)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.
Question: $\exists$ string s of size m such that every s_i has a size-m substring s'_i with $\mathsf{d_H}(s, s'_i) \leq d$?

The Closest Substring Problem

Closest Substring (CloseSubstr)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.
Question: $\exists$ string s of size m such that every s_i has a size-m substring s'_i with $\mathsf{d_H}(s, s'_i) \le d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 1$.

s_1	c	b	c	a	b	a	a
s_2	c	b	c	a	b	c	b
s_3	a	b	c	c	c	c	a
s_4	c	c	c	a	b	c	a
s_5	c	b	c	a	a	c	a
s_6	c	b	c	a	a	c	a
s_7	a	b	b	a	b	a	a
s_8	b	b	c	a	a	c	a

The Closest Substring Problem

Closest Substring (CloseSubstr)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.
Question: $\exists$ string s of size m such that every s_i has a size-m substring s'_i with $\mathsf{d_H}(s, s'_i) \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 1$.

s_1	c	b	c	**a**	**b**	**a**	**a**
s_2	c	b	c	**a**	**b**	**c**	**b**
s_3	**a**	**b**	**c**	**c**	c	c	a
s_4	c	c	c	**a**	**b**	**c**	**a**
s_5	c	b	c	**a**	**a**	**c**	**a**
s_6	**c**	**b**	**c**	**a**	a	c	a
s_7	a	b	b	**a**	**b**	**a**	**a**
s_8	**b**	**b**	**c**	**a**	a	c	a

The Closest Substring Problem

Closest Substring (CloseSubstr)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.
Question: $\exists$ string s of size m such that every s_i has a size-m substring s_i' with $\mathsf{d_H}(s, s_i') \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 1$.
Solution string: $s = \texttt{abca}$.

s_1	c	b	c	a	b	a	a
s_2	c	b	c	a	b	c	b
s_3	a	b	c	c	c	c	a
s_4	c	c	c	a	b	c	a
s_5	c	b	c	a	a	c	a
s_6	c	b	c	a	a	c	a
s_7	a	b	b	a	b	a	a
s_8	b	b	c	a	a	c	a

The Closest Substring Problem

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 1$.

c	b	c	a	b	a	a
c	b	c	a	b	c	b
a	b	c	c	c	c	a
c	c	c	a	b	c	a
c	b	c	a	a	c	a
c	b	c	a	a	c	a
a	b	b	a	b	a	a
b	b	c	a	a	c	a

The Closest Substring Problem

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 1$.

c	b	c	a	b	a	a	-	-	-
c	b	c	a	b	c	b	-	-	-
-	-	-	a	b	c	c	c	c	a
c	c	c	a	b	c	a	-	-	-
c	b	c	a	a	c	a	-	-	-
-	-	-	c	b	c	a	a	c	a
a	b	b	a	b	a	a	-	-	-
-	-	-	b	b	c	a	a	c	a

The Closest Substring Problem

Example

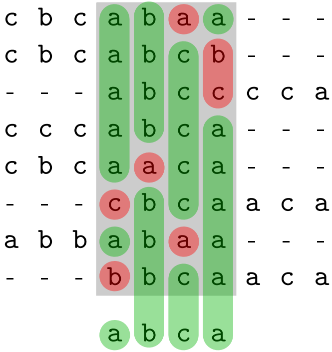
Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 1$.

c	b	c	a	b	a	a	-	-	-
c	b	c	a	b	c	b	-	-	-
-	-	-	a	b	c	c	c	c	a
c	c	c	a	b	c	a	-	-	-
c	b	c	a	a	c	a	-	-	-
-	-	-	c	b	c	a	a	c	a
a	b	b	a	b	a	a	-	-	-
-	-	-	b	b	c	a	a	c	a

The Closest Substring Problem

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 1$.



The Consensus Patterns Problem

Consensus Patterns (ConsPat)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.
Question: $\exists$ string s of size m such that every s_i has a size-m substring s'_i with $\sum_{i=1}^{k} \mathsf{d_H}(s, s'_i) \leq d$?

The Consensus Patterns Problem

Consensus Patterns (ConsPat)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.
Question: $\exists$ string s of size m such that every s_i has a size-m substring s_i' with $\sum_{i=1}^{k} \mathsf{d_H}(s, s_i') \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 5$.

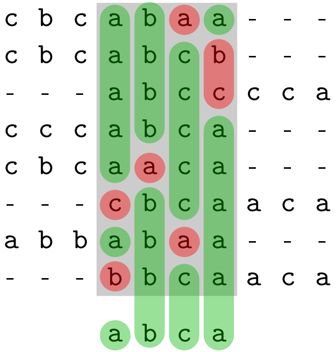
The Consensus Patterns Problem

CONSENSUS PATTERNS (CONSPAT)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.
Question: $\exists$ string s of size m such that every s_i has a size-m substring s'_i with $\sum_{i=1}^{k} \mathsf{d_H}(s, s'_i) \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 5$.



The Consensus Patterns Problem

CONSENSUS PATTERNS (CONSPAT)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.
Question: $\exists$ string s of size m such that every s_i has a size-m substring s'_i with $\sum_{i=1}^{k} \mathsf{d_H}(s, s'_i) \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 5$.

-	-	-	c	b	c	a	b	a	a
-	-	-	c	b	c	a	b	c	b
a	b	c	c	c	c	a	-	-	-
-	-	-	c	c	c	a	b	c	a
-	-	-	c	b	c	a	a	c	a
-	-	-	c	b	c	a	a	c	a
-	-	-	a	b	b	a	b	a	a
-	-	-	b	b	c	a	a	c	a

The Consensus Patterns Problem

CONSENSUS PATTERNS (CONSPAT)

Instance: Strings $s_1, s_2, \ldots, s_k$ of size at most ℓ, $\ell \in \mathbb{N}$, $m, d \in \mathbb{N}$.

Question: $\exists$ string s of size m such that every s_i has a size-m substring s'_i with $\sum_{i=1}^{k} \mathsf{d_H}(s, s'_i) \leq d$?

Example

Input strings: $s_1, s_2, \ldots, s_8$,
$m = 4$,
$d = 5$.
Solution string: $s =$ `cbca`.

-	-	-	c	b	c	a	b	a	a
-	-	-	c	b	c	a	b	c	b
a	b	c	c	c	c	a	-	-	-
-	-	-	c	c	c	a	b	c	a
-	-	-	c	b	c	a	a	c	a
-	-	-	c	b	c	a	a	c	a
-	-	-	a	b	b	a	b	a	a
-	-	-	b	b	c	a	a	c	a
			c	b	c	a			

Parameterised Complexity

Parameterised problem K:
instances are of the form (x, k), where k is the *parameter*

Parameterised Complexity

Parameterised problem K:
instances are of the form (x, k), where k is the *parameter*

- Tractability:
  - K *fixed-parameter tractable* ($K \in \text{FPT}$) $\iff$
    K can be solved in $\mathcal{O}(f(k) \times p(|x|))$ (recursive f and polynomial p).

Parameterised Complexity

Parameterised problem K:
instances are of the form (x, k), where k is the *parameter*

- Tractability:
  - K *fixed-parameter tractable* ($K \in \mathrm{FPT}$) $\iff$
    K can be solved in $\mathcal{O}(f(k) \times p(|x|))$ (recursive f and polynomial p).
- Hardness:
  - K is NP-hard if $k \leq c$ for constant $c \Rightarrow$
    $K \notin \mathrm{FPT}$ (unless $\mathrm{P} = \mathrm{NP}$).

Parameterised Complexity

Parameterised problem K:
instances are of the form (x, k), where k is the *parameter*

- Tractability:
  - K *fixed-parameter tractable* ($K \in \mathrm{FPT}$) $\iff$
    K can be solved in $\mathcal{O}(f(k) \times p(|x|))$ (recursive f and polynomial p).
- Hardness:
  - K is NP-hard if $k \leq c$ for constant $c \Rightarrow$
    $K \notin \mathrm{FPT}$ (unless $\mathrm{P} = \mathrm{NP}$).
  - K is $W[1]$-hard $\Rightarrow$
    $K \notin \mathrm{FPT}$ (under complexity theoretical assumptions).

Known Results - Closest String/Substring Problems

Theorem (Frances, Litman (1997))

CLOSESTR, CLOSESUBSTR *are* NP-*complete, even for binary alphabets.*

Known Results - Closest String/Substring Problems

Theorem (Frances, Litman (1997))

CLOSESTR, CLOSESUBSTR *are* NP-*complete, even for binary alphabets.*

CLOSESTR
(Evans et al. (2003), Gramm et al. (2003))

- ℓ para. $\Rightarrow$ FPT.
- k para. $\Rightarrow$ FPT.
- d para. $\Rightarrow$ FPT.

Known Results - Closest String/Substring Problems

Theorem (Frances, Litman (1997))

CloseStr, CloseSubstr *are* NP-*complete, even for binary alphabets.*

CloseStr
(Evans et al. (2003), Gramm et al. (2003))

- ℓ para. $\Rightarrow$ FPT.
- k para. $\Rightarrow$ FPT.
- d para. $\Rightarrow$ FPT.

CloseSubstr
(Evans et al. (2003), Fellows et al. (2003), Marx et al. (2008))

- ℓ parameter $\Rightarrow$ FPT.
- m, $|\Sigma|$ parameters $\Rightarrow$ FPT.
- k, m, d parameter $\Rightarrow$ $W[1]$-hard.
- k, d parameter, $|\Sigma| = 2 \Rightarrow W[1]$-hard.

Known Results - Consensus Patterns

Theorem (Li et al. (2002))

CONSPAT *is* NP-*complete, even for binary alphabets.*

Known Results - Consensus Patterns

Theorem (Li et al. (2002))

CONSPAT *is* NP-*complete, even for binary alphabets.*

CONSPAT
(Fellows et al. (2003), Marx et al. (2008))

- m, $|\Sigma|$ parameter $\Rightarrow$ FPT.
- d, $|\Sigma|$ parameters $\Rightarrow$ FPT.
- k, m, d parameter $\Rightarrow$ $W[1]$-hard.
- k parameter, $|\Sigma| = 2$ $\Rightarrow$ $W[1]$-hard.

The Parameter $(\ell - m)$

New parameter:
Length difference $(\ell - m)$ between input and solution strings.

The Parameter $(\ell - m)$

New parameter:
Length difference $(\ell - m)$ between input and solution strings.

Motivation:
$(\ell - m) = 0$ for CloseSubstr $\Rightarrow$ CloseStr.
$(\ell - m) = 0$ for ConsPat $\Rightarrow$ trivial.

The Parameter $(\ell - m)$

New parameter:
Length difference $(\ell - m)$ between input and solution strings.

Motivation:
$(\ell - m) = 0$ for CLOSESUBSTR $\Rightarrow$ CLOSESTR.
$(\ell - m) = 0$ for CONSPAT $\Rightarrow$ trivial.

Natural Question

Can we get tractability by bounding $(\ell - m)$/parametrising by $(\ell - m)$?

CloseSubstr Parameterised by $(\ell - m)$

k	m	d	$\|\Sigma\|$	$(\ell - m)$	Results
–	–	–	2	0	NP-hard
p	–	–	–	**p**	FPT
–	**p**	–	–	**p**	FPT
–	–	**p**	–	**p**	FPT

ConsPat Parameterised by $(\ell - m)$

- If $(\ell - m) = 0$, then ConsPat is trivial.

ConsPat Parameterised by $(\ell - m)$

- If $(\ell - m) = 0$, then ConsPat is trivial.
- What if $(\ell - m) \leq c$, for constant $c \geq 1$?

ConsPat Parameterised by $(\ell - m)$

- If $(\ell - m) = 0$, then ConsPat is trivial.
- What if $(\ell - m) \leq c$, for constant $c \geq 1$?
- What if $(\ell - m)$ is a parameter?

ConsPat Parameterised by $(\ell - m)$

- If $(\ell - m) = 0$, then ConsPat is trivial.
- What if $(\ell - m) \leq c$, for constant $c \geq 1$?
- What if $(\ell - m)$ is a parameter?

Theorem

ConsPat *is* NP-*hard, even if* $(\ell - m) \leq 6$ *and* $|\Sigma| \leq 5$.

ConsPat Parameterised by $(\ell - m)$

- If $(\ell - m) = 0$, then ConsPat is trivial.
- What if $(\ell - m) \leq c$, for constant $c \geq 1$?
- What if $(\ell - m)$ is a parameter?

Theorem

ConsPat is NP-hard, even if $(\ell - m) \leq 6$ and $|\Sigma| \leq 5$.

k	m	d	$(\ell - m)$	$\|\Sigma\|$	ℓ	Results
–	–	–	–	**p**	**p**	FPT
p	–	–	–	–	**p**	FPT
–	–	**p**	–	–	**p**	FPT
p	–	–	**p**	–	–	FPT
–	–	**p**	**p**	–	–	FPT
–	**p**	–	**p**	–	**p**	Open

Alphabet Reduction for CloseSubstr and ConsPat

CloseSubstr/ConsPat instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-
s_2	d	b	e	f	-	-
s_3	g	h	a	b	c	f

Alphabet Reduction for CloseSubstr and ConsPat

CloseSubstr/ConsPat instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-
s_2	d	b	e	f	-	-
s_3	g	h	a	b	c	f

solution string

s	x_1	x_2	x_3

Alphabet Reduction for CLOSESUBSTR and CONSPAT

CLOSESUBSTR/CONSPAT instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-	
s_2	d	b	e	f	-	-	
s_3	g	h	a	b	c	f	

solution string

s	x_1	x_2	x_3

Alphabet Reduction for CloseSubstr and ConsPat

CloseSubstr/ConsPat instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-
s_2	d	b	e	f	-	-
s_3	g	h	a	b	c	f

solution string

s	x_1	x_2	x_3

Alphabet Reduction for CloseSubstr and ConsPat

CloseSubstr/ConsPat instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-
s_2	d	b	e	f	-	-
s_3	g	h	a	b	c	f

solution string

s	x_1	x_2	x_3

Alphabet Reduction for CloseSubstr and ConsPat

CloseSubstr/ConsPat instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-
s_2	d	b	e	f	-	-
s_3	g	h	a	b	c	f

solution string

s	x_1	x_2	x_3

Alphabet Reduction for CloseSubstr and ConsPat

CloseSubstr/ConsPat instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-	s'_1	A	B	C	A	C	-
s_2	d	b	e	f	-	-	s'_2	D	B	D	E	-	-
s_3	g	h	a	b	c	f	s'_3	E	F	A	B	C	E

solution string

s $\quad x_1 \; x_2 \; x_3$

Alphabet Reduction for CloseSubstr and ConsPat

CloseSubstr/ConsPat instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-	s'_1	A	B	C	A	C	-
s_2	d	b	e	f	-	-	s'_2	D	B	D	E	-	-
s_3	g	h	a	b	c	f	s'_3	E	F	A	B	C	E

solution string

s x_1 x_2 x_3

Alphabet Reduction for CloseSubstr and ConsPat

CloseSubstr/ConsPat instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-
s_2	d	b	e	f	-	-
s_3	g	h	a	b	c	f

s'_1	A	B	C	A	C	-
s'_2	D	B	D	E	-	-
s'_3	E	F	A	B	C	E

solution string

$s \quad x_1 \quad x_2 \quad x_3$

Alphabet Reduction for CLOSESUBSTR and CONSPAT

CLOSESUBSTR/CONSPAT instance:

$m = 3$,

input strings:

s_1	a	b	c	a	c	-
s_2	d	b	e	f	-	-
s_3	g	h	a	b	c	f

s'_1	A	B	C	A	C	-
s'_2	D	B	D	E	-	-
s'_3	E	F	A	B	C	E

solution string

s x_1 x_2 x_3

Exact Exponential Time Algorithm

Theorem

CloseSubstr and ConsPat can be solved in time $\mathcal{O}^*(\sigma^m)$*,*
where $\sigma = |\max\{|\bigcup_{i=1}^{k} \text{alph}(s_i[j..j + (|s_i| - m)])| \mid 1 \leq j \leq m\}|$.

Exact Exponential Time Algorithm

Theorem

CloseSubstr *and* ConsPat *can be solved in time* $\mathcal{O}^*(\sigma^m)$,
where $\sigma = |\max\{|\bigcup_{i=1}^{k} \mathrm{alph}(s_i[j..j + (|s_i| - m)])| \mid 1 \leq j \leq m\}|$.

$\sigma \leq k(\ell - m + 1)$

$\Rightarrow$

Exact Exponential Time Algorithm

Theorem

CloseSubstr and ConsPat can be solved in time $\mathcal{O}^*(\sigma^m)$, *where* $\sigma = |\max\{|\bigcup_{i=1}^{k} \text{alph}(s_i[j..j + (|s_i| - m)])| \mid 1 \leq j \leq m\}|$.

$\sigma \leq k(\ell - m + 1)$

$\Rightarrow$

CloseSubstr and ConsPat can be solved in time $\mathcal{O}^*((k(\ell - m + 1))^m)$.

Exact Exponential Time Algorithm

Theorem

*Close*Substr *and* ConsPat *can be solved in time* $\mathcal{O}^*(\sigma^m)$, *where* $\sigma = |\max\{|\bigcup_{i=1}^{k} \mathrm{alph}(s_i[j..j + (|s_i| - m)])| \mid 1 \le j \le m\}|$.

$\sigma \le k(\ell - m + 1)$

$\Rightarrow$

CloseSubstr and ConsPat can be solved in time $\mathcal{O}^*((k(\ell - m + 1))^m)$.

We need this alphabet reduction later to bound the alphabet in the hardness reduction for ConsPat with $(\ell - m) \le 6$ and $|\Sigma| \le 5$.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Existing Reductions for ConsPat:

- Represent graph $\mathcal{G}$ as a string $s_{\mathcal{G}}$ (by listing its edges).

Hardness for CONSPAT with Bounded $(\ell - m)$ and $|\Sigma|$

Existing Reductions for CONSPAT:

- Represent graph $\mathcal{G}$ as a string $s_{\mathcal{G}}$ (by listing its edges).
- Copy $s_{\mathcal{G}}$ several times.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Existing Reductions for ConsPat:

- Represent graph $\mathcal{G}$ as a string $s_{\mathcal{G}}$ (by listing its edges).
- Copy $s_{\mathcal{G}}$ several times.
- Solution string selects an edge from each copy, such that the selected edges form a k-clique.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Existing Reductions for ConsPat:

- Represent graph $\mathcal{G}$ as a string $s_{\mathcal{G}}$ (by listing its edges).
- Copy $s_{\mathcal{G}}$ several times.
- Solution string selects an edge from each copy, such that the selected edges form a k-clique.

$\Rightarrow$ "Guessing the clique" is done by "aligning the solution string"

Hardness for CONSPAT with Bounded $(\ell - m)$ and $|\Sigma|$

Existing Reductions for CONSPAT:

- Represent graph $\mathcal{G}$ as a string $s_{\mathcal{G}}$ (by listing its edges).
- Copy $s_{\mathcal{G}}$ several times.
- Solution string selects an edge from each copy, such that the selected edges form a k-clique.

$\Rightarrow$ "Guessing the clique" is done by "aligning the solution string"
$\Rightarrow$ $(\ell - m)$ cannot possibly be bounded with this technique.

Hardness for CONSPAT with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a graph with $V = \{v_1, v_2, \ldots, v_n\}$.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a graph with $V = \{v_1, v_2, \ldots, v_n\}$.

- Closed neighbourhood: $N_{\mathcal{G}}[v_i] = \{v_j \mid \{v_i, v_j\} \in E\} \cup \{v_i\}$.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a graph with $V = \{v_1, v_2, \ldots, v_n\}$.

- Closed neighbourhood: $N_{\mathcal{G}}[v_i] = \{v_j \mid \{v_i, v_j\} \in E\} \cup \{v_i\}$.
- $\mathcal{G}$ is k-regular $\Leftrightarrow$ every v_i has exactly k neighbours.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a graph with $V = \{v_1, v_2, \ldots, v_n\}$.

- Closed neighbourhood: $N_{\mathcal{G}}[v_i] = \{v_j \mid \{v_i, v_j\} \in E\} \cup \{v_i\}$.
- $\mathcal{G}$ is k-regular $\Leftrightarrow$ every v_i has exactly k neighbours.
- $C \subseteq V$ is perfect code $\Leftrightarrow |N_{\mathcal{G}}[v_i] \cap C| = 1$, $v_i \in V$.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a graph with $V = \{v_1, v_2, \ldots, v_n\}$.

- Closed neighbourhood: $N_{\mathcal{G}}[v_i] = \{v_j \mid \{v_i, v_j\} \in E\} \cup \{v_i\}$.
- $\mathcal{G}$ is k-regular $\Leftrightarrow$ every v_i has exactly k neighbours.
- $C \subseteq V$ is perfect code $\Leftrightarrow |N_{\mathcal{G}}[v_i] \cap C| = 1$, $v_i \in V$.

3-Regular Perfect Code (3RPerCode)

Instance: A 3-regular graph $\mathcal{G}$.
Question: Does $\mathcal{G}$ contain a perfect code?

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a graph with $V = \{v_1, v_2, \ldots, v_n\}$.

- Closed neighbourhood: $N_{\mathcal{G}}[v_i] = \{v_j \mid \{v_i, v_j\} \in E\} \cup \{v_i\}$.
- $\mathcal{G}$ is k-regular $\Leftrightarrow$ every v_i has exactly k neighbours.
- $C \subseteq V$ is perfect code $\Leftrightarrow |N_{\mathcal{G}}[v_i] \cap C| = 1$, $v_i \in V$.

3-Regular Perfect Code (3RPerCode)

Instance: A 3-regular graph $\mathcal{G}$.

Question: Does $\mathcal{G}$ contain a perfect code?

Theorem (J. Kratochvíl and M. Křivánek 1988)

3RPerCode *is* NP-*complete.*

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a 3-regular graph, $V = \{v_1, v_2, \ldots, v_n\}$,

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a 3-regular graph, $V = \{v_1, v_2, \ldots, v_n\}$,

$N_i = N_{\mathcal{G}}[v_i] = \{v_{r_{i,1}}, v_{r_{i,2}}, v_{r_{i,3}}, v_{r_{i,4}}\}$.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a 3-regular graph, $V = \{v_1, v_2, \ldots, v_n\}$,

$N_i = N_{\mathcal{G}}[v_i] = \{v_{r_{i,1}}, v_{r_{i,2}}, v_{r_{i,3}}, v_{r_{i,4}}\}$.

$\Sigma = V \cup \{\star\}$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a 3-regular graph, $V = \{v_1, v_2, \ldots, v_n\}$,

$N_i = N_{\mathcal{G}}[v_i] = \{v_{r_{i,1}}, v_{r_{i,2}}, v_{r_{i,3}}, v_{r_{i,4}}\}$.

$\Sigma = V \cup \{\star\}$

Transform each N_i into string

$$s_i = \quad \star^6 \quad t_{i,1} \quad t_{i,2} \quad \ldots \quad t_{i,j} \quad \ldots \quad t_{i,n} \quad \star^6$$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a 3-regular graph, $V = \{v_1, v_2, \ldots, v_n\}$,

$N_i = N_{\mathcal{G}}[v_i] = \{v_{r_{i,1}}, v_{r_{i,2}}, v_{r_{i,3}}, v_{r_{i,4}}\}$.

$\Sigma = V \cup \{\star\}$

Transform each N_i into string

$$s_i = \quad \star^6 \quad t_{i,1} \quad t_{i,2} \quad \ldots \quad \alpha_1 \star \alpha_2 \star \alpha_3 \star \alpha_4 \star \quad \ldots \quad t_{i,n} \quad \star^6$$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Let $\mathcal{G} = (V, E)$ be a 3-regular graph, $V = \{v_1, v_2, \ldots, v_n\}$,

$N_i = N_{\mathcal{G}}[v_i] = \{v_{r_{i,1}}, v_{r_{i,2}}, v_{r_{i,3}}, v_{r_{i,4}}\}$.

$\Sigma = V \cup \{\star\}$

Transform each N_i into string

$$s_i = \quad \star^6 \quad t_{i,1} \quad t_{i,2} \quad \ldots \quad \alpha_1 \star \alpha_2 \star \alpha_3 \star \alpha_4 \star \quad \ldots \quad t_{i,n} \quad \star^6$$

$$\alpha_s = \begin{cases} v_{r_{i,s}} & \text{if } v_{r_{i,s}} \in N_j, \\ \star & \text{else.} \end{cases}$$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

$$s_1 = \star^6 \, t_{1,1} t_{1,2} \ldots t_{1,n} \, \star^6 \text{ with}$$

$$\begin{aligned} N_1 &= (v_1, v_4, v_5, v_8), \\ N_4 &= (v_5, v_9, v_4, v_1), \\ N_5 &= (v_1, v_5, v_{10}, v_4), \\ N_8 &= (v_1, v_8, v_{11}, v_{15}). \end{aligned}$$

$$\begin{array}{lcccccccc}
t_{1,1} = & v_1 & \star & v_4 & \star & v_5 & \star & v_8 & \star, \\
t_{1,4} = & v_1 & \star & v_4 & \star & v_5 & \star & \star & \star, \\
t_{1,5} = & v_1 & \star & v_4 & \star & v_5 & \star & \star & \star, \\
t_{1,8} = & v_1 & \star & \star & \star & \star & \star & v_8 & \star, \\
t_{1,9} = & \star & \star & v_4 & \star & \star & \star & \star & \star, \\
t_{1,10} = & \star & \star & \star & \star & v_5 & \star & \star & \star, \\
t_{1,11} = & \star & \star & \star & \star & \star & \star & v_8 & \star, \\
t_{1,15} = & \star & \star & \star & \star & \star & \star & v_8 & \star .
\end{array}$$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

$|s_i| = 8n + 12 \Rightarrow \ell = 8n + 12$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

$|s_i| = 8n + 12 \Rightarrow \ell = 8n + 12$

Set $m = 8n + 6 \Rightarrow (\ell - m) = 6$.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

$|s_i| = 8n + 12 \Rightarrow \ell = 8n + 12$

Set $m = 8n + 6 \Rightarrow (\ell - m) = 6$.

Lemma

Solution string has form $s = \star^6 x_1 \star^7 x_2 \star^7 \ldots x_n \star^7$ *with* $x_i \in N_i$.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

$|s_i| = 8n + 12 \Rightarrow \ell = 8n + 12$

Set $m = 8n + 6 \Rightarrow (\ell - m) = 6$.

Lemma

Solution string has form $s = \star^6 x_1 \star^7 x_2 \star^7 \ldots x_n \star^7$ *with* $x_i \in N_i$.

We need to show: $x_i \in N_j \Rightarrow x_j = x_i$.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

$|s_i| = 8n + 12 \Rightarrow \ell = 8n + 12$

Set $m = 8n + 6 \Rightarrow (\ell - m) = 6$.

Lemma

Solution string has form $s = \star^6 x_1 \star^7 x_2 \star^7 \dots x_n \star^7$ *with* $x_i \in N_i$.

We need to show: $x_i \in N_j \Rightarrow x_j = x_i$.

s_1: $\star \star \star \star \star \star v_1 \star v_4 \star v_5 \star v_8 \star \cdots$

s: $\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

$|s_i| = 8n + 12 \Rightarrow \ell = 8n + 12$

Set $m = 8n + 6 \Rightarrow (\ell - m) = 6.$

Lemma

Solution string has form $s = \star^6 x_1 \star^7 x_2 \star^7 \ldots x_n \star^7$ *with* $x_i \in N_i$.

We need to show: $x_i \in N_j \Rightarrow x_j = x_i$.

s_1 $\star \star \star \star \star \star v_1 \star v_4 \star v_5 \star v_8 \star \cdots$

s $\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

$\star \star \star \star \star \star x_1 \cdots$

Hardness for CONSPAT with Bounded $(\ell - m)$ and $|\Sigma|$

Lemma

$\mathcal{G}$ has perfect code $\Leftrightarrow$ exists solution string.

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Lemma

$\mathcal{G}$ has perfect code $\Leftrightarrow$ exists solution string.

Alphabet reduction:

$$\begin{array}{lcccccccccc}
s_1 & \star^6 & t_{1,1} & \star^6 & t_{1,2} & \star^6 & \ldots & \star^6 & t_{1,n} & \star^6 \\
s_2 & \star^6 & t_{2,1} & \star^6 & t_{2,2} & \star^6 & \ldots & \star^6 & t_{2,n} & \star^6 \\
s_3 & \star^6 & t_{3,1} & \star^6 & t_{3,2} & \star^6 & \ldots & \star^6 & t_{3,n} & \star^6 \\
\vdots & & & & & & & & & \\
s_n & \star^6 & t_{n,1} & \star^6 & t_{n,2} & \star^6 & \ldots & \star^6 & t_{n,n} & \star^6
\end{array}$$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Lemma

$\mathcal{G}$ has perfect code $\Leftrightarrow$ exists solution string.

Alphabet reduction:

$$
\begin{array}{lcccccccc}
s_1 & \star^6 & t_{1,1} & \star^6 & t_{1,2} & \star^6 & \dots & \star^6 & t_{1,n} & \star^6 \\
s_2 & \star^6 & t_{2,1} & \star^6 & t_{2,2} & \star^6 & \dots & \star^6 & t_{2,n} & \star^6 \\
s_3 & \star^6 & t_{3,1} & \star^6 & t_{3,2} & \star^6 & \dots & \star^6 & t_{3,n} & \star^6 \\
\vdots & & & & & & & & & \\
s_n & \star^6 & t_{n,1} & \star^6 & t_{n,2} & \star^6 & \dots & \star^6 & t_{n,n} & \star^6
\end{array}
$$

Hardness for ConsPat with Bounded $(\ell - m)$ and $|\Sigma|$

Lemma

$\mathcal{G}$ has perfect code $\Leftrightarrow$ exists solution string.

Alphabet reduction:

$$\begin{array}{cccccccccc}
s_1 & \star^6 & t_{1,1} & \star^6 & t_{1,2} & \star^6 & \dots & \star^6 & t_{1,n} & \star^6 \\
s_2 & \star^6 & t_{2,1} & \star^6 & t_{2,2} & \star^6 & \dots & \star^6 & t_{2,n} & \star^6 \\
s_3 & \star^6 & t_{3,1} & \star^6 & t_{3,2} & \star^6 & \dots & \star^6 & t_{3,n} & \star^6 \\
\vdots & & & & & & & & & \\
s_n & \star^6 & t_{n,1} & \star^6 & t_{n,2} & \star^6 & \dots & \star^6 & t_{n,n} & \star^6
\end{array}$$

Theorem

ConsPat *is* NP-*hard, even for* $(\ell - m) \leq 6$ *and* $|\Sigma| = 5$.

Thank you very much for your attention.