Optimal Inapproximability of Max CSPs over large alphabet

Pasin Manurangsi[1] **Preetum Nakkiran**[2] Luca Trevisan[1]

[1]UC Berkeley [2]Harvard

RANDOM-APPROX 2016

Max k-CSP$_R$

Maximum Constraint Satisfaction Problem:

- Variables take values in alphabet of size R.
- Constraints involve k variables each.
- Goal: find assignment maximizing # of satisfied constraints.

MAX k-CSP$_R$

Maximum Constraint Satisfaction Problem:

- Variables take values in alphabet of size R.
- Constraints involve k variables each.
- Goal: find assignment maximizing # of satisfied constraints.

Example

For $k = 2$, $R = 3$, a 2-CSP$_3$ is given by a list of constraints:

$$\begin{cases} (x_1 = 0 \wedge x_2 = 2) \\ (x_1 = 1 \wedge x_3 = 2) \\ \dots \end{cases}$$

Hardness of $\textsc{Max}\ k\text{-CSP}_R$

NP-hard to solve exactly (contains MAX-CUT, MAX 3-SAT).

Hardness of MAX k-CSP$_R$

NP-hard to solve exactly (contains MAX-CUT, MAX 3-SAT).

NP-hard to approximate (PCP theorem).

Hardness of MAX k-CSP$_R$

NP-hard to solve exactly (contains MAX-CUT, MAX 3-SAT).

NP-hard to approximate (PCP theorem).

Boolean CSPs ($R = 2$): Optimal approximation factor is $O(k/2^k)$.

Hardness of $\text{MAX } k\text{-CSP}_R$

NP-hard to solve exactly (contains MAX-CUT, MAX 3-SAT).

NP-hard to approximate (PCP theorem).

Boolean CSPs ($R = 2$): Optimal approximation factor is $O(k/2^k)$.

Non-boolean CSPs ($R > 2$): not resolved prior.

Hardness of Approximation

Trivial $(1/R^k)$-approximation for $\textsc{Max}$ k-CSP_R: Random assignment. Each clause matches the maximizing assignment w.p. $1/R^k$.

Q: Can we do better? Is it hard to do much better?

Prior Work: Non-boolean Max CSP

Approximation factors:[1]

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$		
$k = 3$			
$3 \leq k < O(1)$			

[1] Ignoring constants, and for large R.

Prior Work: Non-boolean Max CSP

Approximation factors:[1]

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$			
$3 \leq k < O(1)$			

[1] Ignoring constants, and for large R.

Prior Work: Non-boolean Max CSP

Approximation factors:[1]

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$	$\frac{1}{R^2}$		
$3 \leq k < O(1)$			

[1] Ignoring constants, and for large R.

Prior Work: Non-boolean Max CSP

Approximation factors:[1]

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$	$\frac{1}{R^2}$		$\frac{1}{R}$
$3 \leq k < O(1)$			

[1] Ignoring constants, and for large R.

Prior Work: Non-boolean Max CSP

Approximation factors:[1]

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$	$\frac{1}{R^2}$		$\frac{1}{R}$
$3 \leq k < O(1)$	$\frac{1}{R^{k-1}}$		$\frac{1}{R^{k-2}}$

[1] Ignoring constants, and for large R.

Prior Work: Non-boolean Max CSP

Approximation factors:[1]

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$	$\frac{1}{R^2}$		$\frac{1}{R}$
$3 \leq k < O(1)$	$\frac{1}{R^{k-1}}$		$\frac{1}{R^{k-2}}$

For constant $k \geq 3$, factor of R gap in hardness vs. approximation.

[1] Ignoring constants, and for large R.

Our results

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$	$\frac{1}{R^2}$		$\frac{1}{R}$
$3 \leq k < O(1)$	$\frac{1}{R^{k-1}}$		$\frac{1}{R^{k-2}}$

2

Our results

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$	$\frac{1}{R^2}$ $\frac{\log R}{R^2}$		$\frac{1}{R}$
$3 \le k < O(1)$	$\frac{1}{R^{k-1}}$ $\frac{\log R}{R^{k-1}}$		$\frac{1}{R^{k-2}}$

2

Our results

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$	$\frac{1}{R^2}$ $\frac{\log R}{R^2}$	$\frac{\log R}{R^2}$	$\frac{1}{R}$
$3 \leq k < O(1)$	$\frac{1}{R^{k-1}}$ $\frac{\log R}{R^{k-1}}$	$\frac{\log R}{R^{k-1}}$	$\frac{1}{R^{k-2}}$

2

Our results

	Algorithm	UG-Hardness	NP-Hardness
$k = 2$	$\frac{\log R}{R}$	$\frac{\log R}{R}$	$\frac{\log R}{\sqrt{R}}$
$k = 3$	$\frac{1}{R^2}$ $\frac{\log R}{R^2}$	$\frac{\log R}{R^2}$	$\frac{1}{R}$
$3 \leq k < O(1)$	$\frac{1}{R^{k-1}}$ $\frac{\log R}{R^{k-1}}$	$\frac{\log R}{R^{k-1}}$	$\frac{1}{R^{k-2}}$

We give matching UG-hardness and approximation algorithms for any k, R. Gap reduced to $O(1)$ for constant k.[2]

[2] Original paper had polylog(R) gap. Improvement suggested by Rishi Saket, Subhash Khot, Venkat Guriswami.

Dictator Testing

UG-Hardness-of-approximation equivalent to dictator testing.

Dictator: $f(x_1, x_2, \ldots, x_n) = x_i$.

Dictator Testing

UG-Hardness-of-approximation equivalent to dictator testing.

Dictator: $f(x_1, x_2, \ldots, x_n) = x_i$.

Problem

Given oracle access to $f : [R]^n \to [R]$, *determine if* f *is a dictator or "far from a dictator".*

Dictator Testing

UG-Hardness-of-approximation equivalent to dictator testing.

Dictator: $f(x_1, x_2, \ldots, x_n) = x_i$.

Problem

Given oracle access to $f : [R]^n \to [R]$, *determine if* f *is a dictator or "far from a dictator".*

- **Completeness** c**:** If f is a dictator, accept w.p. $\geq c$.
- **Soundness** s**:** If f is "far from" a dictator, accept w.p. $\leq s$.

"Far from dictator" $\equiv$ small low-degree influences (Fourier condition)

Examples

$$f : [R]^n \to R$$

"Far from dictator" $\equiv$ small low-degree influences

Example

Plurality on n coordinates is far from a dictator (no influential coordinate).

Examples

$$f : [R]^n \to R$$

"Far from dictator" $\equiv$ small low-degree influences

Example

Plurality on n coordinates is far from a dictator (no influential coordinate).

Example

$$f(x_1, x_2, \ldots x_n) := x_1 \oplus_R x_2$$

is NOT far from a dictator.

UG-Hardness of Approximation

k-query dictator test over alphabet R, with (soundness, completeness) $= (s, c)$

$\iff$

UG-hard to distinguish between k-CSP$_R$ instances where $OPT \approx s$ vs. $OPT \approx c$

UG-Hardness of Approximation

k-query dictator test over alphabet R, with (soundness, completeness) $= (s, c)$

$\iff$

UG-hard to distinguish between k-CSP_R instances where $OPT \approx s$ vs. $OPT \approx c$

$\Downarrow$

UG-hard to approximate $\textsc{Max}$ k-CSP_R better than $\approx (s/c)$.

UG-hardness of boolean 2-CSP

[Khot, Kindler, Mossel, O'Donnell]

2-Query Boolean Dictator test

$f : \{0,1\}^n \to \{0,1\}, \quad \mathbb{E}[f] = 1/2.$

UG-hardness of boolean 2-CSP

[Khot, Kindler, Mossel, O'Donnell]

2-Query Boolean Dictator test

$f : \{0,1\}^n \to \{0,1\}, \quad \mathbb{E}[f] = 1/2.$

- Pick $x \sim \{0,1\}^n$ uniform

UG-hardness of boolean 2-CSP

[Khot, Kindler, Mossel, O'Donnell]

2-Query Boolean Dictator test

$f : \{0,1\}^n \to \{0,1\}, \quad \mathbb{E}[f] = 1/2.$

- Pick $x \sim \{0,1\}^n$ uniform
- Pick "noise" $\eta \sim \{0,1\}^n$, each coordinate *Bernoulli*(p).

UG-hardness of boolean 2-CSP

[Khot, Kindler, Mossel, O'Donnell]

2-Query Boolean Dictator test

$f : \{0,1\}^n \to \{0,1\}, \quad \mathbb{E}[f] = 1/2.$

- Pick $x \sim \{0,1\}^n$ uniform
- Pick "noise" $\eta \sim \{0,1\}^n$, each coordinate *Bernoulli*(p).
- Accept iff $f(x) = f(x \oplus \eta)$

UG-hardness of boolean 2-CSP

[Khot, Kindler, Mossel, O'Donnell]

2-Query Boolean Dictator test

$f : \{0,1\}^n \to \{0,1\}, \quad \mathbb{E}[f] = 1/2.$

- Pick $x \sim \{0,1\}^n$ uniform
- Pick "noise" $\eta \sim \{0,1\}^n$, each coordinate *Bernoulli*(p).
- Accept iff $f(x) = f(x \oplus \eta)$

For $p \approx 0.15$,

- **Completeness:** If f is a dictator, accepts w.p. $\geq 1 - p \approx 0.85$.
- **Soundness:** If f is "far from" a dictator, accepts w.p. $\leq\approx 0.74$.

UG-hardness of boolean 2-CSP

[Khot, Kindler, Mossel, O'Donnell]

2-Query Boolean Dictator test

$f : \{0,1\}^n \to \{0,1\}, \quad \mathbb{E}[f] = 1/2.$

- Pick $x \sim \{0,1\}^n$ uniform
- Pick "noise" $\eta \sim \{0,1\}^n$, each coordinate *Bernoulli*(p).
- Accept iff $f(x) = f(x \oplus \eta)$

For $p \approx 0.15$,

- **Completeness:** If f is a dictator, accepts w.p. $\geq 1 - p \approx 0.85$.
- **Soundness:** If f is "far from" a dictator, accepts w.p. $\leq \approx 0.74$.
- Ratio: $s/c \approx 0.878567 = \alpha_{GW}$

Why it works

Verifier accepts iff

$$f(x) = f(x + \eta)$$

Noise η iid on every coordinate.

Why it works

Verifier accepts iff

$$f(x) = f(x + \eta)$$

Noise η iid on every coordinate.

If f depends on many coordinates, the noise will "add up":
$f(x + \eta)$ will be almost uncorrelated with $f(x)$.

Why it works

Verifier accepts iff

$$f(x) = f(x + \eta)$$

Noise η iid on every coordinate.

If f depends on many coordinates, the noise will "add up":
$f(x + \eta)$ will be almost uncorrelated with $f(x)$.

Example

majority function $maj : \{\pm 1\}^n \to \{\pm 1\}$.

$$maj(x_1, \dots, x_n) = sign(\sum_i x_i)$$

If noise η is high enough, $sign(\sum_i x_i)$ will be almost independent of $sign(\sum_i (x_i + \eta_i))$

Our k-query large alphabet dictator test

$f : [R]^n \to [R]$

f is balanced: All pre-images $f^{-1}(i)$ of same size.

Our k-query large alphabet dictator test

$f : [R]^n \to [R]$

f is balanced: All pre-images $f^{-1}(i)$ of same size.

- Pick $z \sim [R]^n$ uniform

Our k-query large alphabet dictator test

$f : [R]^n \to [R]$

f is balanced: All pre-images $f^{-1}(i)$ of same size.

- Pick $z \sim [R]^n$ uniform
- Pick k iid noise $\eta_1, \dots, \eta_k \in [R]^n$, s.t. each coordinate of η_j is

$$\begin{cases} 0 & \text{w.p. } \rho \\ \text{uniform in } [R] & \text{otherwise} \end{cases}$$

Our k-query large alphabet dictator test

$f : [R]^n \to [R]$

f is balanced: All pre-images $f^{-1}(i)$ of same size.

- Pick $z \sim [R]^n$ uniform
- Pick k iid noise $\eta_1, \ldots, \eta_k \in [R]^n$, s.t. each coordinate of η_j is

$$\begin{cases} 0 & \text{w.p. } \rho \\ \text{uniform in } [R] & \text{otherwise} \end{cases}$$

- Accept iff $f(z+\eta_1) = f(z+\eta_2) = \cdots = f(z+\eta_k)$

Our Results

$$f : [R]^n \to [R]$$

Accept iff $f(z + \eta_1) = f(z + \eta_2) = \cdots = f(z + \eta_k)$

Our Results

$$f : [R]^n \to [R]$$

Accept iff $f(z + \eta_1) = f(z + \eta_2) = \cdots = f(z + \eta_k)$

We show:

- **Completeness:** If f is a dictator ($f(x) = x_j$), accepts w.p. $\approx \frac{1}{(\log R)^{k/2}}$
- **Soundness:** If f is balanced and has small influences, accepts w.p. $\leq\approx \frac{1}{R^{k-1}}$

If f is far from dictator, the k queries $f(z + \eta_1), f(z + \eta_2), \ldots$ look almost independent – all equal w.p. $\approx \frac{1}{R^{k-1}}$.

Soundness Analysis Ideas

Define $f^i : [R]^n \to \{0, 1\}$ as $f^i(x) := \mathbb{1}[f(x) = i]$

Soundness Analysis Ideas

Define $f^i : [R]^n \to \{0,1\}$ as $f^i(x) := \mathbb{1}[f(x) = i]$

$\mathbb{E}[f^i] = 1/R$ since f is balanced.

Soundness Analysis Ideas

Define $f^i : [R]^n \to \{0,1\}$ as $f^i(x) := \mathbb{1}[f(x) = i]$

$\mathbb{E}[f^i] = 1/R$ since f is balanced.

$$\Pr[accept] = \Pr[f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)]$$

Soundness Analysis Ideas

Define $f^i : [R]^n \to \{0,1\}$ as $f^i(x) := \mathbb{1}[f(x) = i]$

$\mathbb{E}[f^i] = 1/R$ since f is balanced.

$$\begin{aligned} \Pr[accept] &= \Pr[f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\ &= \sum_{i \in [R]} \Pr[i = f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \end{aligned}$$

Soundness Analysis Ideas

Define $f^i : [R]^n \to \{0,1\}$ as $f^i(x) := \mathbb{1}[f(x) = i]$

$\mathbb{E}[f^i] = 1/R$ since f is balanced.

$$\begin{aligned}
\Pr[accept] &= \Pr[f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\
&= \sum_{i \in [R]} \Pr[i = f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\
&= \sum_{i \in [R]} \mathop{\mathbb{E}}_{x,\eta}[f^i(x+\eta_1) f^i(x+\eta_2) \ldots f^i(x+\eta_k)]
\end{aligned}$$

Soundness Analysis Ideas

Define $f^i : [R]^n \to \{0,1\}$ as $f^i(x) := \mathbb{1}[f(x) = i]$

$\mathbb{E}[f^i] = 1/R$ since f is balanced.

$$
\begin{aligned}
\Pr[accept] &= \Pr[f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\
&= \sum_{i \in [R]} \Pr[i = f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\
&= \sum_{i \in [R]} \mathop{\mathbb{E}}_{x,\eta}[f^i(x+\eta_1) f^i(x+\eta_2) \ldots f^i(x+\eta_k)]
\end{aligned}
$$

Define $g^i(x) = \mathbb{E}_\eta[f^i(x+\eta)]$ (i.e. $g^i := T_\rho f^i$)

Soundness Analysis Ideas

Define $f^i : [R]^n \to \{0,1\}$ as $f^i(x) := \mathbb{1}[f(x) = i]$

$\mathbb{E}[f^i] = 1/R$ since f is balanced.

$$\begin{aligned}\Pr[accept] &= \Pr[f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\ &= \sum_{i\in[R]} \Pr[i = f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\ &= \sum_{i\in[R]} \mathop{\mathbb{E}}_{x,\eta}[f^i(x+\eta_1)f^i(x+\eta_2)\ldots f^i(x+\eta_k)] \\ &= \sum_{i\in[R]} \mathop{\mathbb{E}}_{x}[(g^i(x))^k]\end{aligned}$$

Define $g^i(x) = \mathbb{E}_\eta[f^i(x+\eta)]$ (i.e. $g^i := T_\rho f^i$)

Soundness Analysis Ideas

Define $f^i : [R]^n \to \{0,1\}$ as $f^i(x) := \mathbb{1}[f(x) = i]$

$\mathbb{E}[f^i] = 1/R$ since f is balanced.

$$\begin{aligned}
\Pr[accept] &= \Pr[f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\
&= \sum_{i\in[R]} \Pr[i = f(x+\eta_1) = f(x+\eta_2) = \cdots = f(x+\eta_k)] \\
&= \sum_{i\in[R]} \mathop{\mathbb{E}}_{x,\eta}[f^i(x+\eta_1) f^i(x+\eta_2) \ldots f^i(x+\eta_k)] \\
&= \sum_{i\in[R]} \mathop{\mathbb{E}}_{x}[(g^i(x))^k] \\
\text{(want)} &\approx \sum_{i\in[R]} \mathop{\mathbb{E}}_{x}[(g^i(x))]^k = \sum_{i} (1/R)^k = 1/R^{k-1}
\end{aligned}$$

Define $g^i(x) = \mathbb{E}_\eta[f^i(x+\eta)]$ (i.e. $g^i := T_\rho f^i$)

Soundness Analysis Ideas

$$f : [R]^n \to [0,1], \quad \mathbb{E}[f] = 1/R$$

Soundness Analysis Ideas

$$f : [R]^n \to [0,1], \quad \mathbb{E}[f] = 1/R$$

$$g(x) = \mathop{\mathbb{E}}_{\eta}[f(x+\eta)] = (T_\rho f)(x)$$

Soundness Analysis Ideas

$$f : [R]^n \to [0,1], \quad \mathbb{E}[f] = 1/R$$

$$g(x) = \mathop{\mathbb{E}}_{\eta}[f(x+\eta)] = (T_\rho f)(x)$$

Want to show:

$$\mathbb{E}[(T_\rho f)^k] \lesssim \mathbb{E}[f]^k \iff ||T_\rho f||_k \lesssim ||f||_1$$

Soundness Analysis Ideas

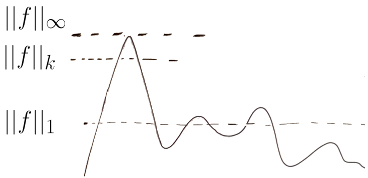
$$f : [R]^n \to [0,1], \quad \mathbb{E}[f] = 1/R$$

$$g(x) = \mathbb{E}_{\eta}[f(x+\eta)] = (T_\rho f)(x)$$

Want to show:

$$\mathbb{E}[(T_\rho f)^k] \lesssim \mathbb{E}[f]^k \iff ||T_\rho f||_k \lesssim ||f||_1$$

This is hypercontractivity



Soundness Analysis Ideas

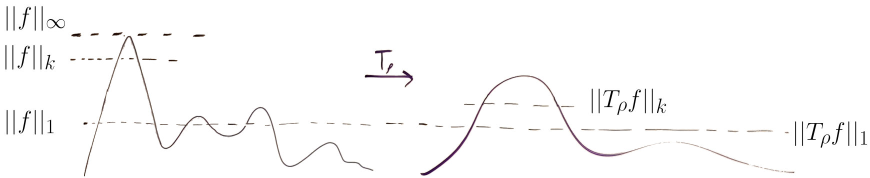
$$f : [R]^n \to [0,1], \quad \mathbb{E}[f] = 1/R$$

$$g(x) = \mathbb{E}_{\eta}[f(x+\eta)] = (T_\rho f)(x)$$

Want to show:

$$\mathbb{E}[(T_\rho f)^k] \lesssim \mathbb{E}[f]^k \iff ||T_\rho f||_k \lesssim ||f||_1$$

This is hypercontractivity



Algorithm Ideas

k-CSP$_R$:= k variables per clause; variables over domain $[R]$

Algorithm Ideas

k-CSP$_R$:= k variables per clause; variables over domain $[R]$

$k = 2$:

Existing $(\frac{\log R}{R})$-approximation algo (SDP-based).

Advantage of $A = R \log R$ over random assignment $(\frac{1}{R^2})$.

Algorithm Ideas

k-CSP$_R$:= k variables per clause; variables over domain $[R]$

$k = 2$:

Existing $(\frac{\log R}{R})$-approximation algo (SDP-based).

Advantage of $A = R \log R$ over random assignment $(\frac{1}{R^2})$.

$k \geq 3$:

Reduction to $k = 2$ case, preserving the Advantage over random assignment.

$\implies (\frac{\log R}{R^{k-1}})$-approximation algo for $k \geq 3$.

Algorithm Ideas

k-CSP$_R$:= k variables per clause; variables over domain $[R]$

$k = 2$:

Existing $(\frac{\log R}{R})$-approximation algo (SDP-based).

Advantage of $A = R \log R$ over random assignment $(\frac{1}{R^2})$.

$k \geq 3$:

Reduction to $k = 2$ case, preserving the Advantage over random assignment.

$\implies (\frac{\log R}{R^{k-1}})$-approximation algo for $k \geq 3$.

Map a constraint

$$(X_1 = a_1) \wedge (X_2 = a_2) \wedge \cdots \wedge (X_k = a_k)$$

to all pairwise constraints $\{(X_i = a_i \wedge X_j = a_j) : 1 \leq i < j \leq k\}$

Conclusions

We know the approximability of $\textsc{Max}\ k\text{-}\mathrm{CSP}_R$ for constant k, assuming Unique Games Conjecture: $\Theta(\frac{\log R}{R^{k-1}})$

Conclusions

We know the approximability of $\textsc{Max}\ k\text{-}\mathrm{CSP}_R$ for constant k, assuming Unique Games Conjecture: $\Theta(\frac{\log R}{R^{k-1}})$

Previously we knew it was between $\frac{1}{R^{k-2}}$ and $\frac{1}{R^{k-1}}$ (NP-hardness).

Conclusions

We know the approximability of $\textsc{Max}\ k\text{-CSP}_R$ for constant k, assuming Unique Games Conjecture: $\Theta(\frac{\log R}{R^{k-1}})$

Previously we knew it was between $\frac{1}{R^{k-2}}$ and $\frac{1}{R^{k-1}}$ (NP-hardness).

Open:

- Tight results for larger $k < R$. (already known for $k \geq R$). Current gap is $\frac{\log R}{R^{k-1}}$ vs. $\frac{k^2 \log(kR)}{R^{k-1}}$

Conclusions

We know the approximability of $\textsc{Max}\ k\text{-}\mathrm{CSP}_R$ for constant k, assuming Unique Games Conjecture: $\Theta(\frac{\log R}{R^{k-1}})$

Previously we knew it was between $\frac{1}{R^{k-2}}$ and $\frac{1}{R^{k-1}}$ (NP-hardness).

Open:

- Tight results for larger $k < R$. (already known for $k \geq R$).
  Current gap is $\frac{\log R}{R^{k-1}}$ vs. $\frac{k^2 \log(kR)}{R^{k-1}}$
- Tight results based on NP-hardness