Limits of Sequential Local Algorithms on the Random k-XORSAT Problem

Kingsley Yung

The Chinese University of Hong Kong

ICALP 2024

Random k-XORSAT

- Framework of random constraint satisfaction problem (random CSP)

Random k-XOR Satisfaction Problem (Random k-XORSAT)

- Instance:
  - Variables: n Boolean variables $x_1, x_2, \ldots, x_n \in \{0,1\} = \mathbb{F}_2$
  - Constraints: m Boolean linear equations of k variables (In $\mathbb{F}_2$, $1+1=0$)
    e.g. $x_1 + x_2 + x_3 = 1$
  - Randomness: Each equation is drawn randomly, from all possibilities.
    R.H.S. values are independent of the rest of instance.
- Task:
  - Assign values to variables so that all constraints are satisfied. (called a **solution**.)
  - **Solution space** $=$ Set of all solutions

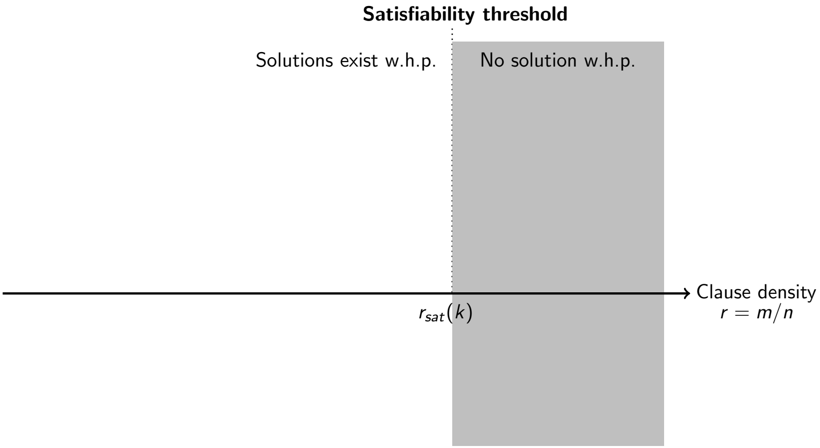
Example

- Example: $\begin{cases} x_1 + x_2 + x_3 & = 1 \\ x_1 \quad + x_3 + x_4 & = 0 \\ \quad x_2 + x_3 \quad + x_5 & = 1 \end{cases}$ with 5 variables, 3 constraints, $k = 3$
- Solving k-XORSAT $\equiv$ Solving a linear system in $\mathbb{F}_2$.
- **Question:** Why interested in some randomly generated linear systems?
- Phase transition (common in random CSPs)

Assumption

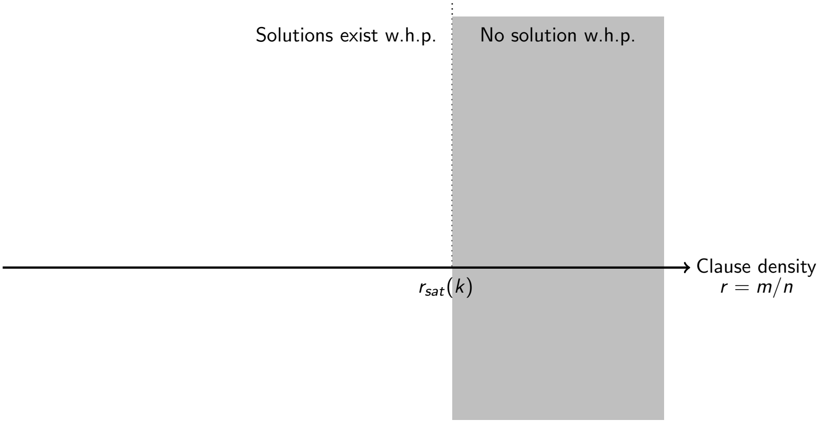
- **Assume:**
- number of equations $m \propto$ number of variables n
- clause density $r = \frac{m}{n}$
- $n \to \infty$

Clause density
$r = m/n$

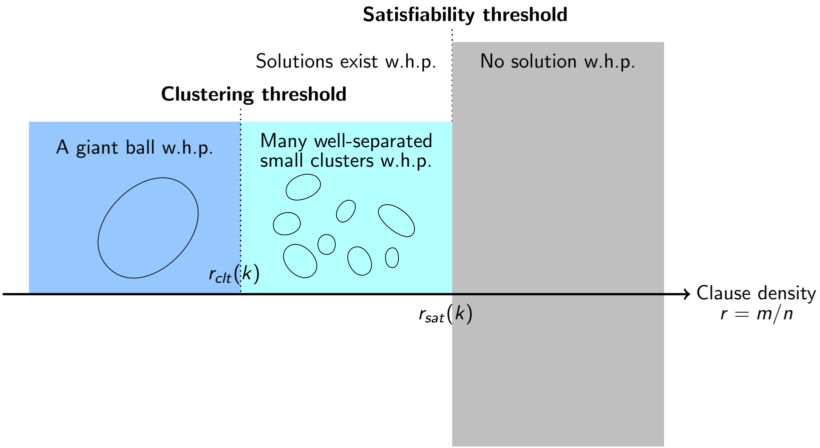


- **Satisfiability threshold**: $r_{\text{sat}}(k) = \frac{\lambda_k}{k(1-e^{-\lambda_k})^{k-1}}$, where λ_k is root of $\frac{x(e^x-1)}{e^x-1-x} = k$
- [Dubois and Mandler 2002; Pittel and Sorkin 2016]

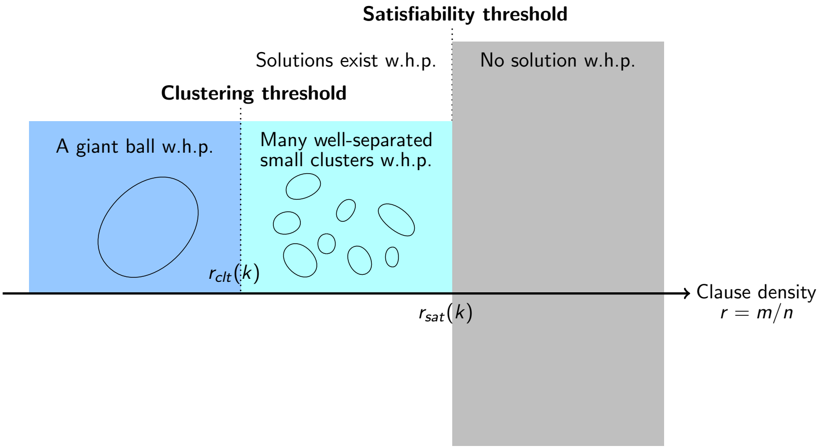
Satisfiability threshold



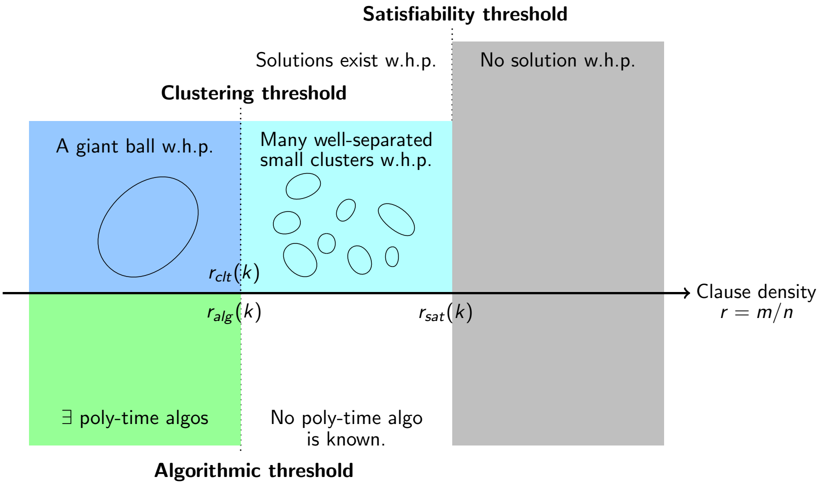
- w.h.p. = with high probability = with probability converging to 1 as $n \to \infty$



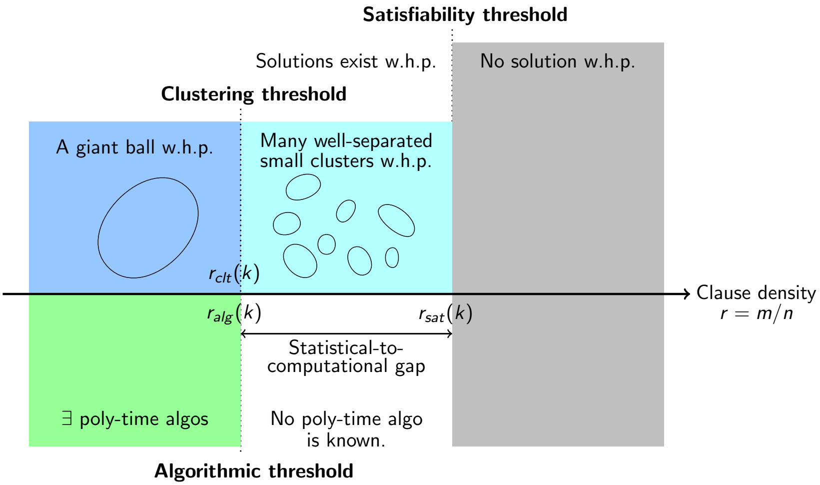
- **Clustering threshold**: $r_{clt}(k) = \min_{\lambda>0} \frac{(k-1)!\lambda}{(1-e^{-\lambda})^{k-1}}$
- [Ibrahimi, et al 2012; Achlioptas and Molloy 2015]



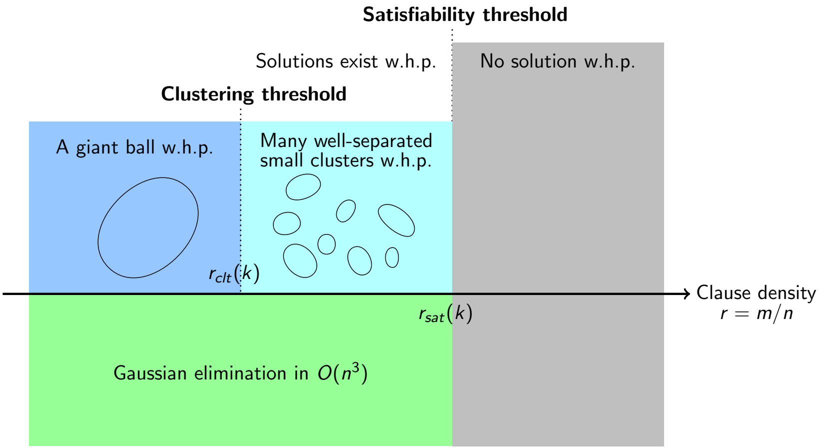
- Common in many random CSPs
- e.g. random k-SAT, random graph coloring, random hypergraph 2-coloring



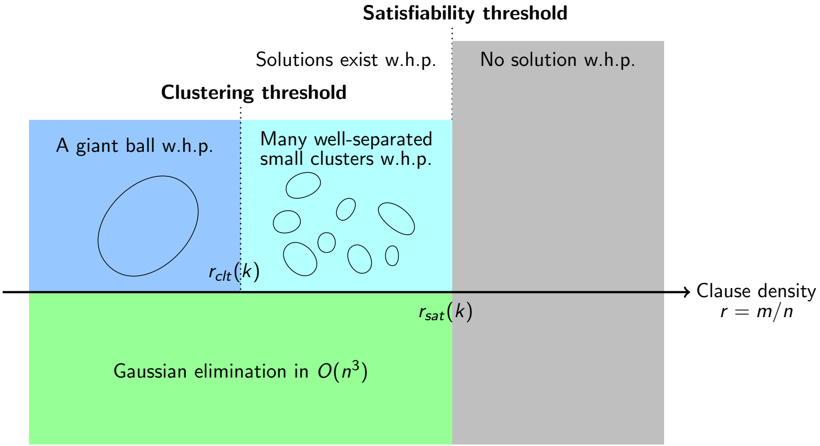
- Those random CSPs: We have poly-time algos to find solutions, with probability $\nrightarrow 0$.
- Only work, when density $<$ clustering threshold



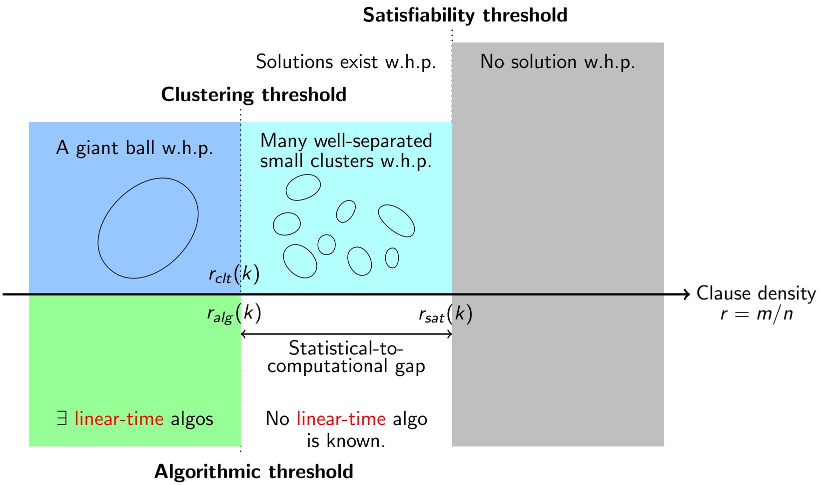
- Those random CSPs: Statistical-to-computational gap
- **Question:** Clustering phenomenon is related to average-case hardness?



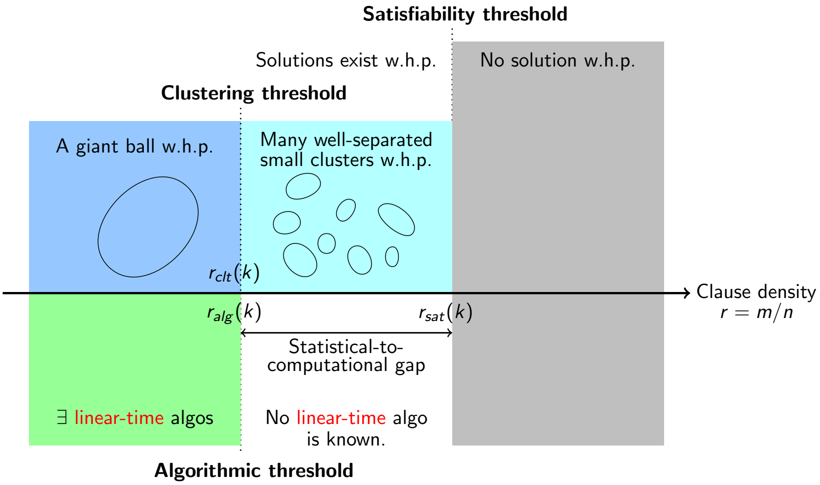
- Random k-XORSAT: We have Gaussian elimination to solve it in $O(n^3)$
- No such gap



- Poly-time = Efficient $\rightarrow$ Linear-time = Efficient
- ~~Gaussian elimination in $O(n^3)$~~



- **Best linear-time algo**: works only for $r < r_{\mathrm{clt}}(k)$. [Ibrahimi, et al 2012]
- Statistical-to-computational gap (linear-time version).



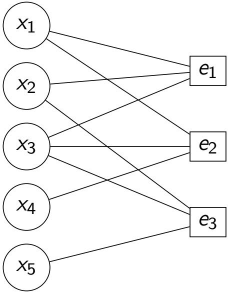
- In this paper, we try to justify the existence of the gap.
- Rule out a natural class of algorithm, from the gap

Our paper

- Sequential Local Algorithms fails to solve random k-XORSAT w.h.p. for density $r_{clt}(k) < r < r_{sat}(k)$ (i.e. in the statistical-to-computational gap).

Sequential local algorithms: Factor graphs

- Graph representation of instances:
  - Variable $\to$ Variable node $\bigcirc$
  - Equation $\to$ Equation node $\square$
  - Connect equation nodes $\bigcirc$ to variable nodes $\square$



$$x_1 + x_2 + x_3 = 1$$

$$x_1 + x_3 + x_4 = 0$$

$$x_2 + x_3 + x_5 = 1$$

- **Distance** between 2 nodes $=$ # edges in the shortest path
- **Local neighborhood** of a variable node, of radius R:
  - Subgraph induced by all nodes of distance $\leq R$ from the node

Sequential Local Algorithms: The algorithm

- **Equip:** Heuristic (called **local rule** τ), postive number $R > 0$
- **Remark 1:** Implementation depends on the choices of τ. It is a class of algorithms.
- **Remark 2:** If local rule τ takes constant time, then algorithm $\mathtt{DEC}_\tau$ takes linear time.

Algorithm Sequential Local Algorithms $\mathtt{DEC}_\tau$

1: **repeat**
2: Pick an unassigned variable randomly, say x_i.
3: $\tau($ Local neighborhood of x_i of radius R $) \to p \in [0, 1]$
4: Assign:
  1 to x_i with probability p
  0 to x_i with probability $1-p$
5: Update instance.
6: **until** Every variable has an assigned value.

Main result

Theorem 1

- For $k \geq 3$ and $r_{clt}(k) < r < r_{sat}(k)$, (i.e. density $\in$ statistical-to-computational gap.) given a sequential local algorithm $\texttt{DEC}_\tau$, with a local rule τ, if $p = \frac{1}{2}$ for $> 2\mu(k, r)$ iterations w.h.p., where
$$\mu(k, r) = \exp(-krQ^{k-1}) + krQ^{k-1}\exp(-krQ^{k-1}) \text{ and}$$
$$Q \text{ is the largest solution of } Q = 1 - \exp(-krQ^{k-1}),$$
- then the algorithm fails to solve random k-XORSAT instance w.h.p.

- The condition is satisfied by some choices of local rules.
- Theorem 2: Same result when using Unit Clause Propagation as local rule, for $k \geq 9$.
- Theorem 3: Same result when using Belief/Survey Propagation as local rule, for $k \geq 13$.

Proof Technique

- **Technique**: Based on Overlap Gap Property OGP (first used by [Gamarnik, Sudan 2017])
- Alternative way to describe clustering.
- An instance exhibits OGP if
  - there exists $0 \leq v_1 < v_2$ s.t.
  - distance between **every pair of solutions** are

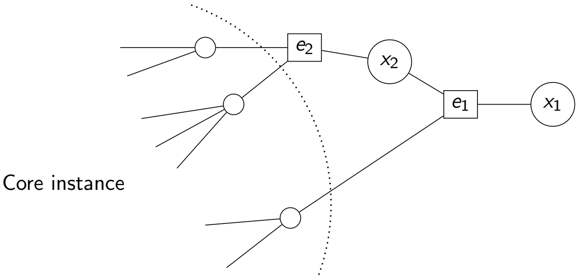
    either $d(\sigma_1, \sigma_2) \leq v_1$ or $d(\sigma_1, \sigma_2) \geq v_2$.

    (close to each other) or (far from each other).
- OGP $\Rightarrow$ Clustering (Converse has not yet confirmed.)
- OGP $\Rightarrow$ Rule out some algorithms. (Average-case hardness)

- Only know random k-XORSAT exhibits OGP for high density.
- Can't cover whole statistical-to-computational gap.
- OGP of sub-instance, instead of entire instance

OGP of sub-instance

- Proof of clustering of random k-XORSAT [Ibrahimi, et al 2012; Achlioptas, Molloy 2015]:
  $\exists$ sub-instance (called **core instance**) that exhibits OGP w.h.p.
- Obtained by:
  Repeatedly removing <u>variables involving ≤ 1 equation</u> and <u>the involved equation</u>.



- OGP of core instance + Modify OGP proof technique $\Rightarrow$ Our result
- Link **clustering phenomenon** and **average-case hardness** together.

Open Problems

- Theorem 2: Same result when using Unit Clause Propagation as local rule, for $k \geq 9$.
- Theorem 3: Same result when using Belief/Survey Propagation as local rule, for $k \geq 13$.
- Extend to lower k, by improving some calculation.

- **Question:** Can we apply the proof on other random CSPs?
- Core instance of random k-SAT ✗
- Good news: Same technique also works for other type of sub-instances with OGP.

- Thank you!