Phase Transitions of Replica Symmetry Breaking for Random Regular NAESAT

Allan Sly, Princeton University

Oxford Discrete Mathematics and Probability Seminar
June 2020

Introduction:
random constraint satisfaction problems;

Combinatorics and Theoretical Computer Science

Constraint satisfaction problem (CSP): is it possible to assign values to a set of *variables* to satisfy a given set of *constraints*?

- Scheduling your appointments for the day
- System of linear equations.
- Colouring a graph or finding a large independent set.
- Satisfying a Boolean formula.

Combinatorics and Theoretical Computer Science

Constraint satisfaction problem (CSP): is it possible to assign values to a set of *variables* to satisfy a given set of *constraints*?

- Scheduling your appointments for the day
- System of linear equations.
- Colouring a graph or finding a large independent set.
- Satisfying a Boolean formula.

Random CSPs

Our focus is to investigate properties when the constraints are chosen randomly.

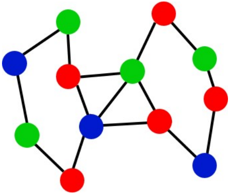
Combinatorial properties of Random Graphs:

- Erdős-Rényi Random Graph: $G(n, \alpha/n)$ with n vertices and edges with probability α/n (average degree α).
- Random α-regular graph: Uniformly chosen from α-regular graphs on n vertices.

Combinatorial properties of Random Graphs:

- Erdős-Rényi Random Graph: $G(n, \alpha/n)$ with n vertices and edges with probability α/n (average degree α).
- Random α-regular graph: Uniformly chosen from α-regular graphs on n vertices.

When is there a proper k-colouring?



K-SAT The random K-SAT problem, a model of a random Boolean formula, is perhaps the canonical random CSP.

K-SAT The random K-SAT problem, a model of a random Boolean formula, is perhaps the canonical random CSP.

Basic Definition:

Variables: $x_1, \ldots, x_n \in \{\mathrm{TRUE}, \mathrm{FALSE}\} \equiv \{+, -\}$

Constraints: m clauses taking the OR of k variables uniformly chosen from $\{+x_1, -x_1, \ldots, +x_n, -x_n\}$.

K-SAT The random K-SAT problem, a model of a random Boolean formula, is perhaps the canonical random CSP.

Basic Definition:

Variables: $x_1, \ldots, x_n \in \{\text{TRUE}, \text{FALSE}\} \equiv \{+, -\}$

Constraints: m clauses taking the OR of k variables uniformly chosen from $\{+x_1, -x_1, \ldots, +x_n, -x_n\}$.

Example: A 3-SAT formula with 4 clauses:

$$\mathscr{G}(\underline{x}) = (+x_1 \text{ OR } +x_2 \text{ OR } -x_3) \text{ AND } \overbrace{(+x_3 \text{ OR } +x_4 \text{ OR } -x_5)}^{\text{clause}}$$
$$\text{AND } (-x_1 \text{ OR } -x_4 \text{ OR } +x_5) \text{ AND } (+x_2 \text{ OR } -x_3 \text{ OR } +x_4)$$

K-SAT The random K-SAT problem, a model of a random Boolean formula, is perhaps the canonical random CSP.

Basic Definition:

Variables: $x_1, \ldots, x_n \in \{\text{TRUE}, \text{FALSE}\} \equiv \{+, -\}$

Constraints: m clauses taking the OR of k variables uniformly chosen from $\{+x_1, -x_1, \ldots, +x_n, -x_n\}$.

Example: A 3-SAT formula with 4 clauses:

$$\mathscr{G}(\underline{x}) = (+x_1 \text{ OR } +x_2 \text{ OR } -x_3) \text{ AND } \overbrace{(+x_3 \text{ OR } +x_4 \text{ OR } -x_5)}^{\text{clause}}$$
$$\text{AND } (-x_1 \text{ OR } -x_4 \text{ OR } +x_5) \text{ AND } (+x_2 \text{ OR } -x_3 \text{ OR } +x_4)$$

Clause density: The K-SAT model is parameterized the problem by the density of clauses $\alpha = m/n$.

K-SAT The random K-SAT problem, a model of a random Boolean formula, is perhaps the canonical random CSP.

Basic Definition:

Variables: $x_1, \ldots, x_n \in \{\text{TRUE}, \text{FALSE}\} \equiv \{+, -\}$

Constraints: m clauses taking the OR of k variables uniformly chosen from $\{+x_1, -x_1, \ldots, +x_n, -x_n\}$.

Example: A 3-SAT formula with 4 clauses:

$$\mathscr{G}(\underline{x}) = (+x_1 \text{ OR } +x_2 \text{ OR } -x_3) \text{ AND } \overbrace{(+x_3 \text{ OR } +x_4 \text{ OR } -x_5)}^{\text{clause}}$$
$$\text{AND } (-x_1 \text{ OR } -x_4 \text{ OR } +x_5) \text{ AND } (+x_2 \text{ OR } -x_3 \text{ OR } +x_4)$$

Clause density: The K-SAT model is parameterized the problem by the density of clauses $\alpha = m/n$.

Variant NAE-SAT: An assignment $\underline{x}$ is a solution if both $\underline{x}$ *and* $-\underline{x}$ are satisfying. It's **regular** if every variable is in the same number of clauses.

Graphical description: We can encode a K-SAT formula as a bipartite hyper-graph:

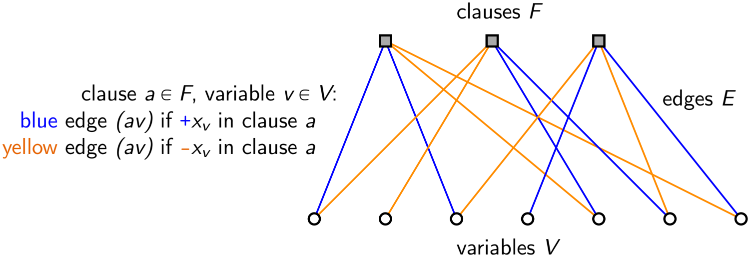
Graphical description: We can encode a K-SAT formula as a bipartite hyper-graph:

Take a 4-SAT formula with 3 clauses: $\mathscr{G}(\underline{x}) =$

$$(+x_1 \text{ OR } +x_3 \text{ OR } -x_5 \text{ OR } -x_7) \text{ AND } (-x_1 \text{ OR } -x_2 \text{ OR } +x_5 \text{ OR } +x_6)$$
$$\text{AND } (-x_3 \text{ OR } +x_4 \text{ OR } -x_6 \text{ OR } +x_7)$$

Graphical description: We can encode a K-SAT formula as a bipartite hyper-graph:
Take a 4-SAT formula with 3 clauses: $\mathscr{G}(\underline{x}) =$
$(+x_1 \text{ OR } +x_3 \text{ OR } -x_5 \text{ OR } -x_7) \text{ AND } (-x_1 \text{ OR } -x_2 \text{ OR } +x_5 \text{ OR } +x_6)$
$\text{AND } (-x_3 \text{ OR } +x_4 \text{ OR } -x_6 \text{ OR } +x_7)$

We can encode the formula as a bipartite graph $\mathscr{G} \equiv (V, F, E)$:

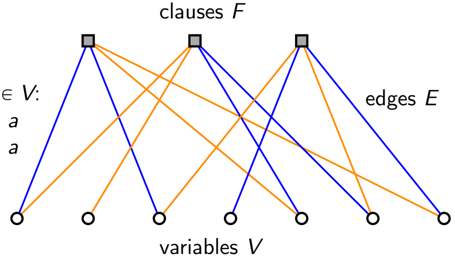


(4-SAT: each clause has degree 4)

Graphical description: We can encode a K-SAT formula as a bipartite hyper-graph:
Take a 4-SAT formula with 3 clauses: $\mathscr{G}(\underline{x}) =$
$(+x_1 \text{ OR } +x_3 \text{ OR } -x_5 \text{ OR } -x_7) \text{ AND } (-x_1 \text{ OR } -x_2 \text{ OR } +x_5 \text{ OR } +x_6)$
$\text{AND } (-x_3 \text{ OR } +x_4 \text{ OR } -x_6 \text{ OR } +x_7)$

We can encode the formula as a bipartite graph $\mathscr{G} \equiv (V, F, E)$:

clause $a \in F$, variable $v \in V$:
blue edge *(av)* if $+x_v$ in clause a
yellow edge *(av)* if $-x_v$ in clause a



(4-SAT: each clause has degree 4)

The resulting random graph is locally tree-like, almost no short cycles and it's local distribution can be described completely.

Main Question:

- *Satisfiability Threshold*: For which α are there satisfying assignments?

Main Question:

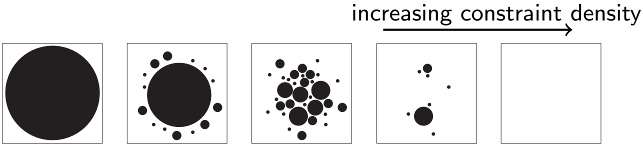
- *Satisfiability Threshold*: For which α are there satisfying assignments?

Other Question:

- *Free Energy*: How many solutions are there?
- *Local Statistics*: Properties of solutions such as how many clauses are satisfied only once?
- *Overlaps*: What does the joint distribution of several solutions look like?
- *Algorithmic*: Can solutions be found efficiently?

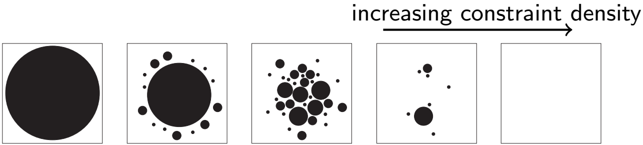
Theoretical Physics

Disordered systems such as *spin glasses* are models of interacting particles/variables with frustrated interactions. Many random constraint satisfaction problems can be recast as dilute mean-field spin glasses.



Theoretical Physics

Disordered systems such as *spin glasses* are models of interacting particles/variables with frustrated interactions. Many random constraint satisfaction problems can be recast as dilute mean-field spin glasses.



One-step Replica Symmetry Breaking Predictions:
Developed to study dense spin-glasses such as the Sherrington-Kirkpatrick model.

- **Replica Symmetry Breaking**: Clustering of assignments.
- **Cavity Method**: Heuristic for analyzing adding one variable.

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

$$\mathbb{E}Z = 2^n(1 - 1/2^{k-1})^m$$

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

$$\mathbb{E}Z = 2^n(1-1/2^{k-1})^m = \exp\{n\,\underbrace{[\ln 2 + \alpha \log(1-1/2^{k-1})]}\}$$

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

$$\mathbb{E}Z = 2^n(1 - 1/2^{k-1})^m = \exp\{n \underbrace{[\ln 2 + \alpha \log(1 - 1/2^{k-1})]}\}$$

exponent decreases in α, crosses zero at $\alpha_1 \approx 2^{k-1} \ln 2 + O(1)$

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

$$\mathbb{E}Z = 2^n(1-1/2^{k-1})^m = \exp\{n \underbrace{[\ln 2 + \alpha \log(1-1/2^{k-1})]}\}$$

exponent decreases in α, crosses zero at $\alpha_1 \approx 2^{k-1} \ln 2 + O(1)$

First moment threshold α_1 separates $\mathbb{E}Z \to \infty \,\big|\, \mathbb{E}Z \to 0$.

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

$$\mathbb{E}Z = 2^n(1-1/2^{k-1})^m = \exp\{n \underbrace{[\ln 2 + \alpha \log(1-1/2^{k-1})]}\}$$

exponent decreases in α, crosses zero at $\alpha_1 \approx 2^{k-1}\ln 2 + O(1)$

First moment threshold α_1 separates $\mathbb{E}Z \to \infty \,\big|\, \mathbb{E}Z \to 0$.

$\alpha_1 \neq \alpha_{\text{sat}}$: At least ϵn unconstrained variables so
$Z > 0 \Rightarrow Z \geqslant 2^{\epsilon n}$.

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

$$\mathbb{E}Z = 2^n(1 - 1/2^{k-1})^m = \exp\{n \underbrace{[\ln 2 + \alpha \log(1 - 1/2^{k-1})]}\}$$

exponent decreases in α, crosses zero at $\alpha_1 \approx 2^{k-1} \ln 2 + O(1)$

First moment threshold α_1 separates $\mathbb{E}Z \to \infty \,\big|\, \mathbb{E}Z \to 0$.

$\alpha_1 \neq \alpha_{\mathsf{sat}}$: At least ϵn unconstrained variables so $Z > 0 \Rightarrow Z \geqslant 2^{\epsilon n}$.

Second Moment method:

$$\mathbb{P}[Z > 0] \geqslant \frac{(\mathbb{E}Z)^2}{\mathbb{E}[Z^2]}$$

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

$$\mathbb{E}Z = 2^n(1 - 1/2^{k-1})^m = \exp\{n \underbrace{[\ln 2 + \alpha \log(1 - 1/2^{k-1})]}\}$$

exponent decreases in α, crosses zero at $\alpha_1 \approx 2^{k-1} \ln 2 + O(1)$

First moment threshold α_1 separates $\mathbb{E}Z \to \infty \,\big|\, \mathbb{E}Z \to 0$.

$\alpha_1 \neq \alpha_{\mathsf{sat}}$: At least ϵn unconstrained variables so $Z > 0 \Rightarrow Z \geqslant 2^{\epsilon n}$.

Second Moment method:

$$\mathbb{P}[Z > 0] \geqslant \frac{(\mathbb{E}Z)^2}{\mathbb{E}[Z^2]}$$

To be useful, requires always $\mathbb{E}[Z^2] \asymp (\mathbb{E}Z)^2$.

First Moment method on $Z \equiv |\{\text{satisfying assignments of } \mathscr{G}\}|$:

$$\mathbb{E}Z = 2^n(1 - 1/2^{k-1})^m = \exp\{n \underbrace{[\ln 2 + \alpha \log(1 - 1/2^{k-1})]}\}$$

exponent decreases in α, crosses zero at $\alpha_1 \approx 2^{k-1}\ln 2 + O(1)$

First moment threshold α_1 separates $\mathbb{E}Z \to \infty \,\big|\, \mathbb{E}Z \to 0$.

$\alpha_1 \neq \alpha_{\mathsf{sat}}$: At least ϵn unconstrained variables so $Z > 0 \Rightarrow Z \geqslant 2^{\epsilon n}$.

Second Moment method:

$$\mathbb{P}[Z > 0] \geqslant \frac{(\mathbb{E}Z)^2}{\mathbb{E}[Z^2]}$$

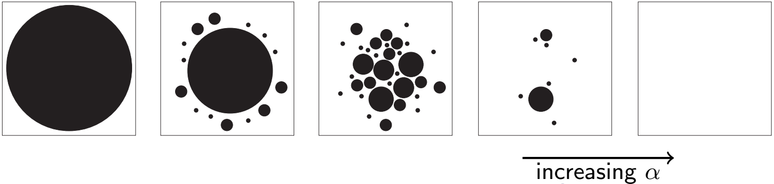
To be useful, requires always $\mathbb{E}[Z^2] \asymp (\mathbb{E}Z)^2$.

For NAE-SAT, colourings, second moment method succeeds up to $\alpha_2 = \alpha_{\mathsf{sat}} - O(1)$. Fails, for K-SAT for **all** $\alpha > 0$.

Some physics perspective:
condensation and replica symmetry breaking

Phase Diagram

Two solutions are connected if they differ by one bit.

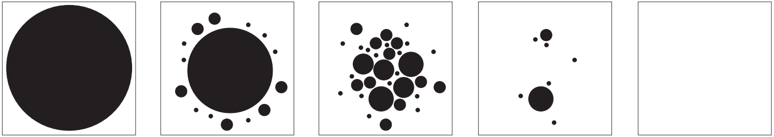


Krząkała–Montanari–Ricci-Tersenghi–Semerjian–Zdeborová '07

Montanari–Ricci-Tersenghi–Semerjian '08

Phase Diagram

well-connected

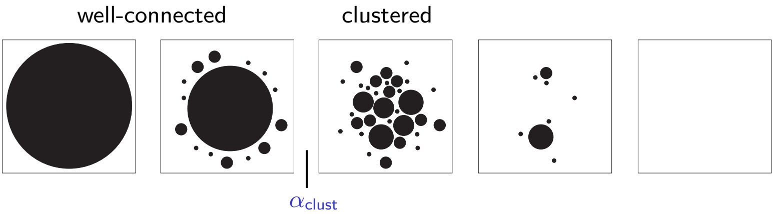


Krząkała–Montanari–Ricci-Tersenghi–Semerjian–Zdeborová '07

Montanari–Ricci-Tersenghi–Semerjian '08

The solution space **SOL** starts out as a well-connected cluster.

Phase Diagram



Krząkała–Montanari–Ricci-Tersenghi–Semerjian–Zdeborová '07

Montanari–Ricci-Tersenghi–Semerjian '08

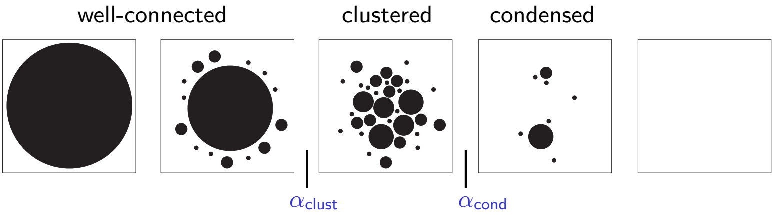
The solution space **SOL** starts out as a well-connected cluster.

After α_{clust}, **SOL** decomposes into exponentially clusters

–Clustering Achlioptas, Coja-Oghlan '10

Two uniformly chosen solutions are near independent. —'RS'

Phase Diagram



Krząkała–Montanari–Ricci-Tersenghi–Semerjian–Zdeborová '07

Montanari–Ricci-Tersenghi–Semerjian '08

The solution space **SOL** starts out as a well-connected cluster.

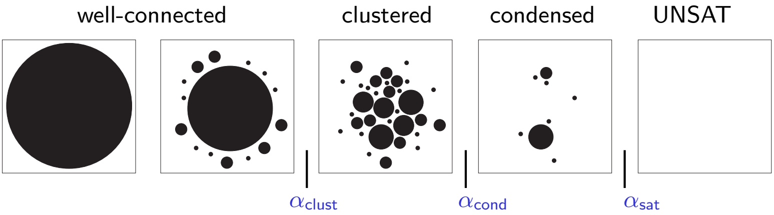
After α_{clust}, **SOL** decomposes into exponentially clusters

–Clustering Achlioptas, Coja-Oghlan '10

Two uniformly chosen solutions are near independent. —'RS'

After α_{cond}, **SOL** is dominated by a few large clusters
Two uniformly chosen solutions may be correlated. —'RSB'

Phase Diagram



Krząkała–Montanari–Ricci-Tersenghi–Semerjian–Zdeborová '07

Montanari–Ricci-Tersenghi–Semerjian '08

The solution space **SOL** starts out as a well-connected cluster.

After α_{clust}, **SOL** decomposes into exponentially clusters

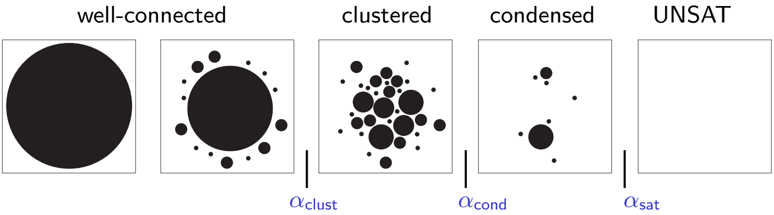
–Clustering Achlioptas, Coja-Oghlan '10

Two uniformly chosen solutions are near independent. —'RS'

After α_{cond}, **SOL** is dominated by a few large clusters
Two uniformly chosen solutions may be correlated. —'RSB'

After α_{sat}, no solutions w.h.p.

Phase Diagram



Krząkała–Montanari–Ricci-Tersenghi–Semerjian–Zdeborová '07

Montanari–Ricci-Tersenghi–Semerjian '08

The solution space **SOL** starts out as a well-connected cluster.

After α_{clust}, **SOL** decomposes into exponentially clusters

–Clustering Achlioptas, Coja-Oghlan '10

Two uniformly chosen solutions are near independent. —'RS'

After α_{cond}, **SOL** is dominated by a few large clusters
Two uniformly chosen solutions may be correlated. —'RSB'

After α_{sat}, no solutions w.h.p.

RSB: These models are called one step replica symmetry breaking (1RSB) because the overlap of two uniformly chosen solutions is concentrated on two points in the condensation regime.

Cluster Model: We represent clusters as a new spin system on $V(\mathscr{G})$.

Cluster Model: We represent clusters as a new spin system on $V(\mathscr{G})$.

- Start from $\underline{x} \in \{+, -\}^{V(\mathscr{G})}$ and explore the cluster $\mathcal{C}$.

Cluster Model: We represent clusters as a new spin system on $V(\mathscr{G})$.

- Start from $\underline{x} \in \{+, -\}^{V(\mathscr{G})}$ and explore the cluster $\mathcal{C}$.
- If a spin can be flipped between + and − without violating any clauses it is set to `f`.
- Iterate until done.

Cluster Model: We represent clusters as a new spin system on $V(\mathscr{G})$.

- Start from $\underline{x} \in \{+, -\}^{V(\mathscr{G})}$ and explore the cluster $\mathcal{C}$.
- If a spin can be flipped between $+$ and $-$ without violating any clauses it is set to $\mathtt{f}$.
- Iterate until done.
- Each variable is mapped to a value from $\{+, -, \mathtt{f}\}$.

Cluster Model: We represent clusters as a new spin system on $V(\mathscr{G})$.

- Start from $\underline{x} \in \{+, -\}^{V(\mathscr{G})}$ and explore the cluster $\mathcal{C}$.
- If a spin can be flipped between + and – without violating any clauses it is set to $\mathtt{f}$.
- Iterate until done.
- Each variable is mapped to a value from $\{+, -, \mathtt{f}\}$.

This resulting configuration on $\{+, -, \mathtt{f}\}^{V(\mathscr{G})}$ is our definition of a cluster. It is a spin system satisfying the following conditions:

- $\mathtt{f}$ are not forced by any clause.
- + and – variables must be forced by at least one clause.
- No violated clause.

Cluster Model: We represent clusters as a new spin system on $V(\mathscr{G})$.

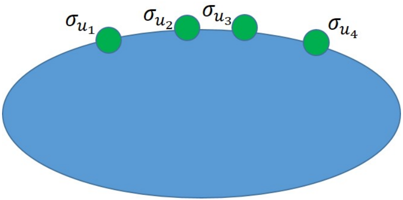
- Start from $\underline{x} \in \{+, -\}^{V(\mathscr{G})}$ and explore the cluster $\mathcal{C}$.
- If a spin can be flipped between + and – without violating any clauses it is set to $\mathtt{f}$.
- Iterate until done.
- Each variable is mapped to a value from $\{+, -, \mathtt{f}\}$.

This resulting configuration on $\{+, -, \mathtt{f}\}^{V(\mathscr{G})}$ is our definition of a cluster. It is a spin system satisfying the following conditions:

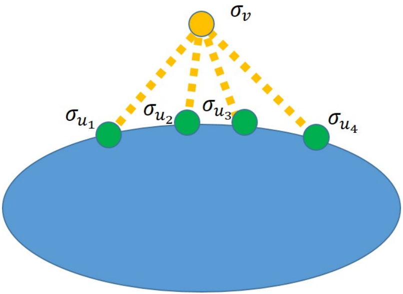
- $\mathtt{f}$ are not forced by any clause.
- + and – variables must be forced by at least one clause.
- No violated clause.

We call this the cluster model. Let Ω_n be the number of $\{+, -, \mathtt{f}\}^{V(\mathscr{G})}$ configurations. Locally rigid resulting in no clustering.

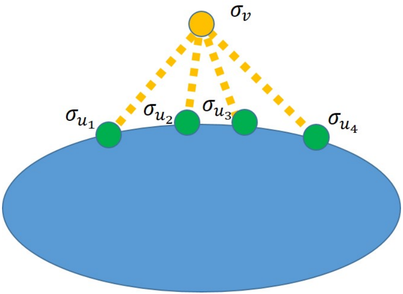
Cavity Method: Adding a new vertex v (or clause).



Cavity Method: Adding a new vertex v (or clause).



Cavity Method: Adding a new vertex v (or clause).

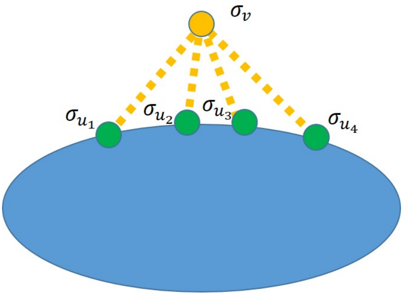


If we know the joint distribution of σ_{u_i} we can:

1. Calculate the law of σ_v
2. Evaluate the change in the partition function from Z_{n+1}/Z_n.

Write $\log Z_n = \sum_{i=1}^{n} \log Z_i/Z_{i-1}$.

Cavity Method: Adding a new vertex v (or clause).



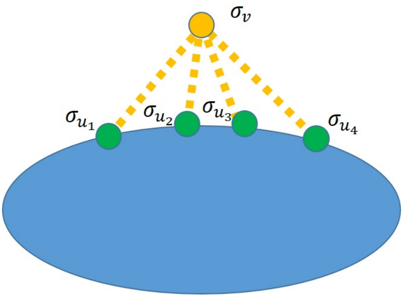
If we know the joint distribution of σ_{u_i} we can:

1. Calculate the law of σ_v
2. Evaluate the change in the partition function from Z_{n+1}/Z_n.

Write $\log Z_n = \sum_{i=1}^{n} \log Z_i/Z_{i-1}$.

The Replica Symmetric heuristic assumes that σ_{u_i} are independent drawn from some law μ.

Cavity Method: Adding a new vertex v (or clause).



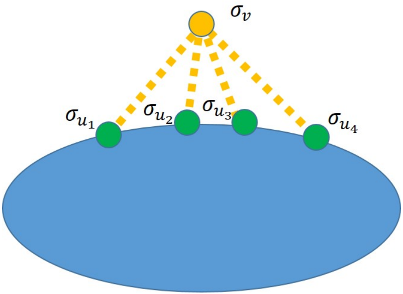
If we know the joint distribution of σ_{u_i} we can:

1. Calculate the law of σ_v
2. Evaluate the change in the partition function from Z_{n+1}/Z_n.

Write $\log Z_n = \sum_{i=1}^{n} \log Z_i/Z_{i-1}$.

The Replica Symmetric heuristic assumes that σ_{u_i} are independent drawn from some law μ.
The 1-RSB heuristic assumes this for the cluster model.

Cavity Method: Adding a new vertex v (or clause).



If we know the joint distribution of σ_{u_i} we can:

1. Calculate the law of σ_v
2. Evaluate the change in the partition function from Z_{n+1}/Z_n.

Write $\log Z_n = \sum_{i=1}^{n} \log Z_i/Z_{i-1}$.

The Replica Symmetric heuristic assumes that σ_{u_i} are independent drawn from some law μ.
The 1-RSB heuristic assumes this for the cluster model.
Self-consistency: The law of σ_v should also be drawn from μ which means μ must satisfy a fixed point equation.

Explicit formula ($k \geqslant 3$)

Explicit formula ($k \geqslant 3$) Let q be the fixed point of the recursion,

$$q = \frac{2 - 2\left(\frac{1-2(q/2)^{k-1}}{1-(q/2)^{k-1}}\right)^{d-1}}{2 - \left(\frac{1-2(q/2)^{k-1}}{1-(q/2)^{k-1}}\right)^{d-1}}$$

This represents the probability that a variable is frozen to + or – in a random cluster.

Explicit formula ($k \geqslant 3$) Let q be the fixed point of the recursion,

$$q = \frac{2 - 2\left(\frac{1-2(q/2)^{k-1}}{1-(q/2)^{k-1}}\right)^{d-1}}{2 - \left(\frac{1-2(q/2)^{k-1}}{1-(q/2)^{k-1}}\right)^{d-1}}$$

This represents the probability that a variable is frozen to + or – in a random cluster.

Define

$$\Phi(\alpha) = \log(2) - \log(2-q) - d(1-k^{-1}-d^{-1})\log[1-2(q/2)^{k-1}] \\ +(d-1)\log[1-(q/2)^{k-1}]$$

Expected change in $\log \Omega_n$ *to* $\log \Omega_{n+1}$.

Explicit formula ($k \geqslant 3$) Let q be the fixed point of the recursion,

$$q = \frac{2 - 2(\frac{1-2(q/2)^{k-1}}{1-(q/2)^{k-1}})^{d-1}}{2 - (\frac{1-2(q/2)^{k-1}}{1-(q/2)^{k-1}})^{d-1}}$$

This represents the probability that a variable is frozen to + or – in a random cluster.

Define

$$\Phi(\alpha) = \log(2) - \log(2-q) - d(1 - k^{-1} - d^{-1}) \log[1 - 2(q/2)^{k-1}] \\ + (d-1)\log[1-(q/2)^{k-1}]$$

Expected change in $\log \Omega_n$ *to* $\log \Omega_{n+1}$.

Then the 1RSB prediction $d_{SAT} \approx 2^{k-1} k \log 2 + O(1)$ is the root of $\Phi(\alpha) = 0$. –Ding, S., Sun '16

Beyond the Satisfiability Threshold

Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

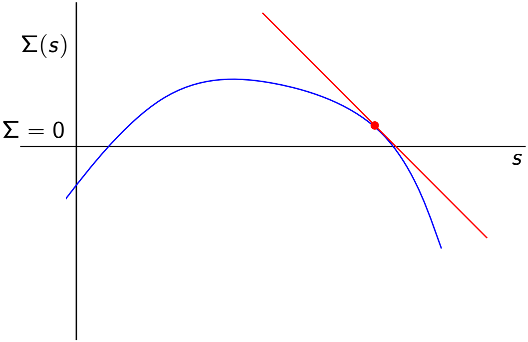
$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{\text{(cluster size)}}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

$\mathbb{E}Z$ is dominated by s where $\Sigma'(s) \equiv -1$ (depending on α).

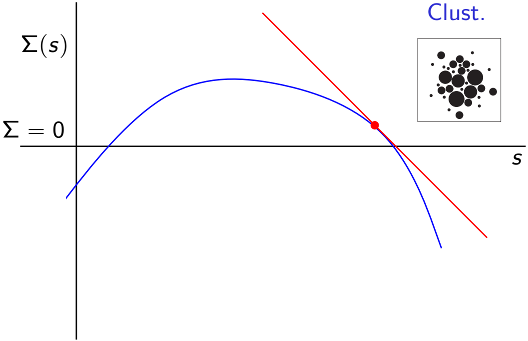


Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

$\mathbb{E}Z$ is dominated by s where $\Sigma'(s) \equiv -1$ (depending on α).

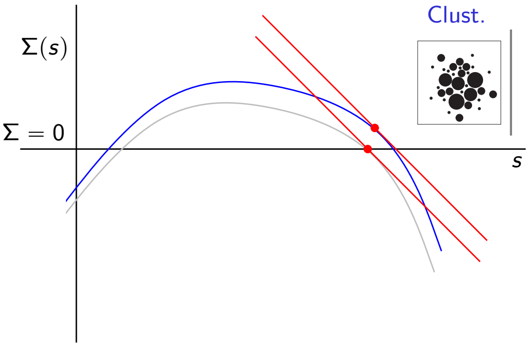


Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

$\mathbb{E}Z$ is dominated by s where $\Sigma'(s) \equiv -1$ (depending on α).

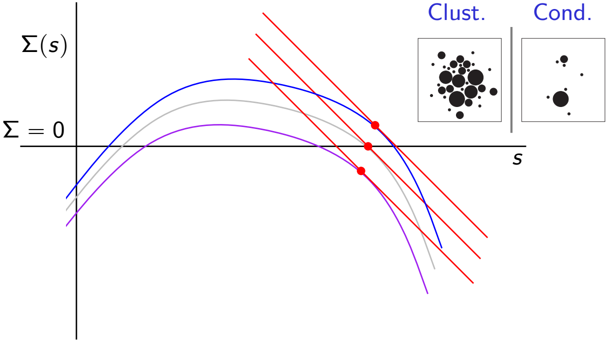


Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

$\mathbb{E}Z$ is dominated by s where $\Sigma'(s) \equiv -1$ (depending on α).

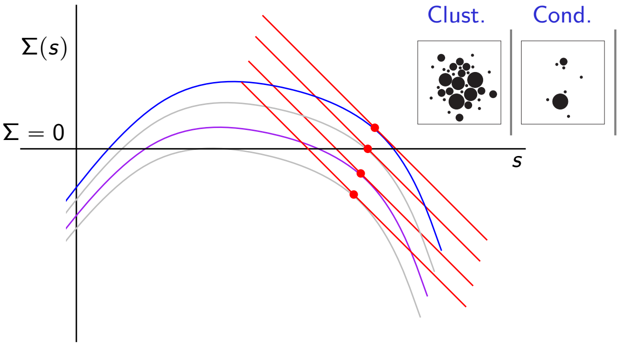


Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

$\mathbb{E}Z$ is dominated by s where $\Sigma'(s) \equiv -1$ (depending on α).

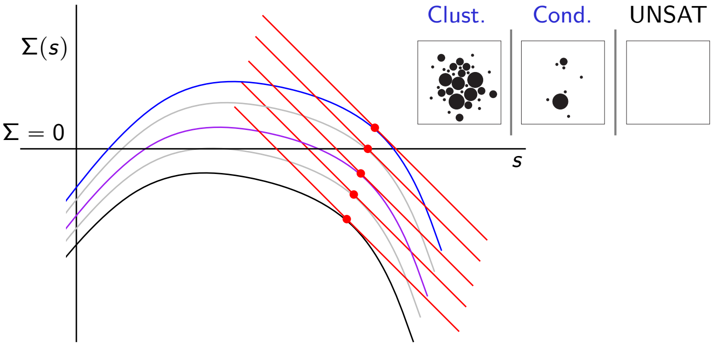


Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

$\mathbb{E}Z$ is dominated by s where $\Sigma'(s) \equiv -1$ (depending on α).

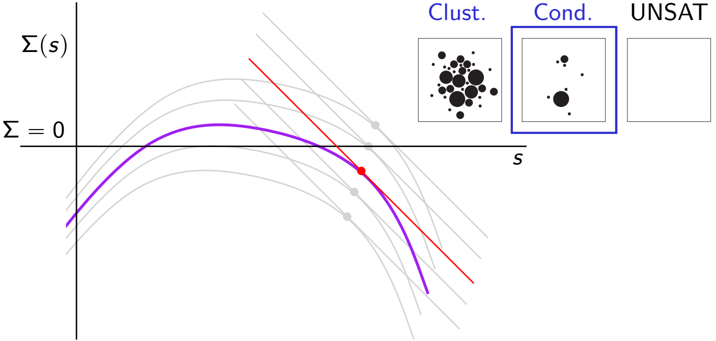


Condensation

Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

$\mathbb{E}Z$ is dominated by s where $\Sigma'(s) \equiv -1$ (depending on α).

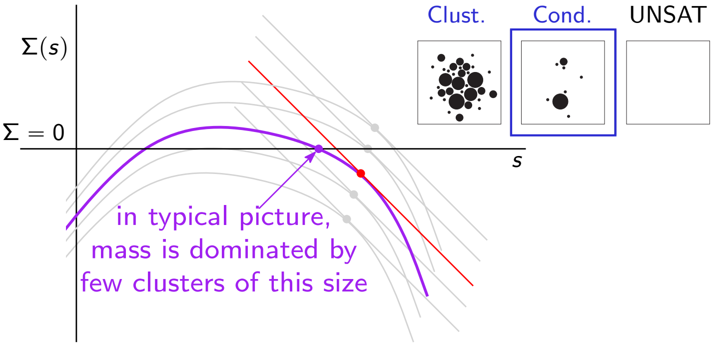


Condensation

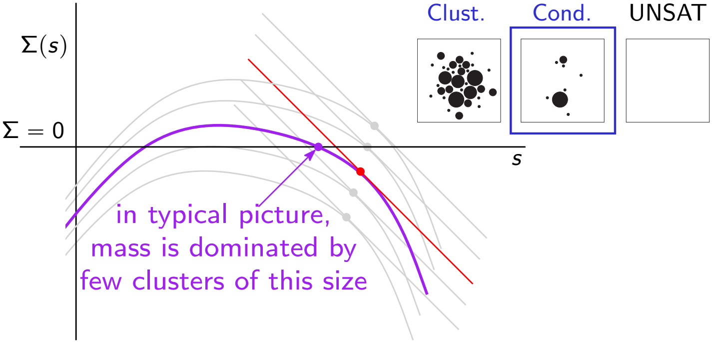
Complexity function $\Sigma \equiv \Sigma_\alpha(s)$ such that:

$$\mathbb{E}Z = \sum \underbrace{(\text{cluster size})}_{\exp\{ns\}} \times \underbrace{\mathbb{E}[\text{number of clusters of that size}]}_{\exp\{n\Sigma(s)\}}$$

$\mathbb{E}Z$ is dominated by s where $\Sigma'(s) \equiv -1$ (depending on α).



Condensation and non-concentration



The 1-RSB prediction:

- Satisfiability Threshold

$$\alpha_{\text{sat}=\sup\{\alpha:\sup_s \Sigma(s)\geqslant 0\}}$$

- Condensation Threshold and free energy

$$\alpha_{\text{cond}} = \sup\left\{\alpha : \sup_s s + \Sigma(s) = \sup_{s:\Sigma(s)\geqslant 0} s + \Sigma(s)\right\}$$

$$\Phi = \lim_{n\to\infty} \frac{1}{n} \log Z = \sup\{s + \Sigma(s) : \Sigma(s) > 0\} = \sup\{s : \Sigma(s) > 0\}$$

Results beyond the condensation threshold:

Results beyond the condensation threshold:

Condensation Threshold:

Random k-Colourings G(n,p) large k [Bapst, Coja-Oghlan, Hetterich, Rassmann, Vilenchik]

Regular k-NAESAT large k [S', Sun, Zhang]

Condensation Regime Free Energy:

Regular k-NAESAT large k [S', Sun, Zhang]

Satisfiability Threshold:

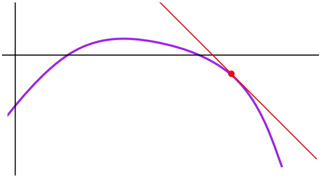
Regular NAESAT large k [Ding, S', Sun]

Maximum Independent Set d-Regular, large d [Ding, S', Sun]

Regular SAT, large k [Coja-Oghlan, Panagiotou]

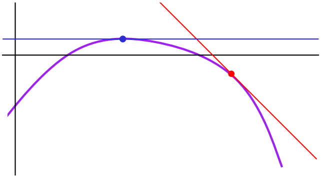
Random k-SAT, large k [Ding, S', Sun]

Free Energy



$$\mathbb{E}Z = \sum_s \exp\{n[1 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -1.$$

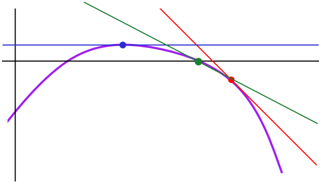
Free Energy



$$\mathbb{E}Z = \sum_s \exp\{n[1 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -1.$$

$$\mathbb{E}|\Omega| = \sum_s \exp\{n[0 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = 0.$$

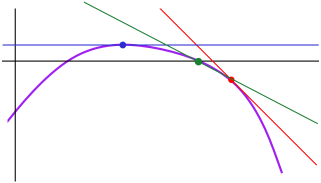
Free Energy



$$\mathbb{E}Z = \sum_s \exp\{n[1 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -1.$$

$$\mathbb{E}|\Omega| = \sum_s \exp\{n[0 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = 0.$$

Free Energy Weight clusters by (their size)$^{\lambda}$

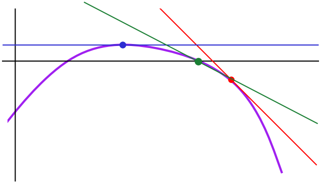


$$\mathbb{E}Z = \sum_s \exp\{n[1 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -1.$$

$$\mathbb{E}|\Omega| = \sum_s \exp\{n[0 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = 0.$$

$$\mathbb{E}Z_\lambda \equiv \sum_s \exp\{n[\lambda \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -\lambda$$

Free Energy Weight clusters by (their size)$^\lambda$



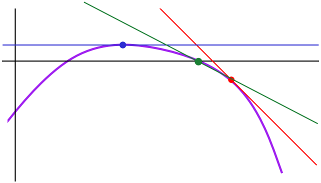
$$\mathbb{E}Z = \sum_s \exp\{n[1 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -1.$$

$$\mathbb{E}|\Omega| = \sum_s \exp\{n[0 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = 0.$$

$$\mathbb{E}Z_\lambda \equiv \sum_s \exp\{n[\lambda \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -\lambda$$

In fact, $\frac{1}{n} \log \mathbb{E}Z_\lambda$ is the Legendre transformation of $\Sigma(s)$.

Free Energy Weight clusters by (their size)$^\lambda$



$$\mathbb{E}Z = \sum_s \exp\{n[1 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -1.$$

$$\mathbb{E}|\Omega| = \sum_s \exp\{n[0 \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = 0.$$

$$\mathbb{E}Z_\lambda \equiv \sum_s \exp\{n[\lambda \cdot s + \Sigma(s)]\}, \qquad \text{maximized at } \Sigma'(s) = -\lambda$$

In fact, $\frac{1}{n} \log \mathbb{E}Z_\lambda$ is the Legendre transformation of $\Sigma(s)$.

The moments of Z_λ may be computed by adding local weights to the free variables in the $\{+, -, \mathtt{f}\}$ configurations.

Counting solutions within a cluster

We divide the subgraph of `f`'s into a forest of O(1)-size trees such that assigning values to one tree does not affect the others.

Counting solutions within a cluster

We divide the subgraph of `f`'s into a forest of O(1)-size trees such that assigning values to one tree does not affect the others.

Every edge encodes the '`f`-tree' it resides in.

$$\Rightarrow \underline{\tau} \in \{+, -, \mathtt{f}\}^{V(\mathscr{G})} \leftrightarrow \underline{\sigma} \in \{\mathtt{f}\text{-trees}\}^{E(\mathscr{G})}$$

Counting solutions within a cluster

We divide the subgraph of `f`'s into a forest of O(1)-size trees such that assigning values to one tree does not affect the others.

Every edge encodes the '`f`-tree' it resides in.

$$\Rightarrow \underline{\tau} \in \{+, -, \mathtt{f}\}^{V(\mathscr{G})} \leftrightarrow \underline{\sigma} \in \{\mathtt{f}\text{-trees}\}^{E(\mathscr{G})}$$

We can use BP algorithm write the number of solutions on trees as a product of weights.

Counting solutions within a cluster

We divide the subgraph of $\mathtt{f}$'s into a forest of O(1)-size trees such that assigning values to one tree does not affect the others.
Every edge encodes the '$\mathtt{f}$-tree' it resides in.

$$\Rightarrow \underline{\tau} \in \{+, -, \mathtt{f}\}^{V(\mathscr{G})} \leftrightarrow \underline{\sigma} \in \{\mathtt{f}\text{-trees}\}^{E(\mathscr{G})}$$

We can use BP algorithm write the number of solutions on trees as a product of weights.

Define weight functions Ψ_v, Ψ_a, Ψ_e accordingly s.t. for each $\underline{\sigma} \in \{\mathtt{f}\text{-trees}\}^{E(\mathscr{G})}$

$$\begin{aligned} w(\underline{\sigma}) &\equiv \prod_v \Psi_v(\underline{\sigma}_{\delta v}) \prod_a \Psi_a(\underline{\sigma}_{\delta a}) \prod_{e=(av)} \Psi_e(\underline{\sigma}_{(av)}) \\ &= \prod_T (\#\text{ of ways of assigning } \mathtt{f}\text{'s. in tree } T) \\ &= (\text{size of cluster}) \end{aligned}$$

Counting solutions within a cluster

We divide the subgraph of $\mathtt{f}$'s into a forest of O(1)-size trees such that assigning values to one tree does not affect the others.
Every edge encodes the '$\mathtt{f}$-tree' it resides in.

$$\Rightarrow \underline{\tau} \in \{+, -, \mathtt{f}\}^{V(\mathscr{G})} \leftrightarrow \underline{\sigma} \in \{\mathtt{f}\text{-trees}\}^{E(\mathscr{G})}$$

We can use BP algorithm write the number of solutions on trees as a product of weights.

Define weight functions Ψ_v, Ψ_a, Ψ_e accordingly s.t. for each $\underline{\sigma} \in \{\mathtt{f}\text{-trees}\}^{E(\mathscr{G})}$

$$\begin{aligned} w(\underline{\sigma}) &\equiv \prod_v \Psi_v(\underline{\sigma}_{\delta v}) \prod_a \Psi_a(\underline{\sigma}_{\delta a}) \prod_{e=(av)} \Psi_e(\underline{\sigma}_{(av)}) \\ &= \prod_T (\#\text{ of ways of assigning } \mathtt{f}\text{'s. in tree } T) \\ &= (\text{size of cluster}) \end{aligned}$$

Then we can define

$$Z_\lambda \equiv \textstyle\sum_{\underline{\sigma}} w^\lambda(\underline{\sigma}).$$

Optimization

We can write

$$\mathbb{E}Z_\lambda = \frac{\sum_{(\mathscr{G},\underline{\sigma})} w^\lambda(\underline{\sigma})}{\#\mathscr{G}}$$

Optimization

We can write

$$\mathbb{E}Z_\lambda = \frac{\sum_{(\mathscr{G},\underline{\sigma})} w^\lambda(\underline{\sigma})}{\#\mathscr{G}}$$

Then partitioning $\underline{\sigma}$ according to its empirical distribution ν,

$$\begin{aligned}\mathbb{E}Z_\lambda[\nu] &= \frac{\binom{n}{n\dot{\nu}}\binom{\alpha n}{\alpha n\hat{\nu}}}{\binom{dn}{dn\bar{\nu}}}{\Psi_v}^{\lambda n\dot{\nu}}{\Psi_a}^{\lambda\alpha n\hat{\nu}}{\Psi_e}^{\lambda dn\bar{\nu}} \\ &\equiv \exp\{n\,[\Sigma(\nu) + \lambda s(\nu)] + o(n)\} \\ &\equiv \exp\{n\,\Phi_\lambda(\nu) + o(n)\}\end{aligned}$$

Optimization

We can write

$$\mathbb{E}Z_\lambda = \frac{\sum_{(\mathscr{G},\underline{\sigma})} w^\lambda(\underline{\sigma})}{\#\mathscr{G}}$$

Then partitioning $\underline{\sigma}$ according to its empirical distribution ν,

$$\begin{aligned}\mathbb{E}Z_\lambda[\nu] &= \frac{\binom{n}{n\dot{\nu}}\binom{\alpha n}{\alpha n\hat{\nu}}}{\binom{dn}{dn\bar{\nu}}} {\Psi_v}^{\lambda n\dot{\nu}} {\Psi_a}^{\lambda\alpha n\hat{\nu}} {\Psi_e}^{\lambda dn\bar{\nu}} \\ &\equiv \exp\{n\,[\Sigma(\nu) + \lambda s(\nu)] + o(n)\} \\ &\equiv \exp\{n\,\Phi_\lambda(\nu) + o(n)\}\end{aligned}$$

Can find optimal $\nu^\star$ by finding fixed points of the Belief Propagation equations (e.g. Dembo–Montanari–Sun '13.)

Optimization

We can write

$$\mathbb{E}Z_\lambda = \frac{\sum_{(\mathscr{G},\underline{\sigma})} w^\lambda(\underline{\sigma})}{\#\mathscr{G}}$$

Then partitioning $\underline{\sigma}$ according to its empirical distribution ν,

$$\begin{aligned}\mathbb{E}Z_\lambda[\nu] &= \frac{\binom{n}{n\dot{\nu}}\binom{\alpha n}{\alpha n\hat{\nu}}}{\binom{dn}{dn\bar{\nu}}} {\Psi_v}^{\lambda n\dot{\nu}} {\Psi_a}^{\lambda\alpha n\hat{\nu}} {\Psi_e}^{\lambda dn\bar{\nu}} \\ &\equiv \exp\{n\,[\Sigma(\nu) + \lambda s(\nu)] + o(n)\} \\ &\equiv \exp\{n\,\Phi_\lambda(\nu) + o(n)\}\end{aligned}$$

Can find optimal $\nu^\star$ by finding fixed points of the Belief Propagation equations (e.g. Dembo–Montanari–Sun '13.)

For regular NAE-SAT and $k \geqslant k_0$, the limit $\Phi(\alpha)$ exists for $\alpha_{\text{cond}<\alpha<\alpha_{\text{sat}}}$, given by an explicit formula matching the 1-RSB prediction from statistical physics. S., Sun, Zhang '16

New Results

Theorem (Nam, S., Sohn 20+) *For $k \geqslant k_0$ (absolute constant), random regular k-NAESAT, WHP the largest and second largest clusters both have a constant fraction of the set total solutions. Two uniformly chosen solutions have normalized hamming distance concentrated on* **two** *points*.

- Requires estimating the partition function up to multiplicative $O(1)$ factor.
- States space of free trees is unbounded.

Thanks!