Large Hinge Width on Sparse Random Hypergraphs *

Tian Liu † Xiaxiang Lin † Chaoyi Wang † Kaile Su † Ke Xu ‡ §

Abstract

Consider random hypergraphs on n vertices, where each k-element subset of vertices is selected with probability p independently and randomly as a hyperedge. By sparse we mean that the total number of hyperedges is $O(n)$ or $O(n \ln n)$. When $k = 2$, these are exactly the classical Erdös-Rényi random graphs $G(n, p)$. We prove that with high probability, hinge width on these sparse random hypergraphs can grow linearly with the expected number of hyperedges. Some random constraint satisfaction problems such as Model RB and Model RD have satisfiability thresholds on these sparse constraint hypergraphs, thus the large hinge width results provide some theoretical evidence for random instances around satisfiability thresholds to be hard for a standard hinge-decomposition based algorithm. We also conduct experiments on these and other kinds of random graphs with several hundreds vertices, including regular random graphs and power law random graphs. The experimental results also show that hinge width can grow linearly with the number of edges on these different random graphs. These results may be of further interests.

1 Introduction

Constraint Satisfaction Problems (CSPs) cover many important $\mathcal{NP}$-hard problems in AI research. In the past, there are two fruitful lines of research on CSPs. One is about structural decomposition; another is about random CSPs. The start point of structural decomposition is to identify tractable subclasses of CSPs. The main results are in this form: CSPs with constraint hypergraphs of bounded structural width are tractable. In particular, all known structural decomposition based solvers run in time $||I||^{O(w)}$, where $||I||$ is input size and w is a structural width such as the size of minimum loop-cutset, hinge width, tree-width, query width, (fractional) hypertree width, spread-cut, etc. A plethora of structural decomposition methods have been developed and compared with each other [Gottlob *et al.*, 2000; Cohen *et al.*, 2008; Greco and Scarcello, 2010]. For a recent survey, see [Dechter, 2006]. On the other hand, the start point of random CSPs is to identify hard instances to serve as benchmarks, such as random instances around the satisfiability threshold of random CSPs [Cheeseman *et al.*, 1991; Mitchell *et al.*, 1992; Selman *et al.*, 1996; Cook and Mitchell, 1997; Xu and Li, 2000; Gao and Culberson, 2007; Xu *et al.*, 2007]. The main results are in this form: there is an easy-hard-easy transition around the satisfiability threshold of random CSPs, and the hardest instances are around the satisfiability thresholds. For a recent survey, see [Gomes and Walsh, 2006].

However, a rigorous link between phase transition and hardness of random instances is still unestablished. By rigorous link we mean that no polynomial time solver can exist at the satisfiability thresholds. This is still out of our current proof techniques. Instead, we might show some theoretical evidence for random instances around the satisfiability thresholds to be hard for some specific solvers. For example, in most structural decomposition methods, after finding a decomposition of the given CSP instance, the join operation is performed on constraint relations contained in each node of the decomposition, to formulate a new solution-equivalent tree-like CSP instance. Each decomposition of a CSP instance of large (i.e. unbounded) width contains a node with a large number of variables. Performing a join on all the variables in such a node is typically of high computational cost. Therefore, a large structural width around the satisfiability threshold can provide some theoretical evidence for these random instances to be hard for that kind of structural decomposition based algorithm.

The most popular structural width is tree-width, see e.g. [Kloks, 1994]. Gao is the first to study tree-width on random hypergraphs from considerations of Constraint Satisfaction and Bayesian Networks [Gao, 2003]. For other structural

*Research supported by the National 973 Program of China (2010CB328103), the NSFC grants (60725207, 60821003 and 60973033), the Open Fund of the State Key Laboratory of Software Development Environment (SKLSDE-2010KF-0X), and the ARC Future Fellowship (FT0991787). We thank Yang Yuan for helpful contributions, and the unknown reviewers for helpful comments.

†Institute of Software, School of Electronics Engineering and Computer Science, Key Laboratory of High Confidence Software Technologies (Peking University), Ministry of Education, Peking University, Beijing 100871, China. Email: {lt,sukl}@pku.edu.cn.

‡State Key Laboratory of Software Development Environment, Beihang University, Beijing 100083, China.

§Corresponding author: Ke Xu (kexu@nlsde.buaa.edu.cn).

width related to CSPs, very little is known on random hypergraphs. In this paper, we show a large hinge width on sparse random hypergraphs. Hinge decomposition was introduced by [Gyssens and Paredaens, 1984; Gyssens *et al.*, 1994] and was further investigated by e.g. [Gottlob *et al.*, 2000; 2001; Cohen *et al.*, 2008; Greco and Scarcello, 2010]. The standard hinge decomposition based algorithm also has a stage of hinge decomposition and a stage of join operation. Compared with tree-width, hinge width has two characters. First, hinge width and tree-width are incomparable, in the sense that there exists family of hypergraphs with bounded tree-width and unbounded hinge width, and vice versa [Gottlob *et al.*, 2000]. Second, hinge width is efficiently computable [Gyssens *et al.*, 1994], while tree-width is not.

We show that for random hypergraphs on n vertices, where each k-element subset of vertices is selected with probability p independently and randomly to be a hyperedge, when the total number of edges is $O(n)$ or $O(n \ln n)$, the hinge width can grow linearly with or even asymptotically equal to the number of hyperedges. In particular, for every fixed positive integer $k \geq 2$, when $p = O\left(\frac{1}{n^{k-1}}\right)$ and the expected number of edges $\overline{m} = O(n)$, the hinge width is $\Omega(\overline{m})$, when $p = O\left(\frac{\ln n}{n^{k-1}}\right)$, the hinge width is $\overline{m} - O(\overline{m})$.

Some random CSPs have satisfiability thresholds on sparse random constraint hypergraphs. Two concrete examples are Model RB and Model RD [Xu and Li, 2000]. Benchmarks based on Model RB and Model RD have been successfully applied in CSP solver competitions since the year 2005, and in many research papers on algorithms. The large hinge width results provide further theoretical evidence on hardness of Model RB and Model RD, besides the known results in [Xu and Li, 2006; Xu *et al.*, 2007].

We also conduct experiments on these and other kinds of random graphs with several hundreds vertices, including regular random graphs and power law random graphs. The experimental results show that hinge width can also grow linearly with the expected number of hyperedges on these different random graphs and match well with our asymptotic analysis on classical sparse random hypergraphs. These results may be of further interests such as in recent practice [Ansótegui *et al.*, 2007; 2008].

2 CSPs and Random Hypergraphs

A CSP instance is a triple $(\mathcal{V}, \mathcal{D}, \mathcal{C})$, where

- $\mathcal{V} = \{v_1, \ldots, v_n\}$ is a set of n variables;
- $\mathcal{D}$ is a finite set of values called *domain*, whose size $|\mathcal{D}|$ can be either fixed or increasing with n;
- $\mathcal{C}$ is a set of constraints, and each constraint C_i is a pair (S_i, R_i), where
  - S_i is a k-tuple of variables, called *constraint scope*;
  - R_i is a subset of $\mathcal{D}^k$, called *constraint relation*.

A constraint $C_i = (S_i, R_i)$ is satisfied if the k-tuple of values assigned to variables in S_i is contained in R_i. A solution to a CSP instance is an assignment of values to all the variables that satisfies all constraints. The *constraint hypergraph* of a CSP instance is a hypergraph with variables as vertices and constraint scopes as hyperedges. For more on CSPs, see e.g. [Dechter, 2003; Lecoutre, 2009].

The constraint hypergraphs of random CSPs are random hypergraphs. For any fixed k, we use $G(n, p, k)$ to denote the probability space of k-uniform random hypergraphs, in which on n vertices, each k-element subset of vertices is selected with probability p independently at random as a hyperedge. When the total number of hyperedges is $O(n)$ or $O(n \ln n)$, they are called *sparse*. $G(n, p, 2)$ is exactly the Classical Erdös-Rényi random graph model $G(n, p)$ [Janson *et al.*, 2000]. We say events $\mathcal{Q}_n$ occur *with high probability* (w.h.p.), if $\lim_{n\to\infty} \Pr(\mathcal{Q}_n) = 1$, usually written as $\Pr\left(\overline{\mathcal{Q}_n}\right) = o(1)$. We will need the following two lemmas.

Definition 1 *Let G be a hypergraph with vertex set $V(G)$ and edge set $E(G)$, then a (χ, s)-*separator *of G is a partition (S, A, B) of $V(G)$, such that*

- *$|S| \leq \chi s$ and $|A|, |B| \geq s$;*
- *there is no edge connecting A and B.*

Lemma 1 *Let $G \in G(n, p = \frac{c}{n^{k-1}}, k)$. If $\epsilon > 0$, ϵn is unbounded, $\chi > 0$, $s = tn$, $0 < t \leq \frac{1}{\chi+2}$, t may change with n, and $c > \frac{k!(\chi t(1-\ln(\chi t))+(\ln 2)(1-\chi t)+\epsilon)}{kt-(1+\chi)^k t^k - k^2 t^2} > 0$, then with high probability, no (χ, s)-separator of G exists.*

Proof Under the given parameters, if there is a (χ, s)-separator with $|S| < \chi s$, then we can move some vertices from A or B to S, such that $S = \chi s$ and the resulting partition is still a (χ, s)-separator. Thus we only need to show that with high probability, there is no (χ, s)-separator with $|S| = \chi s$. Denote by $\mathcal{P}$ the set of all such separators.

Consider a partition $W = (S, A, B)$ with $|S| = \chi s$ and $|A| = a \geq s$. Every edge not connecting A and B is contained in $A \cup S$ or $B \cup S$. By inclusion-exclusion principle, the number of such edges is $\binom{a+\chi s}{k} + \binom{n-a}{k} - \binom{\chi s}{k}$. Let $\lambda(a) = \binom{n}{k} - \binom{a+\chi s}{k} - \binom{n-a}{k} + \binom{\chi s}{k}$ be the number of edges connecting A and B, then $\Pr(W \in \mathcal{P}) = (1-p)^{\lambda(a)}$. By union bound, $\Pr(\mathcal{P} \neq \emptyset) \leq 2\binom{n}{\chi s}\sum_{a=s}^{(n-\chi s)/2}\binom{n-\chi s}{a}(1-p)^{\lambda(a)}$.

Claim 1 $\lambda(a) \geq \frac{skn^{k-1} - k^2 s^2 n^{k-2} - (\chi+1)^k s^k}{k!}$ *for large n.*

Proof Recall that $n^{\underline{k}} = n(n-1)\cdots(n-k+1)$. Approximating by the first two terms in expansion of $n^{\underline{k}}$, we have $\binom{n}{k} = \frac{n^{\underline{k}}}{k!} \geq \frac{n^k - (\sum_{j=0}^{k-1} j)n^{k-1}}{k!} = \frac{n^k - \binom{k}{2}n^{k-1}}{k!}$. Similarly by the first three terms of $(n-a)^{\underline{k}}$, we have $\binom{n-a}{k} = \frac{(n-a)^{\underline{k}}}{k!} \leq \frac{n^k - (\sum_{j=a}^{a+k-1} j)n^{k-1} + \binom{k}{2}a^2 n^{k-2}}{k!} \leq \frac{n^k - (ak+\binom{k}{2})n^{k-1} + k^2 a^2 n^{k-2}}{k!}$. Also $\binom{a+\chi s}{k} = \frac{(a+\chi s)^{\underline{k}}}{k!} \leq \frac{(a+\chi s)^k}{k!}$. So $\lambda(a) \geq \frac{akn^{k-1} - k^2 a^2 n^{k-2} - (\chi s + a)^k}{k!}$. For $a \geq s$ and for large n, the minimum is achieved at $a = s$. □

Now we have $\binom{n}{\chi s} \leq \left(\frac{en}{\chi s}\right)^{\chi s}$ which is $e^{n\chi t(1-\ln(\chi t))}$, $2\sum_{a=s}^{(n-\chi s)/2}\binom{n-\chi s}{a} \leq \sum_{a=0}^{(n-\chi s)}\binom{n-\chi s}{a} = 2^{n-\chi s}$ which is $e^{n(\ln 2)(1-\chi t)}$, and $(1-p)^{\lambda(a)} \leq e^{-p\lambda(a)} \leq e^{-n\frac{c}{k!}(kt-(1+\chi)^k t^k - k^2 t^2)}$. All together, $\Pr(\mathcal{P} \neq \emptyset) \leq e^{n\left(\chi t(1-\ln(\chi t)) + (\ln 2)(1-\chi t) - \frac{c}{k!}(kt-(1+\chi)^k t^k - k^2 t^2)\right)} \leq e^{-\epsilon n} = o(1)$. □

Definition 2 *For hypergraph G and $U \subseteq V(G)$, we define*

$$edges(U) = \{e \in E(G) | e \cap U \neq \emptyset\}.$$

The following lemma gives a lower bound on $|edges(U)|$ in terms of a lower bound on $|U|$. The use of Chernoff Bound (see i.e. [Mitzenmacher and Upfal, 2005; Janson *et al.*, 2000]) is not necessary, but for sharp parameters.

Lemma 2 *Let $G \in G(n, p = \frac{c}{n^{k-1}}, k)$ and the expected number of edges of G is $\overline{m} = \binom{n}{k}\frac{c}{n^{k-1}} = O(n)$. For any $U \subseteq V(G)$, if $|U| \geq n - \tau n$, $0 < \tau < 1$, $\delta > 0$, $\epsilon > 0$, $\epsilon\tau n$ is unbounded and $c > \frac{(1-\ln\tau+\epsilon)k^k}{\tau^{k-1}e^k((1+\delta)\ln(1+\delta)-\delta)}$, then w.h.p., $|edges(U)| \geq \overline{m}(1-n^{-\frac{1}{3}}) - (1+\delta)(\frac{\tau e}{k})^k cn$.*

Proof Let $\mathcal{S} = \{S \subseteq V(G) || S| = \tau n\}$. Let X_S be the number of edges in subgraph induced by S.

Claim 2 *W.h.p. for all $S \in \mathcal{S}$, $X_S \leq (1+\delta)\left(\frac{\tau e}{k}\right)^k cn$.*

Proof The expectation of X_S is $\mathbb{E}(X_S) = \binom{\tau n}{k} p \leq \left(\frac{\tau n e}{k}\right)^k \frac{c}{n^{k-1}} = \left(\frac{\tau e}{k}\right)^k cn$. By the Chernoff bound, we have $\Pr\left(X_S > (1+\delta)\left(\frac{\tau e}{k}\right)^k cn\right) < \left(\frac{e^\delta}{(1+\delta)^{1+\delta}}\right)^{\left(\frac{\tau e}{k}\right)^k cn}$. By the union bound, $\Pr\left(\exists S \in \mathcal{S}, X_S > (1+\delta)\left(\frac{\tau e}{k}\right)^k cn\right) \leq \binom{n}{\tau n}\left(\frac{e^\delta}{(1+\delta)^{1+\delta}}\right)^{\left(\frac{\tau e}{k}\right)^k cn} \leq \left(\frac{en}{\tau n}\right)^{\tau n}\left(\frac{e^\delta}{(1+\delta)^{1+\delta}}\right)^{\left(\frac{\tau e}{k}\right)^k cn} = e^{\left(\ln\frac{e}{\tau} + c\frac{\tau^{k-1}e^k}{k^k}(\delta-(1+\delta)\ln(1+\delta))\right)\tau n} = e^{-\epsilon\tau n} = o(1)$. □

Claim 3 $|E(G)| \geq \overline{m}(1-n^{-\frac{1}{3}})$ *with high probability.*

Proof $\Pr(|E(G)| < \overline{m}(1-n^{-\frac{1}{3}})) \leq e^{-n^{-\frac{2}{3}}\frac{\overline{m}}{2}} = e^{-O(n^{\frac{1}{3}})} = o(1)$ by Chernoff bound. □

By these claims and Definition 2, w.h.p., $edges(U) \geq |E(G)| - X_S \geq \overline{m}(1-n^{-\frac{1}{3}}) - (1+\delta)(\frac{\tau e}{k})^k cn$. □

3 Hinge Width

Definition 3 *[Gyssens* et al.*, 1994; Gottlob* et al.*, 2000] Let G be a hypergraph, $H \subseteq E(G)$, and $F \subseteq E(G) - H$. $\bigcup H$ is the set of all vertices in edges in H. Then F is called* connected with respect to (w.r.t.) *H if, for any two edges, $e, f \in F$, there is a sequence of edges $e_1, \ldots, e_m$, such that:*

- $e_1 = e$;
- $e_i \cap e_{i+1} \not\subseteq \bigcup H$, *for $i = 1, \ldots, m-1$;*
- $e_m = f$.

The maximal connected subsets of $E(G) - H$ w.r.t. H is called connected component of $E(G) - H$ *w.r.t. H.*

Definition 4 *[Gyssens* et al.*, 1994; Gottlob* et al.*, 2000] Let G be a hypergraph, and let H be either $E(G)$ or a subset of $E(G)$ containing at least two edges. Let $H_1, \ldots, H_m$ be the connected components of $E(G) - H$ w.r.t. H. Then H is called a* hinge *if, for $i = 1, \ldots, m$, there exists an edge $h_i \in H$ such that*

$$(\bigcup H) \cap (\bigcup H_i) \subseteq h_i.$$

Definition 5 *[Gyssens* et al.*, 1994; Gottlob* et al.*, 2000] A* hinge decomposition *of hypergraph G is a tree $T = (N, A)$, with nodes N and labeled arcs A, such that*

1. *the tree nodes are minimal hinges of G;*
2. *each edge in $E(G)$ is contained in at least one tree node;*
3. *two adjacent tree nodes share precisely one edge which is the label of the arc connecting the two nodes; moreover, their shared variables are precisely the members of this edges;*
4. *the vertices of G shared by two tree nodes are entirely contained in each tree node on their connecting path.*

It was shown in [Gyssens *et al.*, 1994] that the cardinality of the largest node in any hinge decomposition of G is an invariant of G, which is the cardinality of the largest minimal hinge. We call it *hinge width* of G, and denote it by $hw(G)$.

The following Algorithm 1 is from [Gyssens *et al.*, 1994].

Input: A hypergraph $G = (V, E)$
Output: A hinge decomposition T for G

1: Mark each edge in E as *unused*. Set $i = 0$, $N_0 = \{E\}$ and $A_0 = \emptyset$, and mark the node in N_0 as *non-minimal*.
2: If all nodes of N_i are marked *minimal*, then set $T = (N_i, A_i)$ and stop. Else choose a *non-minimal* node F in N_i.
3: If all edges in F are marked as *used*, then mark F as *minimal* and goto 2. Else, choose an *unused* edge $e \in F$ and mark e as *used*.
4: Let $\Gamma = \{D \cup \{e\} | D$ is a connected component of $F - e$ with respect to $e\}$ and $\gamma : F \to \Gamma$ be any function such that for all $f \in F$, $f \in \gamma(f)$. If $|\Gamma| = 1$, then goto 3.
5: Set
$$N_{i+1} = (N_i - \{F\}) \cup \Gamma$$
$$\begin{aligned} A_{i+1} = (A_i &- \{(\{F, F'\}, f) \,|\, (\{F, F'\}, f) \in A_i)\}) \\ &\cup \{(\{\gamma(f), F'\}, f) \,|\, (\{F, F'\}, f) \in A_i\} \\ &\cup \{(\{\gamma(f), \gamma(e)\}, e) \,|\, f \in F, \gamma(f) \neq \gamma(e)\} \end{aligned}$$
and mark all nodes newly added to N_{i+1} as *non-minimal*.
6: Increment i and goto 2.

Algorithm 1: Computing a hinge decomposition

4 Asymptotic Analysis

A complete run from steps 2 to 6 in Algorithm 1 is called a *round*. For all round i, assume that $N_i = \{H_1^{(i)}, \ldots, H_{p_i}^{(i)}\}$. Let $H_{max}^{(i)}$ be the hinge in N_i with the maximum number of vertices $|\bigcup H_{max}^{(i)}|$. Suppose that Algorithm 1 stops when $i = I$. Then $|H_{max}^{(I)}|$ is a lower bound of $hw(G)$.

Definition 6 *When step 5 is running, the edge e in steps 3 to 5 is called a* separating edge *[Gyssens* et al.*, 1994]. Denote by $\partial(H_j^{(i)})$ the set of all separating edges in $H_j^{(i)}$.*

Proposition 1 *For all i and j, there is no edge in G connecting $V(G) - \bigcup H_j^{(i)}$ and $\bigcup H_j^{(i)} - \bigcup \partial(H_j^{(i)})$.*

Proof By steps 4 and 5 in Algorithm 1, except possibly by a separating edge, there is no other edge connecting vertices in $\bigcup H_{j_1}^{(i)}$ and vertices in $\bigcup H_{j_2}^{(i)}$ for $j_1 \neq j_2$. □

To lower bound $|H_{max}^{(I)}|$, let $U = \bigcup H_{max}^{(I)} - \bigcup \partial(H_{max}^{(I)})$. By Definition 2 and Proposition 1, $edges(U) \subseteq H_{max}^{(I)}$. Thus $hw(G) \geq |H_{max}^{(I)}| \geq |edges(U)|$. Below we will first get an upper bound on I and a lower bound on $|U|$ (Lemma 6), and then by Lemma 2 get a lower bound on $|edges(U)|$.

Lemma 3 *For all i, $|\bigcup H_{max}^{(i)}| \leq n - i$.*

Proof For all i, we have the following three claims.

Claim 4 *For all j, $|\bigcup H_j^{(i)}| \geq k + 1$.*

Proof By Definition 4 about hinge, $|H_j^{(i)}| \geq 2$. Two different k-uniform hyperedges have at least $k + 1$ vertices. □

Claim 5 $\sum_{j=1}^{p_i} |\bigcup H_j^{(i)}| \leq n + (p_i - 1)k$.

Proof Recall that (N_i, A_i) is a tree with p_i hinge nodes and $p_i - 1$ tree edges. By step 5 in Algorithm 1, every tree edge is labeled by a separating edge, and every pair of hinge nodes connected by a tree edge shares the separating edge. So in the sum $\sum_{j=1}^{p_i} |\bigcup H_j^{(i)}|$, separating edges are totally recounted by $p_i - 1$ times and each separating edge has k vertices. □

Claim 6 $p_i \geq i + 1$.

Proof By induction. When $i = 0$, $N_0 = \{E\}$, so $p_0 = 1$. In each round, at step 5 in Algorithm 1, an old hinge is split into at least two new hinges, so $p_{i+1} \geq p_i + 1$. □

Now by Claim 5, $\sum_{j=1}^{p_i} |\bigcup H_j^{(i)}| \leq n + (p_i - 1)k$. By Claim 4, $\sum_{j=2}^{p_i} |\bigcup H_j^{(i)}| \geq (p_i - 1)(k + 1)$. By Claim 6, $p_i \geq i + 1$. Without lose of generality, we may assume that $H_{max}^{(i)} = H_1^{(i)}$, so $|\bigcup H_{max}^{(i)}| = |\bigcup H_1^{(i)}| \leq n + (p_i - 1)k - (p_i - 1)(k + 1) = n - p_i + 1 \leq n - i$. □

Lemma 4 *For all i and s, if $(k + 1)s \leq |\bigcup H_{max}^{(i)}| \leq n - s$ and $i \leq s$, then there is a $(k, s) - separator$ in G.*

Proof Let $S = \bigcup \partial(H_{max}^{(i)})$, $A = \bigcup H_{max}^{(i)} \setminus S$, and $B = V(G) \setminus \bigcup H_{max}^{(i)}$. Clearly, (S, A, B) is a partition of $V(G)$. Below we show that (S, A, B) is a (k, s)-separator of G.

Claim 7 *For all i, $|\partial(H_{max}^{(i)})| \leq |\bigcup_{j=1}^{p_i} \partial(H_j^{(i)})| = i$.*

Proof At each round, exactly one new separating edge e is added to $\bigcup_{j=1}^{p_i} \partial(H_j^{(i)})$ at steps 3 to 5 in Algorithm 1. □

Now by Claim 7 and $i \leq s$, $|\partial H_{max}^{(i)}| \leq i \leq s$. Each hyperedge in $\partial H_{max}^{(i)}$ has k vertices, so $|S| \leq ks$. By $|\bigcup H_{max}^{(i)}| \geq (k + 1)s$, $|A| \geq (k + 1)s - ks = s$. By $|\bigcup H_{max}^{(i)}| \leq n - s$, $|B| \geq n - (n - s) = s$. By Proposition 1, no edge connects A and B. □

Lemma 5 *For all i and s, if $|\bigcup H_{max}^{(i)}| < (k + 1)s < n$ and $i \leq s < \frac{n}{2k+3}$, then there is a $(k, s) - separator$ in G.*

Proof Since $|\bigcup_{j=1}^{p_i} (\bigcup H_j^{(i)})| = |V(G)| = n$, there is a j_0 such that $(k + 1)s \leq |\bigcup_{j=1}^{j_0} (\bigcup H_j^{(i)})|$. Since $s < \frac{n}{2k+3}$, $n - s > 2(k+1)s$. By $|\bigcup H_{max}^{(i)}| < (k+1)s$, we can assume that $|\bigcup_{j=1}^{j_0} (\bigcup H_j^{(i)})| \leq n - s$. Let $S = \bigcup_{j=1}^{j_0} (\bigcup \partial(H_j^{(i)}))$, $A = \bigcup_{j=1}^{j_0} (\bigcup H_j^{(i)}) \setminus S$, and $B = V(G) \setminus \bigcup_{j=1}^{j_0} (\bigcup H_j^{(i)})$. The remaining part is similar to the proof of Lemma 4. □

Lemma 6 *Let $G \in G(n, p = \frac{c}{n^{k-1}}, k)$. For any $s = tn$ and $0 < t < \frac{1}{2k+3}$, if c satisfies the conditions in Lemma 1, then $I < s$ and $|U| > n - (k + 1)s$ with high probability.*

Proof By following three claims.

Claim 8 *For all $i \leq s$, $|\bigcup H_{max}^{(i)}| > n - s$ with high probability.*

Proof Otherwise by Lemma 4 and Lemma 5, with non-vanishing probability there is a (k, s)-separator of G, contradiction to Lemma 1. □

Claim 9 *$I < s$ with high probability.*

Proof By Claim 8, $|\bigcup H_{max}^{(s)}| > n - s$. If $I \geq s$, then by Lemma 3, $|\bigcup H_{max}^{(s)}| \leq n - s$, a contradiction. □

Claim 10 *$|U| > n - (k + 1)s$ with high probability.*

Proof By Claim 9 and Claim 8, $|\bigcup H_{max}^{(I)}| > n - s$. By Claim 7 and Claim 9, $|\partial(H_{max}^{(I)})| \leq I < s$. Each edge in G has k vertices, so $|\bigcup \partial(H_{max}^{(I)})| < ks$. Thus, $|U| = |\bigcup H_{max}^{(I)}| - |\bigcup \partial(H_{max}^{(I)})| > n - (k + 1)s$. □

This finishes the proof of Lemma 6. □

Theorem 1 *Let $G \in G(n, p = \frac{c}{n^{k-1}}, k)$ and the expected number of edges of G is $\overline{m} = \binom{n}{k} \frac{c}{n^{k-1}} = O(n)$. If $s = tn$, $0 < t \leq \frac{1}{2k+3}$, $\epsilon > 0$, ϵtn is unbounded, $\delta > 0$, $c > \frac{k!(kt(1-\ln(kt))+(\ln 2)(1-kt)+\epsilon)}{kt-(1+k)^k t^k - k^2 t^2} > 0$ and $c > \frac{(1-\ln((k+1)t)+\epsilon)k^k}{[(k+1)t]^{k-1} e^k [(1+\delta)\ln(1+\delta)-\delta]}$, then w.h.p., $hw(G) = \Omega(\overline{m})$.*

Proof Set $\tau = (k + 1)t$, we can easily check that the parameters satisfy conditions in Lemma 6 and Lemma 2. Thus by discussions at the beginning of this section, with high probability $hw(G) \geq |H_{max}^{(I)}| \geq |edges(U)| = \Omega(\overline{m})$. □

Theorem 2 *Let $G \in G(n, p = \frac{b \ln n}{n^{k-1}}, k)$ and the expected number of edges of G is $\overline{m} = \binom{n}{k} \frac{b \ln n}{n^{k-1}} = O(n \ln n)$. For any $b > 0$, $hw(G) = \overline{m} - o(\overline{m})$ with high probability.*

Proof Similar to proof of Theorem 1. First, let $s = tn$, $t = \left(\frac{(\ln 2)(k-1)!}{b} + \epsilon\right) \frac{1}{\ln n}$ for any constant $\epsilon > 0$, then all conditions in Lemma 6 are satisfied. Next, let $c = b \ln n$, $\tau = (k + 1)t = \frac{d}{\ln n}$ where $d = (k + 1) \left(\frac{(\ln 2)(k-1)!}{b} + \epsilon\right)$, and $\delta = (\ln n)^{k-1}$, then all conditions in Lemma 2 are satisfied. Finally, $|edges(U)| \geq \overline{m}(1 - n^{-\frac{1}{3}}) - (1+\delta)(\frac{\tau e}{k})^k cn = \overline{m}(1 - n^{-\frac{1}{3}}) - \frac{1+(\ln n)^{k-1}}{(\ln n)^k} (\frac{de}{k})^k cn = \overline{m} - o(\overline{m})$. □

5 Model RB and Model RD

Two concrete random CSPs to apply Theorem 2 are Model RB and Model RD [Xu and Li, 2000]. Model RB is like this:

- given n variables each with domain $\{1, 2, ..., d\}$, where $d = n^\alpha$ and $\alpha > 0$ is constant;
- select with repetition $m = rn \ln n$ random constraint scopes, for each scope select without repetition k of n variables, where $k \geq 2$ is an integer constant;
- select uniformly at random without repetition $(1-p)d^k$ compatible assignments for each constraint, where $0 < p < 1$ is constant.

In Model RD, in the last step above, each assignment for the k variables is selected with probability $1 - p$ as compatible independently. Model RB and Model RD have satisfiability thresholds $r_{cr} = -\frac{\alpha}{\ln(1-p)}$ [Xu and Li, 2000]. A large hinge width result will provide further theoretical evidence on hardness of Model RB and Model RD, besides the known results in [Xu and Li, 2006; Xu *et al.*, 2007].

Clearly Model RB and Model RD have the same space of random constraint hypergraphs, with constraint scopes as hyperedges. Let $G^{RB}(n, r, k)$ denote the probability space of these random constraint hypergraphs. We show an asymptotic equivalence between $G^{RB}(n, r, k)$ and $G(n, p, k)$ for $p = rn \ln n / \binom{n}{k} = O(\ln n)/n^{k-1}$, so that Theorem 2 can be applied to Model RB and Mode RD. We use the Ball and Bin model [Mitzenmacher and Upfal, 2005]. Suppose that we put a balls independently and uniformly at random into b bins. Then the joint distribution of the number of balls in each bin is referred to as in the *exact case*. In another case called the *Poisson case*, the number of balls in each bin is independent and has Poisson distribution with mean a/b.

Lemma 7 *[Mitzenmacher and Upfal, 2005] Any event occurring with probability p in Poisson case occurs with probability at most* $pe\sqrt{a}$ *in exact case.*

Lemma 8 *Let* $\mathcal{Q}_n$ *be an arbitrary graph property,* $r > 0$ *and* $p = rn \ln n / \binom{n}{k}$*. Then* $\Pr_{G^{RB}(n,r,k)}(\mathcal{Q}_n) \leq e\sqrt{rn \ln n} \Pr_{G(n,p,k)}(\mathcal{Q}_n)$.

Proof Treating edges as balls and all k-element subsets of vertices as bins, then $G^{RB}(n, r, k)$ is no more than the exact case where $a = rn \ln n$ and $b = \binom{n}{k}$, and Poisson case is equivalent to $G(n, p, k)$ with $p = rn \ln n / \binom{n}{k}$. Then this lemma follows from Lemma 7. □

Corollary 1 *Let* $G \in G^{RB}(n, r, k)$*. Then for any constant* $r > 0$*, with high probability,* $hw(G) = rn \ln n - O(n)$.

Proof By Theorem 2 and the fact that all $o(1)$'s in its proofs are something like $e^{-\epsilon n}$, which can subsume the $e\sqrt{rn \ln n}$ factor in the Lemma 8. □

6 Experiments

We do experiments on instances of a realistic size for different kinds of random graphs to complement the asymptotic analysis. For simplicity, we only consider graphs, i.e. $k = 2$. We let the number of vertices n increase from 10 to 300 with step length 10, and let the (expected) number of edges m be n, $2n$ and $0.4n \ln n$, respectively. At each pair (n, m), we generate 60 instances to average on hw/m, where hw is hinge width. We compare the results on *classical* random graphs $G(n, p)$, *regular* random graphs and *power law* random graphs. We use the configuration model [Janson *et al.*, 2000] to generate regular or power law instances with a given sequence of degrees. The results are displayed in Figures 1 to 3.

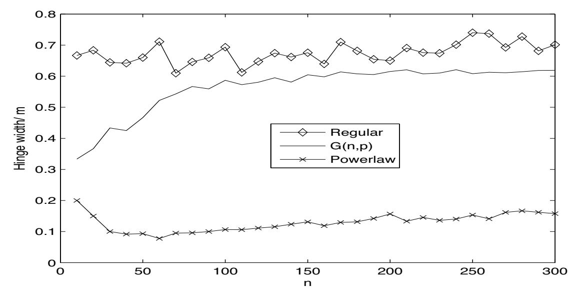

Figure 1: Results for $m = n$

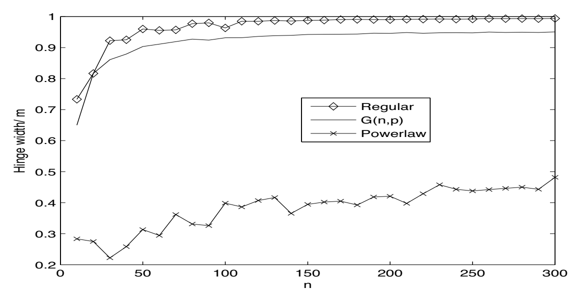

Figure 2: Results for $m = 2n$

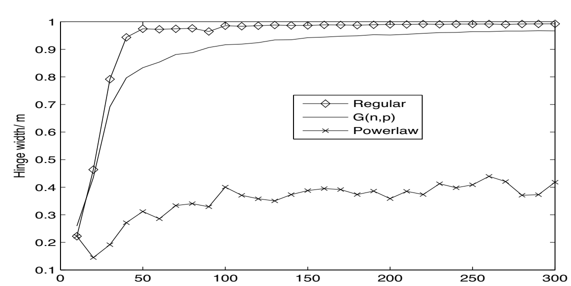

Figure 3: Results for $m = 0.4n \ln n$

From the figures, we can find that on all these random graphs, hinge width hw grows linearly with the number of edges m. For each pair (n, m), regular random graphs have the largest hinge width, while power law random graphs have the smallest hinge width. Hinge width is also called the *degree of cyclicity* of graphs [Gyssens *et al.*, 1994], graphs with more nodes of degree 1 or 2 will have smaller hinge width. Power law graphs do have more nodes of smaller degrees than regular graphs or $G(n, p)$. Moreover, the experimental results match well with asymptotic analysis on $G(n, p)$ already on instances of 100 to 300 vertices.

7 Conclusion and Future Work

We have shown that hinge width can grow linearly on sparse classical random hypergraphs. This provides theoretical evidence for random instances around the satisfiability thresholds to be hard for the standard hinge decomposition method, at least for Model RB and Model RD. We also conduct experiments on instances of a realistic size and the results match well with the asymptotic analysis. The hinge width in regular random graphs is the largest among all three kinds of random graphs, so this result supports the previous suggestions to use balanced random instances as benchmarks [Ansótegui *et al.*, 2007; 2008]. More interestingly, the hinge width in power law graphs is much lower than that in regular/classical random graphs. Since the power law graphs are more common in the real world, this implies that the hinge decomposition is still of potential use in the real-world instances. Our analysis and experiments depend essentially on the efficient hinge algorithm (Algorithm 1) given by [Gyssens *et al.*, 1994]. Thus tools developed in structural decomposition are useful, not only in solving restricted easy CSP instances, but also in analyzing random hard CSP instances. For other structural width, such as (fractional) hypertree width and spread-cut, a similar analysis or experiment on random instances should be an interesting future work. But this seems to be a challenge task. For example, we can easily handle hinge width on $1,000$ vertices, but can hardly handle tree-width exactly on more than 50 vertices, due to a lack of efficient tree-width algorithms.

References

[Ansótegui *et al.*, 2007] C. Ansótegui, R. Béjar, C. Fernàndez, and C. Mateu. On balanced csps with high treewidth. In *Proc. AAAI*, pages 161–166, 2007.

[Ansótegui *et al.*, 2008] C. Ansótegui, M. L. Bonet, J. Levy, and F. Manyà. Measuring the hardness of sat instances. In *Proc. AAAI*, pages 222–228, 2008.

[Cheeseman *et al.*, 1991] P. Cheeseman, R. Kanefsky, and W. Taylor. Where the really hard problems are. In *Proc. IJCAI*, pages 163–169, 1991.

[Cohen *et al.*, 2008] D. Cohen, P. Jeavons, and M. Gyssens. A unified theory of structural tractability for constraint satisfaction problems. *J. Computer and System Sci.*, 74(5):721–743, 2008.

[Cook and Mitchell, 1997] S. A. Cook and D. G. Mitchell. Finding hard instances of the satisfiability problem: a survey. In *DIMACS Series, Vol. 35*, pages 1–17, 1997.

[Dechter, 2003] Rina Dechter. *Constraint Processing*. Morgan Kaufmann, 2003.

[Dechter, 2006] R. Dechter. Tractable structures for constraint satisfaction problems. In *Handbook of constraint programming*, pages 209–244, 2006.

[Gao and Culberson, 2007] Y. Gao and J. Culberson. Consitency and random constraint satisfaction problems. *J. Artif. Intell. Res. (JAIR)*, 28:517–557, 2007.

[Gao, 2003] Y. Gao. Phase transition of tractability in constraint satisfaction and bayesian network inference. In *Proc. UAI*, pages 265–271, 2003.

[Gomes and Walsh, 2006] C. Gomes and T. Walsh. Randomness and structures. In *Handbook of constraint programming*, pages 639–664, 2006.

[Gottlob *et al.*, 2000] G. Gottlob, N. Leone, and F. Scarcello. A comparison of structural csp decomposition methods. *Artif. Intell.*, 124(2):243–282, 2000.

[Gottlob *et al.*, 2001] G. Gottlob, N. Leone, and F. Scarcello. The complexity of acyclic conjunctive queries. *J. American Comput. Mach.*, 43(3):431–498, 2001.

[Greco and Scarcello, 2010] G. Greco and F. Scarcello. On the power of structural decompositions of graph-based representations of constraint problems. *Artif. Intell.*, 174(5-6):382–409, 2010.

[Gyssens and Paredaens, 1984] M. Gyssens and J. Paredaens. A decomposition methodology for acyclic database. In *Advances in Database Theory, Vol. 2*, pages 85–122. Plenum Press, 1984.

[Gyssens *et al.*, 1994] M. Gyssens, P. Jeavons, and D. Cohen. Decomposing constraint satisfaction problems using database techniques. *Artif. Intell.*, 66(1):57–89, 1994.

[Janson *et al.*, 2000] S. Janson, T. Luczak, and A. Rucinski. *Random Graphs*. John Wiley & Sons, Inc., 2000.

[Kloks, 1994] T. Kloks. *Treewidth: computations and approximations, LNCS 842*. Springer, 1994.

[Lecoutre, 2009] Christophe Lecoutre. *Constraint Networks: Techniques and Algorithms*. ISTE/Wiley, 2009.

[Mitchell *et al.*, 1992] D. G. Mitchell, B. Selman, and H. J. Levesque. Hard and easy distributions of sat problems. In *Proc. AAAI*, pages 459–465, 1992.

[Mitzenmacher and Upfal, 2005] M. Mitzenmacher and E. Upfal. *Probability and Computing*. Cambridge University Press, 2005.

[Selman *et al.*, 1996] B. Selman, D. G. Mitchell, and H. J. Levesque. Generating hard satisfiability problems. *Artif. Intell.*, 81:17–29, 1996.

[Xu and Li, 2000] K. Xu and W. Li. Exact phase transitions in random constraint satisfaction problems. *J. Artif. Intell. Res. (JAIR)*, 12:93–103, 2000.

[Xu and Li, 2006] K. Xu and W. Li. Many hard examples in exact phase transitions. *Theoret. Comput. Sci*, 355(3):291–302, 2006.

[Xu *et al.*, 2007] K. Xu, F. Boussemart, F. Hemery, and C. Lecoutre. Random constraint satisfaction: Easy generation of hard (satisfiable) instances. *Artif. Intell.*, 171:514–534, 2007.