ETH_zürich_

Disproof of the Neighborhood Conjecture with Implications to SAT

Journal Article

Author(s):
Gebauer, Heidi

Publication date:
2012-05

Permanent link:
https://doi.org/10.3929/ethz-b-000061671

Rights / license:
In Copyright - Non-Commercial Use Permitted

Originally published in:
Combinatorica 32(5), https://doi.org/10.1007/s00493-012-2679-y

DISPROOF OF THE NEIGHBORHOOD CONJECTURE WITH IMPLICATIONS TO SAT*

HEIDI GEBAUER

We study a special class of binary trees. Our results have implications on Maker/Breaker games and SAT: We disprove a conjecture of Beck on positional games and construct an unsatisfiable k-CNF formula with few occurrences per variable, thereby improving a previous result by Hoory and Szeider and showing that the bound obtained from the Lovász Local Lemma is tight up to a constant factor.

A (k,s)-*CNF formula* is a boolean formula in conjunctive normal form where every clause contains exactly k distinct literals and every variable occurs in at most s clauses. The (k,s)-SAT problem is the satisfiability problem restricted to (k,s)-CNF formulas. Kratochvíl, Savický and Tuza showed that for every $k \geq 3$ there is an integer $f(k)$ such that every $(k, f(k))$-CNF formula is satisfiable, but $(k, f(k)+1)$-SAT is already NP-complete (it is not known whether $f(k)$ is computable). Kratochvíl, Savický and Tuza also gave the best known lower bound $f(k) = \Omega\left(\frac{2^k}{k}\right)$, which is a consequence of the Lovász Local Lemma. We prove that, in fact, $f(k) = \Theta\left(\frac{2^k}{k}\right)$, improving upon the best known upper bound $O\left((\log k) \cdot \frac{2^k}{k}\right)$ by Hoory and Szeider.

Finally we establish a connection between the class of trees we consider and a certain family of positional games. The Maker/Breaker game we study is as follows. Maker and Breaker take turns in choosing vertices from a given n-uniform hypergraph $\mathcal{F}$, with Maker going first. Maker's goal is to completely occupy a hyperedge and Breaker tries to prevent this. The *maximum neighborhood size* of a hypergraph $\mathcal{F}$ is the maximal s such that some hyperedge of $\mathcal{F}$ intersects exactly s other hyperedges. Beck conjectures that if the maximum neighborhood size of $\mathcal{F}$ is smaller than $2^{n-1} - 1$ then Breaker has a winning strategy. We disprove this conjecture by establishing, for every $n \geq 3$, the existence of an n-

---

uniform hypergraph with maximum neighborhood size $3 \cdot 2^{n-3}$ where Maker has a winning strategy. Moreover, we show how to construct, for every n, an n-uniform hypergraph with maximum degree at most $\frac{2^{n+2}}{n}$ where Maker has a winning strategy.

In addition we show that each n-uniform hypergraph with maximum degree at most $\frac{2^{n-2}}{en}$ has a proper halving 2-coloring, which solves another open problem posed by Beck related to the Neighborhood Conjecture.

1. Introduction

1.1. Trees

We first consider a special class of trees, which connect both to Maker/Breaker games and SAT. Throughout this paper, by a *binary tree* we mean a rooted tree where every node has either two or no children. In such a binary tree we say that a leaf v is *l-close* to a node w if w is an ancestor of v, at distance at most l from v. For positive integers k and d, we call a binary tree T a (k,d)-*tree* if (i) every leaf has depth at least $k-1$ and (ii) for every node u of T there are at most d leaves $(k-1)$-close to u. Clearly, every binary tree with all leaves at depth at least $k-1$ is a $(k, 2^{k-1})$-tree. The following lemma, which we will show in Section 4, will be the main ingredient in proving some new results on Maker/Breaker games and SAT.

Lemma 1.1. A $\left(k, \lfloor \frac{2^{k+2}}{k} \rfloor\right)$-*tree exists for every* $k \geq 1$.

Remark 1.2. Lemma 1.1 is not best possible. By using a series of tedious, technical arguments we can push down the value $\frac{2^{k+2}}{k}$ to $\frac{63}{64} \frac{2^{k-1}}{k}$ for infinitely many k. However, we chose not to include it here because more recently (in [8]) the smallest value $r = r(k)$ for which there exists a (k,r)-tree has been settled up to lower order terms: $r(k) = \left(\frac{1}{e} + O\left(\frac{1}{\sqrt{k}}\right)\right) \frac{2^k}{k}$.

1.2. SAT

Lemma 1.1 has implications on SAT. Following the standard notation we denote by (k,s)-*CNF* the set of boolean formulas $\mathcal{F}$ in conjunctive normal form where every clause of $\mathcal{F}$ has exactly k distinct literals and each variable occurs in at most s clauses of $\mathcal{F}$. Moreover, we denote by (k,s)-SAT the satisfiability problem restricted to formulas in (k,s)-CNF. Tovey [18] proved that every (3,3)-CNF formula is satisfiable. For k a positive integer let $f(k)$ be defined as the largest integer, so that every $(k, f(k))$-CNF formula is satisfiable. Tovey also showed that $f(3) = 3$ and that, moreover, (3,4)-SAT

is NP-complete. Kratochvíl, Savický and Tuza [12] generalized this result by showing that for every $k \geq 3$ $(k, f(k)+1)$-SAT is already NP-complete. This phenomenon, that $(k, f(k))$-SAT is trivial while $(k, f(k)+1)$-SAT is already NP-hard, is often referred to as *complexity jump*.

The best known lower bound for $f(k)$, a consequence of the Lovász Local Lemma, is due to Kratochvíl, Savický and Tuza [12].

Theorem 1.3. (Kratochvíl, Savický and Tuza [12]) $f(k) \geq \lfloor \frac{2^k}{ek} \rfloor$.

From the other side Savický and Sgall [15] showed that $f(k) = O(k^{(1-\alpha)} \cdot \frac{2^k}{k})$ where $\alpha = \log_3 4 - 1 \approx 0.26$. This was improved by Hoory and Szeider [9] who proved that $f(k) = O((\log k) \cdot \frac{2^k}{k})$, which is the best known upper bound. We close the gap between upper and lower bound up to a constant factor by showing that $f(k) = \Theta\left(\frac{2^k}{k}\right)$. More precisely, in Section 2 we will show (in the proof of Lemma 1.6) that for every given (k, d)-tree we can construct an unsatisfiable $(k, 2d)$-CNF formula. Together with Lemma 1.1 this directly implies the following.

Theorem 1.4. *For every integer k,*

$$f(k) < \frac{2^{k+3}}{k}.$$

Hence the lower bound in Theorem 1.3 is best possible up to a factor $8e$.

Recently Moser [14] showed that for $s \leq \frac{2^{k-6}}{k}$ not only every (k, s)-CNF formula has a satisfying assignment but there is also an algorithm computing such an assignment efficiently. Theorem 1.4 proves that this bound is asymptotically tight. Indeed, for some $(k, \frac{2^{k+3}}{k})$-CNF formulas we can not find a satisfying assignment efficiently, simply because there is none.

A special class of unsatisfiable formulas. The class MU1 of minimal unsatisfiable CNF formulas $\mathcal{F}$ where $m(\mathcal{F}) - n(\mathcal{F}) = 1$ with $m(\mathcal{F})$ denoting the number of clauses of $\mathcal{F}$ and $n(\mathcal{F})$ denoting the number of variables of $\mathcal{F}$ has been widely studied (see, e.g., [1], [5], [11], [13], [17]). While it is not known whether $f(k)$ is computable Hoory and Szeider investigated $f_1(k)$, the largest integer such that no $(k, f_1(k))$-CNF formula is in MU1. They showed that $f_1(k)$ is computable. With $f(k) \leq f_1(k)$ this allowed them to derive the best known upper bounds for $f(k)$ for small k: $f(5) \leq 7$, $f(6) \leq 11$, $f(7) \leq 17$, $f(8) \leq 29$, $f(9) \leq 51$.

While the derivation of the previous bound of $f(k) = O\left((\log k) \cdot \frac{2^k}{k}\right)$ by Hoory and Szeider did not go via an MU1 formula the construction for

proving Theorem 1.4 resides in MU 1 (c.f. Lemma 1.6). As a consequence, Theorem 1.4 remains true if we replace $f(k)$ with $f_1(k)$.

Corollary 1.5. *For every integer k we have $f_1(k) < \frac{2^{k+3}}{k}$, implying that $f_1(k) = \Theta(\frac{2^k}{k})$ and $f_1(k) = (1 + o(1))f(k)$.*

It is an open question whether $f(k) = f_1(k)$, i.e., whether some unsatisfiable CNF formulas with the smallest possible number of occurrences per variable (i.e. some unsatisfiable $(k, f(k) + 1)$-CNF formulas) are members of MU 1. Scheder [16] showed that for almost disjoint k-CNF formulas (i.e. CNF formulas where any two clauses have at most one variable in common) this is not true, i.e., no almost disjoint unsatisfiable $(k, \tilde{f}(k) + 1)$-CNF formula is in MU 1, with $\tilde{f}(k)$ denoting the maximum s such that every almost disjoint (k, s)-CNF formula is satisfiable.

Bounded neighborhood size. The *neighborhood* $\Gamma(C)$ of a clause C in a CNF formula $\mathcal{F}$ is the set of clauses in $\mathcal{F}$ that share at least one variable with C, excluding C itself. Analogously to $f(k)$ let $l(k)$ denote the largest integer d such that every k-CNF formula for which $|\Gamma(C)| \leq d$ for every clause C of $\mathcal{F}$ is satisfiable. In fact, the proof of Theorem 1.3 shows that every k-CNF formula $\mathcal{F}$ with $|\Gamma(C)| \leq \frac{2^k}{e} - 1$ for all clauses C of $\mathcal{F}$ is satisfiable. (Note that Theorem 1.3 is a direct consequence of this; for, if in a k-CNF formula every variable occurs at most $\lfloor \frac{2^k}{ek} \rfloor$ times then no clause can collect more than $k \left(\lfloor \frac{2^k}{ek} \rfloor - 1 \right) \leq \frac{2^k}{e} - 1$ neighbors.) Thus,

$$(1) \qquad\qquad l(k) \geq \left\lfloor \frac{2^k}{e} \right\rfloor - 1.$$

This bound is asymptotically tight. The simplest reason is that the complete formula consisting of all 2^k clauses of size k over k variables is clearly unsatisfiable and has neighborhoods of size $2^k - 1$ at each clause. In Section 2 we will show (in the proof of Lemma 1.6) that for every given (k, d)-tree we can construct an unsatisfiable k-CNF formula $\mathcal{F}$ where $|\Gamma(C)| \leq kd$ for every clause C of $\mathcal{F}$. So the construction of appropriate (k, d)-trees (as, for example, those mentioned in Remark 1.2) allows to further tighten the gap between the known lower and upper bounds.

Connection to (k, d)-trees. The following lemma establishes a connection between SAT and the (k, d)-trees described above.

Lemma 1.6. *Let T be a (k, d)-tree, k and d positive integers. Then there is an unsatisfiable k-CNF formula $\mathcal{F} = \mathcal{F}(T)$ with the following properties.*

(a) *Every literal occurs in at most d clauses of $\mathcal{F}$.*
(b) *For every two distinct clauses C, D having a variable in common there is a variable that appears in C and D with opposite signs.*
(c) *If T is minimum with respect to the number of leaves then $\mathcal{F}$ belongs to* $\mathrm{MU}1$.
(d) $|\Gamma(C)| \leq kd$ *for all clauses C in $\mathcal{F}$.*

In particular, $f(k), f_1(k) \leq 2d - 1$ and $l(k) \leq kd - 1$.

Lemma 1.6 will be proved in Section 2. Note that Theorem 1.4 and Corollary 1.5 follow directly from Lemma 1.6, Lemma 1.1 and Theorem 1.3.

Implications on (k,d)-trees. By (1) and Lemma 1.6 we obtain the following.

Observation. *There is no (k,d)-tree for $d \leq \frac{2^k}{ek} - 1$.*

1.3. Maker/Breaker Games

A *hypergraph* is a pair (V, E), where V is a finite set whose elements are called *vertices* and E is a family of subsets of V, called *hyperedges*. A hypergraph is *n-uniform* if every hyperedge contains exactly n vertices. We study the following Maker/Breaker game. Maker and Breaker take turns in claiming one previously unclaimed vertex of a given n-uniform hypergraph $\mathcal{F}$, with Maker going first. Maker wins if he claims all vertices of some hyperedge of $\mathcal{F}$, otherwise Breaker wins. We say that Maker uses a *pairing strategy* if, after claiming his first vertex, he divides all but at most one of the remaining vertices of $\mathcal{F}$ into pairs and whenever Breaker claims one vertex of a pair he takes the other one.

Let $\mathcal{F}$ be an n-uniform hypergraph. The *degree* $d(v)$ of a vertex v is the number of hyperedges containing v and the *maximum degree* $\Delta(\mathcal{F})$ of a hypergraph $\mathcal{F}$ is the maximum degree of its vertices. The *neighborhood* $N(e)$ of a hyperedge e is the set of hyperedges of $\mathcal{F}$ which intersect e, excluding e itself, and the *maximum neighborhood size* of $\mathcal{F}$ is the maximum of $|N(e)|$ where e runs over all hyperedges of $\mathcal{F}$.

The famous Erdős-Selfridge Theorem [6] states that for each n-uniform hypergraph $\mathcal{F}$ with less than 2^{n-1} hyperedges Breaker has a winning strategy. This upper bound on the number of hyperedges is best possible as the following example shows. Let $V = \{v\} \cup \{v_1, \ldots, v_{n-1}\} \cup \{w_1, \ldots, w_{n-1}\}$ and let E be the union of all hyperedges containing v and exactly one vertex of every pair (v_i, w_i). $\mathcal{G} := (V, E)$ is an n-uniform hypergraph with 2^{n-1} hyperedges.

By first claiming v and then pairing v_i with w_i for every $i \in \{1, \ldots, n-1\}$, Maker can occupy all vertices of some hyperedge.

Note that the maximum degree of $\mathcal{G}$ is 2^{n-1}, thus equally large as the number of hyperedges of $\mathcal{G}$. This provides some evidence that in order to be a Maker's win a hypergraph must have largely overlapping hyperedges. Moreover, Beck [3] conjectured that the main criterion for whether a hypergraph is a Breaker's win is not the cardinality of the hyperedge set but rather the maximum neighborhood size, i.e., the actual reason why each hypergraph $\mathcal{H}$ with less than 2^{n-1} edges is a Breaker's win is that the maximum neighborhood size of $\mathcal{H}$ is smaller than $2^{n-1} - 1$.

Neighborhood Conjecture (Open Problem 9.1(a), [3]) Assume that $\mathcal{F}$ is an n-uniform hypergraph, and its maximum neighborhood size is smaller than $2^{n-1} - 1$. Is it true that Breaker has a winning strategy on $\mathcal{F}$?

Further motivation for the Neighborhood Conjecture is the well-known Erdős-Lovász 2-coloring Theorem – a direct consequence of the famous Lovász Local Lemma – which states that every n-uniform hypergraph with maximum neighborhood size at most $\frac{2^{n-1}}{e} - 1$ has a proper 2-coloring. An interesting feature of this theorem is that the size of the hypergraph does not matter. By another application of the Local Lemma we prove (in Theorem 1.8) that, moreover, every n-uniform hypergraph with maximum degree at most $\frac{2^{n-2}}{en}$ has a so called proper *halving* 2-coloring, i.e., a proper 2-coloring where the number of red vertices and the number of blue vertices differ by at most 1. This guarantees the *existence* of a course of the game such that at the end Breaker owns at least one vertex of each hyperedge and thus is the winner. Moreover, the existence of a proper halving 2-coloring is a necessary condition for Breaker having a winning strategy. Indeed, assume for a contradiction that Breaker has a winning strategy on a hypergraph $\mathcal{F}$ which does not admit a proper halving 2-coloring. Suppose further that during the game Maker colors his vertices red and Breaker colors his vertices blue. Note that by the end of the game the vertices are colored in such a way that the number of red vertices and the number of blue vertices differ by at most 1. By assumption Breaker has a winning strategy, thus every hyperedge contains at least one blue vertex. Since $\mathcal{F}$ does not admit a proper halving 2-coloring there must be at least one hyperedge containing only blue vertices. Since Maker starts the game he can now "steal" Breaker's strategy by starting with an arbitrary first vertex and then following Breaker's strategy (if this strategy calls for a vertex he occupied before he takes an arbitrary free vertex: no extra move is disadvantageous for him). This enables him to occupy all vertices of some hyperedge, which contradicts the assumption that Breaker has a winning strategy.

We first prove that the Neighborhood Conjecture, in this strongest of its forms, is not true, even if we require Maker to use a pairing strategy.

Theorem 1.7. *For every $n \geq 3$ there is an n-uniform hypergraph $\mathcal{H}$ with maximum neighborhood size $2^{n-2}+2^{n-3}$ where Maker has a winning pairing strategy.*

In Section 3 we prove Theorem 1.7 by constructing an appropriate binary tree T, which gives rise to the required hypergraph. In his book [3] Beck also poses the following weaker version of the Neighborhood Conjecture.

Open problem 1. *(Open Problem 9.1(b), [3]) If the Neighborhood Conjecture is too difficult (or false) then how about if the upper bound on the maximum neighborhood size is replaced by an upper bound $\frac{2^{n-c}}{n}$ on the maximum degree where c is a sufficiently large constant?*

In the hypergraph $\mathcal{H}$ we will construct to prove Theorem 1.7 one vertex has degree $2^{n-2}+1$, which is still high. However, the existence of vertices with high degree is not crucial. We also construct a hypergraph with maximum degree $\frac{2^{n+2}}{n}$ on which Maker has a winning strategy (c.f. Corollary 1.11).

In his book [3] Beck also poses several further weaker versions of the Neighborhood Conjecture. The last one is as follows.

Open problem 2. *(Open Problem 9.1(f), [3]) How about if we just want a proper halving 2-coloring?*

It is already known [3] that the answer to Open Problem 2 is positive if the maximum degree is at most $\left(\frac{3}{2}-o(1)\right)^{n}$. According to Beck [3] the real question is whether or not $\frac{3}{2}$ can be replaced by 2. We prove that the answer is yes.

Theorem 1.8. *For every n-uniform hypergraph $\mathcal{F}$ with maximum degree at most $\frac{2^{n-2}}{en}$ there is a proper halving 2-coloring.*

We will prove Theorem 1.8 in Section 3.

Connection to trees. Let T be a binary tree where every leaf has depth at least $n-1$. By $\mathcal{H}_T = \mathcal{H}_T(n)$ we denote the n-uniform hypergraph whose hyperedges are the vertex sets of paths that start at a leaf and go up $n-1$ levels. The next lemma will be proved in Section 3.

Lemma 1.9. *Let T be a binary tree where every leaf has depth at least $n-1$. Then Maker has a winning pairing strategy on $\mathcal{H}_T$.*

By the construction of $\mathcal{H}_T$ and Lemma 1.9 we can immediately connect (k,d)-trees to the game we study.

Observation 1.10. *Let T be an (n,d)-tree. Then (i) Maker has a winning pairing strategy on $\mathcal{H}_T$ and (ii) every vertex of $\mathcal{H}_T$ occurs in at most d hyperedges.*

The next corollary is then a direct consequence of Observation 1.10 and Lemma 1.1.

Corollary 1.11. *For every n there is an n-uniform hypergraph with maximum degree at most $\frac{2^{n+2}}{n}$ where Maker has a winning pairing strategy.*

Notation. Ceiling and floor signs are routinely omitted whenever they are not crucial for clarity. Throughout this paper log stands for the binary logarithm. Let T be a binary tree. For any integer l we let *level l* denote the set of nodes at depth l. Moreover, a *path* of T is a sequence of vertices $v_1, v_2, \ldots, v_j$ of T where v_k is a child of v_{k-1} for every $k = 2, \ldots, j$. Depending on the context we consider a hyperedge e of a hypergraph $\mathcal{H}_T$ either as a set or as a path in T. So we will sometimes speak of the start or end node of a hyperedge.

Organization of this paper. In Section 2 we establish a connection between the trees we study and SAT by proving Lemma 1.6. In Section 3 we show Lemma 1.9, which connects trees to the Maker/Breaker game we consider, and prove Theorem 1.7 refuting the Neighborhood Conjecture in the strongest of its forms. Moreover, we also show Theorem 1.8.

In Section 4 we finally prove Lemma 1.1 by constructing suitable (k,d)-trees.

2. Constructing Unsatisfiable k-CNF Formulas with Small Neighborhood

In order to prove Lemma 1.6 we will consider the following construction of an unsatisfiable k-CNF formula. Let T be a given (k,d)-tree and let $\widehat{T}$ denote the binary tree obtained by attaching the roots of disjoint copies of T as the two children of a new root. Note that $\widehat{T}$ is a (k,d)-tree where every leaf has depth at least k. For every non-leaf node $w \in V(\widehat{T})$ we create a variable x_w and label one of its children with the literal x_w and the other with the negated version $\bar{x}_w$. Note that the root does not receive a literal. With every leaf v of $\widehat{T}$ we associate a clause C_v by walking along a path of length $k-1$ from v towards the root and taking the disjunction of all labels encountered

on this path (i.e., the labels of all nodes to which v is $(k-1)$-close). Finally, we let $\mathcal{F}(T)$ denote the conjunction of all such clauses C_v.

Clearly, every assignment α over the variables of $\mathcal{F}(T)$ defines a path in T from the root to a leaf, say v, by always proceeding to the unique child whose label is mapped to 0 by α; thus the clause C_v associated with v is violated by α. Hence we have the following.

Observation 2.1. *For every (k,d)-tree T, the corresponding formula $\mathcal{F}(T)$ is unsatisfiable.*

Moreover, in the proof of Lemma 1.6 we will use a powerful characterization of MU 1-formulas given by Davydov, Davydova, and Kleine Büning [5]. (Here a clause $C=(x_1\vee x_2\vee\ldots\vee x_k)$ is represented as the set $\{x_1,\ldots,x_k\}$ of its literals, and a CNF formula $\mathcal{F}=C_1\wedge C_2\wedge\ldots\wedge C_n$ is represented as the set $\{C_1,\ldots,C_n\}$ of its clauses. "vbl$(\mathcal{F})$" denotes the set of variables which occur in $\mathcal{F}$.)

Lemma 2.2 (Davydov, Davydova, and Kleine Büning [5]). $\mathcal{F}\in$MU 1 *if and only if either $\mathcal{F}=\{\emptyset\}$ or $\mathcal{F}$ is the disjoint union of formulas $\mathcal{F}_1',\mathcal{F}_2'$ such that for a variable x we have*

- vbl$(\mathcal{F}_1')\cap$vbl$(\mathcal{F}_2')=\{x\}$ *and* $\{x,\bar{x}\}\subseteq\bigcup_{C\in\mathcal{F}}C$;
- $\mathcal{F}_1:=\{C\backslash\{x\}:C\in\mathcal{F}_1'\}\in$MU 1;
- $\mathcal{F}_2:=\{C\backslash\{\bar{x}\}:C\in\mathcal{F}_2'\}\in$MU 1.

Two nodes of a binary tree are called *siblings* if they share the same parent. Let T be a binary tree and suppose that we label the nodes of T other than the root by literals such that every literal appears exactly once and siblings get complementary literals. We can now construct a CNF formula $\mathcal{G}$ by associating with every leaf v of T a clause C_v that is the disjunction of *some* literals along the path from v to the root, and taking the conjunction of all the C_v. Every CNF formula $\mathcal{G}$ which can be obtained in this way is called a *T-formula.*

Note that for every (k,d)-tree T, the formula $\mathcal{F}(T)$ is a $\widehat{T}$-formula. The following characterization of MU 1-formulas is an immediate (and known) consequence of Lemma 2.2.

Corollary 2.3. *A CNF formula $\mathcal{G}$ is in MU 1 if and only if $\mathcal{G}$ is a T-formula for some binary tree T where every literal associated to a vertex of T does appear in $\mathcal{G}$.*

We are now ready to prove Lemma 1.6.

Proof of Lemma 1.6. Let T be a (k,d)-tree and let $\mathcal{F} = \mathcal{F}(T)$. Recall that $\widehat{T}$ is the binary tree obtained by attaching the roots of disjoint copies of T as the two children of a new root. By Observation 2.1 $\mathcal{F}$ is unsatisfiable. Moroever, the defining property of (k,d)-trees guarantees that no literal appears in more than d clauses (hence (a)). We now settle (b). Let u and v be two leaves of $\widehat{T}$ such that the clauses C_u and C_v associated with u and v share at least one variable. Moreover, let w denote the lowest common ancestor of u and v (i.e. the node of maximum depth in $\widehat{T}$ that appears on both paths from u and v, respectively, to the root). Then one child of w occurs in C_u whereas the other child occurs in C_v. Since siblings have complementary literals (b) is shown.

Next we prove (c). By construction $\mathcal{F}$ is a $\widehat{T}$-formula. Due to the minimality of T, every non-root node w of T has one leaf descendant at distance at most $k-2$, since otherwise the subtree of T rooted at w would be a (k,d)-tree with fewer leaves than T. Hence every node of T (including the root) has one leaf descendant at distance at most $k-1$, and thus every literal associated to a vertex of $\widehat{T}$ does appear in $\mathcal{F}$. So, by Corollary 2.3, $\mathcal{F}$ belongs to $\mathrm{MU}1$.

(d) follows from (a) and (b): Indeed, if we define $\mathrm{occ}(u)$ as the number of clauses of $\mathcal{F}$ containing a literal u, and if we abbreviate „the clause C contains the literal u" by „$u \in C$", then (b) allows us to write $|\Gamma(C)|$ as $\sum_{u \in C} \mathrm{occ}(\bar{u})$, which is at most kd for every clause C of $\mathcal{F}$. ∎

3. Counterexample to the Neighborhood Conjecture

Recall that, for a given binary tree T where every leaf has depth at least $n-1$, we let $\mathcal{H}_T = \mathcal{H}_T(n)$ denote the n-uniform hypergraph whose hyperedges are the vertex sets of paths that start at a leaf and go up $n-1$ levels.

Proof of Lemma 1.9. The set of non-root nodes of T can be divided into pairs of siblings. By first claiming the root of T and then pairing every node with its sibling Maker can finally achieve a path from the root to a leaf, which by assumption contains a hyperedge. ∎

Proof of Theorem 1.7. Due to Lemma 1.9 it suffices to show the next lemma.

Lemma 3.1. *For every $n \geq 3$ there is a binary tree T where every leaf has depth at least n–1 such that $\mathcal{H}_T$ has maximum neighborhood size $2^{n-2} + 2^{n-3}$.*

Proof. Let T' be a full binary tree with $n-1$ levels. For each leaf u of T' we proceed as follows: We add two children v, w to u and let v be a leaf. Then

we attach a full binary tree S with $n-2$ levels to w (such that w is the root of S). For each leaf u' of S we add two children v', w' to u' and let v' be a leaf. Note that the hyperedge ending at v' starts at u. Finally, we attach a full binary tree S' with $n-1$ levels to w' (such that w' is the root of S'), see Figure 1. Let T denote the resulting tree.
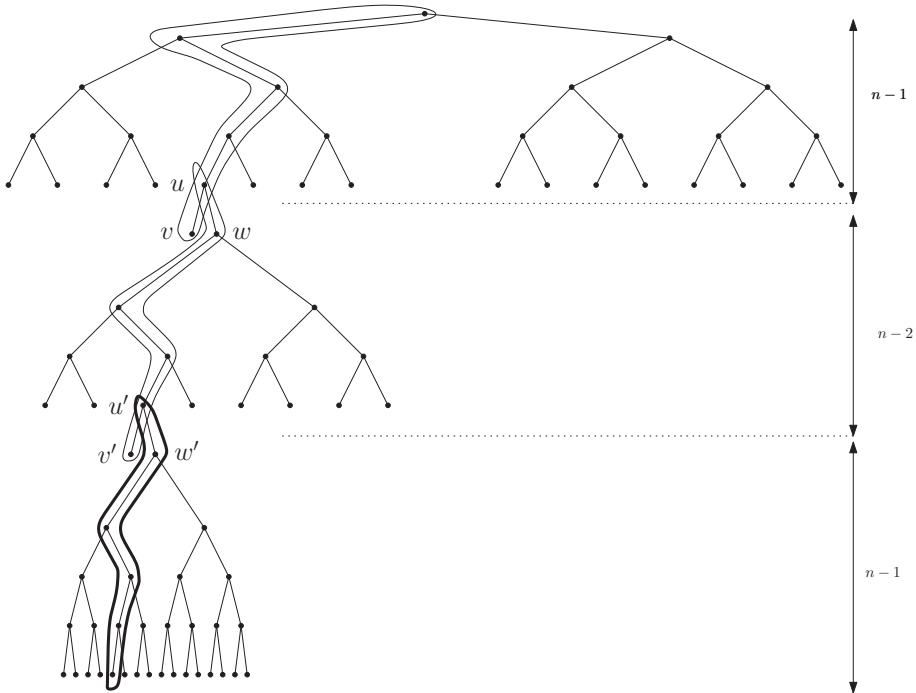

Figure 1. An illustration of $\mathcal{H}_T$. The marked paths represent exemplary hyperedges.

Clearly, every leaf of T has depth at least $n-1$. It remains to show that the maximum neighborhood of $\mathcal{H}_T$ is at most $2^{n-2}+2^{n-3}$.

Claim. *Every hyperedge e of $\mathcal{H}_T$ intersects at most $2^{n-2}+2^{n-3}$ other hyperedges.*

In order to prove this claim, we fix six vertices u, u', v, v', w, w' according to the above description, i.e., u is a node on level $n-2$ whose children are v and w, u' is a descendant of w on level $2n-4$ whose children are v' and w'. Let e be a hyperedge of $\mathcal{H}_T$. Note that the start node of e is either the root of T, a node on the same level as u or a node on the same level as u'. We now distinguish these cases.

(a) The start node of e is the root. By symmetry we assume that e ends at v. According to the construction of T the hyperedge e intersects the $2^{n-2}-1$ other hyperedges starting at the root and the 2^{n-3} hyperedges starting at u. So altogether e intersects $2^{n-2}+2^{n-3}-1$ other hyperedges.

(b) The start node of e is on the same level as u. By symmetry we suppose that e starts at u and ends at v'.

The hyperedges intersecting e can be divided into the following three categories.

- The hyperedge starting at the root and ending at v,
- the $2^{n-3}-1$ hyperedges different from e starting at u, and
- the 2^{n-2} hyperedges starting at u',

implying that e intersects $2^{n-2}+2^{n-3}$ other hyperedges in total.

(c) The start node of e is on the same level as u'. By symmetry we assume that e starts at u'. Then e intersects the $2^{n-2}-1$ other hyperedges starting at u' and the hyperedge starting at u and ending at v', thus 2^{n-2} other hyperedges altogether. ∎

Establishing a Proper Halving 2-Coloring

Proof of Theorem 1.8. Let $\mathcal{F}=(V,E)$. We can assume without loss of generality that $|V|$ is even. (Otherwise we let V' be the vertex set obtained by adding a dummy vertex x to V, and we consider the hypergraph $\mathcal{F}'=(V',E)$. Since every proper halving 2-coloring of $\mathcal{F}'$ yields a proper halving 2-coloring of $\mathcal{F}$, it suffices to show that $\mathcal{F}'$ has as proper halving 2-coloring.) We will show the following, stronger claim.

Proposition 3.2. *Let $\mathcal{F} = (V,E)$ be an n-uniform hypergraph with $2r$ vertices and maximum degree at most $\frac{2^{n-2}}{en}$. Then for every partition $(v_1,v_1'),\ldots,(v_r,v_r')$ of V into pairs, there is a proper 2-coloring such that v_i and v_i' have different colors for every i, $i=1,\ldots,r$.*

Before starting with the proof we need some notation. First, let $P = (v_1,v_1'),\ldots,(v_r,v_r')$ be a partition of V into pairs. By a *(proper) P-2-coloring* we denote a (proper) 2-coloring of $\mathcal{F}$ such that v_i and v_i' have different colors for every i, $i=1,\ldots,r$. Moreover, for every vertex $x\in V$ we denote by $f(x)$ the vertex which is paired with x in P (i.e., $f(v_i)=v_i'$ and $f(v_i')=v_i$).

Proof of Proposition 3.2. We fix a partition $P = (v_1, v_1'), \ldots, (v_r, v_r')$ of V into pairs. Our goal is to show that there is a proper P-2-coloring. For each hyperedge $e = (w_1, \ldots, w_n)$ we add e and $e' = (f(w_1), \ldots, f(w_n))$ and denote the resulting hypergraph by $\mathcal{F}'$. Note that $\Delta(\mathcal{F}') \leq 2 \cdot \Delta(\mathcal{F}) \leq \frac{2^{n-1}}{en}$. We now transform $\mathcal{F}'$ into a SAT instance. For every hyperedge $e = (u_1, \ldots, u_n)$ of $\mathcal{F}'$ we form a clause $C_e = (u_1 \vee u_2 \vee \ldots \vee u_n)$ and set $\mathcal{H} := \wedge_{e \in E(\mathcal{F}')} C_e$ with $E(\mathcal{F}')$ denoting the hyperedge set of $\mathcal{F}'$. Then we replace (in $\mathcal{H}$) v_i and v_i' with x_i and $\bar{x}_i$, respectively, for every i, $i = 1, \ldots, r$. Note that every variable x_i occurs in at most $2\Delta(\mathcal{F}') \leq \frac{2^n}{en}$ clauses of $\mathcal{H}$. Due to Theorem 1.3 there is a satisfying assignment α of $\mathcal{H}$. Note that by construction, there is no clause where *every* literal is set to 1 by α (indeed, every clause where all literals are set to 1 corresponds to a clause where all literals are set to 0). So every clause C of $\mathcal{H}$ contains two literals x, y where $\alpha(x) = 1$ and $\alpha(y) = 0$. For every variable x_i where $i \in \{1, \ldots, r\}$ we proceed as follows. If $\alpha(x_i) = 1$ then we color v_i red and v_i' blue, otherwise we do it the other way round. Clearly, this yields a proper P-2-coloring of $\mathcal{F}'$. Since every proper P-2-coloring of $\mathcal{F}'$ is also a proper P-2-coloring of $\mathcal{F}$ this concludes the proof. ∎

4. Constructing Suitable (k, d)-Trees

We need some notation first. Let T be a binary tree (not necessarily with all leaves having depth at least $k - 1$) and let v be a vertex of T. In the following we denote by the *degree* $d(v)$ of v the number of leaf descendants which have distance at most $k - 1$ from v.

Proof of Lemma 1.1. Note that the full binary tree of height $k - 1$ is a $(k, 2^{k-1})$-tree. We have $\lfloor \frac{2^{k+2}}{k} \rfloor \geq 2^{k-1}$ for $k \leq 8$ so we can assume that $k > 8$. Let $r = 2^{k+1-\lfloor \log k \rfloor}$ and note that $r \leq \lfloor \frac{2^{k+2}}{k} \rfloor$. Let T' be a full binary tree of height $k - 1$. For every vertex v_j on level $k - \lfloor \log k \rfloor$ we denote by the *interval* I_j the set of all leaf descendants of v_j. So we can subdivide the leaves of T' into $s = 2^{k-\lfloor \log k \rfloor}$ intervals $I_1, \ldots, I_s$ of cardinality $2^{\lfloor \log k \rfloor - 1}$ each. For every such interval $I_j = \{v_0, \ldots, v_{2^{\lfloor \log k \rfloor - 1} - 1}\}$ and every $v_i \in I_j$ we attach a full binary subtree of height i to v_i. Let T denote the resulting tree. Figure 2 shows an illustration for the case where k is a power of 2. It suffices to prove the following.

Proposition 4.1. *Let v be a vertex of T. Then $d(v) \leq r$.*

Proof. We apply induction on the depth i of v. For $i = 0$ the claim is clearly true. Indeed, the degree of the root is $\frac{2^{k-1}}{2^{\lfloor \log k \rfloor - 1}} = 2^{k - \lfloor \log k \rfloor} = \frac{r}{2}$. Now suppose
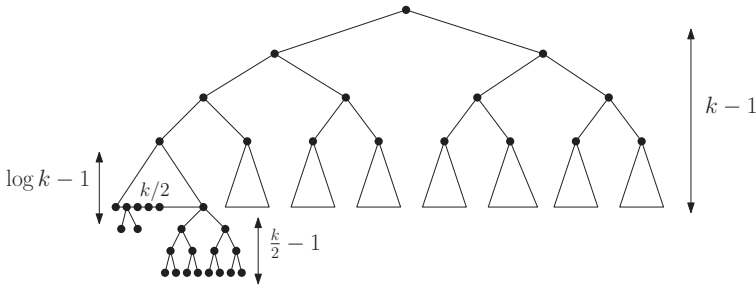
Figure 2. The construction of T where k is a power of 2.

that v has depth $i \in \{1, \ldots, 2^{\lfloor \log k \rfloor - 1} - 1\}$. Note that the set of descendants of v on level $k-1$ can be subdivided into $\frac{2^{k-1-i}}{2^{\lfloor \log k \rfloor - 1}}$ intervals, i.e., at least one interval for the values of i we consider. Let v' denote the parent of v. By construction the number of leaf descendants which have distance at most $k-2$ from v equals $\frac{d(v')}{2}$. Moreover, every interval $\{v_0, \ldots, v_{2^{\lfloor \log k \rfloor - 1} - 1}\}$ gives rise to 2^i leaves on level $k-1+i$, implying that the number of leaf descendants of v which have distance exactly $k-1$ from v equals $\frac{2^{k-1-i}}{2^{\lfloor \log k \rfloor - 1}} \cdot 2^i = 2^{k - \lfloor \log k \rfloor} = \frac{r}{2}$. So altogether, $d(v) \leq \frac{d(v')}{2} + \frac{r}{2} \leq r$. It remains to consider the case where v has depth at least $2^{\lfloor \log k \rfloor - 1}$. By construction no leaf of T has depth larger than $k + 2^{\lfloor \log k \rfloor - 1} - 2$, implying that the degree of v is at most the degree of its parent. ∎

References

[1] R. AHARONI and N. LINIAL: Minimal non-two-colorable hypergraphs and minimal unsatisfiable formulas, *J. Combin. Theory Ser. A* **43**, (1986), 196–204.

[2] N. ALON and J. H. SPENCER: *The Probabilistic Method,* J. John Wiley & Sons (2002).

[3] J. BECK: Combinatorial Games: Tic Tac Toe Theory, *Encyclopedia of Mathematics and its Applications* **114**, 2008.

[4] J. BECK: Remarks on positional games, *Acta Math. Acad. Sci. Hungar.* **40**, (1982), 65–71.

[5] G. DAVYDOV, I. DAVYDOVA and H. KLEINE BÜNING: An efficient algorithm for the minimal unsatisfiability problem for a subclass of CNF, *Artif. Intell.* **23**, (1998), 229–245.

[6] P. ERDŐS and J. L. SELFRIDGE: On a combinatorial game, *J. Combinatorial Theory Ser. A* **14**, (1973), 298–301.

[7] P. ERDŐS and J. SPENCER: Lopsided Lovász local lemma and Latin transversals, *Discrete Appl. Math.* **30**, (1991), 151–154.

[8] H. GEBAUER, T. SZABÓ and G. TARDOS: The Local Lemma is Tight for SAT, *Proc. 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, (2011), 664–674.

[9] S. HOORY and S. SZEIDER: A note on unsatisfiable k-CNF formulas with few occurrences per variable, *SIAM J. Discrete Math* **20** (2), (2006), 523–528.

[10] S. HOORY and S. SZEIDER: Computing Unsatisfiable k-SAT Instances with Few Occurrences per Variable, *Theoretical Computer Science* **337** (1–3), (2005), 347–359.

[11] H. KLEINE BÜNING and X. ZHAO: On the structure of some classes of minimal unsatisfiable formulas, *Discr. Appl. Math.* **130**(2), (2003), 185–207.

[12] J. KRATOCHVÍL, P. SAVICKÝ and Z. TUZA: One more occurrence of variables makes satisfiability jump from trivial to NP-complete, *SIAM J. Comput.* **22** (1), (1993), 203–210.

[13] O. KULLMANN: An application of matroid theory to the SAT problem, *Fifteenth Annual IEEE Conference on Computational Complexity*, (2000), 116-124.

[14] R. MOSER: A constructive proof of the Lovász Local Lemma, *Proc. 41st Annual ACM Symposium on Theory of Computing (STOC)*, (2009), 343–350.

[15] P. SAVICKÝ and J. SGALL: DNF tautologies with a limited number of occurrences of every variable, *Theoret. Comput. Sci.* **238** (1–2), (2000), 495–498.

[16] D. SCHEDER: Unsatisfiable Linear CNF Formulas Are Large and Complex, *27th International Symposium on Theoretical Aspects of Computer Science (STACS)*, (2010), 621–631.

[17] S. SZEIDER: Homomorphisms of conjunctive normal forms, *Discr. Appl. Math.* **130**(2), (2003), 351–365.

[18] C. A. TOVEY: A simplified NP-complete satisfiability problem, *Discr. Appl. Math.* **8** (1), (1984), 85–89.

Heidi Gebauer

Institute of Theoretical Computer Science
ETH Zurich, CH-8092
Switzerland
gebauerh@inf.ethz.ch