Technical Reports on Mathematical and Computing Sciences: TR-C184
title: On Unique Pigeon Hole Principles
author: Marcel Crasmaru and Osamu Watanabe
affiliation:
Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology
Meguro-ku Ookayama, Tokyo 152-8552, Japan
(`watanabe@is.titech.ac.jp`)

Abstract: Based on a variation of the pigeon hole principle, we propose a propositional formula having a unique satisfying assignment and prove an exponential lower bound for computing each bit of its satisfying by the tree-like resolution. Our proof is obtained by following the outline proposed by Ben-Sasson and Wigderson [BW99].

1 Introduction

The resolution has been studied as a fundamental proof system in computational/proof complexity theory. Though the resolution is useful, it is so simple that one may not expect that it *always* provides us with an efficient proof. In fact, the resolution cannot give polynomial-size proofs to very simple assertions such as the pigeonhole principle. Including this fact, the limitation of the resolution has been revealed by a sequence of very impressive investigations; see, e.g., [Raz]. Here we discuss another way of using the resolution — a tool for finding a satisfying assignment. We show again some limitation of the resolution for this application.

The resolution is usually used to prove a propositional assertion given as a DNF formula. An actual input formula F is its negation (hence, a CNF formula), and the resolution is used to derive a contradiction, i.e., the empty clause, from F, which is the proof for the validity of the original assertion $\neg F$. But the resolution may also give us some important information when searching for a satisfying assignment for a given CNF formula F. In general, any clause obtained by a sequence of resolution rules is a necessary condition for assignments satisfying F. Thus, if some of such necessary conditions can be succinctly represented as a clause (which is not always the case) and such clauses can be derived easily by the resolution, then we can use them to reduce the assignment space, thereby simplifying the satisfying assignment search. In fact, this sort of reduction occurs whenever the backtrack step occurs in the DPLL procedure [DDL62], a well-known algorithm for the SAT problem. In this paper, for formulas with some succinct necessary conditions for their satisfying assignments, we investigate whether such conditions can be found easily by the resolution.

As an extreme case, we consider *uniquely satisfiable* formulas, formulas with exactly one satisfying assignment. This unique solution property sometimes can be used to search solutions efficiently, e.g., [PPSZ98]. In our context, the property indeed guarantees very useful necessary conditions for satisfying assignments. For example, consider any uniquely satisfiable formula F over variables $x_1, \ldots, x_n$. Assume that $(a_1, \ldots, a_n) \in \{0,1\}^n$ is its unique satisfying assignment. Then clearly, for every i, $1 \leq i \leq n$, a unit clause $x_i^{a_i}$ is a necessary condition for the satisfying assignment, where $x_i^{a_i}$ is x_i if $a_i = 1$ and $\overline{x_i}$ otherwise. In fact, the set of these necessary conditions determine the satisfying assignment. Thus, if these necessary conditions were derived efficiently, then we could efficiently find a satisfying assignment for uniquely satisfiable formulas.

Here we discuss the possibility of using the resolution for deriving such necessary conditions. We introduce a natural variation of the pigeonhole principle — unique pigeonhole principles — and prove an exponential lower bound for the tree-like resolution. Interestingly, another variation (i.e., a functional version) has a $O(n^2)$-size proof.

Note that we can modify the standard pigeonhole principle PHP_n^m ($m > n$) to a uniquely satisfiable formula P by adding one new variable z; intuitively, P is defined as $P = (\mathrm{PHP}_n^m \vee z) \wedge \bigwedge_{i,j}(x_{ij} \vee \overline{z})$. Clearly, this P has a unique satisfying assignment; furthermore, one can easily obtain the same lower bounds for deriving, e.g., a unit clause z from the existing lower bounds for PHP_n^m. The contribution of this paper is to prove a similar exponential lower bound to a more natural example of unique satisfiable formulas, which could be useful as a model of some computation. Furthermore, we can show the importance of deriving the functional conditions in the pigeon hole principle on our example. Our proof is obtained by simply following the outline proposed by Ben-Sasson and Wigderson [BW99]. Though some detailed combinatorial analysis is necessary, our proof demonstrates the usefulness of their proof outline.

2 Preliminaries

We use notions and notations standard in the proof complexity theory. Here we review those important in the following discussion from [BW99].

A *literal* over a *Boolean variable* x is either x (denoted also as x^1) or $\overline{x}$ (denoted also as x^0). A *clause* is a disjunction of literals. We say that a literal x^a *appears* in a clause C if it is in C; we also say that a *variable* x *appears* in C if either x or $\overline{x}$ is an element of C. We say that a clause A is *contained* in a clause B (denoted $A \subset B$) if the set of literals of A is a subset of the set of literals of B. A CNF formula is a set of clauses, semantically, the conjunction of these clauses.

For a clause C, x a variable and $a \in \{0,1\}$, the *restriction* of of C setting x to a is:

$$C|_{x=a} \stackrel{\mathrm{def}}{=} \begin{cases} 1, & \text{if } x^a \text{ appears in } C; \\ C \setminus \{x^{1-a}\}, & \text{if } x^{1-a} \text{ appears in } C; \text{ and} \\ C, & \text{otherwise.} \end{cases}$$

For a clause C, $\boldsymbol{x} = (x_1, \ldots, x_k)$ a k tuple of variables, and $\boldsymbol{a} = (a_1, \ldots, a_k) \in \{0,1\}^k$, a restriction of of C setting $\boldsymbol{x}$ to $\boldsymbol{a}$ is:

$$C|_{\boldsymbol{x}=\boldsymbol{a}} \stackrel{\mathrm{def}}{=} C|_{x_1=a_1,\ldots,x_k=a_k} \stackrel{\mathrm{def}}{=} (\cdots((C|_{x_1=a_1})|_{x_2=a_2})\cdots)|_{x_k=a_k}.$$

Let $\mathcal{F}$ be a CNF formula over a tuple of variables $\boldsymbol{x} = (x_1, \ldots, x_n)$ (denoted also $\mathcal{F}(x_1, \ldots, x_n)$). We say that $\mathcal{F}$ is *satisfiable* if there exists $\boldsymbol{a} \in \{0,1\}^n$ such that $\mathcal{F}|_{\boldsymbol{x}=\boldsymbol{a}} = 1$.

For clauses $A = (x \vee \alpha)$ and $B = (\overline{x} \vee \beta)$ where α and β are any disjunctions of literals, let $\mathrm{res}(A, B) = (\alpha \vee \beta)$ be the *resolvent* of A and B. The resolution rule is a derivation of $\mathrm{res}(A, B)$ from A and B, which is written as follows:

$$\frac{(x \vee \alpha)(\overline{x} \vee \beta)}{(\alpha \vee \beta)}.$$

What follows is the key property for the resolution rule.

Lemma 2.1 *(Resolution Lemma) Any CNF formula $\mathcal{F} = \{A, B\} \cup E$ is satisfiable if and only if formula $\mathcal{G} = \{res(A, B), A, B\} \cup E$ is satisfiable.*

A *resolution derivation* π of a clause A from $\mathcal{F}$ is a sequence of clauses $\pi = \{D_1, D_2, \dots, D_s\}$ such that the last clause is A and each D_i is either some clause of $\mathcal{F}$ or is derived from previous clauses using one of the following derivation rules:

- The resolution rule: $\frac{A \vee x \quad B \vee \overline{x}}{A \vee B}$, or
- The weakening rule: $\frac{A}{A \vee B}$,

where x be any variable of $\mathcal{F}$ and A and B are any clauses, i.e., disjunctions of any literals.

A graph $G_\pi(A)$ of a derivation π of a clause A is a DAG with the clauses of the derivation as nodes, and for derivation step edges are added from the assumption clauses to the consequence clause. A derivation π is called *tree-like* if G_π is a tree, allowing copies of the original clauses of $\mathcal{F}$ in order to make a proof tree-like. The size of the derivation π is the number of clauses in it, denoted size(π). For any formula $\mathcal{F}$ and clause A, $\mathrm{S_T}(\mathcal{F}, A)$ is the *proof size* for A (by the tree-like resolution), i.e., the *minimal size* of a tree-like derivation of the clause A from the formula $\mathcal{F}$. In particular, $\mathrm{S_T}(\mathcal{F}, \emptyset)$, the proof size for the refutation, is denoted by $\mathrm{S_T}(\mathcal{F})$. If there is a derivation from $\mathcal{F}$ to a clause A, we write this as $\mathcal{F} \vdash A$.

2.1 Unique Pigeon Hole Principles

The pigeon hole principle is an assertion claiming that no one-to-one mapping exists from m pigeons to n holes for any $m > n$. More specifically, for any m and n, let PHP_n^m be the formula over a variable set $\{x_{ij} : 1 \le i \le m, 1 \le j \le n\}$ consisting of the following clauses:

$$\text{(The } i\text{th Pigeon Rule, } 1 \le i \le m) \quad P_i \stackrel{\text{def}}{=} \bigvee_{1 \le q \le n} x_{iq}.$$

$$\text{(The } j\text{th Hole Rule, } 1 \le j \le n) \quad H_j \stackrel{\text{def}}{=} \bigwedge_{1 \le p < p' \le m} \overline{x_{pj}} \vee \overline{x_{p'j}}.$$

Intuitively, P_i asserts that pigeon i stays in some hole, and H_j asserts that hole j is not occupied by more than one pigeon. If we further add the following clauses, then we have a *functional version* FPHP_n^m:

$$\text{(Functional Rule on the } i\text{th Pigeon, } 1 \le i \le m) \quad F_i \stackrel{\text{def}}{=} \bigwedge_{1 \le q < q' \le n} \overline{x_{iq}} \vee \overline{x_{iq'}}.$$

Clearly, PHP_n^m is not satisfiable if $m > n$, but here we are interested in the case $m \le n$ where PHP_n^m is satisfiable. In particular, we consider the case $m = n > 1$, and discuss the problem of finding a satisfying assignment for PHP_n^n — the *pigeon hole problem.* Unfortunately, this problem is not a unique solution problem.

For a pigeon hole problem with the unique solution property, we consider one variation of the above pigeon hole principles — two dimensional pigeon hole principles, or *unique pigeon hole principles*. Roughly speaking, for each $n > 1$, we consider n^2 pigeons and n^2 holes, each indexed by a pair of numbers in $[n]$, and we define a set of constraints so that there is a unique map from n^2 pigeons to n^2 holes. Specifically, the *unique pigeon hole principle* UPHP_n is defined as a formula over $n^2(2n-1)$ variables x_{ijkj} and x_{ijil}, $1 \leq i,j,k,l \leq n$, where x_{ijkj} (resp., x_{ijil}) means[1] that pigeon ij is mapped to hole kj (resp., il). It consists of the following clauses:

(The ijth Pigeon Rule (Row), $1 \leq i,j \leq n$) $\quad P^{\mathrm{R}}_{ij} \stackrel{\mathrm{def}}{=} \bigvee_{1 \leq l \leq n} x_{ijil}.$

(The ijth Pigeon Rule (Column), $1 \leq i,j \leq n$) $\quad P^{\mathrm{C}}_{ij} \stackrel{\mathrm{def}}{=} \bigvee_{1 \leq k \leq n} x_{ijkj}.$

(The klth Hole Rule, $1 \leq k,l \leq n$) $\quad H_{kl} \stackrel{\mathrm{def}}{=} \bigwedge_{\substack{11 \preceq pq \prec p'q' \preceq nn \\ (p=k \vee q=l) \wedge (p'=k \vee q'=l)}} \overline{x_{pqkl}} \vee \overline{x_{p'q'kl}}$

(By $\preceq$ and $\prec$, we mean the lexicographic order on pairs of integers.)

A hole with index kv for some v (resp., ul for some u) is called a hole in row k (resp., column l); pigeons are called in a similar way. Intuitively, P^{R}_{ij} (resp., P^{C}_{ij}) asserts that the ijth pigeon stays in one of the holes in row i (resp., column j). On the other hand, H_{kl} asserts that the klth hole is not occupied by more than one pigeon.

It is easy to see that UPHP_n is satisfiable and it has a unique satisfying assignment, namely, the one mapping each pigeon ij to hole ij. Thus, unit clauses of the form x_{ijij}, $1 \leq i,j \leq n$, are necessary conditions for the satisfying assignment, which altogether indeed determine the satisfying assignment. (The assignment to the other variables x_{ijkl}, where $i \neq k$ or $l \neq j$, is derived easily from the Hole Rules.) The goal of this paper is to estimate proof size for deriving x_{ijij}, namely, $\mathrm{S}_{\mathrm{T}}(\mathrm{UPHP}_n, x_{ijij})$.

Here again we can consider the following functional conditions:

(Functional Rule for the ijth Pigeon) $\quad F_{ij} \stackrel{\mathrm{def}}{=} \bigwedge_{\substack{11 \preceq kl \prec k'l' \preceq nn \\ (k=i \vee l=j) \wedge (k'=i \vee l'=j)}} \overline{x_{ijkl}} \vee \overline{x_{ijk'l'}}$

Let FUPHP_n be a formula adding these clauses to UPHP_n. (In fact, we can replace the Hole Rules by these Functional Rules; a satisfying assignment is uniquely fixed by the Pigeon Rules and the Functional Rules.)

It turns out that a derivation of x_{ijij} becomes easy with the Functional Rules.

Proposition 2.1 *There is a tree-like resolution proof for* $\mathrm{FUPHP}_n \vdash x_{ijij}$ *of size* $O(n^2)$.

Proof. For simplicity, denote $P^{\mathrm{R}}_{ij} = x \vee a_1 \vee \cdots \vee a_{n-1}$ and $P^{\mathrm{C}}_{ij} = x \vee b_1 \vee \cdots \vee b_{n-1}$, where $x = x_{ijij}$, and $a_u = x_{ijiv}$ and $b_v = x_{ijuj}$ for some p and q. Consider the following proof.

[1] For simplifying notations, we write a two dimensional index (i,j) as ij.

$$\frac{\dfrac{\dfrac{\dfrac{x \vee a_1 \vee \cdots \vee a_{n-1} \qquad \overline{a_{n-1}} \vee \overline{b_{n-1}}}{x \vee a_1 \vee \cdots \vee a_{n-2} \vee \overline{b_{n-1}}} \qquad \overline{a_{n-2}} \vee \overline{b_{n-1}}}{x \vee a_1 \vee \cdots \vee a_{n-3} \vee \overline{b_{n-1}}}}{\vdots}}{}$$

$$\frac{\dfrac{x \vee a_1 \vee \overline{b_{n-1}} \qquad \overline{a_1} \vee \overline{b_{n-1}}}{x \vee \overline{b_{n-1}}} \qquad \overline{x} \vee \overline{b_{n-1}}}{\overline{b_{n-1}}}$$

Similarly, the other unit clauses $\overline{b_1}, \ldots, \overline{b_{n-2}}$ are derived. Then resolving clause $x \vee b_1 \vee \cdots \vee b_{n-1}$ with these obtained unit clauses $\overline{b_{n-1}}, \ldots, \overline{b_1}$ leads to the desired unit clause $x = x_{ijij}$. This tree-like resolution proof is of size $O(n^2)$. □

2.2 The Relation between Width and Size

We will prove our lower bound result by following the proof outline proposed in [BW99], where a lower bound for proof size is obtained from a lower bound for the width of the proof. The key is the relation between proof width and proof size; here we recall this relation and confirm it holds for our setting.

The *width* of a clause C, denoted by $|C|$, is the number of literals in it. For a derivation π of a clause A from formula $\mathcal{F}$, the width of π, denoted by $w(\pi)$, is the maximal width of a clause in π. We often write $\mathcal{F} \vdash_u A$ when there is a derivation π of a clause A from formula $\mathcal{F}$ such that $w(\pi) = u$. $w(\mathcal{F} \vdash A)$ denotes $\min_{\pi = \mathcal{F} \vdash A} w(\pi)$.

In [BW99], the following facts are proved.

Lemma 2.2 *For $a \in \{0,1\}$, if $\mathcal{F}|_{x=a} \vdash_w A$, then $\mathcal{F} \vdash_{w+1} A \vee x^{1-a}$.*

Theorem 2.1 *$w(\mathcal{F} \vdash \emptyset) \leq w(\mathcal{F}) + \log \mathrm{S_T}(\mathcal{F})$.*

From these facts, we can easily obtain something similar for satisfiable formulas having a unique solution.

Lemma 2.3 *If $\mathcal{F}(x_1, \ldots, x_n)$ has a unique solution $\boldsymbol{a} = (a_1, \ldots, a_n) \in \{0,1\}^n$, then $w(\mathcal{F} \vdash x_i^{a_i})$ $\leq 1 + w(F) + \log \mathrm{S_T}(\mathcal{F}, x_i)$.*

Proof. Note that $\mathcal{F}|_{x_i=1-a_i}$ is unsatisfiable; hence by Theorem 2.1, we have $w(\mathcal{F}|_{x_i=1-a_i} \vdash \emptyset)$ $\leq w(\mathcal{F}|_{x_i=1-a_i}) + \log \mathrm{S_T}(\mathcal{F}|_{x_i=1-a_i}, \emptyset)$. Let π be a refutation of $\mathcal{F}|_{x_i=1-a_i}$ such that $w(\pi) = w(F|_{x_i=1-a_i} \vdash \emptyset)$. From Lemma 2.2, it follows that $\mathcal{F} \vdash_{w(\pi)+1} x_i^{a_i}$. We then have that $w(\mathcal{F} \vdash x_i^{a_i})$ $\leq w(\pi) + 1 \leq 1 + w(\mathcal{F}|_{x_i=1-a_i}) + \log \mathrm{S_T}(\mathcal{F}|_{x_i=1-a_i}, \emptyset)$. But $w(\mathcal{F}|_{x_i=1-a_i}) \leq w(\mathcal{F})$; henceforth, we have $w(\mathcal{F} \vdash x_i^{a_i}) \leq 1 + w(\mathcal{F}) + \log \mathrm{S_T}(\mathcal{F}|_{x_i=1-a_i}, \emptyset)$.

Let θ be a tree-like derivation of the unit clause $x_i^{a_i}$ from $\mathcal{F}$ so that $size(\theta) = \mathrm{S_T}(\mathcal{F}, x_i)$. Consider the derivation $\theta|_{x_i=1-a}$ obtained from θ by removing all clauses that contain the literal $x_i^{1-a_i}$ and erasing all occurrences of the literal $x_i^{a_i}$. More precisely, we modify θ as follows on each of the following situations:

1. A resolution of the form $\mathrm{res}(A \vee y, B \vee \overline{y}) = (A \vee B)$ in θ, where literals $x_i^{a_i}, x_i^{1-a_i}$ do not appear in A nor in B: remain unchanged in $\theta|_{x_i=1-a}$.

2. A resolution of the form $\mathrm{res}(A \vee x^{a_i} \vee y, B \vee x^{a_i} \vee \overline{y}) = (A \vee B \vee x^{a_i})$ in θ: change it to $\mathrm{res}(A \vee y, B \vee \overline{y}) = (A \vee B)$ in $\theta|_{x_i=1-a}$.

3. A resolution of the form $\mathrm{res}(A \vee x^{1-a_i}, B \vee x^{a_i}) = (A \vee B)$ in θ: change it to the weakening rule $B \vdash (A \vee B)$ in $\theta|_{x_i=1-a}$.

4. A resolution of the form $\mathrm{res}(A \vee x^{1-a_i} \vee y, B \vee \overline{y}) = (A \vee B \vee x^{1-a_i})$ in θ: remove it in $\theta|_{x_i=1-a}$.

5. other cases: reduce trivially to the already listed ones.

It is easy to see that $\theta|_{x_i=1-a}$ is a tree-like derivation of the empty clause from $\mathcal{F}|_{x_i=1-a_i}$ and $\mathrm{size}(\theta|_{x_i=1-a}) \leq \mathrm{size}(\theta)$. We can conclude that $\mathrm{S_T}(\mathcal{F}|_{x_i=1-a_i}, \emptyset) \leq \mathrm{S_T}(\mathcal{F}, x_i)$. Recalling the inequality $w(\mathcal{F} \vdash x_i^{a_i}) \leq 1 + w(\mathcal{F}) + \log \mathrm{S_T}(\mathcal{F}|_{x_i=1-a_i}, \emptyset)$, the theorem follows. □

Corollary 2.2 $\mathrm{S_T}(\mathcal{F}, x_i) \geq 2^{w(\mathcal{F} \vdash x_i^{a_i}) - w(\mathcal{F}) - 1}$.

3 Lower Bound Proof

We prove our lower bound result, that is, the following theorem.

Theorem 3.1 *For any sufficiently large n and for any i and j, $1 \leq i, j \leq n$, we have* $\mathrm{S_T}(\mathrm{UPHP}_n, x_{ijij}) = 2^{\Omega(n)}$.

First, as in [BW99], we modify UPHP_n to a formula consisting of short clauses. Let EUPHP_n be a nondeterministic extension of UPHP_n as defined in [BW99] for PHP_n^m; EUPHP_n is obtained from UPHP_n by replacing P_{ij}^{R} and P_{ij}^{C} with the following equivalent nondeterministic extension CNF formulas EP_{ij}^{R} and EP_{ij}^{C}:

$$\begin{aligned} EP_{ij}^{\mathrm{R}} &\stackrel{\mathrm{def}}{=} (z_{ij,1}^{\mathrm{R}} \equiv (x_{iji1} \vee x_{iji2})) \wedge (z_{ij,2}^{\mathrm{R}} \equiv (z_{ij,1}^{\mathrm{R}} \vee x_{iji3})) \wedge \cdots \wedge (z_{ij,n-3}^{\mathrm{R}} \vee x_{ijin-1} \vee x_{ijin}), \quad \text{and} \\ EP_{ij}^{\mathrm{C}} &\stackrel{\mathrm{def}}{=} (z_{ij,1}^{\mathrm{C}} \equiv (x_{ij1j} \vee x_{ij2j})) \wedge (z_{ij,2}^{\mathrm{C}} \equiv (z_{ij,1}^{\mathrm{C}} \vee x_{ij3j})) \wedge \cdots \wedge (z_{ij,n-3}^{\mathrm{C}} \vee x_{ijn-1j} \vee x_{ijnj}). \end{aligned}$$

Intuitively, new variables $z_{ij,t}^{\mathrm{R}}$ (resp., $z_{ij,t}^{\mathrm{C}}$) express a disjunction of $x_{iji1}, \ldots, x_{iji(t+1)}$ (resp., $x_{ij1j}, \ldots, x_{ij(t+1)j}$. Note that EUPHP_n now consists of clauses with at most 3 literals; that is, $w(\mathrm{EUPHP}_n) \leq 3$. Observe also that for any $1 \leq i, j \leq n$ and $U \in \{\mathrm{C}, \mathrm{R}\}$, any assignment satisfying P_{ij}^U can be extended uniquely to the one satisfying EP_{ij}^U; thus, EUPHP_n is uniquely satisfiable.

Furthermore, it is easy to check that the following lemma holds; it then follows that any exponential lower bound proved for EUPHP_n immediately implies the same exponential lower bound for UPHP_n. Hence, in the following we will investigate EUPHP_n.

Lemma 3.1 *There exists a polynomial p such that for every tree-like derivation $\pi = [\mathrm{UPHP}_n \vdash x^a]$ of a unit clause x^a, there exists correspondingly a tree-like derivation $\phi = [\mathrm{EUPHP}_n \vdash x^a]$ satisfying* $\mathrm{S_T}(\phi) \leq p(n) \cdot \mathrm{S_T}(\pi)$.

In the following, let α_0 denote the unique satisfying assignment of EUPHP_n. That is, for any $1 \leq i, j, k, l \leq n$,

$$x_{ijkl}|_{\alpha_0} = \begin{cases} 1, & \text{if } kl = ij, \text{ and} \\ 0, & \text{otherwise.} \end{cases}$$

To the nondeterministic variables $z^{\mathrm{R}}_{ij,t}$ and $z^{\mathrm{C}}_{ij,t}$, we define α_0 so that values consistent with their interpretation are assigned. (Whenever a new assignment is defined, in the following, we follow this rule to extend assignments to the nondeterministic variables.)

From now on and throughout this section, let us fix any sufficiently large n and any index ij, $1 \leq i, j \leq n$, and prove $\mathrm{S_T}(\mathrm{UPHP}_n, x_{ijij}) = 2^{\Omega(n)}$. From the relation between proof size and width, it suffices to show that $w(\mathrm{EUPHP}_n \vdash x_{ijij}) = \Omega(n)$. To prove this bound, we follow [BW99] and argue as follows.

1. Define a complexity measure $\mu : \{Clauses\} \longrightarrow N$ such that $\mu(C) \leq 1$ for any clause C in the axioms of EUPHP_n.
2. Prove that $\mu(x_{ijij})$ is large.
3. Infer that in *any* derivation $\mathrm{EUPHP}_n \vdash x_{ijij}$, there is some medium size $\mu(C)$.
4. Prove that if $\mu(C)$ is medium, then $w(C)$ is large.

Define $\mathcal{A}_0 = \{A_{pq} : 1 \leq p, q \leq n\}$ where A_{pq} is the conjunction of all clauses on pigeon pq, more specifically, the conjunction of EP^{R}_{pq}, EP^{C}_{pq}, and all Hole Rules H_{pqkl} for $1 \leq k, l \leq n$. For any clause C and $\mathcal{A} \subseteq \mathcal{A}_0$, we say that $\mathcal{A}$ *implies* C, denoted by $\mathcal{A} \models C$, if every assignment satisfying every Boolean function $A \in \mathcal{A}$ satisfies also C. Define $\mu_{\mathcal{A}_0}(C) \overset{\text{def}}{=} \min\{|\mathcal{A}| : \mathcal{A} \subseteq \mathcal{A}_0 \,\text{and}\, \mathcal{A} \models C\}$.

Now we show the properties mentioned above for this $\mu_{\mathcal{A}_0}$. First we observe that the $\mu_{\mathcal{A}_0}$-measure of any clause of EUPHP_n is small.

Lemma 3.2 *For any clause in* EUPHP_n*, we have* $\mu_{\mathcal{A}_0}(C) \leq 1$.

Next by the following three lemmas, we compute the value of $\mu_{\mathcal{A}_0}(x_{ijij})$.

Lemma 3.3 *For $\mathcal{A} \subseteq \mathcal{A}_0$, if there are A_{iq} and A_{pj} such that $A_{iq} \notin \mathcal{A}$ and $A_{pj} \notin \mathcal{A}$, then $\mathcal{A} \not\models x_{ijij}$.*

Proof. If $q = j$, then we consider an assignment α that differs from α_0 only on x_{ijij}; that is, $x_{ijij}|_\alpha = 1 - x_{ijij}|_{\alpha_0}$ $(= 0)$. Then it satisfies $\mathcal{A}$, i.e., $\mathcal{A}|_\alpha = 1$, implying $A \not\models x_{ijij}$. The case $p = i$ is treated similarly. Otherwise we have $p \neq i$ and $q \neq j$. Let β be an assignment satisfying

$$x_{k_1k_2k_3k_4}|_\beta = \begin{cases} 0, & \text{if } k_1k_2 = ij \text{ and } k_3k_4 = ij, \\ 1, & \text{if } k_1k_2 = ij \text{ and } (k_3k_4 = iq \text{ or } k_3k_4 = pj), \\ 0, & \text{if } k_1k_2 = k_3k_4 = iq \text{ or } k_1k_2 = k_3k_4 = pj, \text{ and} \\ x_{k_1k_2k_3k_4}|_{\alpha_0}, & \text{otherwise.} \end{cases}$$

That is, the pigeon ij is ubiquitously staying alone in the holes iq and pj but not in the ij hole. Although $A_{iq}|_\beta = A_{pj}|_\beta = 0$, we have that $\mathcal{A}|_\beta = 1$, because A_{iq} and A_{pj} do not belong to $\mathcal{A}$. It follows that $\mathcal{A} \not\models x_{ijij}$. □

Lemma 3.4 $\mu_{\mathcal{A}_0}(x_{ijij}) \;\geq\; 2n-1.$

Proof. Suppose, for the sake of contradiction, that there exist $\mathcal{A} \subseteq \mathcal{A}_0$ such that $|\mathcal{A}| < 2n-1$ and $\mathcal{A} \models x_{ijij}$.

From Lemma 3.3 we may assume that either $\{A_{iq} : 1 \leq q \leq n\} \subseteq \mathcal{A}$ or $\{A_{pj} : 1 \leq p \leq n\} \subseteq \mathcal{A}$. W.l.o.g. assume that $\{A_{iq} : 1 \leq q \leq n\} \subseteq \mathcal{A}$. From $|\mathcal{A}| < 2n-1$, it follows that there exist $A_{p_1 1}, A_{p_2 2}, \ldots, A_{p_n n} \in \mathcal{A}_0$ such that $A_{p_1 1}, A_{p_2 2}, \ldots, A_{p_n n} \notin \mathcal{A}$. Consider then an assignment γ that satisfies

$$x_{k_1k_2k_3k_4}|_\gamma \;=\; \begin{cases} 0, & \text{if } k_1k_2 = iv \text{ and } k_3k_4 = iv \text{ for some } v,\ 1 \leq v \leq n, \\ 1, & \text{if } k_1k_2 = iv \text{ and } k_3k_4 = i(v+1) \text{ for some } v,\ 1 \leq v \leq n-1, \\ 1, & \text{if } k_1k_2 = in \text{ and } k_3k_4 = i1, \\ 1, & \text{if } k_1k_2 = iv \text{ and } k_3k_4 = p_v v \text{ for some } v,\ 1 \leq v \leq n, \\ 0, & \text{if } k_1k_2 = k_3k_4 = p_v v \text{ for some } v,\ 1 \leq v \leq n, \text{ and} \\ x_{k_1k_2k_3k_4}|_{\alpha_0}, & \text{otherwise.} \end{cases}$$

In words, every pigeon iv, $1 \leq v \leq n$, initially at the intersection of row i and column v is shifted right to the next right hole on the row i; also we have pigeon iv ubiquitously occupy alone the holes $p_v v$, where $p_v v$ is the index such that $A_{p_v v} \notin \mathcal{A}$. The fact that $A_{p_1 1}, A_{p_2 2}, \ldots, A_{p_n n} \notin \mathcal{A}$ assures us that these changes do not affect to the constraints in $\mathcal{A}$; hence, it still holds that $\mathcal{A}|_\gamma = 1$. But we have $x_{ijij}|_\gamma = 0$, contradicting $\mathcal{A} \models x_{ijij}$. $\square$

In fact, this bound is tight, as shown by the following lemma.

Lemma 3.5 $\mu_{\mathcal{A}_0}(x_{ijij}) = 2n-1.$

Proof. It suffices to find $\mathcal{A} \subseteq \mathcal{A}_0$ such that $|\mathcal{A}| = 2n-1$ and $\mathcal{A} \models x_{ijij}$. Let $\mathcal{A} = \{A_{iq} : 1 \leq q \leq n\} \cup \{A_{pj} : 1 \leq p \leq n\}$. Consider any assignment δ satisfying $\mathcal{A}|_\delta = 1$. Then under this assignment δ, there are precisely n distinct pigeons staying in holes of row i; similarly, there are precisely n distinct pigeons in column j, each in a different hole also.

Suppose that the pigeon ij is not in the hole ij. In order to satisfy the constraint A_{ij}, the pigeon ij has to ubiquitously occupy a hole on row i and a hole on column j. It follows that there exist two pigeons, say iv and uj, that are staying in the hole ij, contradicting the Hole Rule H_{ivij} and H_{ujij}. Therefore, it must hold that $x_{ijij}|_\delta = 1$, whence $\mathcal{A} \models x_{ijij}$. $\square$

The third step is provable by using the following sub-additive property w.r.t. resolution, which is clear from the nature of the resolution rule: for any A, B, and C such that $\mathrm{res}(A,B) = C$, we have $\mu_{\mathcal{A}_0}(C) \leq \mu_{\mathcal{A}_0}(A) + \mu_{\mathcal{A}_0}(B)$. Using this property, and from Lemma 3.2 and Lemma 3.4, the following fact is immediate.

Lemma 3.6 *In any derivation* $\mathrm{EUPHP}_n \vdash x_{ijij}$*, there must be a clause* C *such that* $n/3 \leq \mu_{\mathcal{A}_0}(C) \leq 2n/3$.

The final step is to show that the above clause C with medium measure $\mu_{\mathcal{A}_0}(C)$ should have a large width.

Lemma 3.7 *If a clause* C *satisfies* $n/3 \leq \mu_{\mathcal{A}_0}(C) \leq 2n/3$*, then* $|C| \geq n/3$.

Proof. The proof again resembles the proof of Lemma 4.9 of [BW99], but its combinatorial argument is a bit more involved.

Suppose that $|C| < n/3$. Let $\mathcal{A} \subseteq \mathcal{A}_0$ be a minimal set such that $\mathcal{A} \models C$; thus, from the assumption on C, we have $n/3 \leq |\mathcal{A}| \leq 2n/3$. Let $S(C)$ be the set of pigeons pq such that a variable of the type x_{pqkl}, $z^{\mathrm{R}}_{pq,t}$, or $z^{\mathrm{C}}_{pq,t}$ appears in C; it is the set of all pigeons that the clause C touches. Let $\mathcal{A}(C) \subseteq \mathcal{A}_0$ be the set of constraints A_{pq} such that $pq \in S(C)$. As $|C| < |\mathcal{A}|$ (since $|C| < n/3 \leq |\mathcal{A}|$), there is a constraint $A_{i_0 j_0} \in \mathcal{A} \setminus \mathcal{A}(C)$. By minimality of $\mathcal{A}$, there exists an assignment η such that $(\mathcal{A} \setminus \{A_{i_0 j_0}\})|_\eta = 1$ and $C|_\eta = A_{i_0 j_0}|_\eta = 0$. Without lose of generality, we may assume that η sets all variables outside of $\mathcal{A} \cup \mathcal{A}(C)$ to 0.

We will show that the assignment η can be modified so that the values $(\mathcal{A} \setminus \{A_{i_0 j_0}\})|_\eta$ and $C|_\eta$ remain invariant and moreover:

- there exists an empty hole $i_0 v$ on row i_0, i.e., for all pigeons pq we have $x_{pqi_0v}|_\eta = 0$, and
- there exists an empty hole uj_0 on column j_0, i.e., for all pigeons pq we have $x_{pquj_0}|_\eta = 0$.

Then consider an assignment ϕ defined as

$$x_{k_1k_2k_3k_4}|_\phi \;=\; \begin{cases} 1, & \text{if } k_1k_2 = i_0j_0 \text{ and } (k_3k_4 = i_0v \text{ or } k_3k_4 = uj_0); \\ 0, & \text{if } k_1k_2 = i_0j_0 \text{ and } k_3k_4 \neq i_0v \text{ and } k_3k_4 \neq uj_0; \text{ and} \\ x_{k_1k_2k_3k_4}|_\eta, & \text{otherwise.} \end{cases}$$

That is, the pigeon i_0j_0 ubiquitously occupies the holes (and only the holes) i_0v and uj_0. We claim that ϕ satisfies $\mathcal{A}|_\phi = 1$ and $C|_\phi = 0$, contradicting the fact that $\mathcal{A} \models C$. The claim follows easily recalling that $A_{i_0j_0} \notin \mathcal{A}(C)$ (hence $C|_\phi = 0$) and the holes i_0v and uj_0 were empty under the (modified) assignment η.

Below we show that it is possible to modify η only on row i_0 so that an empty hole i_0v is created while keeping $(\mathcal{A} \setminus \{A_{i_0j_0}\})|_\eta = 1$ and $C|_\eta = 0$. An argument for column j_0 is the same, and it is omitted here. Note that modifications are made only for holes on the row i_0 and on the column j_0 respectively; hence, they can be made independently except for the hole i_0j_0, which is treated in the same way.

Before explaining our actual modification on η, we show three claims for preparing our modification strategies. For any set S of pigeons and assignment ψ, denote by $H(S, \psi)$ the set of holes of row i_0 occupied by pigeons of S under the assignment ψ; that is, $H(S, \psi) = \{\, i_0l :$ there exists x_{pqi_0l} such that $pq \in S$ and $x_{pqi_0l}|_\psi = 1 \,\}$.

Claim 1 *For any assignment ψ such that $C|_\psi = 0$, there exists an assignment ξ obtained from ψ by freeing some holes of row i_0 so that $C|_\xi = 0$ and $|H(S(C), \xi)| \leq |C|$.*

Proof. Let $L(C)$ be the set of literals on the pigeon ij appearing negatively in C, i.e., literals in C of type $\overline{x_{pqi_0l}}$, $\overline{z^{\mathrm{R}}_{i_0q,t}}$, or $\overline{z^{\mathrm{C}}_{pq,t}}$. It is easy to see that $L(C)$ is the set of *all* literals y of C such that $y|_\psi = 0 \Longrightarrow \psi$ maps at least one pigeon in some hole on row i_0; assigning 0 to the other literals does not force any hole of row i_0 to be occupied. We modify ψ to ξ so that $C|_\xi = 0$ and at most $|L(C)|$ ($\leq |C|$) holes in row i_0 are occupied by some pigeon under ξ.

Consider first a literal $y \in L(C)$ of type $\overline{x_{pqi_0l}}$ or $\overline{z^{\mathrm{C}}_{pq,t}}$. For this literal, the condition $y|_\psi = 0$ is satisfied by mapping some pigeon to *at most one* hole on row i_0. Let $h(y)$ denote this hole. ($h(y)$ is left undetermined, if no such hole exists.) Next consider literals $y = \overline{z^{\mathrm{R}}_{i_0 q,t}}$ in $L(C)$. Recall that $\overline{z^{\mathrm{R}}_{i_0 q,t}} = \overline{x_{i_0 q i_0 1}} \wedge \cdots \wedge \overline{x_{i_0 q i_0 (t+1)}}$. Since $y|_\psi = 0$, some variable in the set $\{x_{i_0 q i_0 1}, \ldots, x_{i_0 q i_0 (t+1)}\}$ that must be assigned 1 by ψ; in other words, ψ maps the pigeon $i_0 q$ to some holes in $\{i_0 1, \ldots, i_0(t+1)\}$. Let $h(y)$ be, say, lexicographically the first one among such holes. Then let ξ be an assignment defined as follows:

$$x_{k_1 k_2 k_3 k_4}|_\xi = \begin{cases} 1, & \text{if } k_3 = i_0 \text{ and } k_3 k_4 = h(y) \text{ for some } y \in L(C); \\ 0, & \text{if } k_3 = i_0 \text{ but } k_3 k_4 \neq h(y) \text{ for all } y \in L(C); \text{ and} \\ x_{k_1 k_2 k_3 k_4}|_\psi, & \text{otherwise.} \end{cases}$$

In other words, on the row i_0, we keep the minimum number of pigeons $i_0 j$ for having the clause C unsatisfiable. It is easy to check that ξ has the required properties of our claim. □

In the following, for any $\mathcal{A} \subseteq \mathcal{A}_0$, let $S(\mathcal{A})$ denote the set of pigeons pq such that $A_{pq} \in \mathcal{A}$.

Claim 2 *For any $\mathcal{A} \subseteq \mathcal{A}_0$, let ψ be an assignment satisfying $\mathcal{A}|_\psi = 1$. Then there exists an assignment ξ obtained from ψ by freeing of some holes of row i_0 so that we have both $\mathcal{A}|_\xi = 1$ and $|H(S(\mathcal{A}), \xi)| \leq |S(\mathcal{A})|$.*

Proof. Consider any pigeon $pq \in S(\mathcal{A})$. If $|H(\{pq\}, \psi)| \leq 1$, then pq contributes at most one to $|H(S(\mathcal{A}), \psi)|$, and we do not need to worry about the pigeon pq. What we should consider is the other case, i.e., $|H(\{pq\}, \psi)| > 1$. Note that we may assume that $p = i_0$, because otherwise, we have $|H(\{pq\}, \psi)| \leq 1$. Now for any pigeon $i_0 q \in S(\mathcal{A})$ such that $|H(\{i_0 q\}, \psi)| > 1$, let $h(i_0 q)$ be the hole $i_0 q$ if it exists in $H(\{i_0 q\}, \psi)|$; otherwise $h(i_0 q) = i_0 l$ for any $i_0 l \in H(\{i_0 q\}, \psi)|$. Modify now ψ to the following assignment ξ:

$$x_{k_1 k_2 k_3 k_4}|_\xi = \begin{cases} 1, & \text{if } k_3 k_4 = h(i_0 q) \text{ for some } i_0 q \in S(\mathcal{A}); \\ 0, & \text{if } k_3 k_4 \in H(S, \psi) \text{ but } k_3 k_4 \neq h(i_0 q) \text{ for any } i_0 q \in S(\mathcal{A}); \text{ and} \\ x_{k_1 k_2 k_3 k_4}|_\psi, & \text{otherwise.} \end{cases}$$

Roughly speaking, we force each pigeon $i_0 q$ that occupies more than one hole on row i_0 to free the unnecessary holes. Only the point we must take care is not to set $x_{i_0 q i_0 q} = 0$ (if it were set 1 by ψ); this is for the column constraint on the pigeon $i_0 q$. One can check that ξ satisfies $\mathcal{A}|_\xi = 1$ and $|H(S(\mathcal{A}), \xi)| \leq |S(\mathcal{A})|$. □

Claim 3 *For any $\mathcal{A} \subseteq \mathcal{A}_0 \cap \mathcal{A}(C)$, let ψ be an assignment satisfying $\mathcal{A}|_\psi = 1$ and $C|_\psi = 0$. Then there exists an assignment ξ obtained from ψ by freeing of some holes of row i_0 so that $\mathcal{A}|_\xi = 1$, $C|_\xi = 0$, and $|H(\mathcal{A}, \xi)| \leq |C| + |\mathcal{A}|$.*

Proof. First modify ψ to ξ that is given by Claim 1. That is, ξ is an assignment satisfying that $|H(\mathcal{A}(C), \xi)| \leq |C|$ and $C|_\xi = 0$. If $\mathcal{A}|_\xi = 1$, then we are done. Otherwise, we have some $A_{pq} \in \mathcal{A}$ such that $A_{pq}|_\xi = 0$. Because $A_{pq}|_\psi = 1$, and ξ is obtained from ψ only by freeing some holes on row i_0, the current situation $A_{pq}|_\xi = 0$ is the result of removing the pigeon pq from holes on row i_0. Thus, we can choose one hole $i_0 u'$ from the row i_0 and flip back the value of precisely the variable $x_{pq i_0 u'}$ in order to make A_{pq} true. It is clear that after at most $|\mathcal{A}|$ flips, we obtain an assignment ξ such that $\mathcal{A}|_\xi = 1$ and $C|_\xi = 0$. The bound $|H(\mathcal{A}, \xi)| \leq |C| + |\mathcal{A}|$ holds, since at most one hole is added to $H(\mathcal{A}, \xi)$ at each flip. □

Now we resume the proof of Lemma 3.7. First we modify η to free all holes occupied by pigeon i_0j_0 on row i_0; this modification does not falsify $\mathcal{A} \setminus \{A_{i_0j_0}\}$ nor make C true. To apply the above claims to this η, we let $\mathcal{A}_1 = \mathcal{A} \setminus \{A_{i_0j_0}\}$, $\mathcal{A}_2 = \mathcal{A}_1 \setminus \mathcal{A}(C)$, and $\mathcal{A}_3 = \mathcal{A}_1 \cap \mathcal{A}(C)$. On the other hand, we let C_1 be a subset of C such that $\mathcal{A}(C_1) = \mathcal{A}(C) \setminus \mathcal{A}_1$, and let $C_3 = C \setminus C_1$. Then by applying Claim 1 to C_1 and η, we obtain ξ_1 from η such that at most $|C_1|$ holes on row i_0 are occupied by pigeons of $S(C_1)$. Next Claim 2 for A_2 gives us an assignment ξ_2 from ξ_1 such that at most $|\mathcal{A}_2$ holes are occupied by pigeons of $S(\mathcal{A}_2)$. Finally, Claim 3 for C_3 and A_3 provides an assignment ξ_3 from ξ_2 such that at most $|C_3| + |\mathcal{A}_3|$ holes are occupied by pigeons of $S(C_3) \cup S(\mathcal{A}_3)$. Thus, altogether at most

$$|C_1| + |\mathcal{A}_2| + |C_3| + |\mathcal{A}_3| \;=\; |C| + |\mathcal{A} \setminus \{A_{i_0j_0}\}| \;<\; n$$

holes on row i_0 are occupied by pigeons of $S(C_1) \cup S(\mathcal{A}_2) \cup S(C_3) \cup S(\mathcal{A}_3)$, which is equal to $S(\mathcal{A}) \cup S(C) \setminus \{i_0j_0\}$. Hence, there exists some hole i_0u on row i_0 that is not occupied by any pigeons in $S(\mathcal{A}) \cup S(C) \setminus \{i_0j_0\}$. Recall that the original η sets all variables not in $\mathcal{A} \cup \mathcal{A}(C)$ to 0 and that the pigeon i_0j_0 has been removed from the row i_0, which properties are inherited by ξ_3. Therefore, we can conclude that no pigeon stays in the hole i_0u under ξ_3. $\square$

Now we summarize our results. We have obtained an exponential lower bound for $\mathrm{S_T}(\mathrm{EUPHP}_n, x_{ijij})$. More precisely, from Corollary 2.2 and Lemma 3.7 and from the fact that $w(\mathrm{EUPHP}_n) \leq 3$, we have the following bound.

Theorem 3.2 *For any i and j, $1 \leq i, j \leq n$, we have* $\mathrm{S_T}(\mathrm{EUPHP}_n, x_{ijij}) \;\geq\; 2^{n/3-4}$.

This also implies similar bounds for the other unit clauses.

Corollary 3.3 *For any i, j, k, and l such that $1 \leq i, j, k, l \leq n$, $k \neq i$, and $l \neq j$, we have* $\mathrm{S_T}(\mathrm{EUPHP}_n, \overline{x_{ijil}}) \;\geq\; 2^{n/3-4}/n$ *and* $\mathrm{S_T}(\mathrm{EUPHP}_n, \overline{x_{ijkj}}) \;\geq\; 2^{n/3-4}/n$.

Proof. Consider any i and j, $1 \leq i, j \leq n$. If either $\mathrm{S_T}(\mathrm{EUPHP}_n, \overline{x_{ijil}}) < 2^{n/3-4}/n - 1$ for all l, $1 \leq l \neq j \leq n$, or $\mathrm{S_T}(\mathrm{EUPHP}_n, \overline{x_{ijkj}}) < 2^{n/3-4}/n - 1$ or all k, $1 \leq k \neq i \leq n$, then we would have a derivation of x_{ijij} of size $\leq 2^{n/3-4}$. Thus, there should be some x_{ijil} or x_{ijkj} for which $\mathrm{S_T}(\mathrm{EUPHP}_n, \overline{x_{ijkj}}) > 2^{n/3-4}/n - 1$. Then by the symmetry, the other variables should have the same bound. $\square$

Recall that x_{ijij} is easily derived from the Functional Rules; this fact and the above hardness implies the hardness of deriving any clause in the Functional Rules.

Corollary 3.4 *For some constant $d > 0$, and for any i, j, k, and l such that $11 \preceq kl \prec k'l' \preceq nn$, $k = i \vee l = j$, and $k' = i \vee l' = j$, we have* $\mathrm{S_T}(\mathrm{EUPHP}_n, \overline{x_{ijkl}} \vee \overline{x_{ijk'l'}}) \;\geq\; 2^{n/3}/n^d$.

References

[BW99] E. Ben-Sasson and A. Wigderson, Short proofs are narrow - resolution made simple, in *Proc. 31st ACM Sympos. on Theory of Comput.* (STOC'99), ACM, 517–526, 1999.

[DDL62] M. Davis, G. Logemann, and D. Loveland, A machine program for theorem proving, *Comm. of the ACM*, Vol.5, 394–397, 1962.

[DP60] M. Davis and H. Putnam, A computing procedure for quantification theory, *Journal of the ACM*, 7(3), 201–215, 1960.

[PPSZ98] R. Paturi, P. Pudlák, M.E. Saks, and F. Zane, An imporoved exponential-time algorithm for k-SAT, in *Proc. 39th IEEE Symposium on Foundations of Computer Science* (FOCS'98), IEEE, 628–637, 1998.

[Raz] A.A. Razborov, Proof complexity of pigeonhole principles, a manuscript.