

Placing quantified variants of 3-SAT and NOT-ALL-EQUAL 3-SAT in the polynomial hierarchy

Janosch Döcker, Britta Dorn, Simone Linz, and Charles Semples

August 22, 2019

Abstract

The complexity of variants of 3-SAT and NOT-ALL-EQUAL 3-SAT is well studied. However, in contrast, very little is known about the complexity of the problems' quantified counterparts. In the first part of this paper, we show that $\forall\exists$ 3-SAT is Π_2^P -complete even if (1) each variable appears exactly twice unnegated and exactly twice negated, (2) each clause is a disjunction of exactly three distinct variables, and (3) the number of universal variables is equal to the number of existential variables. Furthermore, we show that the problem remains Π_2^P -complete if (1a) each universal variable appears exactly once unnegated and exactly once negated, (1b) each existential variable appears exactly twice unnegated and exactly twice negated, and (2) and (3) remain unchanged. On the other hand, the problem becomes NP-complete for certain variants in which each universal variable appears exactly once. In the second part of the paper, we establish Π_2^P -completeness for $\forall\exists$ NOT-ALL-EQUAL 3-SAT even if (1') the Boolean formula is linear and monotone, (2') each universal variable appears exactly once and each existential variable appears exactly three times, and (3') each clause is a disjunction of exactly three distinct variables that contains at most one universal variable. On the positive side, we uncover variants of $\forall\exists$ NOT-ALL-EQUAL 3-SAT that are co-NP-complete or solvable in polynomial time.

Keywords: 3-Sat, Not-All-Equal 3-Sat, quantified satisfiability, polynomial hierarchy, bounded variable appearances, computational complexity.

1 Introduction

The Boolean satisfiability problem SAT plays a major role in the theory of NP-completeness. It was the first problem shown to be complete for the class NP (Cook's Theorem [3]) and many NP-hardness results are established by using this problem, or restricted variants thereof, as a starting point for polynomial-time reductions. Restricted variants of a problem that remain NP-complete are particularly useful as they allow for the possibility of simpler proofs and stronger results.

The most prominent NP-complete variant of the Boolean satisfiability problem is 3-SAT. Here we are given a conjunction of clauses such that each clause contains exactly three literals, where a literal is a propositional variable or its negation. An instance of 3-SAT is a yes-instance if there is a truth assignment to the propositional variables¹ such that at least one literal of each clause evaluates to true. Interestingly, even within 3-SAT, we can restrict the problem further. For example, for instances of 3-SAT in which each clause contains exactly three distinct variables, Tovey [17, Theorem 2.3] proved that 3-SAT remains NP-complete if each variable appears in at most four clauses. Furthermore, this result also holds if each variable appears exactly twice unnegated and exactly twice negated [1, Theorem 1]. On the other hand, the problem becomes trivial, i.e., the answer is always yes, if each variable appears at most three times [17, Theorem 2.4].

A popular NP-complete variant of 3-SAT called NOT-ALL-EQUAL 3-SAT (NAE-3-SAT) asks whether we can assign truth values to the variables such that at least one, but not all, of the literals of each clause evaluate to true. Schaefer [15] first established NP-completeness of NAE-3-SAT in the setting where each clause contains at most three literals. Subsequently, Karpinski and Piecuch [9, 10] showed that NAE-3-SAT is NP-complete if each variable appears at most four times. Furthermore, Porschen et al. [12, Theorem 3] showed that NAE-3-SAT remains NP-complete if (i) each literal appears at most once in any clause, and (ii) the input formula is *linear* and *monotone*, that is, each pair of distinct clauses share at most one variable and no clause contains a literal that is the negation of some variable. Following on from this last result, Darmann and Döcker [5] showed recently that NAE-3-SAT remains NP-complete if, in addition to (i) and (ii), each variable appears exactly four times. By contrast, if a monotone conjunction of clauses has the property that each variable appears at most three times, NAE-3-SAT can be decided in linear time [11, Theorem 4, p. 186].

¹From now on, we simply say variable instead of propositional variable since all variables used in the paper take only values representing true and false.

In this paper, we consider generalized variants of 3-SAT and NAE-3-SAT, namely $\forall\exists$ 3-SAT and $\forall\exists$ NAE-3-SAT, respectively. Briefly, $\forall\exists$ 3-SAT is a quantified variant of 3-SAT, where each variable is either *universal* or *existential*. The decision problem $\forall\exists$ 3-SAT asks if, for every assignment of truth values to the universal variables, there exists an assignment of truth values to the existential variables such that at least one literal of each clause evaluates to true. Observe that, if an instance of $\forall\exists$ 3-SAT does not contain a universal variable, then this instance reduces to an instance of 3-SAT. Analogously, we can think of $\forall\exists$ NAE-3-SAT as a generalized variant of NAE-3-SAT. Formal definitions of both problems are given in the next section.

Stockmeyer [16] and Dahlhaus et al. [6] showed, respectively, that $\forall\exists$ 3-SAT and $\forall\exists$ NAE-3-SAT are complete for the second level of the polynomial hierarchy or, more precisely, complete for the class Π_2^P . In this paper, we establish Π_2^P -completeness for restricted variants of these two quantified problems. In particular, we show that $\forall\exists$ 3-SAT is Π_2^P -complete if each universal variable appears exactly once unnegated and exactly once negated, and each existential variable appears exactly twice unnegated and exactly once negated or each existential variable appears exactly once unnegated and exactly twice negated. Although we do not consider approximation aspects in this paper, by way of comparison, Haviv et al. [8] showed that approximating a particular optimization version of $\forall\exists$ 3-SAT is Π_2^P -hard even if each universal variable appears at most twice and each existential variable appears at most three times. Whether optimization versions of the Π_2^P -complete problems presented in this paper are Π_2^P -hard to approximate is a question that we leave for future research. Furthermore, we establish Π_2^P -completeness for $\forall\exists$ 3-SAT if each universal variable appears exactly s_1 times unnegated and exactly s_2 times negated, each existential variable appears exactly t_1 times unnegated and exactly t_2 times negated, and the following three properties are satisfied: (i) $s_1 = s_2$, (ii) $s_1 \in \{1, 2\}$, and (iii) $t_1 = t_2 = 2$. These latter completeness results hold even if each clause is a disjunction of exactly three distinct variables and the number of universal and existential variables is *balanced*, that is, the number of universal and existential variables are the same.

Turning to $\forall\exists$ NAE-3-SAT, we show that the problem remains Π_2^P -complete if each universal variable appears exactly once, each clause contains at most one universal variable, each existential variable appears exactly three times, and the conjunction of clauses is both linear and monotone. Interestingly, while one appearance of each universal variable is enough to obtain a Π_2^P -hardness result in this setting, the same is not true for $\forall\exists$ 3-SAT unless the polynomial hierarchy collapses [8, p. 55].

The remainder of the paper is organized as follows. The next section introduces notation and terminology, and formally states three variants of $\forall\exists$ 3-SAT and $\forall\exists$ NOT-ALL-EQUAL 3-SAT that are the main focus of this paper. Section 3 (resp. Section 4) investigates the computational complexity of $\forall\exists$ 3-SAT (resp. $\forall\exists$ NOT-ALL-EQUAL 3-SAT). Both of Sections 3 and 4 start with a subsection on enforcers that are needed for the subsequent hardness proofs and, in terms of future work, we expect to be of independent interest in their own right.

2 Preliminaries

This section introduces notation and terminology that is used throughout the paper.

Let $V = \{x_1, x_2, \dots, x_n\}$ be a set of variables. A *literal* is a variable or its negation. We denote the set $\{x_i, \bar{x}_i : i \in \{1, 2, \dots, n\}\}$ of all literals that correspond to elements in V by \mathcal{L}_V . A *clause* is a disjunction of a subset of \mathcal{L}_V . If a clause contains exactly k distinct literals for $k \geq 1$, then it is called a k -*clause*. For example, $(x_1 \vee \bar{x}_2 \vee x_4)$ is a 3-clause. A *Boolean formula in conjunctive normal form* (CNF) is a conjunction of clauses, i.e., an expression of the form $\varphi = \bigwedge_{j=1}^m C_j$, where C_j is a clause for all j . In what follows, we refer to a Boolean formula in conjunctive normal form simply as a *Boolean formula*. Now, let φ be a Boolean formula. We say that φ is *linear* if any pair of distinct clauses share at most one variable and that it is *monotone* if no clause contains an element in $\{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n\}$. Furthermore, if each clause contains at most k literals, it is said to be in k -*CNF*. For each variable $x_i \in V$, we denote the number of appearances of x_i in φ plus the number of appearances of \bar{x}_i in φ by $a(x_i)$. A *variable assignment* or, equivalently, *truth assignment* for V is a mapping $\beta: V \rightarrow \{T, F\}$, where T represents the truth value True and F represents the truth value False. A truth assignment β *satisfies* φ if at least one literal of each clause evaluates to T under β . If there exists a truth assignment that satisfies φ , we say that φ is *satisfiable*. For a truth assignment β that satisfies φ and has the additional property that at least one literal of each clause evaluates to F , we say that β *nae-satisfies* φ . Lastly, let V and V' be two disjoint sets of variables, let β be a truth assignment for V , and let β' be a truth assignment for $V \cup V'$. We say that β' *extends* β (or, alternatively, that β extends to $V \cup V'$) if $\beta(x_i) = \beta'(x_i)$ for each $x_i \in V$.

A *quantified Boolean formula* Φ over a set $V = \{x_1, x_2, \dots, x_n\}$ of vari-

ables is a formula of the form

$$\forall x_1 \cdots \forall x_p \exists x_{p+1} \cdots \exists x_n \bigwedge_{j=1}^m C_j.$$

The variables x_1, x_2, \dots, x_p are *universal* variables of Φ and the variables $x_{p+1}, x_{p+2}, \dots, x_n$ are *existential* variables of Φ . Furthermore, for variables $x_i, x_{i+1}, \dots, x_{i'}$ with $1 \leq i < i' \leq p$ and $x_{i''}, x_{i''+1}, \dots, x_{i'''}$ with $p+1 \leq i'' < i''' \leq n$, we define

$$\forall X_i^{i'} := \forall x_i \cdots \forall x_{i'} \quad \text{and} \quad \exists X_{i''}^{i'''} := \exists x_{i''} \cdots \exists x_{i'''},$$

respectively and, similarly,

$$X_i^{i'} := \{x_i, \dots, x_{i'}\} \quad \text{and} \quad X_{i''}^{i'''} := \{x_{i''}, \dots, x_{i'''}\},$$

respectively.

We next introduce notation that transforms a Boolean formula φ into another such formula. Specifically, we use $\varphi[x \mapsto y]$ to denote the Boolean formula obtained from φ by replacing each appearance of variable x with variable y (i.e., replace x with y and replace \bar{x} with \bar{y}). For pairwise distinct pairs $(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)$ of variables, we use $\varphi[x_1 \mapsto y_1, \dots, x_k \mapsto y_k]$ to denote the Boolean formula obtained from φ by simultaneously replacing each appearance of variable x_i by variable y_i for $1 \leq i \leq k$. Since the variables are pairwise distinct, note that this operation is well-defined. Lastly, for a constant $b \in \{T, F\}$, the Boolean formula $\varphi[x \mapsto b]$ is obtained from φ by replacing each appearance of variable x by b .

The polynomial hierarchy. An *oracle* for a complexity class A is a black box that, in constant time, outputs the answer to any given instance of a decision problem contained in A . The *polynomial hierarchy* is a system of nested complexity classes that are defined recursively as follows. Set

$$\Sigma_0^P = \Pi_0^P = P,$$

and define, for all $k \geq 0$,

$$\Sigma_{k+1}^P = \text{NP}^{\Sigma_k^P} \quad \text{and} \quad \Pi_{k+1}^P = \text{co-NP}^{\Sigma_k^P},$$

where a problem is in $\text{NP}^{\Sigma_k^P}$ (resp. $\text{co-NP}^{\Sigma_k^P}$) if we can verify an appropriate certificate of a yes-instance (resp. no-instance) in polynomial time when given access to an oracle for Σ_k^P . By definition, $\Sigma_1^P = \text{NP}$ and $\Pi_1^P = \text{co-NP}$. We say that the classes Σ_k^P and Π_k^P are on the *k-th level* of the polynomial hierarchy.

For all $k \geq 0$, there are complete problems under polynomial-time many-one reductions for Σ_k^P and Π_k^P . However, while, for example, the complexity class Π_2^P generalizes both NP and co-NP, it remains an open question whether $\Sigma_k^P \neq \Sigma_{k+1}^P$ or $\Pi_k^P \neq \Pi_{k+1}^P$ for any $k \geq 0$. For further details of the polynomial hierarchy, we refer the interested reader to Garey and Johnson's book [7], an article by Stockmeyer [16], as well as to the compendium by Schaefer and Umans [14] for a collection of problems that are complete for the second or higher levels of the polynomial hierarchy.

The following two Π_2^P -complete problems are the starting points for the work presented in this paper.

$\forall\exists$ 3-SAT

Input. A quantified Boolean formula

$$\forall x_1 \cdots \forall x_p \exists x_{p+1} \cdots \exists x_n \bigwedge_{j=1}^m C_j$$

over a set $V = \{x_1, x_2, \dots, x_n\}$ of variables, where each clause C_j is a disjunction of at most three literals.

Question. For every truth assignment for $\{x_1, x_2, \dots, x_p\}$, does there exist a truth assignment for $\{x_{p+1}, x_{p+2}, \dots, x_n\}$ such that each clause of the formula is satisfied?

$\forall\exists$ NOT-ALL-EQUAL 3-SAT ($\forall\exists$ NAE-3-SAT)

Input. A quantified Boolean formula

$$\forall x_1 \cdots \forall x_p \exists x_{p+1} \cdots \exists x_n \bigwedge_{j=1}^m C_j$$

over a set $V = \{x_1, x_2, \dots, x_n\}$ of variables, where each clause C_j is a disjunction of at most three literals.

Question. For every truth assignment for $\{x_1, x_2, \dots, x_p\}$, does there exist a truth assignment for $\{x_{p+1}, x_{p+2}, \dots, x_n\}$ such that each clause of the formula is nae-satisfied?

Stockmeyer [16], and Eiter and Gottlob [6] established Π_2^P -completeness for $\forall\exists$ 3-SAT and $\forall\exists$ NAE-3-SAT, respectively.

The main focus of this paper are the following three restricted variants of $\forall\exists$ 3-SAT and $\forall\exists$ NAE-3-SAT.

BALANCED $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2)

Input. Four non-negative integers s_1, s_2, t_1, t_2 , and a quantified Boolean

formula

$$\forall x_1 \cdots \forall x_p \exists x_{p+1} \cdots \exists x_n \bigwedge_{j=1}^m C_j$$

over a set $V = \{x_1, x_2, \dots, x_n\}$ of variables such that (i) $n = 2p$, (ii) each C_j is a 3-clause that contains three *distinct* variables, and (iii), amongst the clauses, each universal variable appears unnegated exactly s_1 times and negated exactly s_2 times, and each existential variable appears unnegated exactly t_1 times and negated exactly t_2 times.

Question. For every truth assignment for $\{x_1, x_2, \dots, x_p\}$, does there exist a truth assignment for $\{x_{p+1}, x_{p+2}, \dots, x_n\}$ such that each clause of the formula is satisfied?

In addition, we also consider the decision problem that is obtained from BALANCED $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) by omitting property (i) in the statement of the input. We refer to the resulting problem as $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) . Lastly, we consider the following problem.

MONOTONE $\forall\exists$ NAE-3-SAT- (s, t)

Input. Two non-negative integers s and t , and a monotone quantified Boolean formula

$$\forall x_1 \cdots \forall x_p \exists x_{p+1} \cdots \exists x_n \bigwedge_{j=1}^m C_j$$

over a set $V = \{x_1, x_2, \dots, x_n\}$ of variables such that (i) each C_j is a 3-clause that contains three *distinct* variables and (ii), amongst the clauses, each universal variable appears exactly s times and each existential variable appears exactly t times.

Question. For every truth assignment for $\{x_1, x_2, \dots, x_p\}$, does there exist a truth assignment for $\{x_{p+1}, x_{p+2}, \dots, x_n\}$ such that each clause of the formula is nae-satisfied?

Enforcers. To establish the results of this paper, we will frequently use the concept of enforcers. An *enforcer* (sometimes also called a *gadget*) [1] is a Boolean formula, where the formula itself and each truth assignment that satisfies it have a certain structure. Enforcers are used in polynomial-time reductions to add additional restrictions on how yes-instances can be obtained.

We next detail two unquantified enforcers that were introduced by Berman et al. [1, p.3] and that lay the foundation for several other enforcers that are new to this paper and will be introduced in the following sections. First, let ℓ_1, ℓ_2 and ℓ_3 be three, not necessarily distinct, literals. Without loss of generality, we may assume that $\ell_1 \in \{x_1, \bar{x}_1\}$, $\ell_2 \in \{x_2, \bar{x}_2\}$, and $\ell_3 \in \{x_3, \bar{x}_3\}$. Now consider the following enforcer to which we refer to as

\mathcal{S} -enforcer:

$$\mathcal{S}(\ell_1, \ell_2, \ell_3) = (\ell_1 \vee \bar{a} \vee b) \wedge (\ell_2 \vee \bar{b} \vee c) \wedge (\ell_3 \vee a \vee \bar{c}) \wedge (a \vee b \vee c) \wedge (\bar{a} \vee \bar{b} \vee \bar{c}),$$

where a , b , and c are new variables such that $\{x_1, x_2, x_3\} \cap \{a, b, c\} = \emptyset$. Let $\beta: \{x_1, x_2, x_3, a, b, c\} \rightarrow \{T, F\}$ be a truth assignment. The next observation is an immediate consequence from the fact that, if $\beta(\ell_1) = \beta(\ell_2) = \beta(\ell_3) = F$, then, as the first three clauses form a cyclic implication chain which can only be satisfied by setting $\beta(a) = \beta(b) = \beta(c)$, either the fourth or fifth clause is not satisfied.

Observation 2.1. *Consider the boolean formula $\mathcal{S}(\ell_1, \ell_2, \ell_3)$, where $\ell_i \in \{x_i, \bar{x}_i\}$, and let V be its associated set of variables. A truth assignment β for the variables $\{x_1, x_2, x_3\}$ can be extended to a truth assignment β' for V that satisfies $\mathcal{S}(\ell_1, \ell_2, \ell_3)$ if and only if $\beta(\ell_i) = T$ for some $i \in \{1, 2, 3\}$.*

Remark. We note that Observation 2.1 holds, even if x_1 is a universal variable and all other variables in $\{x_2, x_3, a, b, c\}$ are existential in which case we will write $\mathcal{S}_u(\ell_1, \ell_2, \ell_3)$ to denote the gadget.

In what follows, we will use enforcers that are built of several copies of the \mathcal{S} -enforcer. In such a case, for each pair of \mathcal{S} -enforcer copies, the two 3-element sets of new variables are disjoint.

Again following the constructions from Berman et al. [1], consider a second enforcer:

$$x^{(2)} = \mathcal{S}(x, y, y) \wedge \mathcal{S}(x, \bar{y}, \bar{y}).$$

Note that $x^{(2)}$ is a Boolean formula over eight variables. Moreover, each clause contains three distinct variables since the copies of y and \bar{y} are distributed over different clauses in $\mathcal{S}(x, y, y)$ and $\mathcal{S}(x, \bar{y}, \bar{y})$, respectively. Lastly, each variable, except for x , appears exactly twice unnegated and twice negated in $x^{(2)}$. Now, the next observation follows by construction and Observation 2.1.

Observation 2.2. *Consider the Boolean formula $x^{(2)}$ over a set V of eight variables, where $x, y \in V$. A truth assignment β for $\{x\}$ can be extended to a truth assignment β' for V that satisfies $x^{(2)}$ if and only if $\beta(x) = T$.*

We will use the \mathcal{S} -enforcer and $x^{(2)}$ as well as extensions thereof in the proofs of several results established in this paper.

3 Hardness of balanced and unbalanced versions of $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2)

3.1 New enforcers

We start by describing three new enforcers, with the first one building upon the enforcers introduced in the previous section. Consider the following gadget:

$$E(x) = \mathcal{S}(x, y, y) \wedge \mathcal{S}(x, \bar{y}, \bar{y}) \wedge \mathcal{S}(\bar{x}, z, \bar{z}) \wedge \mathcal{S}(z, \bar{z}, u) \wedge \mathcal{S}(u, \bar{u}, \bar{u})$$

which is an extended variant of the enforcer $x^{(2)}$. We call x the *enforcer variable* of $E(x)$. Note that every variable in $\{u, y, z\}$ appears exactly twice unnegated and exactly twice negated in $E(x)$, and that x appears exactly twice unnegated and exactly once negated in $E(x)$. Moreover, by construction and Observation 2.2, it follows that $E(x)$ is satisfiable by setting x to be T , and by setting all remaining 18 variables appropriately.

Observation 3.1. *Consider the gadget $E(x)$, and let V be its associated set of variables. A truth assignment β for $\{x\}$ can be extended to a truth assignment β' for V that satisfies $E(x)$ if and only if $\beta(x) = T$.*

We now turn to two quantified enforcers whose purpose is to increase the number of universal variables by one and three, respectively, relative to the number of existential variables. First, let

$$\begin{aligned} Q^1 = & (u \vee v \vee a) \wedge (u \vee v \vee b) \wedge (\bar{u} \vee \bar{v} \vee \bar{a}) \wedge (\bar{u} \vee \bar{v} \vee \bar{b}) \wedge \\ & (a \vee \bar{b} \vee r) \wedge (\bar{a} \vee b \vee r) \wedge (c \vee \bar{d} \vee \bar{r}) \wedge (\bar{c} \vee d \vee \bar{r}) \wedge \\ & (w \vee q \vee c) \wedge (w \vee q \vee d) \wedge (\bar{w} \vee \bar{q} \vee \bar{c}) \wedge (\bar{w} \vee \bar{q} \vee \bar{d}), \end{aligned}$$

where u, v, w, q, r are universal variables, and a, b, c, d are existential variables. Observe that each variable of Q^1 appears exactly twice unnegated and exactly twice negated. Second, let

$$\begin{aligned} Q^3 = & (u \vee r \vee a) \wedge (\bar{u} \vee \bar{b} \vee \bar{a}) \wedge (v \vee q \vee b) \wedge \\ & (\bar{v} \vee \bar{r} \vee \bar{a}) \wedge (w \vee a \vee b) \wedge (\bar{w} \vee \bar{q} \vee \bar{b}), \end{aligned}$$

where u, v, w, q, r are universal variables and a, b are existential variables. Observe that each universal variable of Q^3 appears exactly once unnegated and exactly once negated, and that each existential variable of Q^3 appears exactly twice unnegated and exactly twice negated.

Lemma 3.1. *The quantified Boolean formula*

$$\forall u \forall v \forall w \forall q \forall r \exists a \exists b \exists c \exists d Q^1$$

is a yes-instance of $\forall\exists$ 3-SAT.

Proof. Let $U = \{u, v, w, q, r\}$, and let $E = \{a, b, c, d\}$. Furthermore, let $\beta: U \rightarrow \{T, F\}$ be a truth assignment for U . We extend β to $\beta': U \cup E \rightarrow \{T, F\}$ as follows:

$$\beta'(a) = \beta'(b) = \overline{\beta(u)}, \quad \beta'(c) = \beta'(d) = \overline{\beta(w)}.$$

It is now easy to verify that β' satisfies all clauses. Thus, Q^1 is a yes-instance of $\forall\exists$ 3-SAT. \square

Lemma 3.2. *The quantified Boolean formula*

$$\forall u \forall v \forall w \forall q \forall r \exists a \exists b Q^3$$

is a yes-instance of $\forall\exists$ 3-SAT.

Proof. Let $U = \{u, v, w, q, r\}$, and let $E = \{a, b\}$. Furthermore, let $\beta: U \rightarrow \{T, F\}$ be a truth assignment for U . We extend β to a truth assignment $\beta': U \cup E \rightarrow \{T, F\}$ for Q^3 as follows:

$$\beta'(a) = \overline{\beta(u)} \wedge \overline{\beta(r)}, \quad \beta'(b) = \overline{\beta(w)} \vee \overline{\beta(q)}.$$

It is now straightforward to check that Q^3 is a yes-instance of $\forall\exists$ 3-SAT. \square

3.2 Hardness of BALANCED $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2)

In this section, we show that BALANCED $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) is Π_2^P -complete when

$$(s_1, s_2, t_1, t_2) \in \{(2, 2, 2, 2), (1, 1, 2, 2)\}.$$

To this end, for a clause C , we use \overline{C} to denote the clause obtained from C by replacing each literal with its negation and call \overline{C} the *complement* of C . For example, if $C = (x_1 \vee \overline{x_2} \vee \overline{x_3})$, then $\overline{C} = (\overline{x_1} \vee x_2 \vee x_3)$.

Theorem 3.1. BALANCED $\forall\exists$ 3-SAT- $(2, 2, 2, 2)$ is Π_2^P -complete.

Proof. Noting that BALANCED $\forall\exists$ 3-SAT- $(2, 2, 2, 2)$ is a special case of $\forall\exists$ 3-SAT, we deduce that the problem is in Π_2^P . We show that the problem is Π_2^P -hard by a reduction from $\forall\exists$ NAE-3-SAT. Let

$$\Phi_1 = \forall X_1^p \exists X_{p+1}^n \varphi$$

be an instance of $\forall\exists$ NAE-3-SAT, where

$$\varphi = \bigwedge_{j=1}^m C_j$$

is a Boolean formula over a set $V_1 = \{x_1, x_2, \dots, x_n\}$ of variables such that C_j is a disjunction of at most three literals for all $j \in \{1, 2, \dots, m\}$ and no C_j contains a single literal since, otherwise, Φ_1 is a no-instance. Following Schaefer [13, p. 298] and noting that his reduction translates without changes to $\forall\exists$ NAE-3-SAT, we first modify Φ_1 using the following transformation that turns every universal variable x_i of Φ_1 into an existential variable y_i and introduces the set of new universal variables $\{z_1, z_2, \dots, z_p\}$:

$$\begin{aligned}\Phi_2 &= \forall Z_1^p \exists X_{p+1}^n \exists Y_1^p \varphi[x_1 \mapsto y_1, \dots, x_p \mapsto y_p] \wedge \bigwedge_{i=1}^p ((\bar{z}_i \vee y_i) \wedge (z_i \vee \bar{y}_i)). \\ &= \forall Z_1^p \exists X_{p+1}^n \exists Y_1^p \varphi',\end{aligned}$$

where $\varphi' = \varphi[x_1 \mapsto y_1, \dots, x_p \mapsto y_p] \wedge \bigwedge_{i=1}^p ((\bar{z}_i \vee y_i) \wedge (z_i \vee \bar{y}_i))$. Let V_2 be the set of variables of Φ_2 .

3.1.1. Φ_1 is a yes-instance of $\forall\exists$ NAE-3-SAT if and only if Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT.

Proof. First, suppose that Φ_1 is a yes-instance of $\forall\exists$ NAE-3-SAT. Let β_1 be a truth assignment for V_1 that nae-satisfies Φ_1 , and let β_2 be the following truth assignment for V_2 :

- (i) set $\beta_2(x_i) = \beta_1(x_i)$ for each $i \in \{p+1, p+2, \dots, n\}$;
- (ii) set $\beta_2(y_i) = \beta_1(x_i)$ for each $i \in \{1, 2, \dots, p\}$;
- (iii) set $\beta_2(z_i) = \beta_2(y_i)$ for each $i \in \{1, 2, \dots, p\}$.

By (iii), it is straightforward to check that β_2 nae-satisfies Φ_2 and that, for every truth assignment for Z_1^p , there exists a truth assignment for $X_{p+1}^n \cup Y_1^p$ that nae-satisfies Φ_2 . Hence, Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT.

Second, suppose that Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT. Let β_2 be a truth assignment for V_2 that nae-satisfies Φ_2 . By construction of Φ_2 , it follows that $\beta_2(z_i) = \beta_2(y_i)$ for each $i \in \{1, 2, \dots, p\}$. Hence β_1 with $\beta_1(x_i) = \beta_2(y_i)$ for each $i \in \{1, 2, \dots, p\}$, and $\beta_1(x_i) = \beta_2(x_i)$ for each $i \in \{p+1, p+2, \dots, n\}$ is a truth assignment for V_1 that nae-satisfies Φ_2 . Thus Φ_1 is a yes-instance of $\forall\exists$ NAE-3-SAT \square

For each $w_i \in X_{p+1}^n \cup Y_1^p$, we use $a(w_i)$ to denote the number of appearances of w_i in φ' throughout the remainder of this proof. Next, we apply, in turn, the following transformation adapted from Berman et al. [1, p. 4] yielding an instance of $\forall\exists$ 3-SAT.

1. Replace $\exists X_{p+1}^n$ in Φ_2 with the following list of existential variables:

$$\exists x_{p+1,1} \exists x_{p+1,2} \cdots \exists x_{p+1,a(x_{p+1})} \cdots \exists x_{n,1} \exists x_{n,2} \cdots \exists x_{n,a(x_n)}$$

Similarly, replace $\exists Y_1^p$ in Φ_2 with the following list of existential variables:

$$\exists y_{1,1} \exists y_{1,2} \cdots \exists y_{1,a(y_1)} \cdots \exists y_{p,1} \exists y_{p,2} \cdots \exists y_{p,a(y_p)}.$$

Lastly, for each existential variable $w_i \in X_{p+1}^n \cup Y_1^p$ and all $k \in \{1, \dots, a(w_i)\}$, replace the k -th appearance of w_i in φ' by $w_{i,k}$.

2. Replace each clause C_j with $C_j \wedge \overline{C}_j$.
3. For each $w_i \in X_{p+1}^n \cup Y_1^p$, introduce the clauses

$$(\overline{w_{i,1}} \vee w_{i,2}) \wedge (\overline{w_{i,2}} \vee w_{i,3}) \wedge \cdots \wedge (\overline{w_{i,a(w_i)-1}} \vee w_{i,a(w_i)}) \wedge (\overline{w_{i,a(w_i)}} \vee w_{i,1}).$$

4. Replace each 2-clause $(\ell_1 \vee \ell_2)$ by $(\ell_1 \vee \ell_2 \vee \overline{u}) \wedge E(u)$, where u and all 18 variables introduced by $E(u)$ are new existential variables. Append all 19 new variables to the list of existential variables.

Let Φ_3 denote the formula constructed by the preceding four-step procedure, and let V_3 be the set of variables of Φ_3 .

3.1.2. Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT if and only if Φ_3 is a yes-instance of $\forall\exists$ 3-SAT.

Proof. First, suppose that Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT. Let β_2 be a truth assignment for V_2 that nae-satisfies Φ_2 . Obtain a truth assignment β_3 for V_3 as follows:

- (i) set $\beta_3(z_i) = \beta_2(z_i)$ for each $i \in \{1, 2, \dots, p\}$;
- (ii) set $\beta_3(x_{i,k}) = \beta_2(x_i)$ for each $i \in \{p+1, p+2, \dots, n\}$ and $k \in \{1, 2, \dots, a(x_i)\}$;
- (iii) set $\beta_3(y_{i,k}) = \beta_2(y_i)$ for each $i \in \{1, 2, \dots, p\}$ and $k \in \{1, 2, \dots, a(y_i)\}$.

Additionally, for each 2-clause $C = (\ell_1 \vee \ell_2)$ that is replaced with the \mathcal{S} -enforcer $(\ell_1 \vee \ell_2 \vee \overline{u}) \wedge E(u)$ in Step 4, set $\beta_3(u) = T$, and set all 18 existential variables introduced by $E(u)$ such that the 25 clauses of $E(u)$ are satisfied. By construction of $E(u)$ and Observation 3.1, this is always possible. If C is a 2-clause of Φ_2 , then, as C is nae-satisfied by β_2 , it follows that β_3 satisfies $(\ell_1 \vee \ell_2 \vee \overline{u}) \wedge E(u)$. If C is initially a 2-clause introduced in Step 3 and then replaced in Step 4, it follows by (ii) and (iii) that β_3 satisfies $(\ell_1 \vee \ell_2 \vee \overline{u}) \wedge E(u)$. Noting that if a truth assignment nae-satisfies a clause,

then it also nae-satisfies its complement, it is now straightforward to check that β_3 satisfies Φ_3 and, hence, Φ_3 is a yes-instance of $\forall\exists$ 3-SAT.

Second, suppose that Φ_3 is a yes-instance of $\forall\exists$ 3-SAT. Let β_3 be a truth assignment that satisfies Φ_3 . Let u be an enforcer variable such that the 25 clauses associated with $E(u)$ are clauses of Φ_3 but not of Φ_2 . By construction of $E(u)$ and Observation 3.1, we have $\beta_3(u) = T$. Now let $w_i \in X_{p+1}^n \cup Y_1^p$. As β_3 satisfies Φ_3 and each enforcer variable that is contained in V_3 is assigned to T under β_3 , it follows from the clauses introduced in Step 3 that

$$\beta_3(w_{i,1}) = \beta_3(w_{i,2}) = \cdots = \beta_3(w_{i,a(w_i)}).$$

Let β_2 be the truth assignment for Φ_2 that is obtained from β_3 as follows:

- (i) $\beta_2(z_i) = \beta_3(z_i)$ for each $i \in \{1, 2, \dots, p\}$,
- (ii) $\beta_2(x_i) = \beta_3(x_{i,1})$ for each $i \in \{p+1, p+2, \dots, n\}$, and
- (iii) $\beta_2(y_i) = \beta_3(y_{i,1})$ for each $i \in \{1, 2, \dots, p\}$.

As β_3 satisfies Φ_3 , it immediately follows that β_2 satisfies Φ_2 . We complete the proof by showing that β_2 nae-satisfies Φ_2 . Assume that there exists a clause C in Φ_2 whose literals all evaluate to T under β_2 . Let D be the clause obtained from C by applying Step 1. If C contains exactly three literals, then all three literals of \overline{D} evaluate to F ; thereby contradicting that β_3 satisfies Φ_3 . On the other hand, if C contains exactly two literals, then D is replaced with a 3-clause, say D' , and an enforcer, say $E(u')$, in Step 4 and, similarly, \overline{D} is replaced with a 3-clause, say D'' , and an enforcer, say $E(u'')$, in Step 4. Note that D'' is not the complement of D' . Furthermore, again by Observation 3.1, we have $\beta_3(u') = \beta_3(u'') = T$. Now, as each literal of C evaluates to T , each literal of D'' evaluates to F under β_3 ; a contradiction. Hence β_2 nae-satisfies Φ_2 , and so Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT. \square

We next obtain a quantified Boolean formula Φ_4 from Φ_3 such that the number of universal variables in Φ_4 is equal to the number of existential variables in Φ_4 . Let p_e be the number of existential variables in V_3 , and let p_u be the number of universal variables in V_3 . By construction, observe that $p_u = p \geq 0$. Since a new existential variable y_i has been introduced for each universal variable x_i in V_1 with $i \in \{1, 2, \dots, p\}$, we have $p_e \geq p_u$. Let Q_k^1 be the enforcer with variables $\{a_k, b_k, c_k, d_k, q_k, r_k, u_k, v_k, w_k\}$ as introduced in Section 3.1. Obtain Φ_4 from Φ_3 by adding Q_k^1 to the boolean formula, appending $\exists a_k \exists b_k \exists c_k \exists d_k$ to the list of existential variables, and appending $\forall q_k \forall r_k \forall u_k \forall v_k \forall w_k$ to the list of universal variables for each $k \in \{1, 2, \dots, p_e - p_u\}$. It now follows that Φ_4 contains $p_e + 4(p_e - p_u) = 5p_e - 4p_u$ existential variables and $p_u + 5(p_e - p_u) = 5p_e - 4p_u$ universal variables. Moreover, by

Lemma 3.1, we have that Φ_3 is a yes-instance of $\forall\exists$ 3-SAT if and only if Φ_4 is a yes-instance of $\forall\exists$ 3-SAT.

We complete the proof by showing that Φ_4 is an instance of BALANCED $\forall\exists$ 3-SAT-(2, 2, 2, 2). Let V_4 be the set of variables of Φ_4 . By the transformation of Φ_1 into Φ_3 and the construction of Q_k^1 , it is easily checked that each universal variable in V_4 appears exactly twice unnegated and exactly twice negated in Φ_4 . Now, consider the following three sets of existential variables:

- (I) $S_1 = \bigcup_{k=1}^{pe-pu} \{a_k, b_k, c_k, d_k\}$,
- (II) $S_2 = \bigcup_{i=p+1}^n \{x_{i,1}, x_{i,2}, \dots, x_{i,a(x_i)}\}$ and
- (III) $S_3 = \bigcup_{i=1}^p \{y_{i,1}, y_{i,2}, \dots, y_{i,a(y_i)}\}$.

It follows again from the construction of Q_k^1 that each variable in S_1 appears exactly twice unnegated and exactly twice negated in Φ_4 . Furthermore, by Steps 1–3 in the construction of Φ_3 , it follows that each variable in $S_2 \cup S_3$ appears exactly twice unnegated and exactly twice negated in Φ_4 . Lastly, each existential variable in $V_4 - (S_1 \cup S_2 \cup S_3)$ has been introduced by replacing a 2-clause $(\ell_1 \vee \ell_2)$ with $(\ell_1 \vee \ell_2 \vee \bar{u}) \wedge E(u)$ in Step 4 of the construction of Φ_3 . Recall that u appears unnegated exactly twice and negated exactly once in $E(u)$, and that each of the 18 remaining variables introduced by $E(u)$ appears exactly twice unnegated and exactly twice negated in $E(u)$. It now follows that Φ_4 is an instance of BALANCED $\forall\exists$ 3-SAT-(2, 2, 2, 2). We complete the proof of this theorem by noting that each clause of Φ_4 is a 3-clause that contains three distinct variables and that the size of Φ_4 is polynomial in the size of Φ_1 . \square

In Theorem 3.1, we have imposed the same bound on existential and universal variables, i.e. $s_1 = s_2 = t_1 = t_2$. By allowing separate bounds, i.e. $s_1 = s_2$ and $t_1 = t_2$, we obtain the following stronger result.

Theorem 3.2. BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2) is Π_2^P -complete.

Proof. Clearly, BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2) is in Π_2^P . We establish Π_2^P -hardness via a reduction from BALANCED $\forall\exists$ 3-SAT-(2, 2, 2, 2). Let

$$\Phi_1 = \forall X_1^p \exists X_{p+1}^{2p} \varphi.$$

be an instance of BALANCED $\forall\exists$ 3-SAT-(2, 2, 2, 2). Let m be the number of 3-clauses of φ . As $3m = 2p \cdot 4$, observe that p is divisible by 3. Following a similar strategy as in the proof of Theorem 3.1, we apply the following 4-step process to transform Φ_1 into an instance Φ_4 of BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2).

1. Obtain

$$\Phi_2 = \forall C_1^p \exists X_{p+1}^{2p} \exists Y_1^p \exists Z_1^{6p} \varphi[x_1 \mapsto y_1, \dots, x_p \mapsto y_p] \wedge \bigwedge_{i=1}^p (\mathcal{S}_u(\overline{c}_i, y_i, y_i) \wedge \mathcal{S}_u(c_i, \overline{y}_i, \overline{y}_i)),$$

by turning each universal variable in $x_i \in X_1^p$ into an existential variable y_i , adding new universal variables c_1, c_2, \dots, c_p , and adding new existential variables z_1, z_2, \dots, z_{6p} that are introduced as new variables by copies of the \mathcal{S} -enforcer. By construction, each $y_i \in Y_1^p$ appears exactly four times unnegated and exactly four times negated in Φ_2 .

2. For each $y_i \in Y_1^p$ and $k \in \{1, 2, 3, 4\}$, replace the k -th negated appearance of y_i with $\overline{y_{i,k}}$ and replace the k -th unnegated appearance of y_i with $y_{i,k}$. Then replace $\exists Y_1^p$ in Φ_2 with the following list of existential variables

$$\exists y_{1,1} \exists y_{1,2} \exists y_{1,3} \exists y_{1,4} \cdots \exists y_{p,1} \exists y_{p,2} \exists y_{p,3} \exists y_{p,4}.$$

3. Add the following clauses to the Boolean formula resulting from Step 2:

$$\bigwedge_{i=1}^p \left[(\overline{y_{i,1}} \vee y_{i,2} \vee \overline{d_{i,1}}) \wedge (\overline{y_{i,2}} \vee y_{i,3} \vee \overline{d_{i,1}}) \wedge d_{i,1}^{(2)} \wedge (\overline{y_{i,3}} \vee y_{i,4} \vee \overline{d_{i,2}}) \wedge (\overline{y_{i,4}} \vee y_{i,1} \vee \overline{d_{i,2}}) \wedge d_{i,2}^{(2)} \right],$$

where $d_{i,1}$ and $d_{i,2}$ are new existential variables with $i \in \{1, 2, \dots, p\}$, and $d_{i,1}^{(2)}$ and $d_{i,2}^{(2)}$ are the corresponding enforcers as introduced in Section 2. Then append

$$\exists d_{1,1} \exists d_{1,2} \exists d_{2,1} \exists d_{2,2} \cdots \exists d_{p,1} \exists d_{p,2} \exists E_1^{14p}$$

to the list of existential variables, where E_1^{14p} is the set of new variables introduced by these enforcers (each of $d_{i,1}^{(2)}$ and $d_{i,2}^{(2)}$ introduces seven such variables). Let Φ_3 denote the resulting quantified Boolean formula.

4. Note that each universal variable of Φ_3 appears exactly once unnegated and exactly once negated, and that each existential variable of Φ_3 appears exactly twice unnegated and exactly twice negated. Let p_e (resp. p_u) be the number of existential (resp. universal) variables in Φ_3 . Then

$$p_e = p + 4p + 6p + 2p + 14p = 27p \quad \text{and} \quad p_u = p.$$

Evidently, $p_e \geq p_u$. Furthermore, as p is divisible by 3, it follows that p_e and p_u are both divisible by 3. Let $\Delta = (p_e - p_u)/3$. Now, for each

$k \in \{1, 2, \dots, \Delta\}$, add the enforcer Q_k^3 as introduced in Section 3.1 to Φ_3 , append $\exists a_k \exists b_k$ to the list of existential variables, and append $\forall q_k \forall r_k \forall u_k \forall v_k \forall w_k$ to the list of universal variables.

Let Φ_4 denote the formula resulting from the preceding 4-step process. By construction, each clause in Φ_4 is a 3-clause that contains three distinct variables. Moreover, since, for each k , the enforcer Q_k^3 increases the number of universal variables by five and the number of existential variables by two, it follows that Φ_4 is an instance of BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2).

Noting that the size of Φ_4 is polynomial in the size of Φ_1 , we complete the proof by establishing the following statement.

3.2.1. Φ_1 is a yes-instance of BALANCED $\forall\exists$ 3-SAT-(2, 2, 2, 2) if and only if Φ_4 is a yes-instance of BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2).

Proof. Let V_1 be the set of variables of Φ_1 , and let V_4 be the set of variables of Φ_4 . First, suppose that Φ_1 is a yes-instance of BALANCED $\forall\exists$ 3-SAT-(2, 2, 2, 2). Let β_1 be a truth assignment that satisfies Φ_1 . We obtain a truth assignment β_4 for a subset of V_4 , say V_4' , from β_1 as follows:

- (i) for each $i \in \{1, 2, \dots, p\}$, set $\beta_4(c_i) = \beta_1(x_i)$;
- (ii) for each $i \in \{p+1, p+2, \dots, 2p\}$, set $\beta_4(x_i) = \beta_1(x_i)$;
- (iii) for each $i \in \{1, 2, \dots, p\}$ and $k \in \{1, 2, 3, 4\}$, set $\beta_4(y_{i,k}) = \beta_4(c_i)$;
- (iv) for each $i \in \{1, 2, \dots, p\}$ and $k \in \{1, 2\}$, set $\beta_4(d_{i,k}) = T$.

It is straightforward to check that each clause in Φ_4 that does not contain a variable in

$$(A_1^\Delta \cup B_1^\Delta \cup E_1^{14p} \cup Q_1^\Delta \cup R_1^\Delta \cup U_1^\Delta \cup V_1^\Delta \cup W_1^\Delta \cup Z_1^{6p})$$

is satisfied by β_4 . We next extend β_4 in three steps. First, by (iv) and Observation 2.2, it follows that β_4 extends to $V_4' \cup E_1^{14p}$ such that, for each $i \in \{1, 2, \dots, p\}$, the clauses of $d_{i,1}^{(2)}$ and $d_{i,2}^{(2)}$ are satisfied. Second, by Lemma 3.2, β_4 also extends to

$$V_4' \cup A_1^\Delta \cup B_1^\Delta \cup Q_1^\Delta \cup R_1^\Delta \cup U_1^\Delta \cup V_1^\Delta \cup W_1^\Delta$$

such that each clause in $Q_1^3 \wedge Q_2^3 \wedge \dots \wedge Q_\Delta^3$ is satisfied. Third, by (i), (iii), and Observation 2.1 together with its subsequent remark, it follows that β_4 extends to $V_4' \cup Z_1^{6p}$ such that the clauses in

$$\bigwedge_{i=1}^p (\mathcal{S}_u(\bar{c}_i, y_i, y_i) \wedge \mathcal{S}_u(c_i, \bar{y}_i, \bar{y}_i))$$

are satisfied. We deduce that Φ_4 is satisfiable.

Second, suppose that Φ_4 is a yes-instance of BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2). Let β_4 be a truth assignment that satisfies Φ_4 . It follows from Observation 2.2, that $\beta_4(d_{i,1}) = \beta_4(d_{i,2}) = T$ for each $i \in \{1, 2, \dots, p\}$. Hence, the clauses introduced in Step 3 imply that

$$\beta_4(y_{i,1}) = \beta_4(y_{i,2}) = \beta_4(y_{i,3}) = \beta_4(y_{i,4}).$$

It is now easy to check that the truth assignment β_1 for V_1 obtained from β_4 by setting

- (i) $\beta_1(x_i) = \beta_4(y_{i,1})$ for each $i \in \{1, 2, \dots, p\}$ and
- (ii) $\beta_1(x_i) = \beta_4(x_i)$ for each $i \in \{p+1, p+2, \dots, 2p\}$

satisfies Φ_1 . Thus, statement 3.2.1 holds. □

This completes the proof of Theorem 3.2. □

We end this section by remarking that Haviv et al. [8, p. 55] showed that $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) is in NP if $s_1 + s_2 \leq 1$ and in co-NP if $t_1 + t_2 \leq 2$. The latter result implies that BALANCED $\forall\exists$ 3-SAT-(1, 1, 1, 1) is in co-NP. Hence, unless the polynomial hierarchy collapses, the balanced bounds on the number of appearances of universal and existential variables established in Theorems 3.1 and 3.2 are the best possible ones (i.e., for smaller values, the problems can be placed on a lower level of the polynomial hierarchy).

3.3 Hardness of $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2)

Following on from the results by Haviv et al. [8, p. 55] mentioned in the last paragraph, $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) with $s_1 + s_2 \leq 1$ or $t_1 + t_2 \leq 2$ is not Π_2^P -hard unless the polynomial hierarchy collapses. In this section, we show which instances of $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) are NP-complete and which are Π_2^P -complete. Specifically, we show that $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) is NP-complete for when $s_1 + s_2 = 1$ and $(t_1, t_2) \in \{(1, 2), (2, 1)\}$, and Π_2^P -complete for when $s_1 = s_2 = 1$ and $(t_1, t_2) \in \{(1, 2), (2, 1)\}$.

Let Φ be an instance of $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) with $s_1 + s_2 = 1$, and let Y_1^p be the set of universal variables of Φ . As noted by Haviv et al. [8, p. 55], we can obtain an equivalent unquantified Boolean formula from Φ by deleting all literals in $\{y_i, \bar{y}_i : i \in \{1, 2, \dots, p\}\}$ in the clauses of Φ . Hence, if Φ has the additional property that $t_1 + t_2 \leq 2$, it follows from

results by Tovey [17, Section 3] that it can be decided in polynomial time whether or not Φ is a yes-instance. Hence, $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) with $s_1 + s_2 = 1$ and $t_1 + t_2 \leq 2$ is polynomial-time solvable. The next theorem shows that $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) with $s_1 + s_2 = 1$ becomes NP-complete if $(t_1, t_2) \in \{(1, 2), (2, 1)\}$. To establish this result, we use a variant of 3-SAT in which each clause is either a 2-clause or a 3-clause, and each variable appears exactly twice unnegated and exactly once negated, or exactly once unnegated and exactly twice negated. We refer to this variant as 3-SAT-(3). It was shown by Dahlhaus et al. [4, p. 877f] that 3-SAT-(3) is NP-complete. To establish the next theorem, we impose the following two restrictions on an instance φ of 3-SAT-(3).

- (R1) Each 2-clause (resp. 3-clause) contains 2 (resp. 3) distinct variables.
- (R2) Amongst the clauses, each variable appears exactly twice unnegated and exactly once negated.

Indeed, it follows immediately from Dahlhaus et al.'s [4, p. 877f] construction that φ satisfies (R1). Moreover, standard pre-processing that replaces each literal of a variable that appears exactly once unnegated and exactly twice negated with its negation can be used to obtain an instance φ' from φ that satisfies (R2) and that is equivalent to φ . We hence obtain the following theorem.

Theorem 3.3. $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) is NP-complete if $s_1 + s_2 = 1$ and $(t_1, t_2) \in \{(1, 2), (2, 1)\}$.

Proof. It was shown by Haviv et al. [8, p. 55] that $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) with $s_1 + s_2 = 1$ is in NP. We first establish NP-completeness for $\forall\exists$ 3-SAT- $(1, 0, 2, 1)$ via a reduction from 3-SAT-(3).

Let

$$\varphi = \bigwedge_{j=1}^p C_j^2 \wedge \bigwedge_{j=p+1}^m C_j^3$$

be an instance of 3-SAT-(3) over a set X_1^n of variables and such that each clause C_j^k is a k -clause with $k \in \{2, 3\}$. As described prior to the statement of Theorem 3.3, we may assume that φ satisfies (R1) and (R2). Construct the following quantified Boolean formula Φ from φ :

$$\Phi = \forall Y_1^p \exists X_1^n \left(\bigwedge_{j=1}^p (C_j^2 \vee y_j) \wedge \bigwedge_{j=p+1}^m C_j^3 \right).$$

Since φ satisfies (R2), Φ is an instance of $\forall\exists$ 3-SAT- $(1, 0, 2, 1)$. First, suppose that φ is satisfiable. Then there is a truth assignment β that satisfies

each clause in φ . In particular, β satisfies each clause C_j^2 and, hence, any extension of β to Y_1^p with $i \in \{1, 2, \dots, p\}$ is a truth assignment that satisfies Φ . Second, suppose that Φ is satisfiable. Let β' be a truth assignment for Φ such that $\beta'(y_i) = F$ for each $i \in \{1, 2, \dots, p\}$. By the existence of β' , it follows that $\beta(x_i) = \beta'(x_i)$ for each $i \in \{1, 2, \dots, n\}$ is a truth assignment that satisfies each clause in φ . As the size of Φ is polynomial in the size of φ , NP-completeness of $\forall\exists$ 3-SAT-(1, 0, 2, 1) now follows. To see that $\forall\exists$ 3-SAT- (s_1, s_2, t_1, t_2) is also NP-complete for when

- (I) $(s_1, s_2, t_1, t_2) = (0, 1, 2, 1)$,
- (II) $(s_1, s_2, t_1, t_2) = (1, 0, 1, 2)$, or
- (III) $(s_1, s_2, t_1, t_2) = (0, 1, 1, 2)$,

observe that an argument analogous to the one above applies if each literal of a universal variable in Φ is replaced with its negation to establish (I), if each literal of an existential variable in Φ is replaced with its negation to establish (II), and if each literal in Φ is replaced with its negation to establish (III). \square

Theorem 3.4. $\forall\exists$ 3-SAT-(1, 1, t_1, t_2) with $(t_1, t_2) \in \{(1, 2), (2, 1)\}$ is Π_2^P -complete.

Proof. We first establish the theorem for $(t_1, t_2) = (2, 1)$. Throughout the proof, we make use of the following quantified enforcer for an existential variable $d_{i,k}$:

$$E_{\forall}(d_{i,k}) = (d_{i,k} \vee u_{i,k} \vee v_{i,k}) \wedge (d_{i,k} \vee \overline{u_{i,k}} \vee \overline{v_{i,k}}),$$

where $u_{i,k}$ and $v_{i,k}$ are new universal variables for some $i, k \in \mathbb{Z}^+$. The following property of $E_{\forall}(d_{i,k})$ is easy to verify.

- (P) The Boolean formula $\forall u_{i,k} \forall v_{i,k} \exists d_{i,k} E_{\forall}(d_{i,k})$ is a yes-instance of $\forall\exists$ 3-SAT. In particular, if a truth assignment β for $\{d_{i,k}, u_{i,k}, v_{i,k}\}$ has the property that $\beta(d_{i,k}) = T$, then β satisfies $E_{\forall}(d_{i,k})$. Furthermore, if β satisfies $E_{\forall}(d_{i,k})$ and $\beta(u_{i,k}) = \beta(v_{i,k})$, then this implies that $\beta(d_{i,k}) = T$.

As $\forall\exists$ 3-SAT-(1, 1, 2, 1) is a special case of $\forall\exists$ 3-SAT, it follows that the former problem is in Π_2^P . We show Π_2^P -hardness by a reduction from BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2), for which Π_2^P -completeness was established in Theorem 3.2. Let

$$\Phi_1 = \forall X_1^p \exists Y_{p+1}^{2p} \varphi$$

be an instance of BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2). The reduction has two steps:

1. For each existential variable y_i of Φ_1 with $i \in \{p+1, p+2, \dots, 2p\}$, replace the first (resp. second) unnegated appearance of y_i with $y_{i,1}$ (resp. $y_{i,2}$), replace the first (resp. second) negated appearance of y_i with the negation of $y_{i,3}$ (resp. $y_{i,4}$), and add the new clauses

$$\begin{aligned} & (\overline{y_{i,1}} \vee y_{i,2} \vee \overline{d_{i,1}}) \wedge E_{\forall}(d_{i,1}) \wedge (\overline{y_{i,2}} \vee y_{i,3} \vee \overline{d_{i,2}}) \wedge E_{\forall}(d_{i,2}) \wedge \\ & (\overline{y_{i,3}} \vee y_{i,4} \vee \overline{d_{i,3}}) \wedge E_{\forall}(d_{i,3}) \wedge (\overline{y_{i,4}} \vee y_{i,1} \vee \overline{d_{i,4}}) \wedge E_{\forall}(d_{i,4}), \end{aligned}$$

to Φ_1 , where each $d_{i,k}$ with $k \in \{1, 2, 3, 4\}$ is a new existential variable. For all $i \in \{p+1, p+2, \dots, 2p\}$ and $k \in \{1, 2, 3, 4\}$, append $y_{i,k}$ and $d_{i,k}$ to the list of existential variables and append $u_{i,k}$ and $v_{i,k}$ to the list of universal variables.

2. For each existential variable $y_{i,k}$ with $k \in \{3, 4\}$, replace each literal $y_{i,k}$ with $\overline{y_{i,k}}$ and each literal $\overline{y_{i,k}}$ with $y_{i,k}$.

Let Φ_2 be the resulting quantified Boolean formula, and let V_2 be the set of variables of Φ_2 . Note that each existential variable $y_{i,k}$ with $k \in \{3, 4\}$ appears exactly once unnegated and exactly twice negated in the Boolean formula resulting from Step 1. Hence, due to Step 2, it follows that Φ_2 is an instance of $\forall\exists$ 3-SAT-(1, 1, 2, 1). Furthermore, for each $i \in \{p+1, p+2, \dots, 2p\}$, the clauses introduced in Step 1 are replaced with the following clauses in the course of Step 2:

$$\begin{aligned} & (\overline{y_{i,1}} \vee y_{i,2} \vee \overline{d_{i,1}}) \wedge E_{\forall}(d_{i,1}) \wedge (\overline{y_{i,2}} \vee \overline{y_{i,3}} \vee \overline{d_{i,2}}) \wedge E_{\forall}(d_{i,2}) \wedge \\ & (y_{i,3} \vee \overline{y_{i,4}} \vee \overline{d_{i,3}}) \wedge E_{\forall}(d_{i,3}) \wedge (y_{i,4} \vee y_{i,1} \vee \overline{d_{i,4}}) \wedge E_{\forall}(d_{i,4}). \end{aligned}$$

We complete the proof for $(t_1, t_2) = (2, 1)$ by establishing the following statement.

3.4.1. Φ_1 is a yes-instance of BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2) if and only if Φ_2 is a yes-instance of $\forall\exists$ 3-SAT-(1, 1, 2, 1).

Proof. First, suppose that Φ_1 is a yes-instance of BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2). Let β_1 be a truth assignment that satisfies Φ_1 . For every truth assignment β'_2 for the universal variables in

$$\{u_{i,k}, v_{i,k} : i \in \{p+1, p+2, \dots, 2p\} \text{ and } k \in \{1, 2, 3, 4\}\},$$

we extend β'_2 to a truth assignment β_2 for V_2 as follows:

- (i) set $\beta_2(x_i) = \beta_1(x_i)$ for each $i \in \{1, 2, \dots, p\}$;
- (ii) set $\beta_2(y_{i,k}) = \beta_1(y_i)$ for each $i \in \{p+1, p+2, \dots, 2p\}$ and $k \in \{1, 2\}$;

- (iii) set $\beta_2(y_{i,k}) = \overline{\beta_1(y_i)}$ for each $i \in \{p+1, p+2, \dots, 2p\}$ and $k \in \{3, 4\}$;
- (iv) set $\beta_2(d_{i,k}) = T$ for each $i \in \{p+1, p+2, \dots, 2p\}$ and $k \in \{1, 2, 3, 4\}$.

Due to (iv) and Property (P), it is now easily checked that Φ_2 is a yes-instance of $\forall\exists$ 3-SAT-(1, 1, 2, 1).

Second, suppose that Φ_2 is a yes-instance of $\forall\exists$ 3-SAT-(1, 1, 2, 1). Let β_2 be a truth assignment that satisfies Φ_2 such that $\beta_2(u_{i,k}) = \beta_2(v_{i,k})$ for each $i \in \{p+1, p+2, \dots, 2p\}$ and $k \in \{1, 2, 3, 4\}$. Since Φ_2 is a yes-instance, this implies that $\beta_2(d_{i,k}) = T$ by Property (P). Moreover, by construction, we have

$$\beta_2(y_{i,1}) = \beta_2(y_{i,2}) \quad \text{and} \quad \overline{\beta_2(y_{i,1})} = \beta_2(y_{i,3}) = \beta_2(y_{i,4})$$

for each $i \in \{p+1, p+2, \dots, 2p\}$. It now follows that β_1 with

- (i) $\beta_1(x_i) = \beta_2(x_i)$ for each $i \in \{1, 2, \dots, p\}$ and
- (ii) $\beta_1(y_i) = \beta_2(y_{i,1})$ for each $i \in \{p+1, p+2, \dots, 2p\}$

is a truth assignment for the set of variables of Φ_1 that satisfies each clause in Φ_1 and, thus, Φ_1 is a yes-instance of BALANCED $\forall\exists$ 3-SAT-(1, 1, 2, 2). \square

Noting that the size of Φ_2 is polynomial in the size of Φ_1 , the theorem now follows for $(t_1, t_2) = (2, 1)$. Moreover, replacing $k \in \{3, 4\}$ with $k \in \{1, 2\}$ in Step 2 of the reduction and, subsequently, applying an argument that is analogous to 3.4.1, establishes the theorem for $(t_1, t_2) = (1, 2)$. \square

4 Hardness of MONOTONE $\forall\exists$ NAE-3-SAT-(s, t) with bounded variable appearances

4.1 Enforcers

In this section, we describe four monotone enforcers that have recently been introduced in an unquantified context by Darmann and Döcker [5]. For the purposes of this section, we use their enforcers in a quantified setting. Specifically, for the first three enforcers, x can be a universally or existentially quantified variable.

Auxiliary non-equality gadget. First, consider the auxiliary non-equality

gadget

$$\text{NE}_{\text{aux}}(x, y) = (x \vee y \vee a) \wedge (x \vee y \vee b) \wedge (a \vee b \vee u) \wedge \\ (a \vee b \vee v) \wedge (a \vee b \vee w) \wedge (u \vee v \vee w),$$

where a, b, u, v, w are five new existential variables, y is an existential variable, and x is a universal or existential variable. To nae-satisfy the last clause, at least one variable in $\{u, v, w\}$ is set to be T and at least one is set to be F . Then, by the three preceding clauses, we have that a truth assignment that nae-satisfies $\text{NE}_{\text{aux}}(x, y)$ assigns different truth values to a and b . The next observation follows by construction of the first two clauses.

Observation 4.1. *Consider the gadget $\text{NE}_{\text{aux}}(x, y)$, and let V be its associated set of variables. A truth assignment β for $\{x, y\}$ can be extended to a truth assignment β' for V that nae-satisfies $\text{NE}_{\text{aux}}(x, y)$ if and only if $\beta(x) \neq \beta(y)$.*

Equality gadget. The second enforcer is the equality gadget

$$\text{EQ}(x, y) = \text{NE}_{\text{aux}}(p, q) \wedge \text{NE}_{\text{aux}}(p, r) \wedge (x \vee q \vee r) \wedge (y \vee q \vee r),$$

where p, q, r are three new existential variables, y is an existential variable, and x is a universal or existential variable. By construction and Observation 4.1, a truth assignment that nae-satisfies $\text{EQ}(x, y)$ assigns the same truth value to q and r . The next observation follows by construction of the last two clauses in the equality gadget.

Observation 4.2. *Consider the gadget $\text{EQ}(x, y)$, and let V be its associated set of variables. A truth assignment β for $\{x, y\}$ can be extended to a truth assignment β' for V that nae-satisfies $\text{EQ}(x, y)$ if and only if $\beta(x) = \beta(y)$.*

Non-equality gadget. Combining the first and second enforcer, we now obtain another non-equality gadget:

$$\text{NE}(x, y) = \text{EQ}(x, p) \wedge \text{EQ}(y, q) \wedge \text{NE}_{\text{aux}}(p, q),$$

where p and q are two new existential variables, y is an existential variable, and x is a universal or existential variable. The next observation follows immediately by construction, and Observations 4.1 and 4.2.

Observation 4.3. *Consider the gadget $\text{NE}(x, y)$, and let V be its associated set of variables. A truth assignment β for $\{x, y\}$ can be extended to a truth assignment β' for V that nae-satisfies $\text{NE}(x, y)$ if and only if $\beta(x) \neq \beta(y)$.*

The next observation follows by construction of the last three enforcers.

Observation 4.4. *Let \mathcal{E} be an enforcer in $\{\text{NE}_{\text{aux}}(x, y), \text{EQ}(x, y), \text{NE}(x, y)\}$. Then each variable introduced by \mathcal{E} appears at most four times in \mathcal{E} .*

Padding gadget. The fourth enforcer is the gadget

$$\begin{aligned} P1(x) = & (x \vee a \vee b) \wedge (a \vee c \vee d) \wedge (a \vee b \vee e) \wedge \\ & (a \vee d \vee e) \wedge (b \vee c \vee d) \wedge (b \vee c \vee e) \wedge (c \vee d \vee e), \end{aligned}$$

where x is an existential variable, and a, b, c, d, e are five new existential variables each of which appears exactly four times in the gadget. For the truth assignment β for $\{a, b, c, d, e\}$ with $\beta(a) = \beta(c) = \beta(e) = T$ and $\beta(b) = \beta(d) = F$, the next observation follows immediately by construction.

Observation 4.5. *The gadget $P1(x)$ is nae-satisfiable. Moreover, every truth assignment for $\{x\}$ can be extended to a truth assignment for $\{a, b, c, d, e, x\}$ that nae-satisfies $P1(x)$.*

Intuitively, $P1(x)$ is used to increase the number of appearances of existential variables in a Boolean formula until each variable appears exactly four times.

4.2 Hardness of MONOTONE $\forall\exists$ NAE-3-SAT- (s, t)

In this section, we establish that a monotone and linear Boolean formula φ of $\forall\exists$ NAE-3-SAT is complete for the second level of the polynomial hierarchy even if each clause in φ contains at most one universal variable and, amongst the clauses in φ , each universal variable appears exactly once and each existential variable appears exactly three times. We start by establishing a slightly weaker result without linearity.

Proposition 4.1. *MONOTONE $\forall\exists$ NAE-3-SAT- $(1, 4)$ is Π_2^P -complete if each clause contains at most one universal variable.*

Proof. Clearly, the decision problem MONOTONE $\forall\exists$ NAE-3-SAT- $(1, 4)$ as described in the statement of the proposition is in Π_2^P . We show that it is Π_2^P -complete by a reduction from $\forall\exists$ NAE-3-SAT. For the latter problem, Π_2^P -completeness was established by Eiter and Gottlob [6]. Let

$$\Phi_1 = \forall X_1^p \exists Y_{p+1}^n \varphi$$

be an instance of $\forall\exists$ NAE-3-SAT over a set $V_1 = X_1^p \cup Y_{p+1}^n$ of variables. We may assume that each clause contains exactly three literals and at most one duplicate literal.

In what follows, we construct two quantified Boolean formulas that include copies of the enforcers introduced in Section 4.1. Each such enforcer adds several new existential variables. For ease of exposition throughout this proof, we use A to denote the set of all new existential variables that are introduced by a copy of an enforcer in

$$\{\text{NE}_{\text{aux}}(x, y), \text{EQ}(x, y), \text{NE}(x, y), \text{P1}(x)\}.$$

In particular, A is initially empty and, each time we use a new enforcer copy, we add the newly introduced variables to A and append them to the list of existential variables without mentioning it explicitly. We remark that it will always be clear from the context that the number of elements in A is polynomial in the size of Φ_1 .

Now, let

$$\Phi_2 = \forall Z_1^p \exists Y_1^n \exists A \varphi[x_1 \mapsto y_1, \dots, x_p \mapsto y_p] \wedge \bigwedge_{i=1}^p \text{EQ}(z_i, y_i)$$

be the quantified Boolean formula obtained from Φ_1 by first creating a copy z_i of each of the universal variables x_i , replacing each universal variable x_i of Φ_1 with a new existential variable y_i , and then, for all $i \in \{1, 2, \dots, p\}$, adding the enforcer $\text{EQ}(z_i, y_i)$, where z_i is a new universal variable. Furthermore, let $V_2' = Z_1^p \cup Y_1^n$, and let $V_2 = V_2' \cup A$. By construction, each clause in Φ_2 contains at most one universal variable and each universal variable appears exactly once in Φ_2 .

4.1.1. Φ_1 is a yes-instance of $\forall \exists$ NAE-3-SAT if and only if Φ_2 is a yes-instance of $\forall \exists$ NAE-3-SAT.

Proof. First, suppose that Φ_1 is a yes-instance of $\forall \exists$ NAE-3-SAT. Let β_1 be a truth assignment for V_1 that nae-satisfies Φ_1 , and let β_2' be the following truth assignment for V_2' :

- (i) set $\beta_2'(y_i) = \beta_1(y_i)$ for each $i \in \{p+1, p+2, \dots, n\}$;
- (ii) set $\beta_2'(z_i) = \beta_1(x_i)$ for each $i \in \{1, 2, \dots, p\}$;
- (iii) set $\beta_2'(y_i) = \beta_1(x_i)$ for each $i \in \{1, 2, \dots, p\}$.

By (iii) and Observation 4.2, it follows that there is a truth assignment β_2 for V_2 that extends β_2' such that, for each $i \in \{1, 2, \dots, p\}$, the clauses of $\text{EQ}(z_i, y_i)$ are nae-satisfied. Furthermore, since Φ_1 is nae-satisfiable for every truth assignment for X_1^p , it follows that Φ_2 is nae-satisfiable for every truth assignment for Z_1^p . Hence, Φ_2 is a yes instance of $\forall \exists$ NAE-3-SAT.

Second, suppose that Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT. Let β_2 be a truth assignment for V_2 that nae-satisfies Φ_2 . By Observation 4.2, it follows that $\beta_2(z_i) = \beta_2(y_i)$ for each $i \in \{1, 2, \dots, p\}$. Hence β_1 with $\beta_1(x_i) = \beta_2(z_i)$ for each $i \in \{1, 2, \dots, p\}$, and $\beta_1(y_i) = \beta_2(y_i)$ for each $i \in \{p+1, p+2, \dots, n\}$ is a truth assignment for V_1 that nae-satisfies Φ_1 . Furthermore, since Φ_2 is nae-satisfiable for every truth assignment for Z_1^p , it follows that Φ_1 is a yes-instance of $\forall\exists$ NAE-3-SAT. \square

Next, following Darmann and Döcker [5, Theorem 1], we transform Φ_2 into a new quantified Boolean formula in four steps:

1. To remove all negated variables, we start by replacing each appearance of an existential variable in Y_1^n with a new unnegated variable. Specifically, for each existential variable $y_i \in Y_1^n$, let $u(y_i)$ and $n(y_i)$ be the number of unnegated and negated appearances, respectively, of y_i in the Boolean formula of Φ_2 . Recall that $u(y_i) + n(y_i) = a(y_i)$. Now, for each $j \in \{1, 2, \dots, u(y_i)\}$, replace the j -th unnegated appearance of y_i in Φ_2 with $y_{i,j}$. Similarly, for each $j \in \{1, 2, \dots, n(y_i)\}$, replace the j -th negated appearance of y_i in Φ_2 with $y_{i,u(y_i)+j}$. Lastly, for all $i \in \{1, 2, \dots, n\}$, append

$$\exists y_{i,1} \exists y_{i,2} \cdots \exists y_{i,a(y_i)}$$

to the list of existential variables and remove the obsolete variables $\exists Y_1^n$.

2. If $u(y_i) > 1$, introduce the clauses

$$\bigwedge_{i=1}^n \bigwedge_{j=1}^{u(y_i)-1} \text{EQ}(y_{i,j}, y_{i,j+1}).$$

Similarly, if $n(y_i) > 1$, introduce the clauses

$$\bigwedge_{i=1}^n \bigwedge_{j=u(y_i)+1}^{a(y_i)-1} \text{EQ}(y_{i,j}, y_{i,j+1}).$$

3. For each $i \in \{1, 2, \dots, n\}$ with $u(y_i) \notin \{0, a(y_i)\}$, add the gadget

$$\text{NE}(y_{i,u(y_i)}, y_{i,u(y_i)+1}).$$

4. Let Φ'_2 be the quantified Boolean formula resulting from the last three steps. For $i \in \{1, 2, \dots, n\}$, consider an existential variable y_i in Φ_2 . If y_i appears exactly once in Φ_2 , then $y_{i,1}$ only appears once in Φ'_2 because Steps 2 and 3 do not introduce any gadget that adds an additional

appearance of $y_{i,1}$. Otherwise, if y_i appears at least twice in Φ_2 , then the enforcers introduced in the previous two steps increase the number of appearances for each variable $y_{i,j}$ with $j \in \{1, 2, \dots, a(y_i)\}$ by at least one and at most two. Hence, each variable $y_{i,j}$ appears at most three times in Φ'_2 . Moreover, by construction and Observation 4.4, each variable in A appears at most four times in Φ'_2 . Now, for each existential variable v in Φ'_2 (this includes all variables in A), add the clauses

$$\bigwedge_{k=1}^{4-a(v)} \text{P1}(v)$$

to Φ'_2 , where $a(v)$ denotes the number of appearances of v in Φ'_2 .

Let Φ_3 be the quantified Boolean formula constructed by the preceding four-step procedure. Furthermore, let V_3 be the set of variables of Φ_3 , and let $V'_3 = V_3 - A$.

4.1.2. Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT if and only if Φ_3 is a yes-instance of $\forall\exists$ NAE-3-SAT.

Proof. First, suppose that Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT. Let β_2 be a truth assignment for V_2 that nae-satisfies Φ_2 . Obtain a truth assignment β'_3 for V'_3 as follows:

- (i) set $\beta'_3(z_i) = \beta_2(z_i)$ for each $i \in \{1, 2, \dots, p\}$;
- (ii) set $\beta'_3(y_{i,j}) = \beta_2(y_i)$ for each $i \in \{1, 2, \dots, n\}$ and $j \in \{1, 2, \dots, u(y_i)\}$;
- (iii) set $\beta'_3(y_{i,j}) = \beta_2(y_i)$ for each $i \in \{1, 2, \dots, n\}$ and $j \in \{u(y_i)+1, u(y_i)+2, \dots, a(y_i)\}$.

By (ii) and (iii) as well as Observations 4.2, 4.3, and 4.5, it follows that there is a truth assignment β_3 for V_3 that extends β'_3 and nae-satisfies Φ_3 . Moreover, it follows by construction that for every truth assignment for Z_1^p , there exists a truth assignment for $V_3 - Z_1^p$ that nae-satisfies Φ_3 . Hence, Φ_3 is a yes-instance of $\forall\exists$ NAE-3-SAT.

Second, suppose that Φ_3 is a yes-instance of $\forall\exists$ NAE-3-SAT. Let β_3 be a truth assignment for V_3 that nae-satisfies Φ_3 . By Steps 2 and 3 of the construction, and by Observations 4.2 and 4.3, we have

- (I) $\beta_3(y_{i,1}) = \beta_3(y_{i,2}) = \dots = \beta_3(y_{i,u(y_i)})$,
- (II) $\beta_3(y_{i,u(y_i)}) \neq \beta_3(y_{i,u(y_i)+1})$, and
- (III) $\beta_3(y_{i,u(y_i)+1}) = \beta_3(y_{i,u(y_i)+2}) = \dots = \beta_3(y_{i,a(y_i)})$

for all $i \in \{1, 2, \dots, n\}$. Now, obtain a truth assignment β_2 for V_2 as follows:

- (i) set $\beta_2(z_i) = \beta_3(z_i)$ for each $i \in \{1, 2, \dots, p\}$;
- (ii) set $\beta_2(y_i) = \beta_3(y_{i,1})$ for each $i \in \{1, 2, \dots, n\}$ with $u(y_i) \geq 1$;
- (iii) set $\beta_2(y_i) = \beta_3(y_{i,1})$ for each $i \in \{1, 2, \dots, n\}$ with $u(y_i) = 0$;
- (iv) set $\beta_2(a) = \beta_3(a)$ for each $a \in A$ with $a \in V_2$.

It is now straightforward to check that β_2 nae-satisfies Φ_2 and, hence, Φ_2 is a yes-instance of $\forall\exists$ NAE-3-SAT. \square

We complete the proof by showing that Φ_3 has the desired properties. First, since all enforcers introduced in Section 4.1 are monotone, it follows from Step 1 in the construction of Φ_3 from Φ_2 that Φ_3 is monotone. Second, again by Step 1 in the construction of Φ_3 from Φ_2 , it follows that each clause in Φ_3 is a 3-clause that contains three distinct variables. Third, turning to the universal variables in Φ_3 and as mentioned in the construction of Φ_2 , each clause in Φ_2 , and hence in Φ_3 , contains at most one universal variable and each universal variable in Φ_2 , and hence in Φ_3 , appears exactly once. Fourth, recalling Step 4 in the construction of Φ_3 from Φ_2 and that each new existential variable of $P1(v)$ appears exactly four times in the seven clauses associated with $P1(v)$, it follows that each existential variable appears exactly four times in Φ_3 . Noting that the size of Φ_3 is polynomial in the size of Φ , this establishes the proposition. \square

We are now in a position to establish the main result of this section.

Theorem 4.1. *MONOTONE $\forall\exists$ NAE-3-SAT-(1, 3) is Π_2^P -complete if the Boolean formula is linear and each clause contains at most one universal variable.*

Proof. Clearly, the decision problem MONOTONE $\forall\exists$ NAE-3-SAT-(1, 3) as described in the statement of the theorem is in Π_2^P . We show Π_2^P -completeness by a reduction from MONOTONE $\forall\exists$ NAE-3-SAT-(1, 4). Let

$$\Phi_1 = \forall Z_1^p \exists Z_{p+1}^n \varphi$$

be an instance of MONOTONE $\forall\exists$ NAE-3-SAT-(1, 4). By Proposition 4.1, we may assume that each clause in Φ_1 contains at most one universal variable.

We start by defining the following four sets of variables which we use to construct an instance Φ_2 of MONOTONE $\forall\exists$ NAE-3-SAT-(1,3). Let

$$U = \{u_{i,k} : i \in \{p+1, p+2, \dots, n\} \text{ and } k \in \{1, 2, \dots, 8\}\} \text{ and}$$

$$V = \{v_{i,k} : i \in \{p+1, p+2, \dots, n\} \text{ and } k \in \{1, 2, \dots, 8\}\}$$

be two sets of universal variables, and let

$$E = \{e_{i,k} : i \in \{p+1, p+2, \dots, n\} \text{ and } k \in \{1, 2, \dots, 8\}\} \text{ and}$$

$$Z = \{z_{i,k} : i \in \{p+1, p+2, \dots, n\} \text{ and } k \in \{1, 2, \dots, 8\}\}$$

be two sets of existential variables. Now, for each $i \in \{p+1, p+2, \dots, n\}$, we replace the j -th appearance of z_i with $z_{i,j}$ for all $j \in \{1, 2, 3, 4\}$ and introduce the clauses

$$\bigwedge_{k=1}^7 ((z_{i,k} \vee e_{i,k} \vee u_{i,k}) \wedge (e_{i,k} \vee z_{i,k+1} \vee v_{i,k})) \wedge$$

$$(z_{i,8} \vee e_{i,8} \vee u_{i,8}) \wedge (e_{i,8} \vee z_{i,1} \vee v_{i,8}) \wedge$$

$$(z_{i,5} \vee e_{i,1} \vee e_{i,2}) \wedge (z_{i,6} \vee e_{i,7} \vee e_{i,8}) \wedge (z_{i,7} \vee e_{i,3} \vee e_{i,4}) \wedge (z_{i,8} \vee e_{i,5} \vee e_{i,6}).$$

Furthermore, we append each element in $U \cup V$ to the list of universal variables, append each element in $E \cup Z$ to the list of existential variables, and delete the obsolete variables Z_{p+1}^n . Let Φ_2 denote the resulting formula. By construction, it is straightforward to check that Φ_2 is an instance of MONOTONE $\forall\exists$ NAE-3-SAT-(1,3) with at most one universal variable per clause and whose set of variables is

$$U \cup V \cup Z_1^p \cup E \cup Z.$$

Moreover, if any pair of clauses in Φ_1 have two variables in common, then both are existential variables and, hence, again by construction, Φ_2 is linear. Since Φ_2 has all desired properties and the size of Φ_2 is polynomial in the size of Φ_1 , it remains to show that the following statement holds.

4.1.1. Φ_1 is a yes-instance of MONOTONE $\forall\exists$ NAE-3-SAT-(1,4) if and only if Φ_2 is a yes-instance of MONOTONE $\forall\exists$ NAE-3-SAT-(1,3).

Proof. First, suppose that Φ_1 is a yes-instance of MONOTONE $\forall\exists$ NAE-3-SAT-(1,4). Let β_1 be a truth assignment for $Z_1^p \cup Z_{p+1}^n$ that nae-satisfies Φ_1 . Obtain a truth assignment β_2 for $Z_1^p \cup E \cup Z$ as follows:

- (i) set $\beta_2(z_i) = \beta_1(z_i)$ for each $i \in \{1, 2, \dots, p\}$;
- (ii) set $\beta_2(z_{i,k}) = \beta_1(z_i)$ for each $i \in \{p+1, p+2, \dots, n\}$ and $k \in \{1, 2, \dots, 8\}$;

- (iii) set $\beta_2(e_{i,k}) = \overline{\beta_1(z_i)}$ for each $i \in \{p+1, p+2, \dots, n\}$ and $k \in \{1, 2, \dots, 8\}$.

It is easily checked that every truth assignment for $U \cup V \cup Z_1^p \cup E \cup Z$ that extends β_2 nae-satisfies Φ_2 , and thus Φ_2 is a yes-instance of MONOTONE $\forall\exists$ NAE-3-SAT-(1, 3).

Second, suppose that Φ_2 is a yes-instance MONOTONE $\forall\exists$ NAE-3-SAT-(1, 3). Let β_2 be a truth assignment that nae-satisfies Φ_2 such that $\beta_2(u_{i,k}) = F$ and $\beta_2(v_{i,k}) = T$ for each $i \in \{p+1, p+2, \dots, n\}$ and $k \in \{1, 2, \dots, 8\}$. Since $u_{i,k}$ and $v_{i,k}$ are universal variables, β_2 exists. We next show that β_2 satisfies the property

$$\beta_2(z_{i,1}) = \beta_2(z_{i,2}) = \beta_2(z_{i,3}) = \beta_2(z_{i,4})$$

for each $i \in \{p+1, p+2, \dots, n\}$. To this end, consider the subset of clauses

$$(z_{i,8} \vee e_{i,8} \vee F) \wedge (e_{i,8} \vee z_{i,1} \vee T) \wedge \bigwedge_{k=1}^7 ((z_{i,k} \vee e_{i,k} \vee F) \wedge (e_{i,k} \vee z_{i,k+1} \vee T))$$

of Φ_2 , where the universal variables are set according to β_2 . If $\beta_2(z_{i,1}) = F$, then the clause $(z_{i,1} \vee e_{i,1} \vee F)$ implies that $\beta_2(e_{i,1}) = T$ and, hence, by the aforementioned subset of clauses, $\beta_2(z_{i,j}) = F$ for each $j \in \{1, 2, 3, 4\}$. Otherwise, if $\beta_2(z_{i,1}) = T$, then the clause $(e_{i,8} \vee z_{i,1} \vee T)$ implies that $\beta_2(e_{i,8}) = F$ and, hence, again by the aforementioned subset of clauses, $\beta_2(z_{i,j}) = T$ for each $j \in \{1, 2, 3, 4\}$. It now follows that the truth assignment β_1 for $Z_1^p \cup Z_{p+1}^n$ with

- (i) $\beta_1(z_i) = \beta_2(z_i)$ for each $i \in \{1, 2, \dots, p\}$ and
- (ii) $\beta_1(z_i) = \beta_2(z_{i,1})$ for each $i \in \{p+1, p+2, \dots, n\}$

nae-satisfies Φ_1 , and so Φ_1 is a yes-instance of MONOTONE $\forall\exists$ NAE-3-SAT-(1, 4). \square

This completes the proof of Theorem 4.1. \square

4.3 Restrictions that alleviate the complexity of MONOTONE $\forall\exists$ NAE-3-SAT-(s, t)

In this section, we discuss variants of MONOTONE $\forall\exists$ NAE-3-SAT-(s, t) that are in co-NP or solvable in polynomial time. More precisely, we investigate the complexity of MONOTONE $\forall\exists$ NAE-3-SAT-($s, 2$). First note

that MONOTONE $\forall\exists$ NAE-3-SAT-(0, t) is a special case of NAE-3-SAT and therefore in NP. Furthermore, MONOTONE $\forall\exists$ NAE-3-SAT-($s, 1$) can be solved in polynomial time since an instance of this problem is a yes-instance if and only if each clause contains at least one existential variable. Now consider the following decision problem that allows for a set of variables and the set $\{F, T\}$ of constants.

MONOTONE-WITH-CONSTANTS-NAE-3-SAT- t (MC-NAE-3-SAT- t)

Input. A positive integer t , a set $V = \{x_1, x_2, \dots, x_n\}$ of variables and a monotone Boolean formula

$$\bigwedge_{j=1}^m C_j$$

such that each clause contains exactly three distinct elements in $V \cup \{F, T\}$ and, amongst the clauses, each element in V appears exactly t times.

Question. Does there exist a truth assignment $\beta: V \rightarrow \{T, F\}$ such that each clause of the formula is nae-satisfied?

Boolean formulas that include variables and the two constants F and T were, for example, previously considered in the context of NAE-3-SAT [2, p. 275f]. In particular, let φ be an instance of NAE-3-SAT that allows for constants. Bonet et al. [2] showed that, given a solution to φ , it is NP-complete to decide if a second solution to φ exists. This result was in turn used to prove that two problems arising in computational biology are NP-complete. Note that in the special case in which φ does not contain any constant, a second solution can always be obtained from a given truth assignment that nae-satisfies φ by simply interchanging T and F .

We next show that MC-NAE-3-SAT- t is solvable in polynomial time if $t = 2$.

Proposition 4.2. MC-NAE-3-SAT-2 is in P.

Proof. Let $\varphi = \bigwedge_{j=1}^m C_j$ be an instance of MC-NAE-3-SAT-2 over a set $V \cup \{F, T\}$ of variables and constants, where $V = \{x_1, x_2, \dots, x_n\}$. To establish the proposition, we adapt ideas presented by Porschen et al. [11] who developed a linear-time algorithm to decide if an instance of NAE-SAT, i.e. a Boolean formula in CNF, is nae-satisfiable if each variable appears at most twice.

Using the notation $C_j = (\ell_{j,1} \vee \ell_{j,2} \vee \ell_{j,3})$ to denote the j -th clause in φ for each $j \in \{1, 2, \dots, m\}$, we next present an algorithm to decide whether or not φ is a yes-instance of MC-NAE-3-SAT-2. At each step of the algorithm, φ is transformed into a simpler Boolean formula.

1. For each clause C_j with $\ell_{j,k} = x_i$ for some $i \in \{1, 2, \dots, n\}$ and $\ell_{j,k'}, \ell_{j,k''} \in \{F, T\}$ with $\{k, k', k''\} = \{1, 2, 3\}$, do the following.
 - (I) If $\ell_{j,k'} \neq \ell_{j,k''}$, remove C_j from φ .
 - (II) If $\ell_{j,k'} = \ell_{j,k''} = F$, remove C_j and reset φ to be $\varphi[x_i \mapsto T]$.
 - (III) If $\ell_{j,k'} = \ell_{j,k''} = T$, remove C_j and reset φ to be $\varphi[x_i \mapsto F]$.
2. For each pair of variables $x_i, x_{i'} \in V$ with $i \neq i'$ that both appear in two distinct clauses C_j and $C_{j'}$, remove C_j and $C_{j'}$ from φ .
3. For each variable x_i that appears in exactly one clause C_j , remove C_j from φ .
4. For each clause C_j such that $\ell_{j,1}, \ell_{j,2}, \ell_{j,3} \in \{T, F\}$, do the following.
 - (I) If $\{\ell_{j,1}, \ell_{j,2}, \ell_{j,3}\} = \{T, F\}$, remove C_j from φ .
 - (II) Otherwise, stop and return “ φ is a no-instance”.
5. Stop and return “ φ is a yes-instance”.

The algorithm clearly terminates within polynomial time. Moreover, as a variable that appears exactly once in a Boolean formula can be assigned to either T or F without affecting any other clause, it is straightforward to check that each step in the algorithm returns a Boolean formula that is equivalent to φ . Hence, if a clause contains three equal constants, then the algorithm correctly returns that φ is a no-instance in Step 4. Now, suppose that the algorithm returns “ φ is a yes-instance”. Let φ' be the Boolean formula that is obtained at the end of the last iteration of Step 4(I). Then, φ' is an instance of MC-NAE-3-SAT-2 such that each clause contains at most one constant and each pair of clauses have at most one variable in common. Hence, φ' is linear. It remains to show that φ' is a yes-instance of MC-NAE-3-SAT-2.

Before continuing with the proof, we pause to give an overview of a result established by Porschen et al. [11]. Let $\psi = \bigwedge_{j=1}^{m'} C'_j$ be a linear and monotone Boolean formula where each variable appears exactly twice and each clause contains at least two distinct variables. Furthermore, let G_ψ be the *clause graph* for ψ whose set of vertices is $\{C'_1, C'_2, \dots, C'_{m'}\}$ and, for each pair $j, j' \in \{1, 2, \dots, m'\}$ with $j \neq j'$, there is an edge $\{C'_j, C'_{j'}\}$ in G_ψ precisely if C'_j and $C'_{j'}$ have a variable in common. Then ψ is nae-satisfiable if and only if there exists an edge coloring of G_ψ that uses exactly two colors c_1 and c_2 such that each vertex is incident to an edge that is colored c_1 and incident to an edge that is colored c_2 . Moreover, if ψ does not have a connected component that is isomorphic to a cycle of odd length, then such an edge coloring exists.

We now continue with the proof of the proposition. Let φ'' be the Boolean formula obtained from φ' by omitting all constants. By construction, each clause in φ'' contains either two or three distinct variables. It follows that, if φ'' is nae-satisfiable, then φ' is nae-satisfiable. Let $G_{\varphi''}$ be the clause graph for φ'' . First, assume that $G_{\varphi''}$ does not have a connected component that is isomorphic to a cycle of odd length. Then it immediately follows from the result by Porschen et al. [11] that φ'' is nae-satisfiable and, hence, φ' is also nae-satisfiable. Second, assume that $G_{\varphi''}$ has a connected component that is isomorphic to a cycle of odd length. Then the vertices of this component are of the form

$$(x_{i_1} \vee x_{i_2}), (x_{i_2} \vee x_{i_3}), \dots, (x_{i_{p-1}} \vee x_{i_p}), (x_{i_p} \vee x_{i_1}),$$

where $p \geq 3$ is an odd integer and $x_{i_j} \in V$. In other words,

$$\mathcal{C}_{\varphi''} = \bigwedge_{j=1}^{p-1} (x_{i_j} \vee x_{i_{j+1}}) \wedge (x_{i_p} \vee x_{i_1}),$$

is contained in φ'' . Although $\mathcal{C}_{\varphi''}$ is not nae-satisfiable, we next show that the corresponding clauses $\mathcal{C}_{\varphi'}$ in φ' are nae-satisfiable since each such clause contains exactly one constant.

Consider

$$\mathcal{C}_{\varphi'} = \bigwedge_{j=1}^{p-1} (x_{i_j} \vee x_{i_{j+1}} \vee b_j) \wedge (x_{i_p} \vee x_{i_1} \vee b_p),$$

where $b_j \in \{T, F\}$ for each $j \in \{1, 2, \dots, p\}$. Let β be the following truth assignment for $\{x_{i_1}, x_{i_2}, \dots, x_{i_p}\}$:

- (i) set $\beta(x_{i_j}) = \overline{b_p}$ for each $j \in \{1, 2, \dots, p\}$ with j being odd;
- (ii) set $\beta(x_{i_j}) = b_p$ for each $j \in \{1, 2, \dots, p\}$ with j being even.

It follows that β nae-satisfies $\mathcal{C}_{\varphi'}$. An analogous argument can be applied to every other connected component in $G_{\varphi''}$ that is isomorphic to a cycle of odd length. Furthermore, it again follows from Porschen et al.'s result [11] that the edge set of each connected component in $G_{\varphi''}$ that is not isomorphic to a cycle of odd length corresponds to a subset of clauses in φ'' that is nae-satisfiable. Altogether, φ'' is nae-satisfiable and, hence, φ' is also nae-satisfiable. This completes the proof of the proposition. \square

We next establish three corollaries that pinpoint the complexity of MONOTONE $\forall \exists$ NAE-3-SAT- $(s, 2)$.

Corollary 4.1. MONOTONE $\forall\exists$ NAE-3-SAT- $(s, 2)$ is in co-NP for any fixed positive integer s .

Proof. A no-instance of MONOTONE $\forall\exists$ NAE-3-SAT- $(s, 2)$ can be identified by taking an assignment of the universal variables and applying the algorithm presented in Proposition 4.2 to verify in polynomial time whether or not the resulting MC-NAE-3-SAT-2 Boolean formula (with omitted lists of universal and existential quantifiers) is not nae-satisfiable. \square

Corollary 4.2. MONOTONE $\forall\exists$ NAE-3-SAT- $(s, 2)$ is trivially a yes-instance for any fixed positive integer s if each clause contains at most one universal variable.

Proof. Let Φ be an instance of MONOTONE $\forall\exists$ NAE-3-SAT- $(s, 2)$ such that each clause contains at most one universal variable. We follow ideas that are similar to those presented in the algorithm described in the proof of Proposition 4.2. First, if there are two existential variables that both appear in two distinct clauses C_j and $C_{j'}$, obtain a new Boolean formula by removing C_j and $C_{j'}$ from Φ . Repeat this step until no such pair of variables remains. Then, if there is an existential variable that appears in exactly one clause C_j , obtain a new Boolean formula by removing C_j . Similar to the proof of Proposition 4.2, it follows that the resulting Boolean formula, say Φ' , is an instance of MONOTONE $\forall\exists$ NAE-3-SAT- $(s, 2)$ such that each clause contains at most one universal variable and the formula is linear. Moreover, Φ is a yes-instance if and only if Φ' is a yes-instance. If Φ' is empty, then Φ is a yes-instance by correctness of the applied transformations. Otherwise, it follows from the properties of Φ' and the proof of Proposition 4.2 that Φ' and, hence, Φ are yes-instances. \square

Corollary 4.3. MONOTONE $\forall\exists$ NAE-3-SAT- $(1, 2)$ is in P.

Proof. Let Φ be an instance of MONOTONE $\forall\exists$ NAE-3-SAT- $(1, 2)$. To decide whether or not Φ is a yes-instance, we apply the following algorithm to Φ .

1. If there exists a clause that contains three distinct universal variables, then stop and return “ Φ is a no-instance”.
2. For a clause C_j that contains one existential variable, say x , and two distinct universal variables, say u and u' , let $C_{j'}$ be the unique clause that contains the second appearance of x . Then remove C_j and turn x in $C_{j'}$ into a new universal variable.

- (I) If $C_{j'}$ now contains three distinct universal variables, stop and return “ Φ is a no-instance”.
 - (II) Otherwise, repeat until there are no clauses with two universal variables.
3. Stop and return “ Φ is a yes-instance”.

Since each universal variable appears exactly once in Φ and each existential variable appears exactly twice in Φ , it follows that the Boolean formula obtained after each iteration of Step 2 is an instance of MONOTONE $\forall\exists$ NAE-3-SAT-(1, 2). Therefore, if the algorithm eventually produces a Boolean formula, Φ' say, then Φ' has at most one universal variable in each clause and, by Corollary 4.2, Φ' is a yes-instance. Hence, to see that the algorithm works correctly, it suffices to show that an iteration of Step 2 preserves yes-instances. Suppose that Φ_1 is the quantified Boolean formula at the start of an iteration of Step 2 and $C_j = (x \vee u \vee u')$ is a clause in Φ_1 as described in Step 2. Let β be a truth assignment that nae-satisfies Φ . If $\beta(u) = \beta(u') = F$, then it follows that $\beta(x) = T$. On the other hand, if $\beta(u) = \beta(u') = T$, then this implies that $\beta(x) = F$. It now follows that the Boolean formula, Φ_2 say, obtained by turning x into a universal variable is also a yes-instance. Conversely, by reversing this argument, if Φ_2 is a yes-instance, then Φ_1 is a yes-instance. Thus Step 2 preserves yes-instances. We now establish the corollary by noting that the described algorithm has a running time that is polynomial in the size of Φ . \square

Acknowledgements. The third and fourth authors thank the New Zealand Marsden Fund for their financial support.

References

- [1] P. Berman, M. Karpinski, and A. D. Scott (2003). Approximation hardness of short symmetric instances of MAX-3SAT. *Electronic Colloquium on Computational Complexity*, Report No. 49.
- [2] M. L. Bonet, S. Linz, and K. St. John (2012). The complexity of finding multiple solutions to betweenness and quartet compatibility. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 9(1), 273–285.
- [3] S. A. Cook (1971), The complexity of theorem-proving procedures. In Proceedings of the Third Annual ACM Symposium on Theory of Computing, 151–158.

- [4] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis (1994). The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23(4), 864–894.
- [5] A. Darmann and J. Döcker (2019). On simplified NP-complete variants of Not-All-Equal 3-SAT and 3-SAT. arXiv preprint arXiv:1908.04198.
- [6] T. Eiter and G. Gottlob (1995). Note on the complexity of some eigenvector problems. Technical Report CD-TR 95/89, Christian Doppler Laboratory for Expert Systems, TU Vienna.
- [7] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, 1979.
- [8] I. Haviv, O. Regev, and A. Ta-Shma (2007). On the hardness of satisfiability with bounded occurrences in the polynomial-time hierarchy. *Theory of Computing*, 3, 45–60.
- [9] M. Karpinski and K. Piecuch (2017). On vertex coloring without monochromatic triangles. arXiv preprint arXiv:1710.07132.
- [10] M. Karpinski and K. Piecuch (2018). On vertex coloring without monochromatic triangles. In: F. Fomin., V. Podolskii (eds) *Computer Science – Theory and Applications. CSR 2018. Lecture Notes in Computer Science*, vol 10846. Springer, Cham
- [11] S. Porschen, B. Randerath, E. Speckenmeyer (2004). Linear time algorithms for some not-all-equal satisfiability problems. In: Giunchiglia E., Tacchella A. (eds) *Theory and Applications of Satisfiability Testing. SAT 2003. Lecture Notes in Computer Science*, vol 2919. Springer, Berlin, Heidelberg
- [12] S. Porschen, T. Schmidt, E. Speckenmeyer and A. Wotzlaw (2014). XSAT and NAE-SAT of linear CNF classes. *Discrete Applied Mathematics*, 167, 1–14.
- [13] M. Schaefer (2001). Graph Ramsey theory and the polynomial hierarchy. *Journal of Computer and System Sciences*, 62, 290–322.
- [14] M. Schaefer and C. Umans (2002). Completeness in the polynomial-time hierarchy: A compendium. *SIGACT News*, 33(3), 32–49.
- [15] T. J. Schaefer (1978), The complexity of satisfiability problems. In *Proceedings of the Tenth annual ACM Symposium on Theory of Computing*, 216–226.
- [16] L. J. Stockmeyer, The polynomial-time hierarchy. *Theoretical Computer Science*, 3 (1976), pp. 1–22.

- [17] C. A. Tovey (1984). A simplified NP-complete satisfiability problem.
Discrete Applied Mathematics, 8(1), 85–89.