TFNP Intersections Through the Lens of Feasible Disjunction

Pavel Hubáček
Institute of Mathematics, Czech Academy of Sciences, Prague, Czech Republic
Charles University, Faculty of Mathematics and Physics, Prague, Czech Republic

Erfan Khaniki
Institute of Mathematics, Czech Academy of Sciences, Prague, Czech Republic
Charles University, Faculty of Mathematics and Physics, Prague, Czech Republic

Neil Thapen
Institute of Mathematics, Czech Academy of Sciences, Prague, Czech Republic

Abstract

The complexity class CLS was introduced by Daskalakis and Papadimitriou (SODA 2010) to capture the computational complexity of important TFNP problems solvable by local search over continuous domains and, thus, lying in both PLS and PPAD. It was later shown that, e.g., the problem of computing fixed points guaranteed by Banach’s fixed point theorem is CLS-complete by Daskalakis et al. (STOC 2018). Recently, Fearnley et al. (J. ACM 2023) disproved the plausible conjecture of Daskalakis and Papadimitriou that CLS is a proper subclass of PLS ∩ PPAD by proving that CLS = PLS ∩ PPAD.

To study the possibility of other collapses in TFNP, we connect classes formed as the intersection of existing subclasses of TFNP with the phenomenon of feasible disjunction in propositional proof complexity; where a proof system has the feasible disjunction property if, whenever a disjunction F ∨ G has a small proof, and F and G have no variables in common, then either F or G has a small proof. Based on some known and some new results about feasible disjunction, we separate the classes formed by intersecting the classical subclasses PLS, PPA, PPAD, PPADS, PPP and CLS. We also give the first examples of proof systems which have the feasible interpolation property, but not the feasible disjunction property.

2012 ACM Subject Classification Theory of computation → Proof complexity; Theory of computation → Problems, reductions and completeness; Theory of computation → Complexity classes

Keywords and phrases TFNP, feasible disjunction, proof complexity, TFNP intersection classes

Digital Object Identifier 10.4230/LIPIcs.ITCS.2024.63

Funding Pavel Hubáček: Supported by the Czech Academy of Sciences (RVO 67985840). Erfan Khaniki: Supported by the Czech Academy of Sciences (RVO 67985840), GACR grant 19-27871X and SVV-2023-200721. Neil Thapen: Supported by the Czech Academy of Sciences (RVO 67985840) and GAČR grant 23-04825S.

Acknowledgements We would like to thank Lukáš Folwarczný, Tuomas Hakoniemi, Yuhao Li, William Pires, Pavel Pudlák and Robert Robere for helpful discussions related to this work.

1 Introduction

Since the foundational work of Megiddo and Papadimitriou [40], various subclasses of total search problems have been introduced to characterize the computational complexity of problems for which a solution is guaranteed to exist [32, 43, 36, 31, 49, 34, 30, 26, 35]. Such problems are naturally clustered based on the kind of reasoning, or “combinatorial lemma”, used to prove that all instances have a solution; or viewed alternatively, based on the type of an inefficient algorithm that can be used to solve them. A prime example is the class PLS.
of all total search problems solvable via a simple local search algorithm, which can also be
defined as the class of problems for which reasoning by induction over NP formulas is enough
to prove that a solution exists [9].

The complexity class CLS was introduced by Daskalakis and Papadimitriou [14] to capture
the computational complexity of important TFNP problems solvable by local search over
continuous domains and, thus, lying in both PLS and PPAD. It was later shown that,
e.g., the problem of computing fixed points guaranteed by Banach’s fixed point theorem is
CLS-complete by Daskalakis et al. [15]. Recently, Fearnley et al. [18] disproved the plausible
conjecture of Daskalakis and Papadimitriou that CLS is a proper subclass of PLS ∩ PPAD by
establishing that CLS = PLS ∩ PPAD. Their work motivates the natural question whether
there will be other surprising collapses of TFNP intersections; as noted by Fearnley et al. [18],
CLS is currently the only intersection class that has natural complete problems in the sense
that input is explicit and not represented succinctly via a Boolean circuit.

Among the well-studied subclasses of TFNP, a prime candidate for being an intersection
class is PPAD which is contained in the intersection of all the PPAq classes as well as
PPADS. As for other intersections of classical subclasses of TFNP, Göös et al. [21] gave
simple combinatorial complete problems for the intersections of PLS with PPADS and PPP
(respectively). Ishizuka [27] gave a combinatorial characterization of PLS ∩ PPA.

TFNP and proof complexity. Our understanding of the structural properties of subclasses of
TFNP is largely based on its interaction with proof complexity. The seminal paper of Beame
et al. [2] demonstrated that lower bounds in proof complexity can illuminate the relative
complexity of TFNP subclasses, which was subsequently developed further [41, 42, 4]. Recent
work pushed this connection further and characterized various subclasses of TFNP in terms
of corresponding proof systems, thus leading to oracle separations between various subclasses
of TFNP when conflicting bounds were known for the corresponding proof systems [22]. In
this work, we build upon the recent characterizations of subclasses of TFNP. We take a
generic approach to intersections of standard subclasses of TFNP employing the classical
notion of feasible disjunction in proof complexity.

Suppose \( F \) and \( G \) are propositional formulas with no variables in common. Then if the
disjunction \( F \lor G \) is a tautology, it follows that either \( F \) is a tautology or \( G \) is. Thus a
natural property that a proof system may have is that, given a proof \( \pi \) of \( F \lor G \) in the
system, there exists another proof, of a similar length to \( \pi \) or shorter, of either \( F \) or \( G \) by
itself. This property was called weak feasible disjunction by Pudlák [46] (see also [38, 48]); in
this paper we call it simply feasible disjunction\(^1\). It has been shown to hold for proof systems
such as resolution and cutting planes [44], Lovász-Schrijver with the rounding rule [45],
and polynomial calculus and sum-of-squares [24]. Up until now, the only negative result is
recent work by Garlík [19] showing that for \( k \geq 2 \) the system Res(\( k \)), a small extension of
resolution, does not have feasible disjunction. Beyond [19], nothing was known (in classical
logic), even conditional on some hardness assumption. In particular, it is widely open for
strong proof systems such as Frege and extended Frege (more is known for non-classical logic,
see e.g. [10], [29]).

A related property of a proof system is feasible interpolation [37]. This holds if, given a
proof of \( F \lor G \), where \( F \) and \( G \) may share variables, one can construct a small circuit which,
from any assignment to the common variables, computes which of \( F \) and \( G \) is a tautology

\(^1\) “Feasible disjunction”, without the word “weak”, is also used in the literature to mean the stronger
property, that there is a polynomial time algorithm which produces a proof of either \( F \) or \( G \) from a
proof of \( F \lor G \).
under this assignment. This has been intensively studied because one can often guarantee that the circuit is monotone, and thus obtain proof complexity lower bounds from monotone circuit lower bounds. It was observed by Pudlák [46] that systems known to have feasible disjunction also have feasible interpolation, and vice versa; this remains true for the more recent work mentioned above, largely because the proof method for one typically also gives the other.

The proof systems we use below are more naturally used to refute contradictions, rather than to prove tautologies. For this reason when we use feasible disjunction, we actually use the “dual” version of it which is appropriate for such systems, involving conjunction rather than disjunction: if \( F \land G \) has a small refutation, then either \( F \) has a small refutation, or \( G \) does.

1.1 Our Results and Technical Overview

Below, we give a high-level discussion of our results and techniques. Detailed definitions and the necessary background are provided in Section 2 and the formal details of our results in the subsequent Sections 3–5.

1.1.1 Separation of TFNP intersections (Section 4)

Consider the six best-studied TFNP classes PLS, PPA, PPP, PPADS, PPAD, and CLS depicted in Figure 1. We show that, other than the equality \( \text{CLS} = \text{PLS} \cap \text{PPAD} \), no nontrivial intersection of two of these classes equals a third, and no two nontrivial intersections are equal to each other (Theorem 32). This remains true if we include also all classes \( \text{PPA}_q \), for all primes \( q \) (Theorem 33). Our method relies on the well-known connection between natural TFNP classes and proof systems (in particular, as it is set out in [22]).

For a suitable family \( Q_n \) of contradictory CNF formulas, we write \( \text{Search}(Q_n) \) for the search problem of finding a false clause in \( Q_n \), given an assignment. For a natural TFNP class \( A \), there typically exists a proof system \( P \) such that \( \text{Search}(Q_n) \in A \) if and only if \( Q_n \) has small refutations in \( P \), for some appropriate meaning of “small”. Now suppose that \( B, C \not\subseteq A \) are TFNP classes and \( B \cap C \subseteq A \). Take CNF families \( Q \) and \( R \), in disjoint variables, such that \( \text{Search}(Q_n) \in A \) and \( \text{Search}(R_n) \in B \). It is easy to see that \( \text{Search}(Q_n \land R_n) \in B \cap C \), and, thus, \( Q_n \land R_n \) has small refutations in the system \( P \). Therefore, if \( P \) has a suitable form of feasible disjunction, we know that for each \( n \) either \( Q_n \) has a small refutation in \( P \), or \( R_n \) does. Now if we had the stronger conclusion that either \( Q_n \) has small refutations for every \( n \) or \( R_n \) does, for every \( n \), and we had chosen \( Q \) and \( R \) to be “complete” for the classes \( B \) and \( C \) in some appropriate sense, it would follow that either \( B \subseteq A \) or \( C \subseteq A \), contradicting our assumption on \( B \) and \( C \). In fact, whether it is \( Q_n \) or \( R_n \) can in principle
be different for each \( n \), so we use more ad hoc arguments which directly use proof complexity lower bounds for \( Q_n \) and/or \( R_n \) in \( P \) to get a contradiction (where these may be the same lower bounds that were used to show \( B, C \not\subseteq A \) in the first place).

We note that [16] used a similar approach, implicitly relying on the feasible disjunction property, to show (independently of our work) that a certain TFNP intersection is not in PLS. The independent work of Li, Pires, and Robere [39] develops some new characterizations of intersection classes by proof systems.

1.1.2 Feasible disjunction (Section 3)

We define the feasible disjunction property for a general measure \( \mu \) of proof complexity:

▶ Definition 1. Let \( P \) be a proof system and let \( \mu \) be a measure for \( P \). We say that \( P \) has \( \mu \)-feasible disjunction, or feasible disjunction for \( \mu \), if, for CNFs \( F, G \) in disjoint variables,

\[
\min(\mu(F), \mu(G)) \leq \text{poly}(\mu(F \land G), \log n)
\]

where \( n \) is the total number of variables in \( F \) and \( G \).

For our results on intersections, we use that the Nullstellensatz proof system has feasible disjunction for degree, over any field, and that the Sherali-Adams system has feasible disjunction for the measure \( \mu_{SA}^* \), defined as \( \log(\text{magnitude}) + \text{degree} \). The first result appears already in [22] and the second is new (Corollary 26).

Both things are proved in a similar way: if a CNF \( F \) has no small proofs, then there exists an object that behaves like an approximation of an assignment that satisfies \( F \). If \( G \) also has no small proofs, we can combine the object for \( F \) with the object for \( G \) to get something that behaves like an approximation of an assignment satisfying \( F \land G \), exploiting the fact that \( F \) and \( G \) share no variables. This is then enough to show that \( F \land G \) has no small proofs. Hakoniemi [25] used this approach to show feasible disjunction for degree for Sherali-Adams, and feasible disjunction for combined degree and magnitude for the sum-of-squares proof system. The same approach can show feasible disjunction for width in the resolution proof system, as in [16].

For our result for Sherali-Adams, we develop a new “semantic” characterization of Sherali-Adams in small magnitude and degree, modelled on the result for sum-of-squares in [25]. Precisely, we define an \( \varepsilon \)-pseudoexpectation over degree \( d \) for a set of polynomials \( Q \) and show that such an object exists, for a small error \( \varepsilon \), if and only if \( Q \) has no degree-\( d \) Sherali-Adams refutation of small magnitude (Lemmas 21 and 23). We note that a characterization of size in unary Sherali-Adams of a similar nature will appear in [17].

Finally we return to the standard meaning of feasible disjunction, which is for the size measure. We show that neither of the systems reversible resolution and reversible resolution with terminals has feasible disjunction in this sense (Lemma 30). This is interesting because both of them have feasible interpolation, which they inherit as subsystems of resolution [37]. As far as we know, these are the first examples of systems with feasible interpolation but not feasible disjunction, addressing the issue raised by Pudlák [46]. Our proof uses known upper and lower bounds from proof complexity, together with the characterization from [22] of \( \text{RevResT} \) as a kind of intersection of resolution and unary Nullstellensatz.

1.1.3 Combinatorial characterization of \( \text{PPA} \cap \text{PPADS} \) (Section 5)

Our separations, in particular, show that \( \text{PPAD} \subset \text{PPA} \cap \text{PPADS} \subset \text{PPA} \cap \text{PPP} \), i.e., the two intersections \( \text{PPA} \cap \text{PPADS} \) and \( \text{PPA} \cap \text{PPP} \) that were not explicitly studied in the literature yet might be of independent interest. As a first step towards improving our understanding of
these intersection classes, we introduce a complete combinatorial problem for \( \text{PPA} \cap \text{PPADS} \).

The new total search problem, which we call \text{Leaf-Or-Sink}, is close to \text{End-Of-Line} and other similar combinatorial problems characterizing the standard classes \text{PPA}, \text{PPAD}, and \text{PPADS}. It is an interesting open problem to provide a similar combinatorial problem for the higher intersection \( \text{PPA} \cap \text{PPP} \). In particular, it is known that (under randomized reductions) \( \text{PPA} \cap \text{PPP} \) contains integer factoring \([5, 30]\), and, thus, a concise complete problem for \( \text{PPA} \cap \text{PPP} \) could, for example, help in characterizing the search complexity of integer factoring.

For the formal definition of \text{Leaf-Or-Sink}, see Definition 35. Here, we give an informal description of the problem. Recall that an instance of \text{End-Of-Line} is given via a succinctly represented directed graph with vertices of in-degree and out-degree at most one (i.e., a collection of directed paths and cycles) with a known distinguished source. A solution is any sink in the graph or a source distinct from the distinguished source. Our \( \text{PPA} \cap \text{PPADS} \)-complete problem \text{Leaf-Or-Sink} is defined on a generalization of an \text{End-Of-Line} graph. The instance is given via a succinctly represented directed graph, where each vertex has at most two neighbors (i.e., we allow vertices of in-degree or out-degree two) with a distinguished source of degree one. A solution is any sink (of degree one or two) or a leaf (i.e., a vertex with a single neighbor) distinct from the distinguished source.

The containment of \text{Leaf-Or-Sink} in \( \text{PPA} \cap \text{PPADS} \) follows via straightforward reductions to the \( \text{PPA} \)-complete problem \text{Leaf} (i.e., the undirected variant of \text{End-Of-Line}) and \( \text{PPADS} \)-complete problem \text{Sink} (i.e., the variant of \text{End-Of-Line} where sources do not constitute solutions). We establish the hardness for \( \text{PPA} \cap \text{PPADS} \) via a reduction from the canonical \( \text{PPA} \cap \text{PPADS} \)-complete problem, where we are given an instance of \text{Leaf} and an instance of \text{Sink} and we are asked to solve either one of them. Our reduction orients the edges of the \text{Leaf} instance arbitrarily and resolves the “useless” solutions at sinks of degree two that are unrelated to leaves of the \text{Leaf} instance (that are likely introduced by the arbitrary orientation of the edges) using the instance of \text{Sink}. On a high level, for each predecessor of any sink of degree two, we can create a copy of the \text{Sink} graph and, instead of the sink of degree two, connect the predecessor to the distinguished source in its copy of the \text{Sink} graph. In this way, any sink of in-degree two becomes isolated and is no longer a solution to \text{Leaf-Or-Sink}. Unfortunately, even though we have resolved sinks of in-degree two, the copies of the \text{Sink} graph might have added new sources of degree one, which would constitute a solution to \text{Leaf-Or-Sink} but not to \text{Sink}. However, it is easy to take care of these sources due to the fact that there are exactly two copies of the \text{Sink} graph for each sink of in-degree two. For each sink of degree two that we were resolving, we can simply add a new source of degree two with outgoing edges to the two copies of the source. By such “gluing” of the \text{Leaf} graph with the copies of the \text{Sink} graph, we would construct an instance of \text{Leaf-Or-Sink} without sinks of degree two and where the leaves are either leaves in the \text{Leaf} graph or sinks in the \text{Sink} graph.

## 2 Preliminaries

### 2.1 Proof Systems and Measures

We recall some standard definitions. A \textit{clause} is a disjunction of propositional literals, where we write \( \bar{x}_i \) for the negation of the variable \( x_i \). A \textit{CNF} is a conjunction of clauses. A refutation of a CNF \( F \) in a proof system \( P \) is a witness that \( F \) is unsatisfiable, where the form of witness that is allowed depends on the system \( P \).
Definition 2. A measure $\mu$ for a proof system $P$ is a function which assigns a non-negative real number $\mu(\pi)$ to each $P$-refutation $\pi$.

For a CNF $F$ we define $\mu(F) := \min\{\mu(\pi) : \pi$ is a $P$-refutation of $F\}$.

The standard measure for a system is the size (for example in bits) of a refutation, and “feasible” proofs typically mean those of polynomial size in the size of the CNF $F$. However in this paper, following e.g. [22], we use a different scaling, using for example the logarithm of size, rather than size, and typically working with polylogarithmic measures (an exception is Section 3.3).

We work mostly with the algebraic systems Nullstellensatz and Sherali-Adams and their variants, which we introduce now. We also use two variants of the standard propositional proof system resolution, which we describe at the end of this section.

In the algebraic systems we work with polynomials in some field $F$ over some set of variables $x_1, \ldots, x_n$, together with twin/negated variables $\bar{x}_1, \ldots, \bar{x}_n$. Usually we constrain each $x_i$ to be 0/1-valued and constrain $\bar{x}_i$ to take the opposite value. With this in mind we define $I_n$ to be the ideal generated by the Boolean axioms $x^2 - x$ and the negation axioms $1 - x - \bar{x}$. We write $p \equiv q$ to mean that polynomials $p$ and $q$ are equal modulo this ideal.

A monomial is a product of variables, without any coefficient. We write $S_d$ for the set of all monomials of degree $d$ or less and $F[S_d]$ for the set of polynomials of degree $d$ or less, which we sometimes think of as a vector space spanned by $S_d$. We sometimes use the bound $|S_d| \leq (3n)^d$. When working over the reals, we write $\mathbb{R}^+[S_d]$ for the set of polynomials in $\mathbb{R}[S_d]$ which only have positive coefficients. For a real polynomial $p$, we write $||p||$ for the maximum absolute value of any coefficient in $p$.

Over the ideal $I_n$ we can convert any CNF $F$ into a set of polynomials $Q$ expressing the same thing as $F$, by translating a clause $C$ into the polynomial equation $\prod_{j \in J} \bar{x}_j \prod_{k \in K} x_k = 0$ where $J$ is the set of variables appearing positively in $C$ and $K$ is the set of variables appearing negatively. In this way we can use algebraic proof systems to refute CNFs.

Definition 3. Let $F$ be a field and let $Q$ be a set of polynomials over $F$. A nullstellensatz refutation over $F$, or $\text{NS}_F$ refutation, of $Q$ is an identity

$$1 \equiv \sum_{q \in Q} t_q q$$

where each $t_q$ is a polynomial over $F$.

Our main measure for an $\text{NS}_F$ refutation is degree, sometimes written as $\text{deg}$, which we define to be the maximum degree of the expressions $t_q q$. Note that we do not explicitly include the degree of the element of $I_n$ hidden in the notation $\equiv$ in this measure; but we may assume without loss of generality that it is bounded by the degree of the expressions $t_q q$, by the following argument. Suppose $p \equiv p'$ where $p'$ is the multilinearization of $p$. Suppose we multilinearize $p$ term-by-term, starting with some term $rx^2$ which we replace with $rx$. This is the same as adding $-r(x^2 - x)$ from the ideal, and to do this we never need an element of the ideal with degree higher than $\text{deg}(p)$. A similar argument works for dealing with the negation axioms.

Another natural measure we could consider here is the log of the monomial size of a refutation, where monomial size counts the total number of monomials that appear. However, since we can bound the number of monomials in a polynomial by $|S_d| \leq (3n)^d$, from our point of view this would be essentially the same as the degree measure.
Definition 4. Let $Q$ be a set of real polynomials. A Sherali-Adams refutation, or SA refutation, of $Q$ is an identity

$$-1 \equiv v + \sum_{q \in Q} t_q q$$

where $v$ and the $t_q$ are real polynomials, and $v$ only has positive coefficients.

A unary Sherali-Adams or uSA refutation is an SA refutation in which only integer coefficients are allowed.

The degree of such a refutation is the maximum of the degrees of the expressions $t_q q$ and $v$. The magnitude is the maximum of the absolute values of the coefficients appearing in these expressions. The size of an SA refutation is the total number of monomials that appear (counting repetitions). The unary size of a uSA refutation is the sum of the magnitudes of the coefficients in front of all the monomials. We also define combined measures for SA as

$$\mu^*_{SA}(\pi) := \log(\text{magnitude of } \pi) + \deg \pi$$

$$\mu_{uSA}(\pi) := \log(\text{unary size of } \pi) + \deg \pi$$

where $\mu_{uSA}(\pi)$ is only defined if $\pi$ is a uSA refutation. Note that for our purposes, the sum in each measure could be replaced with the maximum, with essentially no change.

Turning to purely propositional systems, a resolution refutation of a CNF $F$ is a sequence of clauses, concluding with the empty clause, in which each clause is either from $F$, or follows from earlier clauses either by the resolution rule, which derives $C \lor D$ from $C \lor x$ and $D \lor \bar{x}$, for any variable $x$, or by the weakening rule, which derives any clause $D \supset C$ from $C$. The width of such a refutation is the size of the largest clause, and is analogous to the degree measure for algebraic systems. In Section 3.3 we need the following variants of resolution, introduced in [22]. For these, we only use the standard size measure.

A reversible resolution, or RevRes, refutation of a CNF $F$ is a sequence of multisets $M_1, \ldots, M_t$ of clauses, where $M_1$ contains only clauses in $F$; $M_t$ contains the empty clause; and each $M_{i+1}$ is derived from $M_i$ either by the reversible resolution rule, which deletes one clause of the form $C \lor x$ and one clause of the form $C \lor \bar{x}$ from $M_i$, replacing them both with a single clause $C$; or by the reversible weakening rule, which is the same thing in reverse.

A reversible resolution with terminals, or RevResT, refutation of $F$ is a RevRes refutation of $F$ with the extra condition that $M_t$ contains exactly one copy of the empty clause and every other clause in $M_i$ is a weakening of a clause of $F$ (in the usual sense of weakening).

2.2 Decision-Tree TFNP and Proof Complexity

We work with the non-uniform “decision tree” model of TFNP and model our definitions and notation on [22]. A slight exception is Section 5, where we define a new class in terms of graphs given by circuits, without any explicit reference to an oracle; but this can be made into a definition in the decision-tree style without any difficulties.

Definition 5. A total search problem is a sequence $R = (R_n)_{n \in \mathbb{N}}$ of relations $R_n \subseteq \{0,1\}^m \times \mathcal{O}_n$, where $m_n \in \mathbb{N}$, each $\mathcal{O}_n$ is a finite set, and for every input $x \in \{0,1\}^m$ there is some solution $y \in \mathcal{O}_n$ such that $(x,y) \in R_n$.

Definition 6 (TFNP$^\text{dt}$). A total search problem, defined as above, is in TFNP$^\text{dt}$ if $m_n$ is at most quasipolynomial in $n$ and, for each $y \in \mathcal{O}_n$, there is a decision tree $T_y$ of depth $\text{poly}(\log n)$, querying $x$ and deciding whether $(x,y) \in R_n$. 

ITCS 2024
Definition 7. Let $R_m \subseteq \{0,1\}^r \times \mathcal{O}_R$ and $S_n \subseteq \{0,1\}^s \times \mathcal{O}_S$ be two total search problems. A reduction of $R_m$ to $S_n$ is a pair of functions $f : \{0,1\}^r \rightarrow \{0,1\}^s$ and $g : \{0,1\}^r \times \mathcal{O}_S \rightarrow \mathcal{O}_R$ satisfying, for all $x \in \{0,1\}^r$ and all $y' \in \mathcal{O}_S$,

$$(f(x), y') \in S_n \Rightarrow (x, g(x, y')) \in R_m.$$  

The reduction has depth $d$ if each bit of $f(x)$ and for every $y' \in \mathcal{O}_S$, the function $x \mapsto g(x, y')$ is computable by a depth-$d$ decision tree.

Definition 8. Let $R$ and $S$ be total search problems in TFNP$^{\text{dt}}$. We say $R$ is reducible to $S$ if for each $n$ there is $m$ quasipolynomial in $n$ such that there is a depth-$\text{poly}(\log n)$ reduction of $R_n$ to $S_m$. We write this as $R \leq S$. If also $S \leq R$ we say $R$ and $S$ are equivalent.

This expresses in our non-uniform setting the idea that we can use solutions to $S$ to find solutions to $R$. Note that $R \leq S$ is equivalent to the condition $S^{\text{dt}}(R) = \text{poly}(\log n)$ of [22]. Unless stated otherwise, we use only the model of TFNP described above, so we omit the superscripts $^{\text{dt}}$ in the names of classes that would otherwise be used to indicate this.

Definition 9. A narrow family of CNFs is a sequence $F = (F_n)_{n \in \mathbb{N}}$ in which in each $F_n$ the number of variables and clauses is quasipolynomial in $n$, and the width is polylogarithmic in $n$.

As usual, we sometimes use the notation like “a narrow CNF $F_n$” to mean a family.

Definition 10. Given an unsatisfiable CNF $G$, define Search($G$) to be the problem of finding a false clause in $G$, given a total assignment to the variables of $G$.

Now let $F = (F_n)_{n \in \mathbb{N}}$ be a narrow family of unsatisfiable CNFs. Then the sequence $(\text{Search}(F_n))_{n \in \mathbb{N}}$ is in TFNP.

Definition 11. Let $P$ be a proof system and let $\mu_P$ be a measure for $P$. We say that a TFNP class $A$ is characterized by $P$ under $\mu_P$ if, for every narrow family $(F_n)_{n \in \mathbb{N}}$ of unsatisfiable CNFs, we have $(\text{Search}(F_n))_{n \in \mathbb{N}} \in A$ if and only if $\mu_P(F_n) \leq \text{poly}(\log n)$.

We use the word “characterized” because it is possible to present any TFNP problem as a false clause search problem of this kind, as follows.

Lemma 12. Let $(R_n)_{n \in \mathbb{N}}$ be a TFNP problem. Let CNF($R_n$) be a CNF asserting that $x$ is an input with no solution in $R_n$ — that is, it asserts that for every $y$ in the solution space, the decision tree for $y$ does not accept $x$. Then CNF($R_n$) is a narrow family of unsatisfiable CNFs and, as TFNP problems, Search(CNF($R_n$)) and $R_n$ are equivalent.

We list some connections between TFNP classes and proof systems; see [22] for a more complete list.

Proposition 13.
1. $\text{PPA}_q$ is characterized by $\text{NS}_F_{q_n}$ under degree, for each prime $q$ [20, 33]
2. $\text{PPADS}$ is characterized by $\text{uSA}$ under $\mu_{\text{SA}}$ [22]
3. PLS is characterized by resolution under width [8].
2.3 Some Upper and Lower Bounds

We describe three families of unsatisfiable, narrow CNFs we use.

The bit pigeonhole principle $\text{BPHP}_n$ has variables for the bits of $2^k + 1$ binary strings, each of length $k$, where $k = \lfloor \log n \rfloor$. It consists of a clause for each pair of strings, asserting that those two strings are distinct. We think of total assignments to these variables as maps from $[n+1]$ pigeons (the indices of the strings) to $[n]$ holes (the values of the strings), so the principle asserts that no two pigeons go to the same hole.

▶ Proposition 14.
1. The problem $\text{Search}(\text{BPHP}_n)$ is in $\text{PPP}$
2. $\text{BPHP}_n$ requires degree $n$ to refute in $\text{SA}$.

Proof. For 1, we use that we have essentially a 1-1 correspondence between the solutions of the usual pigeonhole principle search problem and the clauses of $\text{BPHP}_n$.

Item 2 is shown (implicitly) in [12]. Their way of encoding CNFs into $\text{SA}$ refutations is slightly different from ours, since they translate clauses into linear inequalities rather than monomial equations, so we confirm that their proof works with our encoding. We translate the principle $\text{BPHP}_n$ into a system of equations $m_{i,i',j} = 0$, for each pair $i, i'$ of distinct pigeons and each hole $j$, where $m_{i,i',j}$ is the natural monomial, of degree $2k$, which is 1 if and only if both $i$ and $i'$ go to hole $j$. Suppose we have a $\text{SA}$ refutation of this system of degree at most $n$. We may write it as an equality

$$-1 = v + \sum_{i,i',j \neq i'} t_{i,i',j} m_{i,i',j} + B$$

where $v$ is a sum of monomials with positive coefficients, $B$ is in the ideal generated by the Boolean and negation axioms, and everything has degree at most $n$. We define an evaluation $E : S_n \to \mathbb{R}$ as follows. For any $m \in S_n$, let $X$ be the set of pigeons whose variables appear in $m$. Extend $X$ arbitrarily to a set $X'$ of exactly $n$ pigeons, and let $E(m)$ be the probability that $m$ evaluates to 1 under a random matching of $X'$ to the set $[n]$ of all holes. Observe that $E(m)$ is independent of the choice of $X'$. We extend $E$ to a linear map $\mathbb{R}[S_n] \to \mathbb{R}$ by linearity.

Then $E(1) = 1$, since the empty monomial is always satisfied. Also $E(v) \geq 0$, and for each $i,i',j$ in the sum we have $E(t_{i,i',j} m_{i,i',j}) = 0$, since every matching with $i$ and $i'$ in its domain sets $m_{i,i',j}$ to 0. Finally let $z$ be any single variable and let $m$ be any monomial of suitable degree. Then $E(mz^2) = E(mz)$ by construction, and also $E(mz) + E(mz) = E(m)$, since we may assume that the pigeon associated with $z$ is in $X'$, so every matching under consideration which satisfies $m$ satisfies exactly one of $mz$ and $mz$. Thus $E(B) = 0$. Together these contradict the displayed equality above. (Such an $E$ is called in the literature a pseudoepectation.)

The sink-of-DAG principle $\text{SoD}_n$ has variables for a directed graph on $[n]$, with a distinguished source node, such that edges only go from smaller to larger nodes in $[n]$. It asserts that this graph must have a sink.

▶ Proposition 15.
1. The problem $\text{Search}(\text{SoD}_n)$ is in $\text{PLS}$
2. For any prime $q$, any $\text{NS}_{\mathbb{F}_q}$ refutation of $\text{SoD}_n$ requires degree $\Omega(n)$.
3. Any $n^{o(1)}$ degree $\text{SA}$ refutation of $\text{SoD}_n$ has exponential magnitude.
Proof. Item 1 is straightforward. Item 2 follows from lower bounds in [11, 6] on the closely-related house-sitting principle. Item 3 is one of the main results of [22], but we must show that it still holds with our definitions, which are not quite the same—in particular the version of $SA$ in [22] does not have negated variables, and they count the coefficients in the ideal $I_n$ towards the magnitude.

Let $Q$ be the set of polynomials encoding $SoD_n$. Suppose we have an $SA$ refutation of $Q$ of degree $d \in n^{o(1)}$ and magnitude $R$, of the form

$$-1 \equiv v + \sum_{q \in Q} t_q q.$$  

We let $a$ and $b$ be the polynomials respectively in the ideal of Boolean axioms and the ideal of negation axioms witnessing this equivalence. So we have the following equation, where we may assume that any polynomial appearing in it has degree at most $d$.

$$\sum_{q \in Q} t_q q + a + b = 1 + v.$$  

Now we get rid of the negated variables by replacing each occurrence of $\bar{x}_i$ with $(1 - x_i)$. After this transformation $b$ becomes 0, and we can write the result as

$$\sum_{q \in Q} t'_q q' + a' = 1 + v'.$$  

By the proof of Lemma 3 of [22] there is a total assignment to the variables of (1) under which the right-hand-side $1 + v'$, and thus also $1 + v$, evaluates to at least $1.4^n$. But the maximum possible value of $v$ is $R(3n)^d$, since $v$ contains at most $(3n)^d$ monomials. Therefore $R$ is exponential. ◀

For a prime $p$, the mod-$p$ counting principle $\text{Count}^p_n$ has variables describing a partition of $[pn + 1]$. It asserts that every set in this partition has size precisely $p$.

Proposition 16.
1. The problem $\text{Search}(\text{Count}^q_n)$ is in PPA$_q$, or in PPA for $q = 2$.
2. For distinct primes $p,q$, any $\text{NSF}_q$ refutation of $\text{Count}^p_n$ requires degree $n^{\Omega(1)}$.

Proof. Item 1 is straightforward. Item 2 is from [7]. ◀

3 Feasible disjunction

We show in this section that Nullstellensatz has feasible disjunction for degree; that Sherali-Adams has feasible disjunction for the measure $\mu_{SA}$; and that feasible disjunction for size fails for RevRes and RevResT. What we would really like to show in the second case is that $uSA$ has feasible disjunction for the measure $\mu_{uSA}$, but the result we prove is enough for our argument about TFNP intersections, since the relevant lower bounds are on magnitude rather than explicitly on unary size.

3.1 Feasible Disjunction for Degree for Nullstellensatz

This is shown in Lemma 2, Claim 2 of [22], but we include a proof for completeness. We first need a “semantic” characterization of systems of polynomials with no low degree refutation.
Proposition 17 ([6]). Let \( F \) be a field. Then a set of polynomials \( Q \) does not have \( \text{NS}_F \)-refutations of degree \( d \) if and only if there is a \( d \)-design for \( Q \), that is, a linear function \( D : \mathbb{F}[S_d] \rightarrow \mathbb{F} \) satisfying firstly that \( D(1) = 1 \) and secondly that \( D(q) = 0 \) for \( q, r \) polynomials with \( \deg(q) + \deg(r) \leq d \), where \( q \) is either an axiom from \( F \) or a Boolean or negation axiom, and \( r \) is any polynomial.

Theorem 18 ([22]). For any field \( F \), \( \text{NS}_F \) has \( \deg \)-feasible disjunction.

Proof. Suppose \( Q(\bar{x}) \) and \( P(\bar{y}) \) are two families of polynomials over disjoint variables \( \bar{x} \) and \( \bar{y} \) and moreover that \( \text{NS}_F \) has a degree \( d \) refutation of \( Q \cup P \). For the sake of contradiction assume neither \( Q \) nor \( P \) has an \( \text{NS}_F \) refutation of degree \( d \). By Proposition 17 degree \( d \)-designs \( D_Q \) and \( D_P \) exist for \( Q \) and \( P \) respectively.

Any \( r \in \mathcal{F}[\bar{x}, \bar{y}] \) can be written uniquely as a polynomial \( \sum_{i=1}^{k} a_i m^i_0 m^i_1 \) where each \( m^i_0 \in \mathcal{F}[\bar{x}] \) and \( m^i_1 \in \mathcal{F}[\bar{y}] \). We define a map from such polynomials, of degree at most \( d \), to \( \mathcal{F} \) as follows:

\[
D(r) := \sum_{i=1}^{k} a_i D_Q(m^i_0) \cdot D_P(m^i_1).
\]

We claim that \( D \) is a \( d \)-design for \( Q \cup P \). It is clear from the definition that \( D \) is linear and \( D(1) = 1 \). For the second property, let \( q \) be an axiom and let \( r \in \mathcal{F}[\bar{x}, \bar{y}] \) be as above, such that \( \deg(q) + \deg(r) \leq d \). By the assumptions, \( q \) contains either only \( x \) variables or only \( y \) variables. Let us assume it is \( x \) variables; the case for \( y \) variables is similar. From the definition of \( D \) we obtain

\[
D(qr) = \sum_{i=1}^{k} a_i D_Q(qm^i_0) \cdot D_P(m^i_1).
\]

Since \( D_Q \) is a \( d \)-design for \( Q \), we get that \( D_Q(qm^i_0) = 0 \) for each \( i \), which implies that \( D(qr) = 0 \), as required. Thus \( D \) is a \( d \)-design for \( Q \cup P \), which implies that \( Q \cup P \) does not have \( \text{NS}_F \) refutations of degree \( d \), contradicting our initial assumption.

3.2 Feasible Disjunction for Size and Degree for Sherali-Adams

We show \( \mu^*_\text{SA} \)-feasible disjunction for \( \text{SA} \). We base our approach closely on the proof of feasible disjunction for sum-of-squares in [25]. We need the notion of an \( \text{SA} \) derivation:

Definition 19. An \( \text{SA} \) derivation of a polynomial \( p \) from a set of polynomials \( Q \) is a polynomial identity

\[
p \equiv v + \sum_{q \in Q} t_q q
\]

where \( v \) only has positive coefficients. We think of such an expression as a witness that \( p \geq 0 \) on every solution \( \bar{x} \) of \( Q \). A refutation of \( Q \) is then a derivation of \( -1 \) from \( Q \).

Again the degree of the derivation is the maximum degree of the expressions on the right hand side, without any cancellations. We say that the derivation is \( R \)-bounded if \( ||v|| \leq R \) and \( ||t_q|| \leq R \) for every \( q \in Q \), in other words, if it has magnitude bounded by \( R \).

Note that we do not bound the coefficients appearing in the polynomial in the ideal \( I_n \) which witnesses the equivalence \( \equiv \) in the definition of \( R \)-boundedness.
Definition 20. For $\varepsilon > 0$, an $\varepsilon$-pseudoexpectation for $Q$ over degree $d$ is a linear function $E : \mathbb{R}[S_d] \to \mathbb{R}$ satisfying
1. $E(1) = 1$
2. $E(p) = E(q)$ if $p \equiv q$
3. $E(m) \geq -\varepsilon$ for any $m \in S_d$
4. $|E(mq)| \leq \varepsilon$ for any $m \in S_d$ and $q \in Q$.

We show now that the existence of an $\varepsilon$-pseudoexpectation over degree $d$ is a semantic characterization of when $Q$ has no degree-$d$ $SA$ refutations of small magnitude, and thus can play a similar role to how $d$-designs characterized degree-$d$ $NS_F$ in Section 3.1.

Lemma 21 (Soundness). Over degree $d$, if there is an $\varepsilon$-pseudoexpectation for $Q$, then there is no $R$-bounded refutation of $Q$ for $R < 1/2\varepsilon|Q|(3n)^d$.

Proof. Let $E$ be an $\varepsilon$-pseudoexpectation for $Q$ over degree $d$ and suppose

$$-1 \equiv v + \sum_{q \in Q} t_q q$$

is a degree-$d$ $SA$ refutation of $Q$ with coefficients bounded by $R < 1/2\varepsilon|Q|(3n)^d$. By items 1 and 2 in Definition 20, we have $E(v) + E(\sum t_q q) = -1$.

Recall $v$ is a sum $\sum \lambda_i m_i$, where each $m_i \in S_d$ and $0 < \lambda_i < R$. Since $E(m_i) \geq -\varepsilon$ we have $E(\lambda_i m_i) \geq -R\varepsilon$, and thus $E(v) \geq -(3n)^dR\varepsilon > -1/2|Q|$, since $(3n)^d$ is a bound on the size of $S_d$. Similarly each $t_q$ is the sum of at most $(3n)^d$ monomials, so $|E(t_q)| \leq R(3n)^d\varepsilon < 1/2|Q|$, and thus in particular $E(\sum_{q \in Q} t_q q) > -1/2$. It follows that $E(v) + E(\sum t_q q) > -1$, a contradiction.

Lemma 22. For any $m \in S_d$ there is a sum $v$ of at most $d$ monomials in $S_d$, in which every coefficient is 1, such that $1 - m \equiv v$.

Proof. We prove this by induction. The base case is simply $1 - x_1 \equiv x_i$. For the inductive step, suppose $1 - m \equiv v$. Then

$$1 - mx_1 \equiv \bar{x}_1 + x_i - mx_i = \bar{x}_1 + x_i(1 - m) \equiv \bar{x}_1 + x_i v.$$  

We may assume that $m$ is multilinear, so that we need to consider each variable at most once.

Lemma 23 (Completeness). If there is no $R$-bounded refutation of $Q$ of degree $d$, then there is a $(1/R)$-pseudoexpectation for $Q$ over degree $d$.

Proof. Suppose there is no $R$-bounded refutation of $Q$ of degree $d$. Define sets $A, B \subseteq \mathbb{R}[S_d]$ by

$$A := \{ p : p \equiv v \text{ some } v \in \mathbb{R}^+[S_d] \text{ with } ||v|| \leq R \}$$
$$B := \{ -1 + \sum_{q \in Q} t_q q : t_q q \in \mathbb{R}[S_d] \text{ with } ||t_q|| \leq R \text{ for all } q \}.$$  

Then $A$ and $B$ are disjoint, since otherwise there would be an $R$-bounded refutation of $Q$ of degree $d$. Furthermore they are both nonempty and, considered as subsets of the vector space $\mathbb{R}[S_d]$, are both convex.

Now let $K$ be the quotient map $\mathbb{R}[S_d] \to \mathbb{R}[S_d]/I_n$. Then $K[A]$ and $K[B]$ are still nonempty and convex and, by the $\equiv$ in the definition of $A$, even disjoint. Thus by the hyperplane separation theorem in $\mathbb{R}[S_d]/I_n$ there is a nontrivial linear function $L' : \mathbb{R}[S_d]/I_n \to \mathbb{R}$
and a scalar $e \in \mathbb{R}$ such that $L'(p) \geq e$ for $p \in K[A]$ and $L'(p) \leq e$ for $p \in K[B]$. Composing $K$ with $L'$, we obtain a linear function $L : \mathbb{R}[S_d] \to \mathbb{R}$ with $L(p) \geq e$ for $p \in A$ and $L(p) \leq e$ for $p \in B$, with the extra property that $L(p) = L(q)$ if $p \equiv q$.

Since $0 \in A$ we have $e \leq L(0) = 0$ and since $-1 \in B$ we have $L(-1) \leq e$ and thus $L(1) \geq -e$. We claim $L(1) > 0$. If $e < 0$ this is clear, so suppose $L(1) = 0$ and $e = 0$. Let $m \in S_d$. Then $m \in A$ so $0 \leq L(m)$. On the other hand, by Lemma 22 also $1 - m \in A$, so $0 \leq L(1 - m) = L(1) - L(m) = -L(m)$. Thus $L(m) = 0$ for all $m \in S_d$, contradicting the nontriviality of $L$.

We claim that $E(p) := L(p)/L(1)$ is a $1/R$-pseudoexpectation. By construction $E$ is linear and satisfies conditions 1 and 2 of Definition 20. It remains to show conditions 3 and 4.

We have that $E(p) \geq e/L(1) \geq -1$ for $p \in A$ and $E(p) \leq e/L(1) \leq 0$ for $p \in B$.

For condition 3, let $m \in S_d$. Then $Rm \in A$ so $E(Rm) \geq -1$ hence $E(m) \geq -1/R$. For condition 4, we must show $|E(mq)| \leq 1/R$ for any $m \in S_d$ and $q \in Q$. We have $-1 \pm Rm \in B$ so $E((-1 \pm Rm)) \leq 0$, where we are using $\pm$ to mean that the inequalities hold whether we write $+$ or $-$. Thus $-1 \leq \pm RE(mq)$ giving $|E(mq)| \leq 1/R$ as required.

We can now use our semantic characterization of bounded magnitude, degree-$d$ SA to prove feasible disjunction, similar to the proof of Theorem 18. We first need a technical lemma.

Lemma 24. If $E$ is an $\varepsilon$-pseudoexpectation for $Q$ over degree $d$, then $E(m) \leq 1 + d\varepsilon$ for every $m \in S_d$.

Proof. Let $v$ be the sum of $d$ monomials such that $1 - m \equiv v$ given by Lemma 22. Then $E(1 - m) = E(v) \geq -d\varepsilon$, and then we can use linearity.

Theorem 25. Let $Q(x)$ and $P(y)$ be systems of polynomials in disjoint sets of variables $x,y$ and let $R \geq d$. Suppose that neither $Q$ nor $P$ has an $R$-bounded SA refutation of degree $d$. Then $Q \land P$ has no $R/2$-bounded refutation of degree $d$.

Proof. By the assumption on $P$ and $Q$ and the completeness lemma, there are $1/R$-pseudoexpectations $E_x$ and $E_y$ for respectively $P$ and $Q$ (in their respective variables), both over degree $d$.

Let $S$ be the set of monomials of degree at most $d$ in $x$-variables and also of degree at most $d$ in $y$-variables. We define $E : \mathbb{R}[S] \to \mathbb{R}$ on monomials $m_xm_y \in S$ by $E(m_xm_y) = E_x(m_x)E_y(m_y)$ and extend by linearity to the whole space. Note that for any polynomials $p = \sum_i m_i$ in only $x$-variables and $q = \sum_j n_j$ in only $y$-variables we have

$$E(pq) = E(\sum_i m_i \sum_j n_j) = \sum_i \sum_j E(m_i)E(n_j) = \sum_i \sum_j E_x(m_i)E_y(n_j) = E_x(p)E_y(q).$$

We claim that conditions 1–4 of Definition 20 hold for $E$, with $\varepsilon = 2/R$, which gives the result by the soundness lemma.

Condition 1 is clear. For condition 2, suppose $p \equiv q$ modulo the ideal generated by the Boolean and negation axioms for both $x$ and $y$ variables. Then $p - q$ can be written as a sum of terms of the form $a_xb_xc_y$ or $a_yb_xc_y$, where $a_x$ and $a_y$ are axioms from the ideal for a single $x$ or $y$ variable, and $b_x$ and $c_y$ are polynomials in only $x$ or only $y$ variables. But $E_x(a_x,b_x) = 0$ and $E_y(a_y,c_y) = 0$ for terms of this form, so $E(p - q) = 0$. 

ITCS 2024
For condition 3, let \( m_x m_y \) be a product of an \( x \)-monomial and a \( y \)-monomial. We have 
\(-1/R \leq E_x(m_x), E_y(m_y) \leq 2\), where the upper bound is from Lemma 24 and the assumption that \( R \geq d \). Thus \( E(m_x m_y) = E_x(m_x) E_y(m_y) \geq -2/R \).

For condition 4 we must show \( |E(mq)| \leq 2/R \) for any \( m = m_x m_y \) and any \( q \in P \cup Q \). Suppose without loss of generality that \( q \in Q \), so only has \( y \)-variables. Then \( |E(mq)| = |E_x(m_x)|/R \). As above, by Lemma 24 we have \( |E_x(m_x)| \leq 2 \), and \( |E_y(m_y q)| \leq 1/R \) by condition 4 for \( E_y \).

As an immediate corollary we get

**Corollary 26.** \( \text{SA} \) has \( \mu_{SA}^* \)-feasible disjunction.

### 3.3 Two Failures of feasible Disjunction

In this section, we consider feasible disjunction for size, using the standard definition of feasible disjunction. That is, for every pair \( F, G \) of CNFs over disjoint sets of variables, if there is a refutation \( \pi \) of \( F \land G \) then there is a refutation either of \( F \) or of \( G \) of size polynomial in the size of \( \pi \). We show that this fails for \( \text{RevRes} \) and \( \text{RevRes}^T \).

Let \( \phi \) be the CNF \( \text{SoD}_n \circ \oplus \), that is, the sink-of-DAG principle lifted by replacing each variable with an XOR of size 2. The following proposition is based on a remark in [22].

**Proposition 27.** The family \( \phi \) has quasipolynomial size, polylog width resolution refutations. On the other hand, it requires super-quasipolynomial size refutations in \( \text{uSA} \).

**Proof.** The upper bound is essentially by the robustness of polylog-width resolution. Let \( F(x_1, \ldots, x_n) \) be a \( k \)-CNF with \( m \) clauses and \( n \) variables. Then \( (F \circ \oplus)(y_1, z_1, \ldots, y_n, z_n) \) is a \( 2k \)-CNF with at most \( m2^k \) clauses (each \( x_i \) is substituted by \( y_i \circ \oplus z_i \)). Note that every clause in \( F \) turns into a \( 2k \)-CNF in \( F \circ \oplus \) with at most \( 2^k \) clauses. So if \( k \) is polylogarithmic in \( n \), it is straightforward to see that \( \text{Search}(F \circ \oplus) \leq \text{Search}(F) \). This means that \( \text{Search}(\phi_n) \leq \text{Search}(\text{SoD}_n) \) which implies that \( \text{Search}(\phi_n) \in \text{PLS} \) by Proposition 15. Therefore the family \( \phi_n \) has quasipolynomial size, polylog width resolution refutations by Proposition 13 and the fact that there are at most \( n^{\text{poly}(\log n)} \) many clauses of width \( \text{poly}(\log n) \).

For the lower bound, we follow the proof sketched in [22] (although we do not get as strong a bound as claimed there). We adapt the argument used for resolution in Theorem 4.2 of [3]. Let \( \pi \) be a \( \text{uSA} \) refutation of \( \phi_n \). We apply the following random restriction \( \rho \). Independently for each \( i \), select one of the pair \( (y_i, z_i) \) and set it to either 0 or 1, making each choice uniformly at random. Note that \( \phi_n \upharpoonright \rho \) is equivalent to \( \text{SoD}_n \) up to renaming of variables and literals. Let \( m \) be a monomial in \( \pi \). We may assume that each variable appears in \( m \) at most once. Then \( \Pr[m \upharpoonright \rho \neq 0] \leq \left( \frac{3}{4} \right)^{\deg(m)} \). The reason is that if one of \( y_i \) or \( z_i \) appears in \( m \) (as a positive or negative literal), then this literal is set to 0 with probability \( 1/4 \). If both variables appear in \( m \), then the probability that the product of these two literals is set to 0 is \( \frac{1}{2} < \left( \frac{3}{4} \right)^2 \).

Thus for any \( d \), the expected number of monomials in \( \pi \) of degree at least \( d \) which remain in \( \pi \upharpoonright \rho \) is at most \( \left( \frac{3}{4} \right)^d \) times the number of monomials in \( \pi \), by linearity of expectation. Let \( d := n^{\log \log n} \). Now suppose for a contradiction that the size of \( \pi \) is less than \( \left( \frac{3}{4} \right)^d \). Then there must exist a restriction \( \rho \) such that \( \pi \upharpoonright \rho \) has degree \( d \). But, since \( \phi_n \upharpoonright \rho \) is essentially \( \text{SoD}_n \) and \( \pi \upharpoonright \rho \) still has the form of an \( \text{uSA} \) refutation, this implies that \( \text{SoD}_n \) has \( n^{\log \log n} \) degree, \( \left( \frac{3}{4} \right)^n n^{\log \log n} \) size \( \text{SA} \) refutations, contradicting Proposition 15. ◀
In [28], Theorem 4.1, a family of bipartite graphs $G_n$ is constructed, between $n+1$ pigeons and $n$ holes, with degree bounded by a constant, such that the perfect matching principle $\text{PMP}_{G_n}$ requires exponential size to refute in resolution. Let $\psi_n$ be $\text{PMP}_{G_n}$, which we could also call onto-$\text{FPHP}_{G_n}$, that is, the onto functional pigeonhole principle on $G_n$.

**Proposition 28.** The family $\psi_n$ has polynomial sized, polylog degree refutations in uNS. On the other hand, it requires exponential sized refutations in resolution.

**Proof.** The lower bound is by [28]. The upper bound is by a small adaptation of the standard proof of onto functional PHP in an algebraic system [47]. Suppose $G_n$ has degree $d$, and that $\text{PMP}_{G_n}$ is written in variables $x_e$ for each edge $e$ in $G_n$, where an edge is formally a pair of nodes. $\text{PMP}_{G_n}$ consists of degree $d$ axioms $\prod_{e \ni i} x_e$ for each node $i$ and degree $2$ axioms $x_e x_f$ for each pair $e, f$ of distinct edges sharing a node. For the uNS refutation, observe that

$$\prod_{e \ni i} x_e \equiv \prod_{e \ni i} (1 - x_e) = 1 - \sum_{e \ni i} x_e + B_i$$

where $B_i$ is a degree at most $d$ combination of the $x_e x_f$ axioms. Summing over the set $P$ of $n + 1$ pigeon nodes and rearranging gives

$$n + 1 - \sum_{i \in P} \sum_{e \ni i} x_e \equiv \sum_{i \in P} (\prod_{e \ni i} x_e - B_i),$$

where the double sum on the left is precisely the sum over all variables, and the right hand side is a low-degree combination of axioms. Summing over holes instead of pigeons gives a similar equivalence, with $n$ in place of $n + 1$. Subtracting the second equivalence from the first gives the refutation. ◀

**Proposition 29 ([22]).** uSA simulates RevRes

**Proof.** It was observed in [22] that the constructions for simulating resolution by SA in the literature [13, 1] also give this simulation. We outline how this goes. Suppose we are given a RevRes refutation of a CNF $F$. For each multiset $M_i = \{C_1, \ldots, C_m\}$ of clauses in the refutation, let $p_i$ be the polynomial $r_1 + \cdots + r_m$, where $r_j$ is the standard translation of the clause $C_j$ into a monomial which is zero if and only if $C_j$ is true (where the monomial may used negated variables). Intuitively the equation $p_i = 0$ says the same thing as “every clause in $M_i$ is true”. It is easy to see by the structure of the reversible resolution and reversible weakening rules that $p_i \equiv p_{i+1}$ for each $i$, and in fact $p_{i+1} - p_i$ is simply a monomial times a negation axiom. Moreover it follows from the other conditions on a RevRes refutation that $p_1$ is a sum of (translations into monomials of) axioms, and that the last line $p_n$ has the form $1 + v$ where $1$ is the translation of the empty clause and $v$ is a sum of monomials with positive coefficients. Thus we have $p_1 \equiv 1 + v$, which we can rearrange as $-1 \equiv v - p_1$, which is the required uSA refutation of $F$. ◀

We may assume that $\phi_n$ and $\psi_n$ use disjoint sets of variables. By the next lemma and the fact that RevResT is a subsystem of RevRes we get the failure of feasible disjunction for both systems.

**Lemma 30.**

1. $\phi_n \land \psi_n$ has quasipolynomial sized refutations in RevResT
2. $\phi_n$ requires super-quasipolynomial sized refutations in RevRes
3. $\psi_n$ requires exponential sized refutations in RevRes.
Proof. For item 1 we use Theorem 6 of [22], which characterizes the proof measure of a CNF in \(\text{RevResT}\) (that is, minimum degree plus log of size over all refutations) as the sum of the measures in resolution and \(uNS\). In particular, a CNF has quasipolynomial size, polylog width refutations in \(\text{RevResT}\) if and only if it has such refutations in both resolution and \(uNS\). We have this for \(\phi_n \land \psi_n\) by the two propositions above.

For item 2, if \(\phi_n\) had small refutations in \(\text{RevRes}\) then by Proposition 29 it would also have small refutations in \(uSA\), contradicting Proposition 27. Item 3 follows directly from the lower bound for resolution in Proposition 28. 

\(\square\)

4 Results on intersections

For the purposes of this paper, we define the classical TFNP classes to be the six classes \(\text{PLS}\), \(\text{PPA}\), \(\text{PPP}\), \(\text{PPADS}\), \(\text{PPAD}\), and \(\text{CLS}\) depicted in Figure 1. Among these classes we have the relations

\[
\text{PPP} \supset \text{PPADS} \supset \text{PPAD} \supset \text{CLS}; \quad \text{PPA} \supset \text{PPAD}; \quad \text{PLS} \supset \text{CLS}
\]

and no inclusions hold, other than those implied by the above inclusions [2, 41, 22]. This structure does not change if we replace \(\text{PPA}\) with \(\text{PPA}_q\) for any prime \(q\), and, furthermore, there is no inclusion between classes \(\text{PPA}_q\) and \(\text{PPA}_r\), for distinct primes [31, 26].

We say that a nontrivial intersection is an intersection \(A \cap B\) of two classical classes where neither \(A \subseteq B\) nor \(B \subseteq A\).

\(\square\)

Lemma 31. Suppose \(F_n, G_n\) are narrow CNF families and \(A, B\) are TFNP classes with \(\text{Search}(F_n) \in A\) and \(\text{Search}(G_n) \in B\). Then \(\text{Search}(F_n \land G_n) \in A \cap B\).

Proof. Clearly \(\text{Search}(F_n \land G_n) \leq \text{Search}(F_n)\), since an assignment to the variables of \(F_n \land G_n\) is in particular an assignment to the variables of \(F_n\), and finding a false clause in \(F_n\) gives us a false clause in \(F_n \land G_n\). Similarly \(\text{Search}(F_n \land G_n) \leq \text{Search}(G_n)\). 

\(\square\)

Theorem 32. With the exception of \(\text{PPAD} \cap \text{PLS} = \text{CLS}\), no nontrivial intersection is equal to a classical class. No two distinct nontrivial intersections are equal to each other.

Proof. We start by going through each classical class in turn, and showing that intersecting with that class does not lead to any unexpected collapses.

Intersections with \(\text{PLS}\). We have

\[
\begin{align*}
\text{PLS} & \supset \text{PPP} \cap \text{PLS} \supseteq \text{PPADS} \cap \text{PLS} \supseteq \text{CLS} (= \text{PPAD} \cap \text{PLS}); \\
\text{PLS} & \supset \text{PPA} \cap \text{PLS} \supseteq \text{CLS}.
\end{align*}
\]

Note that the intersection has collapsed \(\text{PPAD}\) and \(\text{CLS}\) together. We claim these classes are all distinct. This follows from the following claims.

1. \(\text{PPP} \cap \text{PLS} \not\subseteq \text{PPADS}\). Suppose this inclusion held. By Propositions 14 and 15 we have \(\text{Search}(\text{BPHP}_n) \in \text{PPP}\) and \(\text{Search}(\text{SoD}_n) \in \text{PLS}\). By Lemma 31 it follows that \(\text{Search}(\text{BPHP}_n \land \text{SoD}_n) \in \text{PPADS}\). By the \(uSA\) characterization of \(\text{PPADS}\) (in Proposition 13) the conjunction \(\text{BPHP}_n \land \text{SoD}_n\) has \(uSA\) refutations of measure \(\mu_{uSA} \leq \text{poly}(\log n)\), and thus in particular has \(SA\) derivations of measure \(\mu_{SA}^* \leq \text{poly}(\log n)\). We may assume that \(\text{BPHP}_n\) and \(\text{SoD}_n\) have no variables in common, so we can apply \(\mu_{SA}^*\)-feasible disjunction for \(SA\) (Corollary 26) to conclude that either
a. for infinitely many \( n \), \( \text{BPHP}_n \) has \( \text{SA} \) refutations of measure \( \mu^\text{SA}_2 \leq \text{poly}(\log n) \). Hence in particular they have degree \( \text{poly}(\log n) \), which is impossible by Proposition 14.

b. for infinitely many \( n \), \( \text{SoD}_n \) has \( \text{SA} \) refutations of measure \( \mu^\text{SA}_2 \leq \text{poly}(\log n) \), that is, simultaneously of degree \( \text{poly}(\log n) \) and of magnitude quasipolynomial in \( n \). This is impossible by Proposition 15.

2. \( \text{PPADS} \cap \text{PLS} \not\subseteq \text{PPA} \). This item was proved in Lemma 2 of [22]. Here we repeat the argument for the sake of completeness. Suppose this inclusion held. By Lemma 12 and Proposition 15 we have \( \text{Search}(\text{CNF}(Q_n)) \in \text{PPADS} \) and \( \text{Search}(\text{SoD}_n) \in \text{PLS} \), where \( Q \) is any complete problem for \( \text{PPADS} \). By Lemma 31 it follows that \( \text{Search}(\text{CNF}(Q_n) \land \text{SoD}_n) \in \text{PPA} \), and hence this conjunction has \( \text{poly}(\log n) \)-degree \( \text{NS}_{F_2} \) refutations, by Proposition 13. We may assume \( \text{CNF}(Q_n) \) and \( \text{SoD}_n \) have no variables in common, so we can apply degree-feasible disjunction for \( \text{NS}_{F_2} \) (Theorem 18) to conclude that either

a. for infinitely many \( n \), \( \text{SoD}_n \) has \( \text{NS}_{F_2} \) refutation of degree \( \text{poly}(\log n) \). This is impossible by Proposition 15.

b. for all sufficiently large \( n \), \( \text{CNF}(Q_n) \) has \( \text{NS}_{F_2} \) refutations of degree \( \text{poly}(\log n) \). Then by Proposition 13 and Lemma 12, we have \( Q \in \text{PPA} \). This is impossible, as \( Q \) is complete for \( \text{PPADS} \).

3. \( \text{PPA} \cap \text{PLS} \not\subseteq \text{PPA}_3 \). This is proved in exactly the same way as the previous item, using \( \text{NS}_{F_2} \) instead of \( \text{NS}_{F_2} \), and taking \( Q \) to be any complete problem for \( \text{PPA} \).

By items 1 and 2 all inclusions in the first row are strict, where to apply 2 we use that \( \text{CLS} \subseteq \text{PPA} \). For \( \text{PPA} \cap \text{PLS} \), by item 2 it is distinct from the two classes in the middle of the first row, and by item 3 it is distinct from \( \text{CLS} \), since \( \text{CLS} \subseteq \text{PPAD} \subseteq \text{PPA}_3 \).

Intersections with \( \text{PPP} \). We have

\[
\text{PPP} \supset \text{PPADS} \supset \text{PPAD} \supset \text{CLS};
\]

\[
\text{PPP} \supset \text{PLS} \cap \text{PPP} \supset \text{CLS};
\]

\[
\text{PPP} \supset \text{PPA} \cap \text{PPP} \supset \text{PPAD}
\]

where \( \text{PPADS}, \text{PPAD} \) and \( \text{CLS} \) are unchanged by intersection with \( \text{PPP} \). We claim the classes above are all distinct. This follows for the classes in the first two rows by item 1 above. For \( \text{PPA} \cap \text{PPP} \) we have

4. \( \text{PPA} \cap \text{PPP} \not\subseteq \text{PPADS} \). Suppose this inclusion held. Then, using a similar argument to item 2, \( \text{CNF}(Q_n) \land \text{BPHP}_n \) has \( \text{SA} \) refutations of measure \( \mu^\text{SA}_2 \leq \text{poly}(\log n) \), where \( Q \) is any complete problem for \( \text{PPA} \). We then get a contradiction from \( \mu^\text{SA}_2 \)-feasible disjunction for \( \text{SA} \), since \( \text{BPHP}_n \) requires large \( \text{SA} \)-degree for all large \( n \) and \( Q \not\subseteq \text{PPADS} \).

Thus \( \text{PPA} \cap \text{PPP} \) is different from anything in the top row. The remaining possible equality is \( \text{PPA} \cap \text{PPP} = \text{PLS} \cap \text{PPP} \), but this is impossible, as \( \text{PPAD} \) is included in the left hand side but not in \( \text{PLS} \).

Intersections with \( \text{PPA} \). We have

\[
\text{PPA} \supset \text{PPAD} \supset \text{CLS};
\]

\[
\text{PPA} \supset \text{PPP} \cap \text{PPA} \supset \text{PPADS} \cap \text{PPA} \supset \text{PPAD};
\]

\[
\text{PPA} \supset \text{PLS} \cap \text{PPA} \supset \text{CLS}
\]

where \( \text{PPAD} \) and \( \text{CLS} \) are unchanged by intersection with \( \text{PPA} \). We claim the classes above are all distinct. We have
5. PPADS \cap PPA \not\subseteq PPA. Suppose this inclusion held. We get that CNF(Q_n) \land \text{Count}_n has \text{poly}(\log n)-degree NS_{F_3} refutations, where Q is complete for PPADS. We then proceed as in item 2, using deg-feasible disjunction for NS_{F_3} and the degree lower bounds for NS_{F_3} from Proposition 16.

Item 5, together with item 4, imply that the first two rows are all distinct. For PLS \cap PPA, by item 3 it is different from anything in the first row, and since PPAD \not\subseteq PLS it is different from anything in the second row.

Intersections with PPADS. We have

PPADS \supset PPAD \supset CLS;
PPADS \supset PPA \cap PPADS \supseteq PPAD;
PPADS \supset PLS \cap PPADS \supset CLS

where PPAD and CLS are unchanged by the intersection, and PPP collapses to PPADS. We claim the classes above are all distinct. PPA \cap PPADS is distinct from everything in the first row by item 5. Then PLS \cap PPADS is distinct from everything else by item 2.

Intersections with PPAD. We have

PPAD \supset CLS.

Under the intersection, everything above PPAD collapses to PPAD, and PLS collapses to CLS. So there is nothing to show.

We have now shown that no nontrivial intersection is equal to a classical class. It remains to show that no two distinct nontrivial intersections are equal to each other. Suppose for a contradiction that we have such an equality A \cap B = C \cap D. We have dealt with all cases where one class appears on both sides, so we may assume the classes A, B, C, D are all distinct. None of them can be CLS, by the condition that the intersections are nontrivial. If we consider the four classes PPP, PPADS, PPAD, PPA, it is similarly impossible to form two nontrivial intersections from these, since PPAD is contained in the other three classes. So without loss of generality we may assume A = PLS and thus C \cap D \subseteq PLS. But this is impossible, as for all remaining choices of C and D we have PPAD \subseteq C \cap D.

\begin{proof}
Everything in the proof of Theorem 32 goes through if we replace PPA with PPAD_q – the only changes we need to make are replacing NS_{F_3} with NS_{F_q} and, if q = 3, replacing NS_{F_3} with NS_{F_3} (or with NS_{F_r} for any prime r different from 3); all the relevant upper and lower bounds still work.

This leaves equalities involving at least two classes PPAD_q, PPA_r, with q, r distinct primes. We consider the possible forms these could take.

The form PPAD_q \cap B = PPA_r. This implies PPA_r \subseteq PPA_q, which is false.

The form PPAD_q \cap PPA_r = C. By the previous case, C must be a classical class, so C can only be PPAD (since we know PPAD \subseteq PPAD_q \cap PPA_r). But by Section 7.2 of [23], this is not the case.
\end{proof}
The form $\text{PPA}_q \cap B = \text{PPA}_r \cap D$. By feasible disjunction for $\text{NS}_{F_q}$, for each $n$ we get a low-degree $\text{NS}_{F_q}$ refutation of either Count$_n$ or $\text{CNF}(D_n)$. Since the first of these is impossible for large $n$, it must asymptotically be a refutation of $\text{CNF}(D_n)$, implying $D \subseteq \text{PPA}_q$. Thus $D$ must be either PPAD or CLS, contradicting that the right hand side was a nontrivial intersection.

The form $\text{PPA}_q \cap \text{PPA}_r = C \cap D$. By the above case neither $C$ nor $D$ can be $\text{PPA}_q$ for a prime $s$. Since $C \cap D \supseteq \text{PPAD}$ and $\text{PLS} \not\subseteq \text{PPAD}$, we must have that $C$ and $D$ are two classes from the list PPP, PPADS, PPAD; but in this case $C \cap D$ is a trivial intersection. ▷

5 A Combinatorial Characterization of PPADS $\cap$ PPA

In this section, we give a combinatorial characterization of the intersection of PPADS and PPA. Unlike the separations for the decision tree variants of TFNP classes established in the previous section, the characterization holds also in the standard “white-box” model of TFNP. As discussed in Section 1.1.3, the search problem Leaf-Or-Sink is to find a leaf (i.e., a vertex with a single neighbor) distinct from $0^n$ or a sink in a directed graph where each vertex is of degree at most two and $0^n$ is a source of degree one.

Next, we give the formal definition of the problem. To avoid cumbersome notation, we first define graph induced by a triple of Boolean circuits.

▸ Definition 34. Given Boolean circuits $N_0, N_1: \{0,1\}^n \rightarrow \{0,1\}^n$, and $D: \{0,1\}^{2n} \rightarrow \{0,1\}$, we define a directed graph $G_{N_0,N_1,D} = (V,E)$, where $V = \{0,1\}^n$ and, for $u, v \in \{0,1\}^n$, $(u,v) \in E$ if and only if $u \neq v$, $u \in \{N_0(v)\} \cup \{N_1(v)\}$, $v \in \{N_0(u)\} \cup \{N_1(u)\}$, $D(u,v) = D(v,u)$, and either
1. $u < v$ and $D(u,v) = 1$, or
2. $u > v$ and $D(u,v) = 0$.

Note that the above definition induces a graph on $\{0,1\}^n$, where each vertex is of degree at most two and, for any pair of vertices $u$ and $v$, there is at most one directed edge adjacent to $u$ and $v$.

▸ Definition 35 (Leaf-Or-Sink). The search problem Leaf-Or-Sink is defined by the relation:
Instance: Boolean circuits $N_0, N_1: \{0,1\}^n \rightarrow \{0,1\}^n$, and $D: \{0,1\}^{2n} \rightarrow \{0,1\}$ such that, in the corresponding graph $G_{N_0,N_1,D}$, the vertex $0^n$ is a source of degree one.

Solution: Either a vertex of degree one distinct from $0^n$ or any sink in $G_{N_0,N_1,D}$.

Next, we show that Leaf-Or-Sink is complete for PPADS $\cap$ PPA. The containment in PPADS $\cap$ PPA is trivial. The high-level overview of our proof of PPADS $\cap$ PPA-hardness is provided in Section 1.1.3.

Below, we recall the formal definitions of the PPA-complete problem Leaf and the PPADS-complete problem Sink. Similarly to the above definition of Leaf-Or-Sink, we phrase the definition in terms of the graph induced by the Boolean circuit(s) given as input to the problem. For a Leaf instance $N: \{0,1\}^n \rightarrow \{0,1\}^{2n}$, $G_N$ is a graph on $\{0,1\}^n$ and there is an undirected edge between $u$ and $v$ iff $u \neq v$, $u \in N(v)$ and $v \in N(u)$. Similarly, for a Sink instance $S, P: \{0,1\}^n \rightarrow \{0,1\}^n$, $G_{S,P}$ is a graph on $\{0,1\}^n$ and there is a directed edge from $u$ to $v$ iff $u \neq v$, $S(u) = v$ and $P(v) = u$. 
Definition 36 (Leaf). The search problem Leaf is defined by the relation:

Instance: Boolean circuit $N: \{0,1\}^n \rightarrow \{0,1\}^{2n}$ such that, in the corresponding undirected graph $G_N$, the vertex $0^n$ is a leaf.

Solution: A leaf in $G_N$ distinct from $0^n$.

Definition 37 (Sink). The search problem Sink is defined by the relation:

Instance: Boolean circuits $S, P: \{0,1\}^n \rightarrow \{0,1\}^n$ such that, in the corresponding directed graph $G_{S,P}$, the vertex $0^n$ is a source of degree one.

Solution: A sink in $G_{S,P}$.

Finally, we state and prove the main theorem of this section.

Theorem 38. Leaf-Or-Sink is complete for PPADS $\cap$ PPA.

Proof. Any instance $(N_0, N_1, D)$ of Leaf-Or-Sink can be reduced to both the PPA-complete problem Leaf and the PPADS-complete problem Sink-Of-Line. First, the circuits $N_0$ and $N_1$ specify an undirected Leaf graph where all the solutions, i.e., leaves distinct from $0^n$, correspond either to a source of degree one distinct from $0^n$ or a sink (of degree one) in $G_{N_0, N_1, D}$, i.e., to a solution of the Leaf-Or-Sink instance $(N_0, N_1, D)$. Second, the directed graph $G_{N_0, N_1, D}$ can be locally transformed into a directed graph satisfying the stronger requirement that any vertex $v \in \{0,1\}^n$ is of in-degree and out-degree at most one: for any vertex of in-degree two or out-degree two, simply erase the adjacent edges. The resulting graph is an instance of Sink-Of-Line where all the solutions, i.e., sinks, are either sinks or neighbors of a sink in $G_{N_0, N_1, D}$ and, thus, can be used to efficiently find a solution to the Leaf-Or-Sink instance $(N_0, N_1, D)$. Therefore, Leaf-Or-Sink is contained in PPADS $\cap$ PPA.\(^2\)

Next, we show that Leaf-Or-Sink is PPADS $\cap$ PPA-hard, i.e., we give a reduction from a PPADS $\cap$ PPA-complete problem to Leaf-Or-Sink. Consider an arbitrary problem in PPADS $\cap$ PPA given by an instance $(S, P)$ of Sink and instance $N$ of Leaf. For ease of exposition, suppose that all the circuits $S, P$, and $N$ take $n$-bit inputs (the general case can be handled easily by padding to the same input length).

Using the instances $(S, P)$ and $N$, we define an instance of Leaf-Or-Sink on $2n$ bits in the following phases, starting with only isolated vertices:

1. Put an undirected edge between vertices $(u, 0^n)$ and $(v, 0^n)$ if there is an edge between $u$ and $v$ in the undirected graph induced by the Leaf instance $N$.

2. Orient the edges in the graph w.r.t. the lexicographic order on $\{0,1\}^n$.

3. Erase both edges adjacent to any sink $(w, 0^n)$ of in-degree two and, for any neighbor $u$ of $w$ in the Leaf instance $N$, embed the instance $(S, P)$ of Sink-Of-Line on the vertices of the form $(u, \cdot)$. Formally, add the edge $((u, x), (u, y))$ if $(x, y)$ is an edge in the directed graph induced by $(S, P)$. For any source $s \neq 0^n$ in the graph induced by $(S, P)$ and neighbors $(u, 0^n)$ and $(v, 0^n)$ of any sink $(w, 0^n)$ of degree two, add edges $((w, s), (u, s))$ and $((w, s), (v, s))$.

\(^2\) In more detail, we need to provide an implicit representation of the PPADS instance via a successor and predecessor circuit. Note that these can be implemented via a constant number of queries to the circuits $(N_0, N_1, D)$. 
Note that we can assume without loss of generality that, in the graph constructed after the second phase, any sink of in-degree two is connected only to vertices of in-degree and out-degree at most one. In particular, no pair of sinks of in-degree two share a predecessor and, therefore, the steps of the third phase are well defined as each embedding performed in the third phase involves distinct vertices.

The resulting graph is clearly an instance of Leaf-Or-Sink. Importantly, neither the new sources of degree two of the form \((w, s)\) nor the balanced vertices \((u, s)\) and \((v, s)\) of in-degree and out-degree one added during the third phase constitute a solution to Leaf-Or-Sink. Let \((u, x)\) be a sink in the resulting graph. By our construction, either \(x = 0^n\) and \(u \neq 0^n\) is a leaf in the Leaf graph induced by \(N\) or \(x\) is a sink in the Sink-Of-Line graph induced by \((S, P)\). Let \((u, x)\) be a source in this graph of degree one. By our construction, it must be the case that \(x = 0^n\) and \(u \neq 0^n\) is a leaf in the Leaf graph induced by \(N\) or \(x\) is a sink in the Sink-Of-Line graph induced by \((S, P)\). ▶

Note that, by allowing fewer or more types of vertices as solutions in the definition of Leaf-Or-Sink, we can provide alternative formulations for the problems defining PPA, PPADS, and PPAD. The variant, where we search only for leaves distinct from the distinguished source is PPA-complete. The variant, where we search for any sink or source (of degree one or two) distinct from the distinguished source is PPAD-complete. The variant where we search only for sinks is PPADS-complete.

References


---

3 This property could be achieved by a simple additional transformation. For every edge \((u, v) \in E\), we can introduce a new vertex \(w_{u,v}\) and add the edges \((u, w_{u,v})\) and \((w_{u,v}, v)\). This transformation is locally computable, preserves the structure of the graph, and ensures the claimed structure w.r.t. degrees of vertices.
TFNP Intersections Through the Lens of Feasible Disjunction


