Separations and characterizations in weak propositional proof systems

Theodoros Papamakarios[1]

[1]Department of Computer Science, University of Chicago, Chicago (papamakarios@uchicago.edu)

January 20, 2022

Abstract

We study weak propositional proof systems as subsystems of sequent calculus. It is well known that allowing more formulas to participate in applications of the cut rule, one gets a hierarchy of systems strictly increasing in strength. We extend this hierarchy at the very bottom to show that even allowing only propositional variables as cuts, one gets a super-polynomially more powerful system. In more customary terms, we show a super-polynomial separation between cut-free sequent calculus and resolution. Furthermore, we identify two new big clusters of proof complexity measures equivalent up to polynomial and $\log n$ factors, in the vicinity of the first levels of the above hierarchy. The first cluster contains, among others, the logarithm of tree-like resolution size, regularized (that is, multiplied by the logarithm of proof length) clause and monomial space, and clause space, both ordinary and regularized, in regular and tree-like resolution. As a consequence, separating clause or monomial space from the (logarithm of) tree-like resolution size is the same as showing a strong trade-off between clause or monomial space and proof length, and is the same as showing a super-critical trade-off between clause space and depth. The second cluster contains width, Σ_2 space (a generalization of clause space to depth 2 Frege systems), both ordinary and regularized, as well as the logarithm of tree-like size in the system $R(\log)$. As an application of some of these simulations, we improve a known size-space trade-off for polynomial calculus with resolution. In terms of separations between the measures above, to show our super-polynomial separation of resolution and cut-free sequent calculus, we show as a first step, a quadratic gap between resolution and cut-free sequent calculus width. This also allows us to get, for the first time, a quadratic separation between resolution width and monomial space in polynomial calculus with resolution. Furthermore, we show a quadratic lower bound on tree-like resolution size for formulas refutable in clause space 4. We introduce on our way yet another proof complexity measure intermediate between depth and the logarithm of tree-like size in resolution that might be of independent interest.

Contents

1 Introduction

The aim of this paper is to present the results of the two reports [38, 37]. Both are concerned with weak propositional proof systems, the relations among them and between complexity measures within them.

The importance of studying weak propositional proof systems is two-fold. On the one hand, they form a concrete basis for understanding lower bounds, the required techniques and their limitations. It is a remarkable fact that many, if not most, known lower bounds in propositional proof complexity rely, or can be shown to rely, on resolution width lower bounds. On the other hand, practical SAT solving, and the surprising success thereof, is based on weak propositional proof systems, and an in depth understanding of these systems, stands as a task of great significance.

One can naturally get weak proof systems as subsystems of the sequent calculus by restricting the formulas allowed to participate in the cut rule. Among these, the weakest is cut-free sequent calculus where no applications of the cut rule are allowed. It has been an open problem whether cut-free sequent calculus can polynomially simulate resolution, a sequent calculus system that only uses atomic cuts, i.e. cuts on propositional variables. It is this problem [37] is occupied with.

The problem was first raised in [18], the paper which essentially established proof complexity as an autonomous area in the computational complexity endeavor, and iterated e.g. in the influential survey [47]. Cook and Reckhow [18] show that in the tree-like case, there are examples where resolution can have exponentially smaller proofs. Arai, Pitassi and Urquhart [3] point out that the answer may heavily depend on how clauses are represented. A clause consisting of the literals, say $\ell_1, \ell_2, \ell_3, \ell_4$, can be seen as either a single disjunction of arity four, or as a series of applications of binary disjunctions, for example $(\ell_1 \vee \ell_2) \vee (\ell_3 \vee \ell_4)$, and this can have a profound impact on the complexity of sequent calculus proofs. The result of Cook and Reckhow above applies in the case where clauses are seen as single disjunctions of unbounded arity, or the case where the order in which the binary disjunctions are applied is fixed. If we are free to choose the order, then tree-like cut-free sequent calculus can quasi-polynomially simulate tree-like resolution, and this is optimal [3]. In the DAG-like case, and if we are free to choose the order in which binary disjunctions are applied, Reckhow [43] shows that cut-free sequent calculus can polynomially simulate regular resolution, and Arai [2] shows that it can polynomially simulate resolution for refuting k-CNF formulas, where $k = O(\log n)$. However, the general question has remained unresolved.

We define the *width* of a sequent calculus proof as the maximum number of formulas occurring in a sequent of the proof. This definition extends in a natural way the concept of the width of a resolution proof to stronger proof systems. Furthermore, it allows for a simple, abstract characterization of sequent calculus width generalizing the characterization of Atserias and Dalmau for resolution width [4]. Using this characterization, we show a quadratic gap between resolution width and cut-free sequent calculus width. This says that including atomic cuts in cut-free sequent calculus can shorten the width of proofs. Utilizing then this gap, we show a super-polynomial separation between cut-free sequent calculus and resolution. To put it in other words, atomic cuts can super-polynomially decrease the size of proofs. This result applies only when clauses are seen as disjunctions of unbounded arity. There is a way to write the clauses in our examples using binary disjunctions, so that the resulting formulas have linear size cut-free sequent calculus refutations. Thus, as it was already known for the tree-like case, the complexity of sequent calculus proofs can depend on how disjunctions are represented.

Towards [38], there is a rich landscape of relations between seemingly unrelated measures of the complexity of proofs for weak proof systems. We already mentioned width and hinted at its importance. Although used at early size lower bounds, its crucial role was only made apparent by Ben-Sasson and Wigderson [11] who, relying on the earlier work of Impagliazzo, Pudlák and Sgall [30] who showed an analogous result for polynomial calculus, showed that width lower bounds directly imply size lower bounds in resolution. Namely, for tree-like resolution proofs,

$$W_R(F \vdash \bot) \leq \log S_{T,R}(F \vdash \bot) + W(F), \tag{1.1}$$

and for general resolution proofs,

$$W_R(F \vdash \bot) \leq O\left(\sqrt{n \log S_R(F \vdash \bot)}\right) + W(F). \tag{1.2}$$

Here $W_R(F \vdash \bot)$, $S_{T,R}(F \vdash \bot)$ and $S_R(F \vdash \bot)$ stand for the minimum width, tree-like size and DAG-like size respectively of refuting an unsatisfiable CNF F in resolution; similar notation is employed throughout the paper. By $W(F)$, we denote the maximum width of a clause in F itself.

In addition to providing size lower bounds, several notions of width have been used to show space lower bounds in different proof systems, demonstrating a close relationship between the two measures [7, 21, 4, 14, 15, 13, 12, 26, 38]. In particular, Atserias and Dalmau [4] show that the minimum width, and the minimum clause space needed to refute F in resolution, $\mathrm{CSpace}(F \vdash \bot)$, satisfy

$$W_R(F \vdash \bot) \leq \mathrm{CSpace}(F \vdash \bot) + W(F), \tag{1.3}$$

and Galesi, Kołodziejczyk and Thapen [26] show a similar relation between resolution width and the minimum monomial space needed to refute F in polynomial calculus with resolution:

$$W_R(F \vdash \bot) \leq O\left((\mathrm{MSpace}(F \vdash \bot))^2\right) + W(F). \tag{1.4}$$

It is worth noting that the relation (1.3) actually refines (1.1). This is due to a result of Esteban and Torán [22] relating the minimum clause space and minimum tree-like size for refuting F:

$$\mathrm{CSpace}(F \vdash \bot) \leq \log S_{T,R}(F \vdash \bot). \tag{1.5}$$

In this way, two *sequential* measures, tree-like size and width, are related with a *space* measure as an intermediate. We will see more examples of such an interplay in this paper.

Regularized versions μ^* of space complexity measures are defined by multiplying the measure in question μ by the logarithm of the proof length; these were considered e.g. by Ben-Sasson [6] and Razborov [42]. The latter paper also contains a suggestion that the "right" level of precision when comparing measures of this kind are up to polynomial and $\log n$ factors;[1] we will henceforth call two measures *equivalent* if they simulate each other in this sense. The paper [42] identified a big cluster of ordinary and regularized space complexity measures, including total and variable space, that are all equivalent to proof depth in resolution. One notable measure that defied this classification was (regularized) clause space.

We identify two other big clusters of equivalent complexity measures not covered by the results in [42]. The cumulative picture combining both previously known and new results is summarized in Figure 1. There, arrows are to be interpreted as

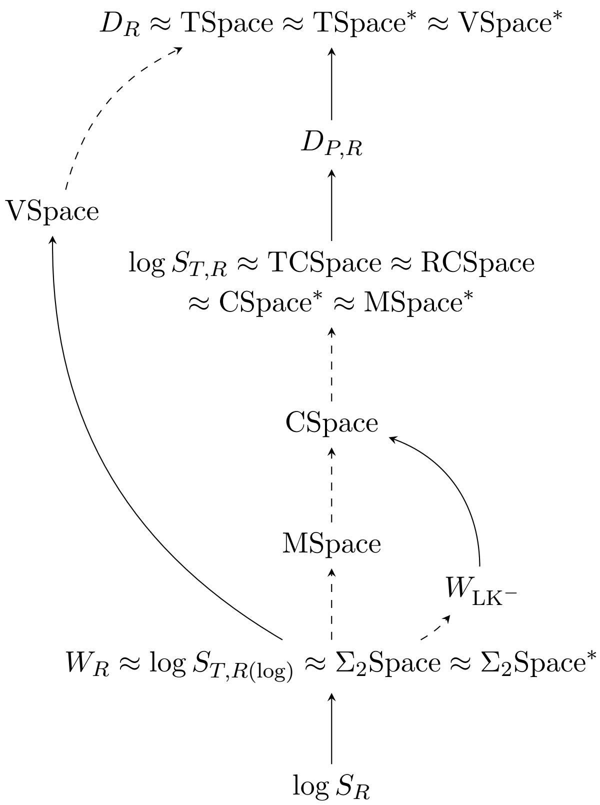

Figure 1: Simulations

inequalities, and $\approx$ as equality, both up to polynomial and $\log n$ factors. A solid arrow from μ_1 to μ_2 indicates that a separation between μ_1 and μ_2 is known, that is, it additionally indicates that there exists a sequence $\{F_n\}$ of unsatisfiable CNFs such that $\mu_2(F_n \vdash \bot) \geq (\mu_1(F_n \vdash \bot) + \log n)^{\omega(1)}$. To improve readability, we have omitted from Figure 1 the argument $F \vdash \bot$. We have refrained from including measures for cut-free sequent calculus apart from the cut free sequent calculus width, denoted by W_{LK^-}, which is directly related to measures for resolution.

But let us explain this picture in more detail. The first new cluster is centered around the logarithm of tree-like resolution size. Given the proof method of the

[1] Note that the size/length measures appear in this set-up under a logarithm. Hence this corresponds to *quasi*-polynomial simulations in the Cook-Reckhow framework.

simulation (1.5) in [22], it can be obviously strengthened in two directions: by replacing the left-hand side with clause space in *tree-like* resolution or by replacing it with *regularized* clause space. Tree-like clause space in resolution was shown to be equivalent to the logarithm of tree-like size in the same paper [22, Corollary 5.1]; in other words, after this replacement in the left-hand side, the bound (1.5) becomes tight, within the precision we are tolerating.

We show that the second variant, that is *regularized* clause space, is also equivalent to the logarithm of tree-like resolution size, and this result extends to also include regularized *monomial* space to the same cluster. Given that [22, Corollary 5.1] also holds for (ordinary) clause space in *regular* resolution [22, Corollary 4.2], this means that all these space measures turn out to be equivalent to each other and to the log of tree-like resolution size. We also remark (given the results above, this readily follows from definitions) that *regularized* versions of the clause space in tree-like or regular resolution are also in this cluster.

The question of whether (ordinary) clause space also belongs here is what we consider to be a major, and most likely very difficult, open problem. But since it has turned out to be closely related to several other threads in proof complexity, we prefer to keep the momentum and defer further discussion to the concluding Section 5.

Our second cluster is presided by resolution width. First, we introduce a natural analogue of clause space in DNF resolution or depth 2 Frege that we call Σ_2 *space*. This can be seen as an extension of clause space to depth 2 systems; indeed, the restriction of Σ_2 space to depth 1 sequent calculus is precisely clause space, and its restriction to k-DNF resolution, for constant k, coincides, up to a constant factor, with the concept of space that has been studied before for such systems (see e.g. [21, 10]). In our model, configurations are *arbitrary* sets of DNFs, and we charge k for every individual k-DNF in the memory. Clearly, $\Sigma_2\text{Space} \leq \text{CSpace}$ and $\Sigma_2\text{Space}^* \leq \text{CSpace}^*$. Then we strengthen the Atserias-Dalmau bound (1.3) by replacing CSpace with Σ_2Space and continue to show that *both* ordinary and regularized versions of Σ_2 space are actually equivalent to resolution width. Thus, remarkably, the difficult open question on whether we have a strong trade-off between space and length for clause space gets a relatively easy negative solution for a stronger proof system. We have also been able to locate in this cluster another interesting size measure: the size of tree-like proofs in the system $R(\log)$, which gives a somewhat unexpected generalization of (1.1). We have not been able to retrieve the equivalence of width and tree-like size in $R(\log)$ from the literature in exactly this form but it is implicit in Lauria [33] and, with a bit of effort, can be traced back as far as Krajíček [31].

It is worth noting that some of the simulations in this cluster work only in the syntactical setting. This comes in contrast with what happens with the other two clusters: all simulations involving clause, monomial, variable and total space, also work in a purely semantic setting. For example, in case of monomial space we can allow arbitrary Boolean functions of monomials as memory configurations and allow any number of sound inferences to be performed at once in each step.

Finally, let us briefly summarize what is known (to the best of our knowledge) in terms of separating the measures in Figure 1 and indicate the new results we show in that regard. Let us start with "true" separations, i.e. separations that work modulo polynomial overheads and $\log n$ factors. From now on, for proof complexity measures μ_1, μ_2 we will use the notation $\mu_1 \preceq \mu_2$ to stand for $\mu_1(F \vdash \bot) \leq (\mu_2(F \vdash \bot) \log n)^{O(1)}$ for any CNF F in n variables; $\mu_1 \approx \mu_2$ is the same as $\mu_1 \preceq \mu_2 \wedge \mu_2 \preceq \mu_1$. Clearly $\preceq$ is transitive, and this implies that $\approx$ is an equivalence relation and $\preceq$ imposes a partial order on its equivalence classes.

Bonet and Galesi [16] prove that $W_R \not\preceq \log S_R$. More precisely, there are constant width formulas F of size $O(n^3)$ such that $S_R(F \vdash \bot) \leq O(n^3)$ and $W_R(F \vdash \bot) \geq \Omega(n)$. Ben-Sasson [6] prove that VSpace $\not\preceq$ CSpace, and after negating the variables in his formulas, this works two more levels up on Figure 1. Namely, there are constant-width formulas F of size $O(n)$ such that $\text{VSpace}(F \vdash \bot) \geq \Omega(n/\log n)$ while $D_{P,R}(F \vdash \bot) \leq O(1)$. $D_{P,R}$ stands for *positive depth*, and it is a one-sided version of resolution depth D_R. This also provides a separation between $D_{P,R}$ and D_R that, though, is much easier to prove directly [48, Theorem 4.6]. Without negating the variables, it is easy to see that Ben-Sasson's proof actually gives $D_{P,R}(F \vdash \bot) \geq \Omega(n/\log n)$, thus separating $D_{P,R}$ from $\log S_{T,R}$ and hence from the whole middle cluster. Ben-Sasson, Håstad and Nordström [36, 9, 10] separate clause space from width; while it is believed that their formulas should also have large monomial space complexity, the questions of separating clause space from monomial space, as well as monomial space from width are widely open. Using their formulas (or rather a small variation of them) however, and the fact that MSpace^* belongs to the same cluster as $\log S_{T,R}$, we are able to get the following:

1. *There are unsatisfiable CNFs F of size $O(n)$ with $S_R(F \vdash \bot) \leq O(n)$, $W_R(F \vdash \bot) \leq O(1)$ and* $\text{MSpace}^*(F \vdash \bot) \geq \Omega(n/\log n)$ (Theorem 4.5).

This improves the previously known bounds $\text{MSpace}^*(F \vdash \bot) \geq \Omega(n^{2/11})$ [5], $\text{MSpace}^*(F \vdash \bot) \geq \Omega(n^{1/4})$ [29] and $\text{MSpace}^*(F \vdash \bot) \geq n^{1/2}/(\log n)^{O(1)}$ [28]. Unlike these previous results, our proof is remarkably simple.

Separating space complexity measures from their own regularized versions appear to be a very daunting task in general. As follows from Figure 1, for variable space this is equivalent to separating it from depth [48]. A quadratic separation between VSpace and VSpace* was proved in [42, Section 6], with a disappointingly elaborate proof. Nothing is known in terms of separating CSpace from (the cluster of) CSpace*. We prove that

2. *there are unsatisfiable CNFs F of size $O(n)$ with* $\text{CSpace}(F \vdash \bot) = 4$ *and* $S_{T,R}(F \vdash \bot) \geq \Omega(n^2/\log n)$ (Theorem 4.8)

making some progress, admittedly rather modest, in that direction. It is for this proof that we need and introduce the entry $D_{P,R}$ on Figure 1. We also remark that the space bound in this result is optimal. More precisely, we make a relatively simple observation (Theorem 4.6) that $\text{CSpace}(F \vdash \bot) \leq 3$ if and only if F is "essentially Horn" in which case it will possess a linear size tree-like resolution refutation.

Nothing seems to be known for CSpace vs. MSpace, and our structural picture provides a good heuristic explanation of the difficulty of this question: a separation between the two would also separate MSpace from MSpace*. Specifically, not even a separation between resolution width and monomial space is known; the techniques of [9, 10] in particular fail to generalize to the case of monomial space. We are able to show a quadratic separation between the two. Namely, we show that

3. *there are unsatisfiable CNFs F of size $O(n^n)$ with $W_R(F \vdash \bot) = O(n)$ and $MSpace(F \vdash \bot) = \Omega(n^2)$ (Theorem 3.2).*

This result is obtained by adapting our separation between resolution width and cut-free sequent calculus width to notions of width tailored for monomial space lower bounds [14, 15].

The paper is organized as follows. After expounding the necessary notions, definitions in Section 2, we first present our super-polynomial separation between resolution and LK^- in Section 3, and prove item 3 above. Then we begin Section 4 with refining (many simulations do not actually involve a polynomial overhead or extra $\log n$ factors) and proving the relations of Figure 1. In Section 4.4 we prove items 1 and 2 above. The paper is concluded with a few remarks and open problems in Section 5.

2 Basic notions

2.1 Sequent calculus

The sequent calculus was introduced by Gentzen [27] to formulate and prove his famous cut-elimination theorem. Many authors describe it as the most elegant proof system, and indeed, it illustrates the symmetries of logic at the level of syntax, like no other system.

Sequent calculus's version for classical logic is often denoted by LK. We shall use LK to denote its propositional part. LK operates with *sequents*. A sequent is a tuple of the form (Γ, Δ), where Γ and Δ are finite sets of formulas. Traditionally, a sequent (Γ, Δ) is written as $\Gamma \to \Delta$. This is to remind us its semantic interpretation: $\Gamma \to \Delta$ is to be interpreted as "if all formulas in Γ are true, then at least one formula in Δ is true". Note that either of the sets Γ, Δ can be empty. Then $\Gamma \to$ means that not all formulas in Γ are true, $\to \Delta$ means that at least one of the formulas in Δ is true, and $\to$ (i.e. both Γ and Δ are empty) is a tautologically false statement.

Let us present the rules of the system. In what follows, A, B represent arbitrary formulas, and $\Gamma, \Delta, \Gamma', \Delta'$ represent finite sets of formulas. Sets are written in a quite plain manner: We write Γ, A instead of $\Gamma \cup \{A\}$, A instead of $\{A\}$, A, B instead of $\{A, B\}$ and so on. The *axioms* of LK are all sequents of the form $A \to A$. Of the inference rules, first we have a rule, which allows us to add formulas to the left or right part of a sequent. This rule is called the *thinning* or *weakening* rule and has the form

$$\frac{\Gamma \to \Delta}{\Gamma' \to \Delta'},$$

where $\Gamma \subseteq \Gamma'$ and $\Delta \subseteq \Delta'$. Next, we have rules for each connective. These come in pairs; a connective is treated differently according to which side of its sequent it appears. The rules for the connectives $\wedge$, $\vee$ and $\neg$ are shown in Table 1. These

$$\neg\mathrm{L}: \frac{\Gamma, \neg A \to \Delta, A}{\Gamma, \neg A \to \Delta} \qquad \neg\mathrm{R}: \frac{\Gamma, A \to \Delta, \neg A}{\Gamma \to \Delta, \neg A}$$

$$\wedge\mathrm{L}_1: \frac{\Gamma, A \wedge B, A \to \Delta}{\Gamma, A \wedge B \to \Delta} \qquad \wedge\mathrm{L}_2: \frac{\Gamma, A \wedge B, B \to \Delta}{\Gamma, A \wedge B \to \Delta} \qquad \wedge\mathrm{R}: \frac{\Gamma \to \Delta, A \wedge B, A \qquad \Gamma \to \Delta, A \wedge B, B}{\Gamma \to \Delta, A \wedge B}$$

$$\vee\mathrm{R}_1: \frac{\Gamma \to \Delta, A \vee B, A}{\Gamma \to \Delta, A \vee B} \qquad \vee\mathrm{R}_2: \frac{\Gamma \to \Delta, A \vee B, B}{\Gamma \to \Delta, A \vee B} \qquad \vee\mathrm{L}: \frac{\Gamma, A \vee B, A \to \Delta \qquad \Gamma, A \vee B, B \to \Delta}{\Gamma, A \vee B \to \Delta}$$

Table 1: The analytic LK rules

rules are called *analytic*, and they already form a complete proof system for proving tautologies; we shall call this system *cut-free* LK or LK^-. Finally, there is the *cut* *rule*:

$$\frac{\Gamma, A \to \Delta \qquad \Gamma \to \Delta, A}{\Gamma \to \Delta}. \tag{2.1}$$

So, already having a proof of, say $\to B$, we may use it to prove $\to A$: $\to A$ can be derived from $B \to A$ and $\to B$ via the cut rule, and now to prove $\to A$, we need to prove the weaker formula $B \to A$. B can be anything. It doesn't need to have any intuitive relation to A, but even as such, it might be the case that a proof of $B \to A$ is much shorter than a proof of $\to A$. Gentzen's cut elimination theorem says that there is always an effective procedure of eliminating all applications of the cut rule from a proof, making it purely analytic. We refer to the formula A in applications of rule (2.1) as the formula being cut, or as the *cut formula*.

An LK proof of a sequent σ is a derivation of σ starting with the axioms and applying the rules of LK. More formally, it is a sequence consisting of sequents that ends with σ, in which every sequent is either an axiom, or results from previous sequents by one of the LK rules. We may view proofs as DAGs, by drawing edges from premises to conclusions in applications of the inference rules. If the DAG corresponding to a proof is a tree, we shall refer to the proof as being *tree-like*.

We write $\vdash \sigma$ if there is a proof of σ, indicating if necessary the underlying proof system with a subscript to $\vdash$. For example, $\vdash_{\mathrm{LK}} \sigma$ means that there is an LK proof of σ and $\vdash_{\mathrm{LK}^-} \sigma$ means that there is a cut-free LK proof of σ. For a set of formulas S, we write $S \vdash \sigma$ if there is a derivation of σ from S, that is a sequence consisting of sequents that ends with σ, in which every sequent is either an axiom, or is of the form $\to A$ for $A \in S$, or results from previous sequents by one of the LK rules. It should be noted that whereas it is true that

$$\vdash_{\mathrm{LK}} A_1, \dots, A_k \to B \iff A_1, \dots, A_k \vdash_{\mathrm{LK}} \to B,$$

it is not true that

$$\vdash_{\mathrm{LK}^-} A_1, \dots, A_k \to B \implies A_1, \dots, A_k \vdash_{\mathrm{LK}^-} \to B.$$

There is no way, without the cut rule, to combine proofs of the A_i's, in order to get a proof of B.

2.2 Sequent calculus as a satisfiability algorithm

It will be particularly convenient to consider the following view of LK. Following Smullyan [45], let us write a sequent $A_1, \dots, A_k \to B_1, \dots, B_\ell$ as

$$T\,A_1, \dots, T\,A_k, F\,B_1, \dots, F\,B_\ell.$$

That is, we annotate the formulas appearing on the left side of a sequent by T, the formulas appearining on its right side by F, and conjoin the two sides to form a single set. T and F stand for true and false respectively — $T\,A$ should be thought of as asserting that A is true and $F\,A$ as asserting that A is false.

Annotated formulas that are not annotated variables are naturally divided into two groups: those of a conjunctive and those of a disjunctive type. Formulas of the form $T\,A \wedge B$, $F\,A \vee B$, $T\,\neg A$ or $F\,\neg A$ belong to the former group, and those of the form $T\,A \vee B$ or $F\,A \wedge B$ to the latter. We use the letter "α" to stand for an arbitrary annotated formula of conjunctive type, and the letter "β" to stand for an arbitrary annotated formula of disjunctive type. We define the components α_i of a formula α and the components β_i of a formula β as shown in Table 2.

α	α_1	α_2	β	β_1	β_2
$T\,A \wedge B$	$T\,A$	$T\,B$	$F\,A \wedge B$	$F\,A$	$F\,B$
$F\,A \vee B$	$F\,A$	$F\,B$	$T\,A \vee B$	$T\,A$	$T\,B$
$T\,\neg A$	$F\,A$				
$F\,\neg A$	$T\,A$				

Table 2: Smullyan's notation

These provisions allow on the one hand for an extremely concise description of the rules of Table 1; they can be written as:

$$\frac{S, \alpha, \alpha_1}{S, \alpha}, \quad \frac{S, \alpha, \alpha_2}{S, \alpha}, \quad \frac{S, \beta, \beta_1 \qquad S, \beta, \beta_2}{S, \beta}.$$

More importantly, they reveal an algorithmic interpretation of LK. An LK proof, seen from the top to the bottom, i.e. from the sequent $\sigma \coloneqq A_1, \ldots, A_k \to B_1, \ldots, B_\ell$ it is proving to the axioms, describes the execution of an algorithm that tries to find a truth assignment (or more generally a model) that falsifies σ. The algorithm begins with σ written as $T\,A_1, \ldots, T\,A_k, F\,B_1, \ldots, F\,B_\ell$, asserting that there is an assignment that makes all A_i true and all B_i false, or equivalently, an assignment that falsifies σ. Then it keeps expanding this set, by applying the LK rules in reverse, that is from the conclusion to the premises. This expansion takes the form of a tree (or a DAG if we identify nodes labelled by the same set). At any point, we may choose a leaf labelled by S, α and add to it a single child labelled by S, α, α_i, as any assignment satisfying α must also satisfy every α_i. Or we may choose a leaf labelled by S, β and add to it two children, one labelled by S, β, β_1 and the other by S, β, β_2, as any assignment satisfying β must either satisfy β_1 or β_2. The thinning rule allows the algorithm to forget information: We may add to a leaf labelled by S, a child labelled by a subset of S. Finally, the cut rule allows us to add to a leaf labelled by S, two children, one labelled by $T\,A$ and the other by $F\,A$ for any formula A, as every assignment must satisfy either A or $\neg A$. This may greatly facilitate the search procedure. If at any point a set of the form $T\,A, F\,A$ is reached, then the search process may terminate at that particular branch, as no assignment can set A to both true and false. Notice that the contradiction $T\,A, F\,A$ corresponds to the axiom $A \to A$. A tree (or DAG) constructed this way, every branch of which ends with a leaf labelled by a set of the form $T\,A, F\,A$, is an LK proof of σ.

A depth-first implementation of the algorithm described above is shown as Algorithm 1 below. Algorithm 1 is called on a sequent represented as a set of annotated formulas. It will return false if there is an LK proof of that sequent and true otherwise. The algorithm chooses at each recursive call non-deterministically what rule to apply and which formula to apply it to. As presented, line 1, corresponding to

Algorithm 1 The LK algorithm

procedure LK(S)
  if S contains both $T\,A$ and $F\,A$ for some formula A **then**
    return false
  if for every $\alpha \in S$, all $\alpha_i \in S$ and for every $\beta \in S$, there is a $\beta_i \in S$ **then**
    return true
  go to either **1**, **2** or **3**
  1. select an $S' \subseteq S$ and **return** LK(S')
  2. select an arbitrary formula A and **return** LK($S, T\,A$) **or** LK($S, F\,A$)
  3. select an $A \in S$
  if $A = \alpha$ **then**
    select a component α_i and **return** LK(S, α_i)
  if $A = \beta$ **then**
    return LK(S, β_1) **or** LK(S, β_k)

the thinning rule, is redundant. However, incorporating memoization, that is the ability to stop the search when a set S has already been encountered in a previous recursive call that has returned, effectively identifying nodes labelled by the same set, this line makes it possible to greatly prune the search for a falsifying assignment. In terms of proofs, DAG-like proofs may be shorter than tree-like proofs. The key point in analyzing the correctness of that algorithm (or equivalently the completeness of LK), is that when at the base case true is returned, we can create an assignment consistent with S by setting for each formula A, A to true if $T\,A \in S$, and false otherwise.

2.3 The width of sequent calculus proofs

We define the *width* of a sequent as the number of formulas it contains, and the width of a sequent calculus proof as the maximum of the widths of the sequents it contains. It is not hard to see that for any provable sequent S_0, there is an LK proof of S_0 of width a constant plus the width of S_0. The concept of the minimum width needed to prove a sequent becomes non-trivial only if we restrict the class of cut formulas we are allowed to use. We shall be mainly interested in the minimum width over all LK$^-$ proofs of S_0, which we denote by $W_{\mathrm{LK}^-}(\vdash S_0)$.

We are going to give a characterization of $W_{\mathrm{LK}^-}(\vdash S_0)$ in terms of the definition below. In what follows, sequents are viewed as in the above section, viz. as sets of annotated formulas.

Definition 2.1. Following the terminology of [45], let us call a sequent S_0 *analytically k-consistent* if there is a set of sequents $\mathcal{S}$ containing S_0 and such that for each $S \in \mathcal{S}$:

1. for any formula A, S does not contain both $T\,A$ and $F\,A$;

2. $S' \subseteq S \implies S' \in \mathcal{S}$;

3. $|S| < k \;\&\; \alpha \in S \implies S, \alpha_i \in \mathcal{S}$ for every component α_i of α;

4. $|S| < k$ & $\beta \in S \implies S, \beta_i \in \mathcal{S}$ for some component β_i of β.

If the following condition is also satisfied, then we call S_0 *synthetically k-consistent* with respect to the set $\mathcal{C}$:

5. $|S| < k \implies S, T\ A \in \mathcal{S}$ or $S, F\ A \in \mathcal{S}$ for any formula $A \in \mathcal{C}$.

It is often helpful to see definitions such as the above, as describing a strategy for the adversary, in a game between a prover and an adversary played on a formula/sequent/set of formulas. In this case, the game is as follows: The configurations of the game are sequents. The initial configuration is S_0. In every round, the prover either deletes some formulas in the current sequent S, or selects an α-formula in S and adds a component of it to S, or selects a β formula, in which case the adversary adds a component of it to S. Allowing condition 5, the prover may choose an arbitrary formula $A \in \mathcal{C}$ and the adversary must respond by adding either $T\ A$ or $F\ A$ to S. The game ends with prover winning once S contains $T\ A$ and $F\ A$ for some formula A. The prover can always win provided that S_0 is provable. The question is: given a bound k, can she win always maintaining that the size of S is at most k? Definition 2.1 describes a strategy for the adversary, permitting the prover from winning when she maintains that bound.

Theorem 2.1. *Suppose that $|S_0| \leq k$. Then S_0 is analytically k-consistent if and only if $W_{\mathrm{LK}^-}(\vdash S_0) > k$. It is synthetically k-consistent with respect to $\mathcal{C}$ if and only if every LK proof of S_0 in which every cut formula belongs to $\mathcal{C}$ has width more than k.*

Proof. Let us only show the former sentence. Suppose first that S_0 is analytically k-consistent, and let $\mathcal{S}$ be the set of sequents witnessing this. We will show that in every tree-like LK$^-$ derivation (not necessarily beginning with axioms) τ of S_0, of width at most k, there is an initial sequent (i.e. one appearing as a leaf) in $\mathcal{S}$. From the first condition of Definition 2.1 that sequent is not an axiom, thus τ is not a proof. It is enough to show this for tree-like derivations, since a DAG-like derivation can be transformed into a tree-like one without increasing the width.

Base case. If τ contains just S_0, then we are done since $S_0 \in \mathcal{S}$.

Inductive step. Take some initial sequents $S_1, \ldots, S_r$ from which a sequent S is derived via an inference rule ρ, and remove them to get the derivation τ'. From the induction hypothesis, there is an initial sequent in τ' that belongs to $\mathcal{S}$. If that sequent is not S, then it also appears in τ and we are done. Otherwise, we have the following cases according to what rule ρ is:

Case 1. If it is the weakening rule, and thus $r = 1$ and $S_1 \subseteq S$, then from the second condition of Definition 2.1, $S_1 \in \mathcal{S}$.

Case 2. If ρ is the α-rule, and thus $r = 1$, $\alpha \in S$ and $S_1 = S, \alpha_1$ for some α_i, then since τ has width at most k, $|S| < k$, and hence from the third condition of Definition 2.1, $S_1 \in \mathcal{S}$.

Case 3. If ρ is the β-rule, and thus $\beta \in S$ and each S_i is of the form S, β_i, then again $|S| < k$, and from the fourth condition of Definition 2.1, some S_i belongs to $\mathcal{S}$.

Now suppose that $W_{\mathrm{LK}^-}(\vdash S_0) > k$. Set

$$\mathcal{S} := \{S \mid |S| \leq k \ \ \& \ \ W_{\mathrm{LK}^-}(\vdash S) > k\}.$$

Clearly $S_0 \in \mathcal{S}$. We will show that $\mathcal{S}$ satisfies the conditions 1–4 of Definition 2.1. For each $S \in \mathcal{S}$, first S cannot contain $T\,A$ and $F\,A$ for some A. This is so, because such a sequent is a weakening of an axiom, and having size at most k, it has a proof of width at most k. For the closure under subsets, if $S' \subseteq S$, then $W_{\mathrm{LK}^-}(\vdash S') > k$, for otherwise $W_{\mathrm{LK}^-}(\vdash S) \leq k$ since S follows from S' via the weakening rule. For the α condition, if $\alpha \in S$ and $|S| < k$, then for each α_i it must be that $S, \alpha_i \in \mathcal{S}$, for otherwise $W_{\mathrm{LK}^-}(\vdash S) \leq k$ since S follows from S, α_i via the α-rule. Finally, if $\beta \in S$ and $|S| < k$, then there must be a β_i such that $S, \beta_i \in \mathcal{S}$, otherwise $W_{\mathrm{LK}^-}(\vdash S) \leq k$, since S follows from all S, β_i via the β-rule. □

2.4 LK$^-$ for refuting CNF formulas, resolution and depth 2 Frege

A *literal* is a propositional variable x, or the negation of propositional variable $\neg x$. We let $\overline{x} \stackrel{\text{def}}{=} \neg x$ and $\overline{\neg x} \stackrel{\text{def}}{=} x$. A *clause* is a disjunction of literals, and a *term* is a conjunction of literals . A *CNF formula* is a conjunction of clauses and a *DNF formula* is a disjunction of terms. The *width*, $W(F)$, of a CNF or DNF formula F is the number of literals in the largest clause or term respectively of F.

Refuting a CNF formula $F = C_1 \wedge \cdots \wedge C_m$ means proving that the clauses C_i cannot be simultanesouly satisfied, that is, it means proving $C_1, \ldots, C_m \to$. LK$^-$ for proving such sequents has the following form. Of the rules in Table 1, the only one that is relevant is $\vee$L, which now, seeing clauses as disjunctions of unbounded arity, has as many premises as the number of literals in the clause it is deriving. Moreover, in such a proof, there is no reason to always carry the clauses C_i in sequents. We may as well delete them from every sequent, but keep in our mind that they are implicitly there. What remains are sequents of the form $\ell_1, \ldots, \ell_r \to$, where the ℓ_i's are literals, and such sequents are nothing other than clauses. To be explicit, the axioms of the resulting system are clauses of the form $x \vee \neg x$, and the inference rules are the weakening rule

$$\frac{C}{C \vee D}$$

and

$$\frac{C \vee \overline{\ell_1} \quad \cdots \quad C \vee \overline{\ell_r}}{C}, \tag{2.2}$$

where C and D are clauses and $\ell_1 \vee \cdots \vee \ell_r$ is a clause of the formula we are refuting. A proof of $C_1, \ldots, C_m \to$, in other words a refutation of $F = C_1 \wedge \cdots \wedge C_m$, in LK$^-$, is a derivation of the empty clause using the above rules. The *size* of such a refutation is the number of clauses it contains, and its width is the size of the largest clause occurring in it. We shall denote by $S_{\mathrm{LK}^-}(\vdash F \to)$ and $W_{\mathrm{LK}^-}(\vdash F \to)$ and the minimum size and the minimum width respectively over all LK$^-$ refutations of F.

Resolution is the system we get by adding to the above system the cut rule (2.1), where the cut formula A is restricted to be a propositional variable. Usually the thinning rule is incorporated into this rule, writing it as

$$\frac{C \vee x \qquad D \vee \neg x}{C \vee D}. \tag{2.3}$$

We may make a resolution proof "cut-only", by pushing all applications of the rule (2.2) at the bottom levels. Namely, we can simulate rule (2.2) by (2.3) as follows: Start with $\ell_1 \vee \cdots \vee \ell_r$, derive from it and $C \vee \overline{\ell_1}$, $C \vee \ell_2 \vee \cdots \vee \ell_r$, then derive from $C \vee \ell_2 \vee \cdots \vee \ell_r$ and $C \vee \overline{\ell_2}$, $C \vee \ell_3 \vee \cdots \vee \ell_r$, and so on, until C is derived (see Figure 2). Now the leaves contain clauses of F and these can

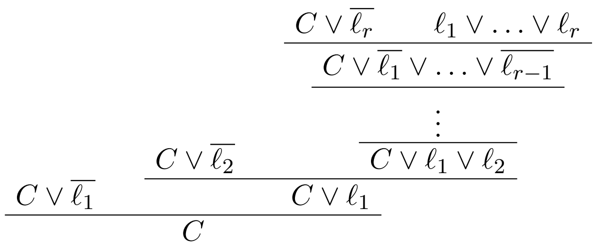

Figure 2: A simulation of (2.2) by the resolution rule

be derived from axioms by (2.2). Deleting all axioms from such a proof we get the usual presentation of resolution, deriving not $C_1, \ldots, C_m \rightarrow$, but the empty sequent (or clause) from $C_1, \ldots, C_m$: A *resolution refutation* of a CNF formula F is a derivation of the empty clause from the clauses of F, using only the rule (2.3). As before, the width of such a derivation is the number of literals in the largest clause occurring in it, and its size is the number of clauses it contains. Its *depth* is the length of the longest root-to-leaf path in its underlying DAG, and its *positive depth* the maximum number of *negative* literals introduced along a root-to-leaf path. We shall denote by $W_R(F \vdash \bot)$, $S_R(F \vdash \bot)$, $D_R(F \vdash \bot)$ and $D_{P,R}(F \vdash \bot)$ the minimum width, minimum size, minimum depth and minimum positive depth respectively, over all resolution refutations of F. It will be convenient to count the size of a tree-like resolution derivation $F \vdash \bot$ as the number of the leaves in the derivation instead of the number of all the nodes. This can only decrease size by a factor of 2. We denote by $S_{T,R}(F \vdash \bot)$ the minimum size, over all tree-like resolution refutations of F.

Notice simulating all applications of (2.2) with (2.3), going from $\vdash C_1, \ldots, C_m \rightarrow$ to $C_1, \ldots, C_m \vdash \bot$, the size of the proof is increased by at most a factor of 2, and width is increased by an additive factor of $W(F)$. That is, we have

$$S_R(F \vdash \bot) \leq 2 S_{\mathrm{LK}^-}(\vdash F \rightarrow)$$

and

$$W_R(F \vdash \bot) \leq W_{\mathrm{LK}^-}(\vdash F \rightarrow) + W(F).$$

Allowing as cuts formulas of depth 1, that is conjunctions or disjunctions of unbounded arity consisting of literlas, we get a system that is known in the literature with several names: depth 1 Frege, depth 2 Frege, DNF resolution or LK_1. As was the case for resolution, for refuting CNF formulas, because we have the cut rule, $\vdash C_1, \dots, C_m \rightarrow$ is the same as $C_1, \dots, C_m \vdash \bot$. As we did for LK^- and resolution, for refuting CNF formulas, we may write sequents as disjunctions. The resulting system will operate on DNF formulas; it has as axioms all formulas $x \vee \neg x$ and its inference rules become:

$$\frac{G}{G \vee H}, \quad \frac{G \vee t_1 \qquad H \vee t_2}{G \vee H \vee (t_1 \wedge t_2)}, \quad \frac{G \vee t \qquad H \vee \bar{t}}{G \vee H}, \tag{2.4}$$

where G and H are DNF formulas and t, t_1, t_2 and $t_1 \wedge t_2$ are terms.

For a non-decreasing function $f : \mathbb{N} \rightarrow \mathbb{N}$, $R(f)$ is the subsystem of depth 2 Frege where each DNF in a proof of size s is required to have width at most $f(s)$. $R(k)$ for k a constant is usually denoted by $\mathrm{Res}(k)$ (thus, resolution is $\mathrm{Res}(1)$). $R(f)$ was first introduced in [32]. The system $R(\log)$ is also known as $\mathrm{LK}_{1/2}$. We will be interested in the minimum size over all (tree-like) derivations $F \vdash \bot$, which we, as usual, denote as $S_{R(f)}(F \vdash \bot)$ ($S_{T,R(f)}(F \vdash \bot)$ respectively).

2.5 Width and the space of proofs

Adapting Definition 2.1 for resolution we get the characterization of [4] for resolution width. Adapting it for LK^- restricted to refuting CNF formulas, we get the definition of dynamic satisfiability from [21]. Namely, let us call sets of literals that do not contain contradictory literals *partial assignments*. We think of the assignment, say $\{x, \neg y, z\}$, as making x true, y false and z true. A partial assignment satisfies a clause C, if it contains a literal of C. It falsifies C if it contains $\bar{\ell}$ for every ℓ in C. We get:

Definition 2.2 [21]**.** Let F be a CNF formula, and let k be a natural number. F is said to be *k-dynamically satisfiable* is there is a non-empty set $\mathcal{A}$ of partial assignments to its variables such that for every assignment $\alpha \in \mathcal{A}$,

1. if $\alpha' \subseteq \alpha$ then $\alpha' \in \mathcal{A}$;
2. if $|\alpha| < k$ and C is a clause of F, then there is an $\alpha' \supseteq \alpha$ in $\mathcal{A}$ that satisfies C.

Theorem 2.1 in particular, becomes:

Theorem 2.2. *A CNF formula F is k-dynamically satisfiable if and only if $W_{\mathrm{LK}^-}(\vdash F \rightarrow) > k$.*

In the game corresponding to Definition 2.2, prover chooses in each round a clause of F, and the adversary responds by choosing a literal in that clause, which adds to the current assignment. Again, the closure under subsets condition corresponds to the ability of the prover to delete at any round literals from the current assignment. The prover wins once the current assignment falsifies a clause of F.

We get a characterization of resolution width by having the prover selecting variables instead of clauses, and the adversary responding by giving values to them. More specifically, in every round the prover selects a variable x of F. Then the adversary selects either x or $\neg x$, and the prover updates the current assignment α by deleting (if she wants) literals and adding the choice of the adversary. Again the prover wins once α falsifies a clause of F. She can win always maintaining $|\alpha| < k$ if and only if $W_R(F \vdash \bot) \leq k$ [4].

Notice that, if $W(F)$ is small, the prover in the second game is more powerful. Namely, we have

$$W_R(F \vdash \bot) \leq W_{\mathrm{LK}^-}(\vdash F \to) + W(F) - 1.$$

We already saw this when we explained how the resolution rule can simulate (2.2). In terms of games, the argument goes as follows: When the prover in the first game selects a clause C, the prover in the second game can start selecting, one by one the variables of C. If the game does not end, then the current assignment satisfies C, and then the prover can delete literals to match the assignments in the two games.

Definition 2.2 was introduced in [21] as a tool for proving space lower bounds in resolution and $R(k)$. The following definition is from [22, 1]. A *memory configuration* in resolution, is a set of clauses. A resolution refutation of a CNF formula F, in configurational form, is a sequence $\mathcal{M}_1, \ldots, \mathcal{M}_t$ of configurations where $\mathcal{M}_1$ is empty, $\mathcal{M}_t$ contains the empty clause and for $i > 1$, $\mathcal{M}_i$ is obtained from $\mathcal{M}_{i-1}$ by one of the following rules:

Axiom download: $\mathcal{M}_i = \mathcal{M}_{i-1} \cup \{C\}$, where C is a clause of F.

Inference: $\mathcal{M}_i = \mathcal{M}_{i-1} \cup \{C\}$, where C is derived from clauses in $\mathcal{M}_{i-1}$ by the resolution rule.

Erasure: $\mathcal{M}_i \subseteq \mathcal{M}_{i-1}$.

The clause space of such a refutation is $\max_{1 \leq i \leq t} |\mathcal{M}_i|$. The clause space of a CNF formula F, denoted by $\mathrm{CSpace}(F \vdash \bot)$, is the minimum clause space, over all refutations, in configurational form, of F. We shall be interested also in the minimum space in subsystems of resolution: $\mathrm{TCSpace}(F \vdash \bot)$ stands for the minimum tree-like configurational resolution refutation of F, i.e. one in which once whenever a formula is used as a premise in an inference rule, it is immediately erased from the memory. $\mathrm{RCSpace}(F \vdash \bot)$ stands for the clause space in *regular* resolution, i.e. the subsystem of resolution where we require that a variable cannot be resolved more than once on any path in (the DAG resulting from the expansion of) the configurational proof π.

Theorem 2.3 [21]**.** *If F is k-dynamically satisfiable, then* $\mathrm{CSpace}(F \vdash \bot) \geq k$*.*

We thus have

$$W_R(F \vdash \bot) - W(F) + 1 \leq W_{\mathrm{LK}^-}(\vdash F \to) \leq \mathrm{CSpace}(F \vdash \bot). \tag{2.5}$$

It is shown in [9] that there are 6-CNF formulas F of size $O(n)$ such that $W_R(F \vdash \bot) = O(1)$ and $\mathrm{CSpace}(F \vdash \bot) = \Omega(n/\log n)$. It is easy to show that $W_{\mathrm{LK}^-}(\vdash F \to) = O(1)$, thus these formulas in fact provide a gap between $W_{\mathrm{LK}^-}(\vdash \to)$ and $\mathrm{CSpace}(F \vdash \bot)$. The question of whether there is a gap between $W_R(F \vdash \bot)$ and $W_{\mathrm{LK}^-}(\vdash F \to)$ has not been addressed, and it is what we will deal in the next section.

A generalization of clause space introduced in [1], is *monomial space*. While configurations in the case of clause space are sets of clauses, for monomial space, arbitrary linear combinations, over a field $\mathbb{F}$, of clauses are allowed as the contents of a configuration, where such a linear combination P is interpreted as the asserting that $P = 0$. As a matter of fact, all known lower bounds for monomial space even hold in the case where arbitrary Boolean functions of clauses are allowed. The term monomial space comes from the fact that this concept captures space in proof systems employing algebraic reasoning.

Namely, seeing clauses as monomials — a clause $\ell_1 \vee \cdots \vee \ell_r$ is seen as the monomial $\overline{\ell_1} \ldots \overline{\ell_r}$ — the question of whether a set of clauses over the variables $x_1, \ldots, x_n$ is unsatisfiable, becomes the question of whether the polynomial 1 belongs to the ideal generated by those clauses and the clauses $x_i^2 - x_i$ and $x_i + \overline{x_i} - 1$ in $\mathbb{F}[x_1, \ldots, x_n, \overline{x_1}, \ldots, \overline{x_n}]$. A systematic way of generating this ideal, in a space oriented model, is the following [1]. Configurations are sets of polynomials over $\mathbb{F}[x_1, \ldots, x_n, \overline{x_1}, \ldots, \overline{x_n}]$. A refutation of a CNF formula F, in configurational form, is a sequence $\mathcal{M}_1, \ldots, \mathcal{M}_t$ of configurations where $\mathcal{M}_1$ is empty, $\mathcal{M}_t$ contains the empty clause and for $i > 1$, $\mathcal{M}_i$ is obtained from $\mathcal{M}_{i-1}$ by one of the following rules:

Axiom download: $\mathcal{M}_i = \mathcal{M}_{i-1} \cup \{C\}$, where C is either a clause of F, $x_i^2 - x$, or $x_i + \overline{x_i} - 1$.

Inference: $\mathcal{M}_i = \mathcal{M}_{i-1} \cup \{P\}$, where P is either a linear combination of polynomials in $\mathcal{M}_{i-1}$ or a literal multiplied by some polynomial in $\mathcal{M}_{i-1}$.

Erasure: $\mathcal{M}_i \subseteq \mathcal{M}_{i-1}$.

The monomial space of such a refutation is the the maximum number of distinct monomials occurring in a configuration. The monomial space, $\mathrm{MSpace}(F \vdash \bot)$, of F, is the mimimum monomial space over all refutations of F.

Both clause and monomial space deal with clauses. Clause space intends to capture the notion space in resolution, and monomial space does the same for algebraic proof systems. We introduce a notion of space in depth 2 Frege system, which we call Σ_2 *space*. A configuration now is a set of DNF formulas, and the Σ_2 space of a configuration $\mathcal{M} = \{G_1, \ldots, G_s\}$ is defined as the sum of the widths in it: $\Sigma_2\mathrm{Space}(\mathcal{M}) \stackrel{\mathrm{def}}{=} W(G_1) + \ldots + W(G_s)$. As above, we see the a refutation from the clauses of a CNF formula F, as a sequence of configurations, where we can download either clauses of F, axioms $x \vee \neg x$, and infer new DNF formulas from the rules of depth 2 Frege (2.4), and define $\Sigma_2\mathrm{Space}(F \vdash \bot)$ as the mimimum Σ_2 space over all refutations of F.

3 Resolution and LK$^-$

We saw that resolution can polynomially simulate cut-free LK for refuting CNF formulas. The aim of this section is to show that the opposite simulation is impossible. We first show a width separation in the two systems, which we then exploit to show a super-polynomial size separation.

3.1 A quadratic gap between LK$^-$ and resolution width

Let $F = \bigwedge_{i=1}^{s} C_i$ and $G = \bigwedge_{i=1}^{t} D_i$ be unsatisfiable CNF formulas. We define

$$F \times G \stackrel{\text{def}}{=} \bigwedge_{i=1}^{s} \bigwedge_{j=1}^{t} (C_i \vee D_j).$$

$F \times G$ is the CNF expansion of the formula $F \vee G$, which is also unsatisfiable.

Remarkably, LK$^-$ width and resolution width exhibit a different behavior with respect to this construction. This disparity ultimately relies on the fact that the cut rule gives us the ability to combine given proofs into a more complicated proof.

On one hand, we have:

Lemma 3.1. *If F and G are over disjoint sets of variables, then*

$$W_{LK^-}(\vdash F \times G \rightarrow) \geq W_{LK^-}(\vdash F \rightarrow) + W_{LK^-}(\vdash G \rightarrow) - 1.$$

Proof. Suppose that F is k-dynamically satisfiable, G is ℓ-dynamically satisfiable, and let $\mathcal{A}$ and $\mathcal{B}$ respectively be sets witnessing this. We need to show that $F \times G$ is $(k+\ell)$-dynamically satisfiable, that is we need to find a set satisfying the conditions of Definition 2.2 for the parameter $k + \ell$. We claim that

$$\mathcal{C} \coloneqq \{\alpha \cup \beta \mid \alpha \in \mathcal{A} \ \ \& \ \ \beta \in \mathcal{B}\}$$

is such a set. Closure under subsets immediately follows from the fact that $\mathcal{A}$ and $\mathcal{B}$ are closed under subsets. For the second condition, suppose that $\gamma \in \mathcal{C}$, $|\gamma| < k+\ell$, and let $C_i \vee D_j$ be a clause of $F \times G$, where C_i is a clause of F and D_j is a clause of G. Since $\gamma \in \mathcal{C}$, there is an $\alpha \in \mathcal{A}$ and a $\beta \in \mathcal{B}$ such that $\gamma = \alpha \cup \beta$. Moreover, since $|\gamma| < k$ and F and G do not share variables, either $|\alpha| < k$ or $|\beta| < \ell$. In the first case there is an $\alpha' \supseteq \alpha$ in $\mathcal{A}$ satisfying C_i, and thus $\alpha' \cup \beta$ is an assignment in $\mathcal{C}$ satisfying $C_i \vee D_j$. In the second case there is a $\beta' \supseteq \beta$ in $\mathcal{B}$ satisfying D_j, and thus $\alpha \cup \beta'$ is an assignment in $\mathcal{C}$ satisfying $C_i \vee D_j$. □

For resolution on the other hand, we have:

Lemma 3.2. $W_R(F \times G \vdash \bot) \leq \max\{W_R(F \vdash \bot) + W(G), W_R(G \vdash \bot)\}$.

Proof. Let π and ρ be resolution refutations of F and G respectively, both of minimum width. Replacing every clause C in π with $C \vee D_i$ we get a resolution proof π_i of D_i from $F \times G$. π_i has width at most $W_R(F \vdash \bot) + W(F)$. Replacing then every clause D_i in ρ with π_i we get a resolution refutation of $F \times G$ with the stated width. □

Choosing an appropriate seed and iterating, we get our result.

Theorem 3.1. *There are CNF formulas G with n^2 variables, size $O(n)^n$, and such that $W_R(G \vdash \bot) = O(n)$ and $W_{\mathrm{LK}^-}(\vdash G \rightarrow) = \Omega(n^2)$.*

Proof. Let F be a formula with constant width, n variables, size $\Theta(n)$, and such that $W_R(F \vdash \bot) = \Theta(n)$. Such formulas exist from e.g. [11]. Consider the formula

$$F^n \coloneqq F_1 \times \cdots \times F_n,$$

where the F_i's are copies of F over mutually disjoint sets of variables. From Lemma 3.2, $W_R(F^n \vdash \bot) = O(n)$. On the other hand $W_{\mathrm{LK}^-}(\vdash F \rightarrow) = \Omega(n)$ from (2.5), and hence from Lemma 3.1, $W_{\mathrm{LK}^-}(\vdash F^n \rightarrow) = \Omega(n^2)$. □

3.2 Separating resolution width from monomial space

The gap shown in the previous section can be extended to stronger versions of dynamic satisfiability that have been used to show monomial space lower bounds, thus showing a gap between resolution width and monomial space. The configurations in those are not assignments anymore, but sets of assignments. They will not be arbitrary sets however; they will have a certain structure. Namely, we call a set H of assignments *admissible*, if it is of the form

$$H = H_1 \times \cdots \times H_r \stackrel{\text{def}}{=} \{\alpha_1 \cup \cdots \cup \alpha_r \mid \alpha_i \in H_i\},$$

where each H_i is a non-empty set of non-empty assignments, for any two assignments $\alpha_i \in H_i$ and $\alpha_j \in H_j$ for $i \neq j$, the domains of α_i and α_j do not intersect, and moreover, if an assignment $\alpha \in H_i$ gives the value ϵ to a variable x, then there is also an assignment $\alpha' \in H_i$ giving to x the value $1 - \epsilon$. The H_i's are called the *factors* of H; we write $\|H\|$ for their number. We write $H' \sqsubseteq H$ if every factor of H' is a factor of H.

Definition 3.1 [14, 15]**.** Let F be a CNF formula and let k be a natural number. We say that F is *k-extendible* if there is a non-empty set of admissible configurations $\mathcal{H}$ such that for each $H \in \mathcal{H}$,

1. if $H' \sqsubseteq H$, then $H' \in \mathcal{H}$;
2. if $\|H\| < k$ and C is a clause of F, then there is an $H' \sqsupseteq H$ in $\mathcal{H}$, such that every $\alpha \in H'$ satisfies C.

Theorem 3.2 [14, 15]**.** *If F is k-extendible, then* $\mathrm{MSpace}(F \vdash \bot) \geq \lfloor k/4 \rfloor$.

Lemma 3.1 with the same proof applies here as well.

Lemma 3.3. *Let F and G be CNF formulas over disjoint sets of variables. If F is k-extendible and G is ℓ-extendible, then $F \times G$ is $(k + \ell)$-extendible.*

Proof. Let $\mathcal{H}$ and $\mathcal{I}$ be sets of admissible configurations witnessing the k and ℓ-extendibility of F and G. Since F and G are over disjoint sets of variables, we may assume that the domains for any of two assignments in $\mathcal{H}$ and $\mathcal{I}$ do not intersect. Set

$$\mathcal{J} \coloneqq \{H \times I \mid H \in \mathcal{H} \;\;\&\;\; I \in \mathcal{I}\}.$$

Clearly, $\mathcal{J}$ is a set of admissible configurations. We claim that it satisfies the conditions of Definition 3.1 for the parameter $k+\ell$. Closure under $\sqsubseteq$ immediately follows from the fact that $\mathcal{H}$ and $\mathcal{I}$ are closed under $\sqsubseteq$. For the second condition, suppose that $J = H \times I \in \mathcal{J}$, $\|J\| < k+\ell$, and let $C_i \vee D_j$ be a clause of $F \times G$, where C_i is a clause of F and D_j is a clause of G. Since $\|J\| < k+\ell$, either $\|H\| < k$ or $\|I\| < \ell$. In the first case, there is an $H' \sqsupseteq H$ in $\mathcal{H}$ such that all assignments in H' satisfy C_i. Then $H' \times I$ is an admissible configuration in $\mathcal{J}$ such that all assignments in it satisfy $C_i \vee D_j$. The second case is analogous. □

Therefore, we get:

Theorem 3.3. *There are CNF formulas G with n^2 variables, size $O(n)^n$, and such that $W_R(G \vdash \bot) = O(n)$ and $\mathrm{MSpace}(G \vdash \bot) = \Omega(n^2)$.*

Proof. Again, let F be a CNF with constant width, n variables and size $\Theta(n)$, that is $\Omega(n)$-extendible. Such formulas exist, see [14, 24, 15, 12]. The formulas

$$F^n := F_1 \times \cdots \times F_n,$$

where the F_i's are copies of F over mutually disjoint sets of variables, have resolution width $O(n)$, and from Lemma 3.3 they are $\Omega(n^2)$-extendible, thus from Theorem 3.2 require $\Omega(n^2)$ monomial space. □

3.3 A super-polynomial separation between resolution and LK$^-$ size

Many of the relations in resolution involving width, can be as well stated for LK$^-$. In fact, they seem to be better suited for LK$^-$; there, the additive $W(F)$ factor that naturally comes with resolution width disappears. We have already seen that $W_{\mathrm{LK}^-}(\vdash F \to) \le \mathrm{CSpace}(F \vdash \bot)$, refining the relation between clause space and width of [4]. But let us give an alternative, constructive proof, here. For sets S and T of formulas, we write $S \models T$ if every total assignment satisfying every formula in S, also satisfies every formula in T.

Theorem 3.4. *For any unsatisfiable CNF formula F,*

$$W_{\mathrm{LK}^-}(\vdash F \to) \le \mathrm{CSpace}(F \vdash \bot).$$

Proof. Let $\mathcal{M}_1, \ldots, \mathcal{M}_t$ be a refutation of F, of clause space s. We shall construct a sequence $\mathbf{T}_1, \ldots, \mathbf{T}_t$ of trees, the vertices of which are labelled by sets of literals, such that for every set S labelling a leaf of $\mathbf{T}_i$, $S \models \mathcal{M}_i$ and $|S| \le |\mathcal{M}_i|$.

We set $\mathbf{T}_1$ to be a tree with one vertex labelled by the empty set. Now, suppose we have constructed $\mathbf{T}_{i-1}$. If $\mathcal{M}_i$ results from $\mathcal{M}_{i-1}$ via an inference step, we set $\mathbf{T}_i := \mathbf{T}_{i-1}$. If $\mathcal{M}_i \subseteq \mathcal{M}_{i-1}$, then we add to every leaf of $\mathbf{T}_{i-1}$ labelled by a satisfiable set S, a child labelled by a subset $S' \subseteq S$ such that $|S'| \le |\mathcal{M}_i|$ and $S' \models \mathcal{M}_i$. Finally, if $\mathcal{M}_i = \mathcal{M}_{i-1} \cup \{C\}$, for a clause $C = \ell_1 \vee \cdots \vee \ell_r$ of F, we add to every child of $\mathbf{T}_i$ labelled by a satisfiable set S, r children labelled by the sets $S \cup \{\ell_j\}$.

Replacing each set $\{\ell_1, \ldots, \ell_k\}$ occurring in $\mathbf{T}_t$, by the clause $\overline{\ell_1} \vee \cdots \vee \overline{\ell_k}$, we get an LK$^-$ refutation of F of width at most s. It is clear from the construction of $\mathbf{T}_t$ that every clause has width at most s and every clause not at a leaf, results from the clauses at its children via either the weakening or the $\vee$L rule. Moreover, since $\mathcal{M}_t$ is unsatisfiable, no set labelling a leaf of $\mathbf{T}_i$ is satisfiable, that is sets at the leaves become weakenings of axioms. $\square$

Let us note that Theorem 3.4 is generalized in [38] to show that $W_{\mathrm{LK}}(F \vdash \bot)$ is equal, up to a constant factor, to the minimum Σ_2 space, a natural analogue of clause space for depth 2 Frege systems, needed to refute F.

Next, we have the size-width relations of [11]. Here the proofs are the same as those in [11]. We shall need the following lemma, saying that LK$^-$ proofs are closed under taking restrictions. For a partial assignment α and an LK$^-$ refutation π of a CNF formula F, the restriction $\pi|_\alpha$ of π to α is obtained from π by deleting all clauses that become true by α, and deleting from all clauses the literals that become false by α.

Lemma 3.4. *$\pi|_\alpha$ is an LK$^-$ refutation of $F|_\alpha$.*

Lemma 3.4 also holds for resolution as long as we add the weakening rule [11].

Theorem 3.5. *For any unsatisfiable CNF formula F,*

$$W_{\mathrm{LK}^-}(\vdash F \to) \le \log S_{\mathrm{TR}}(F \vdash \bot) + 1.$$

Proof. This is in fact a weakened version of Theorem 3.4, as $\mathrm{CSpace}(F \vdash \bot) \le \log S_{\mathrm{TR}}(F \vdash \bot) + 1$ [22]. But let us give a direct construction instead. We shall construct, by induction on s, for every tree-like resolution refutation $\mathbf{T}$ of F of size s, an LK$^-$ refutation of F of width at most $\log s + 1$.

Base cases. If $\mathbf{T}$ has size 1, then it has width 0; if it has size 3 then it has width 2.

Inductive step. Suppose that $\mathbf{T}$ has size $s > 3$. Let $\mathbf{T}_1$ and $\mathbf{T}_2$ be the subproofs of $\mathbf{T}$, deriving $\neg x$ and x respectively for some variable x. One of $\mathbf{T}_1$ and $\mathbf{T}_2$, say $\mathbf{T}_1$, must have size at most $s/2$. $\mathbf{T}_1|_x$ is a refutation of $F|_x$ of size at most $s/2$, and $\mathbf{T}_2|_{\overline{x}}$ is a refutation of $F|_{\overline{x}}$ of size less than s. From the induction hypothesis, there are LK$^-$ refutations π_1 and π_2 of $F|_x$ and $F|_{\overline{x}}$ of width $\log s$ and $\log s + 1$ respectively. Start with π_2. To every application of the rule

$$\frac{C \vee \overline{\ell_1} \quad \cdots \quad C \vee \overline{\ell_r}}{C},$$

where $\ell_1 \vee \cdots \vee \ell_r \vee x$ is a clause of F, add the extra premise $C \vee x$. But x can be derived from π_1, and hence all those $C \vee x$ can be derived via the weakening rule from x: We can do the same to π_1, now adding $C \vee \overline{x}$ as the extra premise, and moreover add to every clause the variable x. The refutation obtained when we combine π_1 and π_2 is a valid LK$^-$ refutation of F of width at most $\log s + 1$. $\square$

Theorem 3.6. *For any unsatisfiable CNF formula F over n variables,*

$$W_{\mathrm{LK}^-}(\vdash F \to) = O\left(\sqrt{n \log S_{\mathrm{LK}^-}(\vdash F \to)}\right).$$

Proof. The construction will be the same as that of Theorem 3.5. The problem here is that it is not clear what variable to choose to recurse on. The trick is to choose the variable that appears more often in the clauses of the proof.

For an LK$^-$ refutation π of a CNF and a $d \geq 0$, let π^* be the set of clauses in π of width greater than d. We call the clauses in π^* the *fat* clauses of π. We show, by induction on n, that for any CNF F in n variables, any LK$^-$ refutation π of F and any integers $d, b \geq 0$,

$$|\pi^*| < a^b \implies W_{\mathrm{LK}^-}(\vdash F \rightarrow) \leq d + b,$$

where

$$a = \left(1 - \frac{d}{2n}\right)^{-1}.$$

The theorem follows taking π to be of minimum size and $d := \left\lceil \sqrt{n \log S(\pi)} \right\rceil$.

Base case. If $n = 1$, then there is an LK$^-$ refutation of F of width 2, and in the case that $b + d < 2$ the implication becomes trivially true.

Inductive step. Suppose that $n > 1$, let $d, b \geq 0$, and let π be an LK$^-$ refutation of F with $|\pi^*| < a^b$. If $b = 0$, then π itself is a refutation of width at most $d + b$. Suppose $b > 0$. There are $2n$ literals, so there must be some literal, say the variable x, appearing in at least $d|\pi^*|/(2n)$ clauses in π^*. $\pi|_x$ is an LK$^-$ refutation of $F|_x$ with at most

$$|\pi^*| \left(1 - \frac{d}{2n}\right) < a^{b-1}$$

fat clauses, so by the induction hypothesis (notice that a is a decreasing function of n), there is an LK$^-$ refutation π' of $F|_x$ of width at most $d+b-1$. Furthermore, $\pi|_{\overline{x}}$ is an LK$^-$ refutation of $F|_{\overline{x}}$ with less than a^b clauses, so by the induction hypothesis, there is an LK$^-$ refutation π'' of $F|_{\overline{x}}$ of width at most $d + b$. Combining π' and π'' as in the proof of Theorem 3.5, we get an LK$^-$ refutation of F of width at most $d + b$. $\square$

Notice that in Theorems 3.4 and 3.5, we have LK$^-$ in the left hand side and resolution in the right hand side. That is to say, cuts are eliminated when constructing the small width proofs. It is tempting to speculate on whether the same is also true for Theorem 3.6, that is whether we can replace $S_{\mathrm{LK}^-}(\vdash F \rightarrow)$ with $S_R(F \vdash \bot)$. After all, the only place where we need LK$^-$ in the right hand side is the case $b = 0$. Theorem 3.1 says that this cannot be true. In fact, the formulas of Theorem 3.1 give the main theorem of this section, which is:

Theorem 3.7. *There is a CNF formula F with n^2 variables and size $O(n)^n$, such that $S_R(F \vdash \bot) = O(n)^n$ but $S_{\mathrm{LK}^-}(\vdash F \rightarrow) \geq \exp(\Omega(n^2))$.*

Proof. The formulas of Theorem 3.1 are such. The upper bound $S_R(F \vdash \bot) = O(n)^n$ follows from the construction of Lemma 3.2. The lower bound follows from the fact that $W_{\mathrm{LK}^-}(\vdash F \rightarrow) = \Omega(n^2)$ and Theorem 3.6. Namely, Theorem 3.6 gives

$$S_{\mathrm{LK}^-}(\vdash F \rightarrow) \geq \exp\left(\Omega\left(\frac{(W_{\mathrm{LK}^-}(\vdash F \rightarrow))^2}{n^2}\right)\right) = \exp\left(\Omega\left(n^2\right)\right). \qquad \square$$

It is important to note that the $\exp\left(\Omega\left(n^2\right)\right)$ lower bound in Theorem 3.7 holds for the version of LK^- operating on clauses, where the clauses of the CNF formula F to be refuted are viewed as disjunctions of unbounded arity. It does not hold when the clauses of F are made up from binary disjunctions and moreover we are free to choose the order in which they are applied. If $\vee$ in the definition

$$F \times G \stackrel{\text{def}}{=} \bigwedge_{i=1}^{s} \bigwedge_{j=1}^{t} (C_i \vee D_j)$$

of $F \times G$, is seen as a binary disjunction, then having derived $C_1, \ldots, C_n \rightarrow$ and $D_1, \ldots, D_t \rightarrow$, it is easy to see that we may derive from these sequents in $s \cdot t$ steps, $C_1 \vee D_1, \ldots, C_1 \vee D_t, \ldots, C_s \vee D_1, \ldots, C_s \vee D_t \rightarrow$, and in this case F^n in Theorem 3.7 has an LK^- refutation of size $n^{O(n)}$. An analogous situation occurs between the tree-like versions of LK^- and resolution [3]. But let us notice, concluding, that with binary disjunctions, LK^- cannot be seen as a system operating on clauses, and it becomes rather unnatural to compare it with resolution — it is not even clear, in this case, whether resolution can polynomially simulate LK^-. LK^- for clauses consisting of binary disjunctions is closer to resolution with limited extension, in which case resolution does polynomially simulate it [46].

4 Space characterizations of complexity measures and size-space trade-offs

The aim of this is section to state and prove all new non-trivial simulations of Figure 1. We start with the simulations in the middle cluster.

4.1 Tree-like resolution size and regularized monomial space

We show that $\log S_{T,R}$, TCSpace, RCSpace, CSpace* and MSpace*, are all equal up to polynomial and $\log n$ factors. Recall that for a space measure μ on configurational proofs, we define its regularized version μ^* as $\mu^*(F \vdash \bot) \stackrel{\text{def}}{=} \min_\pi(\mu(\pi) \log |\pi|)$, where $|\pi|$ is the number of configurations in π. Our main new contribution is the following simulation. (The second item is included since it allows us to shave off the $\log n$ factor with little additional work, and this will give a slightly better bound in Theorem 4.5.)

Theorem 4.1. *For any unsatisfiable CNF formula F over n variables,*

$$\log S_{T,R}(F \vdash \bot) \leq 2\text{MSpace}^*(F \vdash \bot) \log(n+1),$$
$$\text{TCSpace}(F \vdash \bot) \leq 2\left(\text{MSpace}^*(F \vdash \bot) + 1\right).$$

Proof. The proof is analogous to the construction in [42] showing that depth is upper bounded by regularized variable space. Let $\mathcal{M}_1, \ldots, \mathcal{M}_t$ be a refutation of F in configurational form, of monomial space s. We show, by induction on d, that for every interval $[i..j] \subseteq [1..t]$ with $j > i$, $j - i \leq 2^d$, and for every clause D such that $\alpha_D \models \mathcal{M}_i$ and $\alpha_D \models \neg\mathcal{M}_j$, it holds that $S_{T,R}(F \vdash D) \leq (n+1)^{ds}$ and, moreover, the assumed tree-like resolution proof can be carried out in clause space at most $ds + 2$. The theorem follows by taking $[i..j] := [1..t]$, $d := \lceil \log t \rceil$ and $D := \bot$.

Base case. Suppose that $d = 0$, so that $j = i + 1$. The statement is vacuously true except when the step consists in downloading an axiom C from F, simply because in all other cases we have $\mathcal{M}_i \models \mathcal{M}_{i+1}$ and hence D with the specified properties does not even exist. Let D be a clause for which $\alpha_D \models \mathcal{M}_i$ and $\alpha_D \models \neg(\mathcal{M}_i \cup \{C\})$. Then we necessarily must have $\alpha_D \models \neg C$, which is equivalent to saying that D is a weakening of C.

Inductive step. Suppose that $d > 0$, let $[i..j] \subseteq [1..t]$ be any interval with $j - i \leq 2^d$, $j > i + 1$, and let D be a clause such that $\alpha_D \models \mathcal{M}_i$ and $\alpha_D \models \neg\mathcal{M}_j$. Set $k := i + \lceil (j-i)/2 \rceil$, so that $k - i \leq 2^{d-1}$ and $j - k \leq 2^{d-1}$. Let the list $m_1, \ldots, m_s$ contain all monomials occurring in $\mathcal{M}_k$. For a clause A and a monomial $m = \ell_1 \ldots \ell_r$, consider the tree shown in Figure 3 designating a derivation of A in resolution; it is obtained from the obvious decision tree deciding m by reversing edges and weakening the result by A. Denote this tree by $\mathbf{T}_{A;m}$.

Let us now describe the required tree-like resolution proof of D. Start with $\mathbf{T}_{D;m_1}$. To every leaf of $\mathbf{T}_{D;m_1}$ labelled by a clause D', append the tree $\mathbf{T}_{D';m_2}$. Continue this process for all $m_1, \ldots, m_s$. If at any point during this construction, a forbidden disjunction containing a variable and its negation occurs, then we delete

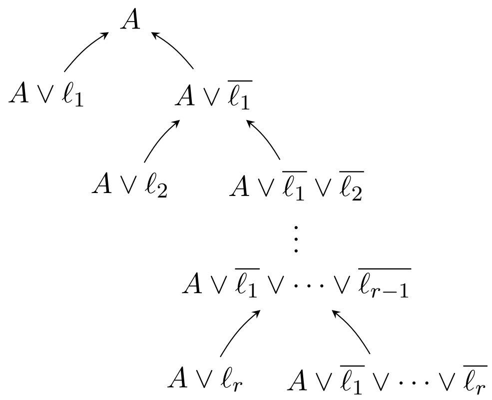

Figure 3: The tree $\mathbf{T}_{A;m}$

that node and contract at its parent. The resulting tree $\mathbf{T}$ has at most $(n+1)^s$ leaves, and each of its leaves is labelled by a clause E such that $\alpha_E \models \mathcal{M}_k$ or $\alpha_E \models \neg\mathcal{M}_k$. From the induction hypothesis, there are tree-like resolution proofs of all those clauses from F, of size $(n+1)^{(d-1)s}$. Therefore, there is a tree-like resolution proof of D from F of size $(n+1)^{ds}$.

To see that this proof can be carried out in clause space at most $ds+2$, notice that the proof designated by $\mathbf{T}$ can be carried out in clause space $s+2$. Proceed with this proof, and whenever a clause at its leaves is downloaded, keep all current clauses in memory (there are at most s of them — the maximum clause space is hit when the parent of two leaves is brought to memory), and derive it in clause space at most $(d-1)s+2$. The fact that such a derivation exists is guaranteed by the induction hypothesis. The resulting proof has clause space at most $s+(d-1)s+2 = ds+2$. $\square$

Remark 4.1. As we already remarked in the introduction, the above construction works for any *sound* system whose configurations are Boolean functions of monomials. In particular, it works for the purely semantic system called functional calculus [1]. Even more generally, it works for any proof system in which small configurations have low decision tree complexity.

For the rest of the relations, we claim that for an unsatisfiable CNF formula F over n variables, we have

$$\begin{aligned}
\mathrm{RCSpace}(F \vdash \bot) &\le \mathrm{TCSpace}(F \vdash \bot) \\
&\le \log S_{T,R}(F \vdash \bot) + 2 \\
&\le 2\,(\mathrm{MSpace}^*(F \vdash \bot)\log(n+1) + 1) \\
&\le 2\,(\mathrm{CSpace}^*(F \vdash \bot)\log(n+1) + 1) \\
&\le 2\,(\mathrm{RCSpace}^*(F \vdash \bot)\log(n+1) + 1) \\
&\le 2\left((\mathrm{RCSpace}(F \vdash \bot))^2 \log(n+1)\log(2n) + 1\right).
\end{aligned}$$

The first inequality follows from the observation that every tree-like refutation can be pruned to the regular form, and this operation does not increase its space. The second inequality is [22, Theorem 2.1], and the third is Theorem 4.1. The fourth and the fifth inequalities are obvious. Finally, the last inequality follows from [22, Corollary 4.2]. Namely, Esteban and Torán showed that if π is a resolution refutation, in configurational form, of clause space s and depth d, then

$$|\pi| \leq \binom{d+s}{s}. \tag{4.1}$$

Taking π to be a regular resolution refutation of minimum clause space, we get, since a regular refutation must have depth at most n,

$$\text{RCSpace}^*(F \vdash \bot) \leq (\text{RCSpace}(F \vdash \bot))^2 \log(2n).$$

As a byproduct, we get that TCSpace $\approx$ RCSpace. This comes in sharp contrast with the situation for size, where there is an exponential separation between tree-like and regular resolution [8].

We also see that, somewhat surprisingly, instead of regularizing clause space by multiplying it by the logarithm of *size*, we could have as well used a much weaker regularization by multiplying by the logarithm (!) of depth, and the resulting measure would still be in this cluster. This allows us to re-cast the main open problem of whether CSpace $\approx$ CSpace* in terms of the existence of a *super-critical* (in the sense of [41]) trade-off between clause space and depth.

The remaining (non-trivial) simulation on Figure 1 involving this cluster is:

Theorem 4.2. *For any unsatisfiable CNF formula F,*

$$\text{TCSpace}(F \vdash \bot) \leq D_{P,R}(F \vdash \bot) + 2.$$

Proof. The argument is a refinement of the argument in [22] showing that tree-like clause space is bounded by depth. We show, by induction on $\mathbf{T}$, that if $\mathbf{T}$ is a tree-like resolution proof of a clause E from F of positive depth d, then there is a tree-like resolution proof, in configurational form, of E from F of clause space at most $d+2$.

If $\mathbf{T}$ has size at most 2, then $d \leq 1$, and TCSpace$(F \vdash \bot) \leq 3$. Otherwise, let $\mathbf{T}_1$ and $\mathbf{T}_2$ be the subproofs of $\mathbf{T}$ proving the two clauses E_1 and E_2 respectively from which E is derived via an application of the resolution rule and possibly applications of the weakening rule. One of $\mathbf{T}_1$ and $\mathbf{T}_2$, say $\mathbf{T}_1$, must have positive depth at most $d-1$. From the induction hypothesis, there is a tree-like proof π_1 of E_1 of clause space at most $d+1$, and a tree-like proof π_2 of E_2 of clause space at most $d+2$. Deriving first E_2 using π_2, and then, keeping E_2 in memory, deriving E_1 using π_1, we get a proof of E of clause space at most $d+2$. □

4.2 Resolution width and Σ_2 space

The simulations for our second cluster will depend upon the following "locality" property of depth 2 Frege.

Lemma 4.1. *Let α be a partial assignment. For each of the inference rules of depth 2 Frege, if both premises contain a term satisfied by α, then α satisfies some term in the conclusion.*

The main theorem of this section says that as long as we transition from depth 1 Frege to depth 2 Frege, then not only width continues to be smaller than space, but in fact it becomes (almost) equal to it. As a historical remark, an extension of the Atserias-Dalmau bound (1.3) for the case of Res(k) is sketched in [25], and, although it is not stated explicitly, it is also apparent in [21].

Theorem 4.3. *For any unsatisfiable CNF formula F,*

$$\frac{1}{5}\Sigma_2\mathrm{Space}(F \vdash \bot) \leq W_R(\vdash F \rightarrow) \leq \Sigma_2\mathrm{Space}(F \vdash \bot).$$

Proof. Let $\mathcal{M}_1, \ldots, \mathcal{M}_t$ be a depth 2 Frege refutation of F, of Σ_2 space s. We will construct an increasing sequence $\mathbf{T}_1, \ldots, \mathbf{T}_t$ of derivations in the system "resolution plus the rule (2.2)". The property we are going to maintain is that for every clause D labelling a leaf of $\mathbf{T}_i$, either D is a weakening of a clause C in F (call such a leaf an *axiom leaf*) or the following holds:

1. for every $G \in \mathcal{M}_i$, α_D satisfies some term of G;
2. $W(D) \leq \Sigma_2\mathrm{Space}(\mathcal{M}_i)$.

$\mathbf{T}_1$ has one vertex labelled by the empty clause. Now suppose we have constructed $\mathbf{T}_{i-1}$ such that 1 and 2 hold for all non-axiom leaves. For every such leaf v labelled by a clause D, do the following.

Axiom Download: Suppose that $\mathcal{M}_i = \mathcal{M}_{i-1} \cup \{C\}$, where $C = \ell_1 \vee \cdots \vee \ell_r$ is either a clause of F (viewed as a 1-DNF) or a logical axiom $x \vee \overline{x}$. If C and D contain conflicting literals, then item 1 is automatically satisfied and we do nothing at this leaf. Next, $C \subseteq D$ would have implied that C is a clause of F which is impossible since we have assumed that the leaf is non-axiom. Thus, there exists at least one $j \in [r]$ such that $\ell_j \notin D$, and for any such j we add to v a child labelled by $D \vee \overline{\ell_j}$. This will be an application of the rule (2.2) if C is a clause or of the resolution rule if C is $x \vee \overline{x}$.

Erasure: Suppose that $\mathcal{M}_i \subseteq \mathcal{M}_{i-1}$. Then add to v a single child labelled by a clause $E \subseteq D$ such that $W(E) \leq \Sigma_2\mathrm{Space}(\mathcal{M}_i)$ and for every $G \in \mathcal{M}_i$, α_E satisfies some term of G.

Inference is immediately taken care of by Lemma 4.1, D does not change.

Since $\bot \in \mathcal{M}_t$, $\mathbf{T}_t$ may not contain any non-axiom leaves and hence defines a refutation. Also, it is clear from the construction and property 2 above that any clause D appearing in it must satisfy $W(D) \leq \max_{1 \leq i \leq t} \Sigma_2\mathrm{Space}(\mathcal{M}_i) = s$. Hence $W(\vdash_F \bot) \leq s$.

For the converse inequality, suppose that $C_1, \ldots, C_t$ is a refutation in the system "resolution plus rule' (2.2)", of width w. For every $i \in [t]$, set

$$G_i := \bigvee_{j=1}^{i} \overline{C_j}.$$

Each G_i is a w-DNF. For our small space refutation, we will first derive G_t and $G_{t-1} \vee C_t$, then cutting on C_t derive from these formulas G_{t-1}, then derive $G_{t-2} \vee C_{t-1}$, and continue this way until we get the empty clause. Notice that $G_{i-1} \vee C_i$ is either a tautology with an obvious derivation in depth 2 Frege, or C_i is a clause of F. In the latter case, we can immediately derive $G_{i-1} \vee C_i$. Otherwise, C_i will be the result of applying either the resolution rule or the weakening rule or rule (2.2) to some clauses among $C_1, \ldots, C_{i-1}$. In either case, it can be checked that $G_{i-1} \vee C_i$ has a tree-like proof of Σ_2 space at most $3w$, and therefore the proof above can be carried in Σ_2 space at most $5w$. □

Remark 4.2. The second part of this proof implies that *a posteriori*, DNF resolution will retain its power in terms of space even if we restrict the *formula space* (the maximum number of DNFs in a configuration) to a constant. This in turn immediately implies, also a posteriori, that we can balance our definition of Σ_2 space replacing in it $W(G_1) + \ldots + W(G_s)$ with $s \cdot \max(W(G_1), \ldots, W(G_s))$, and the resulting measure will still be equivalent to Σ_2 space. We are not aware of a direct proof of this simulation by-passing width.

Next, we show how to control the length of a proof, ruling out for DNF resolution strong length-space trade-offs conjectured for variable, clause and monomial space.

Corollary 4.1. *For any unsatisfiable CNF formula F with n variables,*

$$\Sigma_2\text{Space}^*(F \vdash \bot) \leq O\left((\Sigma_2\text{Space}(F \vdash \bot))^2 \log n\right).$$

Proof. Let $s := \Sigma_2\text{Space}(F \vdash \bot)$. By the first part of Theorem 4.3, F has a width $O(s)$ resolution refutation with the additional rule (2.2). We apply to this refutation the construction from the second part of Theorem 4.3 in which we can clearly assume $t \leq n^{O(s)}$ (since all clauses in the sequence can be assumed to be different). By an easy inspection, the length of the resulting refutation will still be $n^{O(s)}$. Therefore,

$$\Sigma_2\text{Space}^*(F \vdash \bot) \leq O(s^2 \log n).$$ □

The following is known for space in regular or tree-like resolution (see [23] or Section 4.1 above), is trivial for variable or total space and is wide open for clause and monomial space (again, see Section 5 for a more thorough discussion). Remarkably, Corollary 4.2 is about a measure that is *below* (smaller than) clause space unlike the known cases mentioned above.

Corollary 4.2. *If F has a constant Σ_2 space refutation, then it has a refutation of constant Σ_2 space and polynomial length.*

Proof. The refutation constructed in the proof of Corollary 4.1 will in our case also have constant Σ_2 space. □

Let us finally deal with the remaining measure, tree-like proofs in $R(\log)$. (Recall that an $R(\log)$ refutation of a formula F is a depth 2 Frege refutation of size s in which every term has width at most $\log s$.)

Theorem 4.4. *Let F be an unsatisfiable CNF formula over n variables. Then*

$$\Sigma_2\text{Space}(F \vdash \bot)^{1/2} \leq \log S_{T,R(\log)}(F \vdash \bot) \leq O(W_R(\vdash F \rightarrow) \log n).$$

Proof. For the upper bound, let π be a resolution refutation of F of width $w := W_R(\vdash F \rightarrow)$. Apply to it the construction in the second part of the proof of Theorem 4.3 once again. By inspection (cf. the proof of Corollary 4.1), this refutation is tree-like, has size $n^{O(w)}$ and every term occurring in it has width at most w. Padding it with dummy formulas if necessary, we can assume that it has size $\geq 2^w$ which makes it into a tree-like $R(\log)$ refutation of the required size.

For the lower bound, the argument is an adaptation of the argument in [22] showing (1.5). Namely, by pebbling, a tree-like proof $\mathbf{T}$ of size $s > 1$ can be turned into a proof in configurational form, where each configuration contains at most $\log s$ formulas occurring in $\mathbf{T}$. If $\mathbf{T}$ is a refutation in $R(\log)$, then all terms occuring in $\mathbf{T}$ have width at most $\log s$, so the resulting refutation has Σ_2 space $(\log s)^2$. □

4.3 A lower bound on regularized monomial space

One application of the results of the previous section is that they easily allow us to show lower bounds on regularized clause.

It is known [29, 28] that there are formulas F of size $\Theta(n)$ that have a resolution refutation of size $O(n)$, but

$$\text{MSpace}^*(F \vdash \bot) \geq n^{1/2}/(\log n)^{O(1)}.$$

Theorem 4.1 immediately gives the following improvement.

Theorem 4.5. *For every $n \geq 0$, there is a formula F of size $\Theta(n)$ that has a resolution refutation of size $O(n)$, width $O(1)$, and such that*

$$\text{MSpace}^*(F \vdash \bot) \geq \Omega\left(n/\log n\right).$$

Proof. [8] demonstrates the existence of an $O(1)$-CNF F that has resolution refutations of size $O(n)$, width $O(1)$, and such that $\log S_{T,R}(F \vdash \bot) \geq \Omega(n/\log n)$. In fact, [8] shows that $\Omega(n/\log n)$ is also the lower bound on the number of points the Delayer can score in the Prover-Delayer game of [40] played on F. Now, it is proved in [23] that this number of points is precisely equal to $\text{TCSpace}(F \vdash \bot)$ and then the result immediately follows from the second inequality in Theorem 4.1. □

Echoing remarks made in [29, Section 1.2], we do not know of any non-trivial upper or lower bounds on the monomial (or, for that matter, clause) space of the formulas from [8].

4.4 Trade-offs between positive depth and tree-like size for Horn formulas and tree-like size lower bounds

We would like next to focus on tree-like size lower bounds for resolution attained for formulas with small clause space. We will show that a tree-like resolution refutation of a Horn formula actually describes a pebbling strategy, the space and time of the strategy being the positive depth and size respectively of the proof. This gives a more transparent version of the result of [8] used in the proof of Theorem 4.5, which moreover has a natural generalization allowing us to prove some tree-like lower bounds for formulas of small clause space.

4.4.1 Horn formulas — basics

Horn formulas, that include pebbling formulas, have seen a plethora of applications in proof complexity over the past two decades, including separating resolution size from tree-like resolution size [8], separating width from variable space and clause space [6, 9, 10], separating depth from tree-like clause space [48], and giving trade-offs [6, 10, 29, 5], to name a few.

A CNF formula is called *Horn* if every clause in it has at most one non-negated variable. Equivalently, it is a set of atomic sequents, i.e. sequents containing variables, with at most one variable at the right hand side of each sequent.

The following result states that Horn formulas make up, in a certain sense, the easiest class of formulas for proof complexity. For its purposes, it is convenient to define a slightly modified version $\mathrm{CSpace}(\vdash F \to)$ of clause space. Namely, we replace the three standard rules with the following

Three-in-one rule: from a configuration $\mathcal{M}$, infer any configuration $\mathcal{M}^* \subseteq \mathcal{M} \cup F \cup \{C\}$, where C is obtained from clauses in $\mathcal{M}, F$ via the resolution or weakening rule.

Theorem 4.6. *Let F be a CNF formula. The following are equivalent:*

1. *F contains an unsatisfiable CNF sub-formula resulting from a Horn formula by negating some of its variables;*
2. $\mathrm{CSpace}(F \vdash \bot) \leq 3$*;*
3. $\mathrm{CSpace}(\vdash F \to) \leq 1$*;*
4. $W_R(\vdash F \to) \leq 1$*.*

Proof. For $1 \implies 2$, we can w.l.o.g. assume that F itself is an unsatisfiable Horn formula. We show, by induction on the number of variables n, that it can be refuted in clause space 3. The base case is trivial. Now, suppose that $n > 0$, and let y be a variable such that F contains the clause $\to y$. Such a clause must exist, for if every clause contained a negated variable, then we could satisfy F by setting every variable to false. Setting $y \coloneqq 1$, we get an unsatisfiable Horn formula $F|_{y:=1}$ with $n-1$ variables. From the induction hypothesis, there is a clause space 3 refutation of $F|_{y=1}$. Weakening every clause in it by $\overline{y}$ gives us a space 3 proof of $\overline{y}$ from F. Now we only have to download y and infer $\bot$.

For $2 \implies 3$, let $\mathcal{M}_1, \ldots, \mathcal{M}_t$ be a space 3 refutation of the formula F; we can assume w.l.o.g. that it does not contain weakening rules. Consider a path in the corresponding refutation tree of *maximum* possible length, say $C_i \in \mathcal{M}_{t_i}$ $(0 \leq i \leq h)$ are such that $t_0 < \ldots < t_h = t$, C_0 is an axiom, $C_h = \bot$ and for $i \geq 1$, C_i is obtained by resolving C_{i-1} with some $D_{i-1} \in \mathcal{M}_{t_i - 1}$. It remains to show that D_{i-1} is actually an axiom for any $i \geq 1$. For $i = 1$ this follows from the maximality of the chosen path. For $i \geq 2$, we have $\mathcal{M}_{t_{i-1}} = \{C_{i-2}, D_{i-2}, C_{i-1}\}$. Therefore C_{i-1} is consistent (and hence not resolvable) with the two other clauses in $\mathcal{M}_{t_{i-1}}$. All clauses that may have been inferred in $\mathcal{M}_{t_{i-1}+1}, \ldots, \mathcal{M}_{t_i}$ must have C_{i-1} as one of their premises and, as a consequence, are also not resolvable with C_{i-1}. Hence the only clauses in those configurations that may be resolvable with C_{i-1} (in particular, D_{i-2}) are the axioms.

The implication $3 \implies 4$ is proven by an argument similar to the first part of the proof of Theorem 4.3. Namely, a space 1 refutation of minimum length in the three-in-one model must necessarily be a sequence $\{C_1\}, \ldots, \{C_t\}$, where C_1 is obtained by resolving two clauses in F and C_{i+1} is obtained by resolving C_i with a clause in F. The non-axiom leaves of the tree T_i will simply be all those literals among $\overline{\ell_{i1}}, \ldots, \overline{\ell_{i,r_i}}$, where $C_i = \ell_{i1} \vee \ldots \vee \ell_{ir_i}$ that are not axioms of F. It is routinely to check that, as in the proof of Theorem 4.3, T_i is a resolution derivation using the rule (2.2).

Finally, for $4 \implies 1$, we again proceed by induction on the number of variables n of F. The base case is trivial. Suppose that $n > 0$. The fact that there is a width 1 refutation of F, forces F to have a one literal clause (since the refutation must start somewhere), say ℓ. Setting $\ell \coloneqq 1$, we get a width 1 refutation of $F|_{\ell:=1}$. From the induction hypothesis, a sub-formula G of $F|_{\ell:=1}$ is unsatisfiable Horn up to negating some variables. Let $\widehat{G}$ be the corresponding sub-formula of F; $\widehat{G}$ is obtained from G by restoring $\overline{\ell}$ to some of its clauses. Then $\widehat{G} \wedge \ell$ is an unsatisfiable Horn sub-formula of F. □

4.4.2 Tree-like resolution proofs as pebbling strategies

The paper [6] shows that a configurational resolution refutation π of the so-called *pebbling contradiction* Peb_G on a graph G defines a pebbling strategy on G, of time at most $|\pi|$ and space equal to the variable space $\mathrm{VSpace}(\pi)$. These are strategies in the so-called black-white game of [19]. We shall show that a tree-like resolution proof $\mathbf{T}$ of any Horn formula H defines a pebbling strategy of time equal to the size of $\mathbf{T}$ and space essentially equal to the *positive* depth of $\mathbf{T}$. These are strategies in the more basic black-only pebbling game that in the case $H = \mathrm{Peb}_G$ corresponds to the black-only pebbling game on G. They were considered by Urquhart [48] who showed how to relate them to ordinary depth. Thus, in a sense, our Proposition 4.1 below can be viewed as a (far-reaching) refinement of his result.

The rules of the black-only pebbling game, played on a Horn formula H, are as follows. There is a limited amount of pebbles. Pebbles are placed on the variables of H according to the rules:

1. A pebble can be placed on a variable y if $x_1, \ldots, x_k \to y$ is a clause of H, and all $x_1, \ldots, x_k$ have pebbles on them. In particular, a pebble can be always placed on any variable y such that $\to y$ is a clause of H.

2. A pebble can be removed from a variable at any time.

A *configuration* of the pebbling game is a set of the variables of H. A *pebbling strategy* is a sequence of configurations, each resulting from the previous one by one of the rules above. We say that a pebbling strategy *refutes* H if it ends with a configuration where for some clause $x_1, \dots, x_k \to$ of H, all variables $x_1, \dots, x_k$ are pebbled. Note that if H is unsatisfiable, then such a clause must exist.

Proposition 4.1. *Let H be an unsatisfiable Horn formula. A tree-like resolution refutation $\mathbf{T}$ of H of size s and positive depth d can be converted into a pebbling strategy that, starting with the empty configuration, refutes H in at most s steps and using at most $d+1$ pebbles.*

Proof. We begin with a slight modification of our refutation. Namely, viewing $\mathbf{T}$ as a decision tree, its nodes naturally correspond to partial assignments, and for the clause C sitting at the node α, we have $\alpha \models \neg C$. Let us replace C with the *maximal* clause satisfying this property. This will give us a refutation, of the same size and positive depth, in which the resolution rule (2.3) is reduced to

$$\frac{C \vee x \qquad C \vee \overline{x}}{C} \tag{4.2}$$

and leaves are labelled by weakenings of axioms in H.

This refutation need not necessarily consist of Horn formulas even if the original one did so. Nonetheless we will still represent clauses in the sequential form $S \to T$, where S, T are disjoint sets of variables, like at the beginning of Section 4.4.1. Note that $|S| \leq d$ for any clause $S \to T$ appearing in $\mathbf{T}$.

We shall now show by induction that every subtree of $\mathbf{T}$ deriving a clause $S \to T$, leads to a pebbling strategy that, starting with pebbles on all variables of S and using at most $d+1$ pebbles, either refutes H, or ends with a configuration which has pebbles on all variables of S and on one variable of T. Thus, if T is empty then the former must occur and, in particular, the strategy corresponding to the empty sequent $\to$ will start with no pebbles on the variables of H and will refute H.

Suppose that $S \to T$ is at a leaf. If there are variables $x_1, \dots, x_k$ in S such that $x_1, \dots, x_k \to$ is a clause of H, then that leaf describes a strategy that, starting with pebbles on all variables in S, immediately refutes H. Otherwise, there must be variables $x_1, \dots, x_k$ in S and a variable y in T such that $x_1, \dots, x_k \to y$ is a clause of H. Then the strategy of that leaf is to put a pebble on y. Since $|S| \leq d$, the number of pebbles used is at most $d+1$, as required.

If $S \to T$ is not at a leaf, then consider its left and right subtrees $\mathbf{T}_1$ and $\mathbf{T}_2$ proving $S, x \to T$ and $S \to T, x$ respectively (cf. (4.2)). The strategy corresponding to $S \to T$ is defined as follows. First follow $\mathbf{T}_2$'s strategy. If that strategy either refutes H or places a pebble on one of T's variables, then we are done. Otherwise, when the strategy of $\mathbf{T}_2$ is concluded, there are pebbles on S and x. Remove all other pebbles and follow the strategy of $\mathbf{T}_1$. The bound $d+1$ on the number of pebbles used at any moment follows from the same bound for $\mathbf{T}_1$ and $\mathbf{T}_2$.

Clearly, the number of steps of the pebbling strategy corresponding to $\to$ is at most the size of $\mathbf{T}$, and the required bound on the number of pebbles was already noticed. □

Remark 4.3. The proof of Proposition 4.1 relies on an intuitionistic interpretation of the resolution rule. In the intuitionistic tradition, the denotational view of assigning truth values is, philosophically, nonsense. A proposition is "true" if it is provable, and a proof of e.g. a formula $S \to T$ is a construction that given proofs of all the elements of S produces a proof of some element in T. What Proposition 4.1 says is that a tree-like resolution derivation of $S \to T$ precisely describes such a construction, assuming that proofs of all the clauses of H are known. Moreover this construction will be economical in the number of steps and memory if the size and the positive depth respectively of the proof are small.

Let us further notice, that although Proposition 4.1 is stated for Horn formulas, it really is general. A tree-like resolution derivation $\mathbf{T}$ of $S \to T$ from clauses $S_1 \to T_1, \ldots, S_m \to T_m$ describes, in the same way as in the proof of Proposition 4.1, a construction that given proofs for all elements of S, produces a proof of some element in T. Which of the elements in T will a proof be given for, or for that matter the construction itself, will now not be uniquely determined; it will depend on which of the elements of the sets T_i's were proofs given for at the leaves. $\mathbf{T}$ describes at the same time constructions for *any* of these choices.

4.4.3 Tree-like size lower bounds

The following theorem turns pebbling time-space trade-offs for a Horn formula H into tree-like size lower bounds for its substituted version $H[\vee_2]$. We formulate it in a somewhat general form, to account for various kinds of pebbling trade-offs in the literature.

Recall that the substituted version $F[\vee_2]$ of a CNF $F(x_1, \ldots, x_n)$ is defined by replacing x_i with $y_i \vee z_i$ for mutually distinct variables $\{y_1, z_1, \ldots, y_n, z_n\}$, followed by converting the result back to the CNF form in the straightforward way.

Theorem 4.7. *Let H be an unsatisfiable Horn formula on n variables. Suppose that every pebbling strategy that refutes H in s steps and using d pebbles, starting with no pebbles on its variables, satisfies*

$$(d-1) \cdot f(s) \geq g(n)$$

for non-decreasing positive functions f, g. Then

$$f(t) \log t \geq g(n),$$

where $t \stackrel{\text{def}}{=} S_{T,R}(H[\vee_2] \vdash 0)$.

Proof. Create a probability space of partial assignments by choosing independently for every variable x of H, which was substituted by $y \vee z$, one of y and z with probability $1/2$ and setting it to zero. Note that for any α from this space, $H[\vee_2]|_\alpha$ is identical to H up to renaming its variables and hence $\mathbf{T}|_\alpha$ is a refutation of H, again up to renaming variables. Let $\mathbf{T}$ be an arbitrary tree-like resolution refutation of $H[\vee_2]$ of size t represented as in the proof of Proposition 4.1, that is with weakenings at the leaves only. Let $\sigma_1, \ldots, \sigma_k$ be all clauses $S \to T$ with $|S| \geq g(n)/f(t)$ occurring as a leaf in $\mathbf{T}$. We have that

$$P\left[\bigvee_{i=1}^{k} (\sigma_i|_\alpha \neq 1)\right] \leq \sum_{i=1}^{k} P[\sigma_i|_\alpha \neq 1] \leq k 2^{-g(n)/f(t)} \leq t 2^{-g(n)/f(t)}.$$

If $f(t) \log t < g(n)$, then the above probability is smaller than 1, which means that there is a point α in our sample space such that $\mathbf{T}|_\alpha$ is a tree-like resolution refutation of size at most t and positive depth $\leq g(n)/f(t)$. This, from Proposition 4.1, gives a pebbling strategy that refutes H in t steps using d pebbles, where $(d-1) \cdot f(t) < g(n)$. □

Recall that for a DAG G, the pebbling contradiction Peb_G is defined as the Horn formula consisting of all clauses $S \to x$, where $x \in V(G)$ and S is the set of all its immediate predecessors, as well as the clauses $x \to$ for any sink x. Plugging into Theorem 4.7 various DAGs from the literature with known bounds on their pebbling complexity and various functions f, we can get several corollaries.

The first is a simplified proof of the separation by Ben-Sasson et al.

Corollary 4.3 [8]**.** *There are formulas of size $O(n)$ having DAG-like resolution refutations of size $O(n)$, every tree-like resolution refutation of which requires size* $\exp(\Omega(n/\log n))$.

Proof. This is by setting $f := 1$ in Theorem 4.7, and using the graphs of [39] having constant in-degree and requiring $\Omega(n/\log n)$ pebbles to pebble. □

The next result was promised in the introduction. It should be compared with Theorem 4.6.

Theorem 4.8. *There are formulas of size $O(n)$ having tree-like resolution refutations of clause space 4, every tree-like resolution refutation of which has size $\Omega(n^2/\log n)$.*

Proof. This is by setting $f(t) := t$ in Theorem 4.7, and using the graphs of [34, Theorem 2.3.2] having linear size and exhibiting a $dt \geq \Omega(n^2)$ trade-off. These graphs can be pebbled using 3 pebbles, and that immediately gives that $\text{CSpace}(\text{Peb}_{G_n}[\vee_2] \vdash \perp) \leq O(1)$. By being more careful, however, we can bring the space down to the minimum possible value, namely 4, for which a super-linear lower bound on tree-like resolution size is possible.

More precisely, the above graphs have the following form. They contain two directed vertex-disjoint paths, let us call them U and L, and there are additional edges from vertices of U to vertices of L (see Figure 4). Moreover, the indegree of any vertex in D is two, that is, there are no two vertices in U both with edges to the same vertex in D. Let G be such a graph, and let $H \stackrel{\text{def}}{=} \text{Peb}_G$ be the

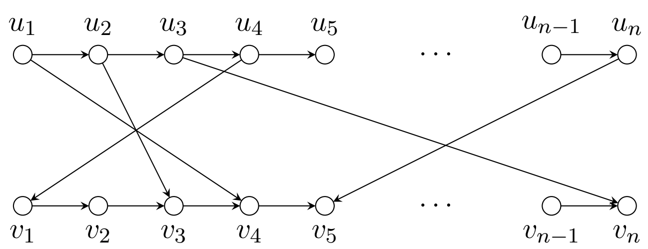

Figure 4: The form of the graphs giving the trade-off in Theorem 4.8

corresponding Horn formula. Call the variables of the path U, $u_1, \ldots, u_n$ and the variables of L, $v_1, \ldots, v_n$ as in Figure 4, and suppose that to obtain $H[\vee_2]$, u_i is substituted by $x_i \vee y_i$ and v_i by $z_i \vee w_i$. We first note that any clause $x_i \vee y_i$ can be derived in clause space 3. This is because the formula resulting by negating all the variables of $\mathrm{Peb}_U[\vee_2]$ is Horn, so by Theorem 4.6, $\mathrm{CSpace}(\mathrm{Peb}_G[\vee_2] \vdash x_i \vee y_i) \leq 3$. Notice that the derivations provided by Theorem 4.6 are tree-like. To show the bound $\mathrm{CSpace}(H[\vee_2] \vdash \bot) \leq 4$, we need to show, having derived $z_{i-1} \vee w_{i-1}$, how to derive $z_i \vee w_i$. Suppose that $\overline{u_j} \vee \overline{v_{i-1}} \vee v_j$ is a clause of H, so that $\overline{x_j} \vee \overline{z_{i-1}} \vee z_i \vee w_i$, $\overline{x_j} \vee \overline{w_{i-1}} \vee z_i \vee w_i$, $\overline{y_j} \vee \overline{z_{i-1}} \vee z_i \vee w_i$ and $\overline{y_j} \vee \overline{w_{i-1}} \vee z_i \vee w_i$ are clauses of $H[\vee_2]$. Notice that

$$\mathrm{CSpace}\left(\begin{array}{c} (z_{i-1} \vee w_{i-1}) \wedge \\ (x_j \vee y_j) \wedge \\ (\overline{x_j} \vee \overline{z_{i-1}} \vee z_i \vee w_i) \wedge \\ (\overline{y_j} \vee \overline{z_{i-1}} \vee z_i \vee w_i) \end{array} \vdash \ w_{i-1} \vee z_i \vee w_i \right) \leq 4.$$

In this derivation, all premises are immediately removed from memory after they are used as premises in an inference. Similarly, we have

$$\mathrm{CSpace}\left(\begin{array}{c} (w_{i-1} \vee z_i \vee w_i) \wedge \\ (x_j \vee y_j) \wedge \\ (\overline{x_j} \vee \overline{w_{i-1}} \vee z_i \vee w_i) \wedge \\ (\overline{y_j} \vee \overline{w_{i-1}} \vee z_i \vee w_i) \end{array} \vdash \ z_i \vee w_i \right) \leq 4.$$

Running the first derivation, deleting everything from memory except $z_{i-1} \vee z_i \vee w_i$ and then running the second derivation, deriving $x_j \vee y_j$ in clause space 3 whenever it is downloaded in these derivations, we get the desired clause space 4 derivation of $z_i \vee w_i$. □

By using the construction from [34, Theorem 4.2.6] and the superconcentrator graphs of [20], Theorem 4.8 can be further generalized to a lower bound $(n/\log n)^{\Omega(k)}$ on the tree-like resolution size of refuting formulas with clause space k. We skip the details as they are straightforward but the result still falls very short of the ultimate goal. Let us further notice that the fact that the space 4 refutation in Theorem 4.8 is tree-like might be interesting, as typically tree-like resolution size lower bounds have been proven in the literature based on the prover-delayer game of [40], which also gives a lower bound for the clause space of tree-like resolution refutations (cf. Theorem 4.5).

5 Conclusion

We showed a quadratic gap between resolution and cut-free sequent calculus width. In terms of the sequent calculus, this says that atomic cuts can shorten the width of proofs. It is well known that cuts can make proofs exponentially shorter. Allowing arbitrary cuts we get a system polynomially equivalent with any Frege system. These are very powerful; proving non-trivial lower bounds for them is completely out of reach of current methods. But even allowing cuts of depth $d + 1$ in an LK system that has cuts of depth d for any constant $d \geq 0$, gives exponentially shorter proofs [31]. And this goes lower: For any constant $k \geq 0$, allowing as cut formulas conjunctions and disjunctions of size $k+1$ in an LK system that has as cuts conjunctions and disjunctions of size at most k, again gives exponentially shorter proofs [44]. We show in this paper that even allowing propositional variables as cuts, gives super-polynomially shorter proofs.

Cut-free sequent width for refuting CNF formulas naturally compares to well studied complexity measures related to resolution: it sits between resolution width and clause space. Our quadratic gap in particular, provides a separation between resolution width and clause space. Stronger such separations are known [9, 10]. Nontheless, our basic construction extends to provide a quadratic gap between resolution width and monomial space. This is to be seen in conjunction with the result of Galesi et al. [26] showing that monomial space provides an upper bound to resolution width:

$$W_R(F \vdash \bot) \leq O\left((\text{MSpace}(F \vdash \bot))^2\right) + W(F). \tag{5.1}$$

Several questions remain open:

- Can cut-free sequent calculus width for refuting CNF formulas be bounded in terms of resolution width? Given the similarity between the two measures, the combination of Lemmas 3.1 and 3.2 giving a quadratic separation might come as a surprise. Can this separation be improved? A strong separation in particular, would give an exponential separation between resolution and cut-free sequent calculus.

- Our super-polynomial separation of resolution and cut-free sequent calculus on the one hand applies only when clauses are seen as disjunctions of unbounded arity. On the other hand, it concerns formulas whose size grows exponentially on the number of variables. Can there be a separation independent of the representation of clauses? Can there be a separation for formulas of size polynomial to the number of variables?

- Cut-free sequent calculus width is bounded by clause space. Can it be bounded in terms of monomial space in a relation similar to (5.1)? This is a good point to also mention that whether (5.1) can be improved to a linear inequality or there are examples where it is tight is unknown as well, and there do not seem to be strong indications for which case is true.

- We show that resolution width and monomial space cannot coincide. Whether they coincide up to polynomial factors however remains open, although it is speculated (cf. [29]) that this is not the case, and moreover, as it is the case for resolution width and clause space [9, 10], there is an $O(1)$ vs $\Omega(n/\log n)$ separation.

We showed that $\log S_{T,R}$, CSpace* and MSpace* are equivalent up to polynomial and $\log n$ factors, demonstrating a picture perfectly analogous to the picture involving D_R, VSpace* and TSpace* in [42]. The most important question remains (widely) open:

Problem 5.1. Is it true that CSpace $\approx \log S_{T,R}$ or MSpace $\approx \log S_{T,R}$? Recall for comparison that $\log S_{T,R} \approx$ CSpace* $\approx$ MSpace*.

Equivalently, do there exist strong trade-offs between clause (or monomial) space and length? It should be contrasted with trade-offs results in e.g. [10, 5], and it is a perfect analogue of Urquhart's question [48] about variable space vs. depth studied in [42, Section 6]. Let us make a few more remarks about this problem.

Firstly, for very small space essentially this question was already asked in the literature before. Namely (see e.g. [35, Open Problem 16]), are there formulas having constant clause space refutations and with the property that any such refutation must necessarily have super-polynomial length? Suitably adjusting it to our framework:

Problem 5.2 (small space variant). Are there formulas that have $(\log n)^{O(1)}$ clause or monomial space refutations and with the property that any such refutation must be of super-quasi-polynomial length $\exp((\log n)^{\omega(1)})$? Equivalently, any tree-like resolution refutation must have super-quasi-polynomial length.

In terms of the perceived difficulty, we do not discern too much of a difference between Problems 5.1, 5.2 and Nordström's question. In fact, we would like to cautiously conjecture that there are formulas F with $\mathrm{CSpace}(F \vdash \bot) \leq 5$ and $\mathrm{CSpace}^*(F \vdash \bot) \geq \exp\left(n^{\Omega(1)}\right)$. But the only result we were able to prove in that direction is the rather weak Theorem 4.8.

Secondly, as suggested by Figure 1, any strong separation between monomial space and clause space would immediately solve Problem 5.1 for monomial space. As we consider the latter to be most likely very difficult, we take it as a good heuristic explanation of why we have not seen any progress on the former problem as well. But let us ask this, and one obviously relevant question, explicitly anyway:

Problem 5.3. Is it true that CSpace $\approx$ MSpace? Is it true that MSpace $\approx W_R$?

We note that by the result from [36, 9], at least one of these must be false. A quadratic separation between width and monomial space has been recently proved by the first author (manuscript in preparation). For a discussion on related topics, see also [17, Section 7.5.5].

Finally, while all these conjectured trade-offs are very strong, they are still not super-critical in the sense of [41] (the required lower bound on length never exceeds 2^n). The inequality (4.1), however, implies that in all these questions refutation length can be replaced with depth. Since the depth, as a stand-alone measure, is always bounded by n, these actually *are* questions about the existence of a *super-critical* trade-off between clause space and *depth*.

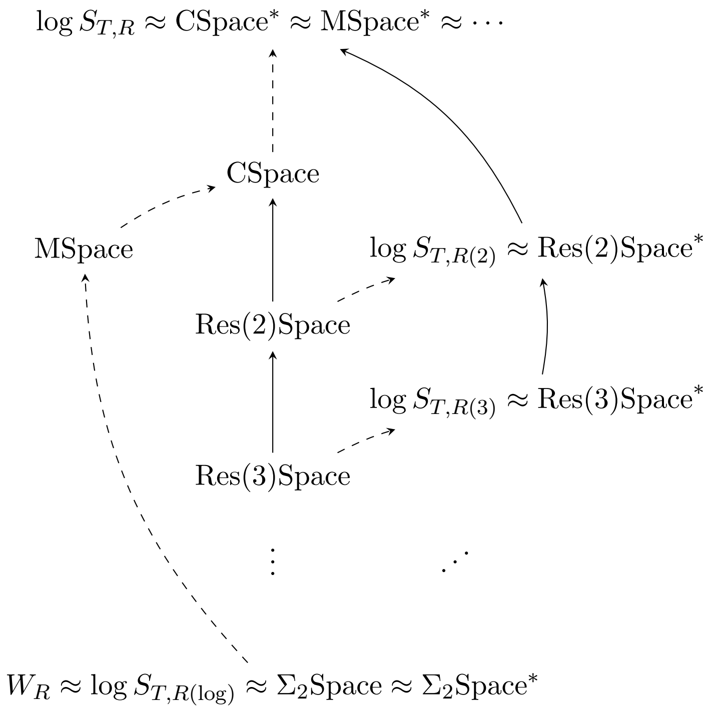

Figure 5: Σ_2 space and tree-like size for subsystems of depth 2 Frege

We have (somewhat surprisingly) proved that depth 2 Frege behaves very differently from resolution with respect to space. Intermediate systems based on Res(k) for a constant k were studied in a similar context before, and it is very natural to wonder what is the situation for those systems.

Let us first remark that for Res(k)-refutations, the definition of space from [21, 10] (formula space) coincides with ours up to a constant factor so we need not distinguish between the two. Then Theorem 4.1 readily generalizes to this regime and gives

$$\log S_{T,\mathrm{Res}(k)} \approx \mathrm{Res}(k)\mathrm{Space}^*,$$

extending the bottom half of Figure 1 as shown in Figure 5. The proof of Corollary 4.1, however, fails for a constant k as badly as it fails for $k = 1$. Hence we have one more question to ask:

Problem 5.4 (Res(k)-variant). Is there a constant $k > 0$ such that $\log S_{T,\mathrm{Res}(k)} \approx \mathrm{Res}(k)\mathrm{Space}$ or at least $\log S_{T,\mathrm{Res}(k)} \preceq \mathrm{CSpace}$?

Let us also mention that as k increases, both hierarchies, $\log S_{T,\mathrm{Res}(k)}$ (and, hence, also Res(k)Space*) and Res(k)Space are proper ([21] and [10] respectively), and this holds even for k up to $(\log n)^c$ where n is the number of variables. Notice that the above excludes the dual version of Remark 4.2: while the formula space of depth 2 Frege refutations can be reduced to constant, this is not true for the widths of individual formulas in the memory.

The relation between VSpace and CSpace is also unknown in one direction (the opposite one is taken care of by [6]). Let us re-iterate the problem posed e.g. in [42]:

Problem 5.5. Is it true that CSpace $\preceq$ VSpace?

Just as with the questions of similar nature discussed above, a negative answer would also imply a separation between VSpace and VSpace*, hence we can expect it to be very difficult.

References

[1] Michael Alekhnovich, Eli Ben-Sasson, Alexander Razborov, and Avi Wigderson. Space complexity in propositional calculus. *SIAM Journal of Computing*, 31:1184–1211, 2002.

[2] Noriko Arai. Relative efficency of propositional proof systems: resolution vs. cut-free LK. *Annals of Pure and Applied Logic*, 104:3–16, 2000.

[3] Noriko Arai, Toniann Pitassi, and Alasdair Urquhart. The complexity of analytic tableaux. *Journal of Symbolic Logic*, 71:777–790, 2006.

[4] Albert Atserias and Victor Dalmau. A combinatorial characterization of resolution width. *Journal of Computer and System Sciences*, 74:323–334, 2008.

[5] Chris Beck, Jakob Nordström, and Bangsheng Tang. Some trade-off results for polynomial calculus: extended abstract. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*, pages 813–822, 2013.

[6] Eli Ben-Sasson. Size-space tradeoffs for resolution. *SIAM Journal of Computing*, 38, 2009.

[7] Eli Ben-Sasson and Nicola Galesi. Space complexity of random formulae in resolution. *Random Structures & Algorithms*, 23:92–109, 2003.

[8] Eli Ben-Sasson, Russell Impagliazzo, and Avi Wigderson. Near optimal separation of tree-like and general resolution. *Combinatorica*, 24:585–603, 2004.

[9] Eli Ben-Sasson and Jakob Nordström. Short proofs may be spacious: An optimal separation of space and length in resolution. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 709–718, 2008.

[10] Eli Ben-Sasson and Jakob Nordström. Understanding space in proof complexity: Separations and trade-offs via substitutions. In *Proceedings of the 2nd Symposium on Innovations in Computer Science*, pages 401–416, 2011.

[11] Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow—resolution made simple. *Journal of the ACM*, 48:149–169, 2001.

[12] Patrick Bennett, Ilario Bonacina, Nicola Galesi, Tony Huynh, Mike Molloy, and Paul Wollan. Space proof complexity for random 3-cnfs. *Information and Computation*, 255:165–176, 2017.

[13] Ilario Bonacina. Total space in resolution is at least width squared. In *Proceedings of the 43rd International Colloquium on Automata, Languages, and Programming*, volume 55, pages 56:1–56:13, 2016.

[14] Ilario Bonacina and Nicola Galesi. Pseudo-partitions, transversality and locality: a combinatorial characterization for the space measure in algebraic proof systems. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, pages 455–472, 2013.

[15] Ilario Bonacina and Nicola Galesi. A framework for space complexity in algebraic proof systems. *Journal of the ACM*, 62:23:1–23:20, 2015.

[16] Maria Luisa Bonet and Nicola Galesi. Optimality of size-width tradeoffs for resolution. *Computational Complexity*, 10:261–276, 2002.

[17] Samuel Buss and Jakob Nordström. Proof complexity and SAT solving. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability*, chapter 7, pages 233–350. IOS Press, 2nd edition, 2021.

[18] Stephen Cook and Robert Reckhow. On the lengths of proofs in the propositional calculus (preliminary version). In *Proceedings of the 6th Annual ACM Symposium on Theory of Computing*, pages 135–148, 1974.

[19] Stephen Cook and Ravi Sethi. Storage requirements for deterministic polynomial time recognizable languages. *Journal of Computer and System Sciences*, 13:25–37, 1976.

[20] Danny Dolev, Cynthia Dwork, Nicholas Pippenger, and Avi Wigderson. Superconcentrators, generalizers and generalized connectors with limited depth. In *Proceedings of the 15th Annual ACM Symposium on Theory of Computing*, pages 42–51, 1983.

[21] Juan Luis Esteban, Nicola Galesi, and Jochen Messner. On the complexity of resolution with bounded conjunctions. *Theoretical Computer Science*, 321:347–370, 2004.

[22] Juan Luis Esteban and Jacobo Torán. Space bounds for resolution. *Information and Computation*, 171:84–97, 2001.

[23] Juan Luis Esteban and Jacobo Torán. A combinatorial characterization of treelike resolution space. *Information Processing Letters*, 87:295–300, 2003.

[24] Yuval Filmus, Massimo Lauria, Mladen Miksa, Jakob Nordström, and Marc Vinyals. Towards an understanding of polynomial calculus: New separations and lower bounds (extended abstract). In *Proceedings of the 40th International Colloquium on Automata, Languages, and Programming*, volume 7965, pages 437–448, 2013.

[25] Yuval Filmus, Massimo Lauria, Mladen Miksa, Jakob Nordström, and Marc Vinyals. From small space to small width in resolution. *ACM Transactions of Computational Logic*, 16:28:1–28:15, 2015.

[26] Nicola Galesi, Leszek Kołodziejczyk, and Neil Thapen. Polynomial calculus space and resolution width. In *Proceedings of the 60th Annual IEEE Symposium on Foundations of Computer Science*, pages 1325–1337, 2019.

[27] Gerhard Gentzen. Untersuchungen über das logische schließen. I. *Mathematische Zeitschrift*, 39:176–210, 1934.

[28] Mika Göös and Toniann Pitassi. Communication lower bounds via critical block sensitivity. *SIAM Journal of Computing*, 47:1778–1806, 2018.

[29] Trinh Huynh and Jakob Nordström. On the virtue of succinct proofs: amplifying communication complexity hardness to time-space trade-offs in proof complexity. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing Conference*, pages 233–248, 2012.

[30] Russell Impagliazzo, Pavel Pudlák, and Jirí Sgall. Lower bounds for the polynomial calculus and the gröbner basis algorithm. *Computational Complexity*, 8:127–144, 1999.

[31] Jan Krajíček. Lower bounds to the size of constant-depth propositional proofs. *Journal of Symbolic Logic*, 59:73–86, 1994.

[32] Jan Krajíček. On the weak pigeonhole principle. *Fundamenta Mathematicae*, 170:123–140, 2001.

[33] Massimo Lauria. A note about k-DNF resolution. *Information Processing Letters*, 137:33–39, 2018.

[34] Thomas Lengauer and Robert Tarjan. Asymptotically tight bounds on time-space trade-offs in a pebble game. *Journal of the ACM*, 29:1087–1130, 1982.

[35] Jakob Nordström. Pebble games, proof complexity, and time-space trade-offs. *Logical Methods in Computer Science*, 9, 2013.

[36] Jakob Nordström and Johan Håstad. Towards an optimal separation of space and length in resolution. *Theory of Computing*, 9:471–557, 2013.

[37] Theodoros Papamakarios. Space characterizations of complexity measures and size-space trade-offs in propositional proof systems. *Electronic Colloqium on Computational Complexity, Report No. 176*, 2021.

[38] Theodoros Papamakarios and Alexander Razborov. Space characterizations of complexity measures and size-space trade-offs in propositional proof systems. *Electronic Colloqium on Computational Complexity, Report No. 74*, 2021.

[39] Wolfgang Paul, Robert Tarjan, and James Celoni. Space bounds for a game on graphs. *Mathematical Systems Theory*, 10:239–251, 1977.

[40] Pavel Pudlák and Russell Impagliazzo. A lower bound for DLL algorithms for k-SAT. In *Proceedings of the 11th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 128–136, 2000.

[41] Alexander Razborov. A new kind of tradeoffs in propositional proof complexity. *Journal of the ACM*, 63:16:1–16:14, 2016.

[42] Alexander Razborov. On space and depth in resolution. *Computational Complexity*, 27:511–559, 2018.

[43] Robert Reckhow. *On the lengths of proofs in the propositional calculus.* PhD thesis, Department of Computer Science, University of Toronto, 1975.

[44] Nathan Segerlind, Samuel Buss, and Russell Impagliazzo. A switching lemma for small restrictions and lower bounds for k-DNF resolution. *SIAM Journal of Computing*, 33:1171–1200, 2004.

[45] Raymond Smullyan. *First-order Logic*. Dover, 1995.

[46] Grigori Tseitin. On the complexity of derivation in propositional calculus. *Studies in constructive mathematics and mathematical logic, part 2*, pages 115–125, 1968.

[47] Alasdair Urquhart. The complexity of propositional proofs. *Bulletin of Symbolic Logic*, 1:425–467, 1995.

[48] Alasdair Urquhart. The depth of resolution proofs. *Studia Logica*, 99:349–364, 2011.