PDMI PREPRINT — 09/2004

First Report on the Semialgebraic Prover [1]

S. S. Fedin[2] A. Kojevnikov[3] B. Konev[3] A. S. Kulikov[2]
S. I. Nikolenko[2] V. P. Orevkov[3]

[1]The research described in this publication was made possible in part by Award No. RM1-2409-ST-02 of the U.S. Civilian Research & Development Foundation for the Independent States of the Former Soviet Union (***CRDF***) and by a grant from ***Intel Corporation*** (CRDF GAP RM0-1373-ST-03).

[2]St.Petersburg State University, Russia. Web: `http://logic.pdmi.ras.ru/~{fedin, kulikov,sergey}`.

Abstract

We describe the basic notions and the algorithm of the semialgebraic prover being developed in Laboratory of Mathematical Logic of Steklov Insitute of Mathematics at St.Petersburg. The prover solves formulas of the first-order logic (with emphasis on formulas having hard propositional part) by translating them into systems of inequalities and searching for a semialgebraic refutation of these systems.

The prover is written in `C++`. The implementation is flexible and allows to modify easily the rules of proof system and even the nature of derivation objects.

This document is being updated continuously. Please visit

`http://logic.pdmi.ras.ru/~saprover/`

to obtain the last version.

Contents

Chapter 1

Introduction

The semialgebraic prover is an implementation of a certain saturation *algorithm* that operates with certain *objects* by means of derivation *rules*. The objects are held in *sets* that have indices that allow to find, add, and remove objects quickly.

The main difference of the algorithm from classical saturation algorithms (cf. OTTER [Kal01], Vampire [RV01]) is the use of two types of objects: *deduction objects* and *modification objects*, with different rules applicable to them. Informally, deduction objects are just normal lines of a proof; modification objects are strong facts that simplify deduction objects (e.g., the fact $P(x) = Q(x) + R(x)$ allows to eliminate the predicate P from all subsequent lines of the proof).

The main algorithm (Sect. 2.2) and the classification of derivation rules (Sect. 2.1) are implemented (and described) in a way that enables to use them with deduction and modification objects of any nature. However, the main idea of the semialgebraic prover is the use of inequalities. Accordingly, the specific types of objects we use are described in Sect. 3.1, and the specific rules are given in Sect. 3.2. The reader may want to look into the (shorter and more intuitive) description of the underlying proof system (Sect. 4.1) to get an informal idea about the essence of the prover. While the completeness of this system follows trivially from the (known) completeness of its propositional counterpart, proving the completeness of the algorithm is much more tricky, because it uses subsumption, simplification, and other things that have no analogies in a proof system. See Sect. 4.2 for a formal proof of completeness for one variant (and not the fastest one) of our prover (invoked using `--complete` option). Examples of proofs found by the prover are given in Ch. 6, and the numerical data from experiments is given in Ch. 5.

The prover is implemented in `C++`, and the minimum amount of details concerning this implementation is given in the corresponding sections. The source code is quite flexible and allows

- easy implementation of provers based on different semialgebraic proof systems;
- easy implementation of provers based on different (not semialgebraic) objects;
- easy (even dynamic) replacement of data structures.

More implementation details are given in the source code and `doxygen` documentation.

Chapter 2

Abstract concept

2.1 Types of derivation rules

We use three main types of derivation rules, each of them has a separate version for deduction and modification objects. There is also a finer classification w.r.t. the number of premises (currently, there are one-premise, two-premise and, sometimes, multi-premise rules). For each rule, it is supposed that the conclusion logically follows from the premises. The rules are implemented as `C++` classes (descendants of the `Rule` class) with an operator `()`[1] that applies the rule to the given objects. For two-premise rules there is also another version of this operator, this version applies the rule to a given object and to each applicable[2] object in a given set of objects (this allows to speed up the program by the use of indices provided by the sets). Some rules have a version operator `()` that accepts two sets.

The reader is referred to Sect. 3.2 for specific examples of rules of the abstract types defined below.

2.1.1 Rules that produce or modify deduction objects

Generation rule.

A two-premise generation rule. A two-premise generation rule is just what a usual two-premise rule is. It accepts two deduction objects (treated as an *un*ordered pair) and derives a new deduction object from them. (In what follows, we denote by `P<type>` a pointer to `<type>`; we use "smart" pointers provided by the `boost` library[3].)

```
PDeductionObjectSet ExpDGenerationRule2::operator()
(PDeductionObject& premise1,
 PDeductionObject& premise2,
 Statistics& statistics)
```

```
PDeductionObjectSet ExpDGenerationRule2::operator()
(PDeductionObject& premise1,
 PDeductionObjectSet& premise2set,
```

[1] Reads "application".
[2] As specified in the algorithm.
[3] `http://www.boost.org/`

```
 Statistics& statistics)
```

(The prefix `Exp` is kept for purely historical reasons and the argument `statistics` is currently unused.) Typically, `DeductionObjectSet` is one of the several sets (with common interface, but different implementation) used in the main algorithm. In what follows, we omit similarly looking code for other rules.

There is also a version of this rule for the case when the second object is a modification object.

```
PDeductionObjectSet ExpDGenerationRule2M::operator()
(PDeductionObjectSet& premise1,
 PModificationObject& premise2,
 Statistics& statistics)
```

The "accelerated" interface of this rule has a set as the *first* argument:

```
PDeductionObjectSet ExpDGenerationRule2M::operator()
(PDeductionObjectSet& premise1set,
 PModificationObject& premise2,
 Statistics& statistics)
```

A one-premise generation rule accepts just one deduction object and generates a new object. We also use another version of such rule that accepts a modification object instead (and generates a deduction object).

Any sequence of applications of such rule to an object is supposed to terminate.

Reduction rule.

A two-premise reduction rule accepts a deduction object (main premise) and a modification object and derives a new deduction object. The differences from generation rules are that a reduction rule accepts different parameters and is supposed to generate only "stronger" objects (i.e., any sequence of applications of such rules to an object is supposed to terminate; this is the difference from the similarly looking version of two-premise generation rules). The "accelerated" interface of this rule has a set as the *first* argument:

```
PDeductionObjectSet ExpDReductionRule2::operator()
(PDeductionObjectSet& premise1set,
 PModificationObject& premise2,
 Statistics& statistics)
```

Simplification rule.

A two-premise simplification rule accepts a deduction object (main premise) and a modification object and simplifies the former using the latter. The new object must be able to play the role of the old object in all possible cases, because it will *replace* the old object in the derivation. Such rule is supposed to generate only "stronger" objects (i.e., any sequence of applications of such rules to an object is supposed to terminate). Such rule is always applied to a set of modification objects in order to simplify the deduction object as long as at least one modification object is applicable. It also returns the resulting status (whether the object was simplified; whether it was simplified to a tautology or to a contradiction).

```
SimplificationResult ExpDSimplificationRule2::operator()
(PDeductionObject& premise1,
 PModificationObjectSet& premise2set,
 Statistics& statistics)
```

This is the only rule that *changes* its premise (the pointer). The "accelerated" interface of this rule has a set also as its first argument; the members of this set are changed.

A one-premise simplification rule accepts just one deduction object and simplifies it; its primary use is to put a deduction object into a *normal form*. There is no interface for such rules; they are implemented in the methods `normalize()` and `proofLine(...)` of an object. (The second method is used if an object really becomes a new line of the derivation.) We assume that objects are held in sets in the normal form.

Subsumption rule.

A two-premise subsumption rule accepts two deduction objects (considered as an *ordered* pair) and (sometimes) figures out whether one of them can be used instead of the other one in the inference. The "weaker" object is eliminated. If two objects are equivalent, the *first* object is eliminated.

In the current implementation there is no "plain" version of operator `()` (that would accept just objects); the "main" version of operator `()` uses a set as its second argument and also says whether the object is subsumed.

```
SimplificationResult ExpDSubsumptionRule2::operator()
(PDeductionObject& premise1,
 PDeductionObjectSet& premise2set,
 Statistics& statistics)
```

This rule has an extra version of the operator `()` that applies it to the elements of the direct product of two sets.

```
void ExpDSubsumptionRule2::operator()
(PDeductionObjectSet& set1,
 PDeductionObjectSet& set2,
 Statistics& statistics)
```

A multi-premise subsumption rule accepts a set and eliminates superfluous elements from it. Currently, it is implemented as a version of operator `()` of a two-premise subsumption rule.

2.1.2 Rules that produce or modify modification objects

Generation rule. Modification objects are generated purely from deduction objects[4]. There are two versions.

A two-premise ... rule.

[4] See also multi-premise simplification rules that may generate new modification objects in order to put a *set* of modification objects into a normal form.

A one-premise . . . rule.

Simplification rule.

A one-premise . . . rule accepts just one modification object and simplifies it; its primary use is to put a modification object into a normal form. There is no interface for such rules; they are implemented in the method `toNormalForm()` of an object. We assume that objects are held in sets in the normal form.

A multi-premise . . . rule accepts two sets of modification objects ("old" and "new" objects) and puts "new" objects into the set of "old" ones keeping the latter set into a "normal form" (in particular, it may eliminate some of the objects and generate new ones):

```
void ExpMSimplificationRuleMulti::operator()
(PModificationObjectSet& oldset,
 PModificationObjectSet& newset,
 Statistics& statistics)
```

The updated set of "new" objects can then be used to process objects that are really new (i.e., either kept or generated by this rule).

2.2 Algorithm

The algorithm is implemented as a `saturate()` method of the `C++` class `ExpInferenceObject`. It uses a collection of sets that are returned by its methods `get<Set>()`. The sets are

- `Active` (set of active deduction objects);
- `Passive` (set of passive deduction objects);
- `NewPassive` (temporary set of deduction objects);
- `Modifiers` (set of modification objects);
- `NewModifiers` (temporary set of modification objects).

The objects are stored in a set in a *normal form*, i.e., after all one-premise simplification rules have been applied to them (and no rule is applicable anymore); if the algorithm attempts to put a contradiction into a set, the execution should terminate (this is *not* a property of sets); if the object violates the current restrictions, it is dropped. There is also a deduction object named

- `current`,

which is the "main" object selected for particular iteration. The algorithm uses certain

- rules

given to it. All rules of each type are given in one meta-rule that applies the inner rules one by one; in the case of simplification rules this process is repeated until no rule is applicable (note that the object is put into a normal form after each change). The meta-rules have the same interface as usual rules and are named below in an intuitive way (for example, "`DExpMetaGenerationRule2`" denotes the two-premise generation rules for deduction objects, and "`DExpMetaGenerationRule`$\langle n \rangle$`M`" denotes the n-premise generation rules that produce deduction objects from (deduction and) modification objects). The algorithm is also given an object of the class

- `Select`

that has an operator `(...)` which takes the next object out of `Passive`.

In the **beginning**, all sets are empty except for `Passive`, which contains the input inequalities (in a normal form).

The algorithm repeats **the main cycle** until a contradiction is derived or no rules are applicable:

0. `NewModifiers = NewPassive =` $\emptyset$.
1. `current = select()`.
2. Preprocess `current`:
   (a) `DExpMetaSimplificationRule2(current,Modifiers)`.
   If `current` becomes trivial, goto 0.
   (b) `DExpMetaSubsumptionRule2(current,Active)`.
   If `current` is subsumed, goto 0.
   (c) `NewModifiers += MExpMetaGenerationRule1(current)`.
   (d) `NewModifiers += MExpMetaGenerationRule2(current,Active)`.
   (e) `MExpMetaSimplificationRuleMulti(Modifiers,NewModifiers)`.
   (f) `NewPassive += DExpMetaGenerationRule1M(NewModifiers)`.
   (g) $\forall a \in$ `Active`, if `DExpMetaSimplificationRule2(`a`,NewModifiers)` changes a, move (modified) a to `NewPassive` (or drop a if it becomes trivial).
   (h) `NewPassive += DExpMetaReductionRule2(Active,NewModifiers)`.
   `NewPassive += DExpMetaGenerationRule2M(Active,NewModifiers)`.
   (i) `DExpMetaSimplificationRule2(current,NewModifiers)`.
   If `current` becomes trivial, goto 6.
   (j) `NewPassive += DExpMetaReductionRule2(current,Modifiers)`.
   `NewPassive += DExpMetaGenerationRule2M(Active,NewModifiers)`.
3. `Active += current`.
4. `NewPassive += DExpMetaGenerationRule1(current)`.
5. `NewPassive += DExpMetaGenerationRule2(current,Active)`.
6. Postprocessing of `NewPassive`:

(a) `DExpMetaSimplificationRule2(NewPassive,Modifiers)`.

(b) `DExpMetaSubsumptionRuleMulti(NewPassive)`.

(c) `DExpMetaSubsumptionRule2(NewPassive,Active)`.

(d) `Passive += NewPassive`.

Remark 2.1. The algorithm must maintain the following invariants:

- Every rule must have a chance to be applied to every pair of *distinct* objects (or single object) unless one of the objects is eliminated. The algorithm above satisfies this requirement as long as objects get into `Active` (which is a requirement on the `Select` heuristic and one can ignore it for trading completeness for speed).

- The distinctness mentioned above must not ruin the completeness of the algorithm (this is a requirement on the set of rules). This is, for example, the case for the ordered resolution. (However, it is easy to modify the algorithm so that it would apply the rules to identical objects too.)

Another version of algorithm. There is another version of algorithm (invoked using the `--earlyreductions` option) which forces reduction and one-premise generation rules to apply as soon as the object is generated. Its use requires that any sequence of these rules (accompanied by the use of subsumption) terminates.

The two differences from the main algorithm are:

Preprocessing step: Apply the rules `ExpMetaDGenerationRule1` followed by `ExpMetaDSubsumptionRuleMulti` to the original objects as long as new objects are generated. The subsumption rules are applied to the "front" (i.e., to objects generated in one step from the original ones; then to objects generated in two steps, etc.). The obtained objects are added to `Passive`.

Step $5\frac{1}{2}$: Apply `ExpMetaDSubsumptionRuleMulti` to `NewPassive`. Then simultaneously apply `ExpMetaDGenerationRule1` and `ExpMetaDReductionRule2` to `NewPassive` and `Modifiers`. Repeat these steps for the set of newly generated objects until there are none. Add the obtained objects to `NewModifiers`.

Chapter 3

Specific prover

3.1 Objects and Notation

- The objects implemented in the semialgebraic prover are constructed of *polynomials*, which are (in theory) lists of monomials (in the implementation, a polynomial is a `C++` class that has certain methods and operators, in particular, arithmetic ones).

- A *monomial* is a product of *distinct* (!) *positive literals* and an integer coefficient (if the coefficient is zero, the product must be empty), a positive literal is of the form $P(\bar{t})$ (P is a predicate, $\bar{t}$ is a list of terms). If the coefficient is positive, the monomial is called a *positive monomial*. The multiplication is always made so that the powers higher than one are eliminated ($P(\bar{t})^2 = P(\bar{t})$).

- A *pseudomonomial* is a positive monomial or (1−positive monomial).

- The positive literals and positive monomials are *ordered* w.r.t. a certain partial order $<$. We assume that all our orderings are *reduction orderings* [BG01]. Recall that a reduction ordering is a well-founded transitive relation satisfying for all terms s, t, l, positions p and substitutions σ than whenever $s > t$ then $l[p/\sigma(s)] > l[p/\sigma(t)]$; here a *position* is an entity uniquely determining a subterm in l, and $l[p/s]$ is the result of substitution the subterm of l in position p with the term s.

- A deduction object is a *polynomial inequality*, i.e., a polynomial. The inequality states that this polynomial is nonnegative. In the low-level implementation, the constant term is stored separately.

- A modification object is a *polynomial equality*, i.e., again a polynomial. The inequality states that this polynomial is zero. The equality is stored (and written in this document) as a rule rewriting a monomial (typically, one of the maximal monomials) into the remaining part of the polynomial.

  The method `canApply()` of equality says whether the equality can be used in simplification and reduction rules (alternatively, it can be used in generation rules only and inside equalities simplification procedure).

In what follows, we use the notation described below (the same rules applies to the same letters with sub/superscripts):

- P, Q, R denote predicates;
- $\widetilde{L}$ denotes any literal;
- L_+ denotes a positive literal;
- M denotes a monomial;
- M_+ denotes a positive monomial;
- $\widetilde{M}$ denotes a pseudomonomial;
- S denotes a linear combination of monomials, including a constant term;
- X, Y, Z denote any sum of monomials, including a constant term, with integer coefficients;
- A, B, C, S_0 denote a linear combination of monomials with integer coefficients and no constant term;
- $\widetilde{S}_0$ denotes a linear combination of pseudomonomials with positive integer coefficients and no constant term;
- ρ, σ denote *substitutions*, which for technical reasons are not just substitute terms for variables but also multiply a polynomial by a positive integer constant;
- $M|_\rho$ is the result of applying ρ to M;
- a, b, c denote integer constants;
- a_+, b_+, c_+ denote positive integer constants;
- $\mathrm{mgu}(t_1, t_2)$ denotes the most general unifier of t_1 and t_2;
- x, y, z denote variables.

When we introduce a new letter by writing a sum containing it (like $M + S$), we suppose the elements of this sum to be disjoint (i.e., M does not occur as a summand of S). The same applies to a product.

3.2 Derivation rules

In what follows, we suppose that variable sets of all objects are disjoint. All derivation rules are applied only if the resulting object[1] fits the restrictions imposed by the user (the maximum degree and the maximum possible values of a non-constant monomial coefficient). We also assume that the premises are multiplied by appropriate positive integers if necessary.

[1] After it is put into a normal form using all one-premise simplification rules.

3.2.1 Rules that produce or modify deduction objects

Generation rules.

Two-premise.

(`Sum/Positive`)

$$\frac{S + M \geqslant 0, \qquad S' - M'M''_+ \geqslant 0}{(SM''_+)|_\rho + S'|_\rho \geqslant 0} \qquad (\rho = \mathrm{mgu}(M, M');\ M \text{ and } -M'M''_+ \text{ are } <\text{-maximal}, M'|_\rho > M''_+|_\rho)$$

(`Sum/Negative`)

$$\frac{S - M \geqslant 0, \qquad S' - M'M''_+ \geqslant 0}{(S(1 - M''_+) - M)|_\rho + S'|_\rho \geqslant 0} \qquad (\rho = \mathrm{mgu}(M, M');\ -M \text{ and } -M'M''_+ \text{ are } <\text{-maximal}, M'|_\rho > M''|_\rho)$$

The following two rules mimic the `Sum` rules, but have an equality as their second premise.

(`MultiplyRewrite/Positive`)

$$\frac{S + M \geqslant 0, \qquad M'M'' = S'}{(SM'')|_\rho + S'|_\rho \geqslant 0,} \qquad (\rho = \mathrm{mgu}(M, M');\ M \text{ is } <\text{-maximal}, M'|_\rho > M''|_\rho)$$

(`MultiplyRewrite/Negative`)

$$\frac{S + M \geqslant 0, \qquad M'M'' = S'}{(S(1 - M'') + M)|_\rho - S'|_\rho \geqslant 0} \qquad (\rho = \mathrm{mgu}(M, M');\ M \text{ is } <\text{-maximal}, M'|_\rho > M''|_\rho)$$

(`LiftAndProject`)

$$\frac{S + M_+ \geqslant 0, \qquad S' - M'_+ \geqslant 0}{(S(1 - M_+) + (S' - 1)M_+)|_\rho \geqslant 0} \qquad (\rho = \mathrm{mgu}(M_+, M'_+);\ M_+ \text{ and } -M'_+ \text{ are } <\text{-maximal})$$

Remark 2.1. Currently, we use only a restricted version of this rule when the premises are linear; for purely propositional instances we also require that the quadratic monomials of the conclusion has only even coefficients.

One-premise.

(`SumToProduct`)

$$\frac{\sum_i \widetilde{M_i} \geqslant a_+}{\prod_i (1 - \widetilde{M_i}) \leqslant 0}$$

Remark 2.2. This rule is a version of Lift-and-Project for the case when one of the premises is trivial.

Remark 2.3. Since this rule can always be immediately followed by converting its result into an equality, in the implementation it is directly applied as a modifier generation rule.

(`MultiplicativeFactorisation`)

$$\frac{S + MM'M'' \geqslant 0}{(S + MM'')|_\rho \geqslant 0} \qquad (\rho = \mathrm{mgu}(M, M'))$$

(`AdditiveFactorisation`)

$$\frac{S + M + M' \geqslant 0}{(S + M + M')|_\rho \geqslant 0} \qquad (\rho = \mathrm{mgu}(M, M'))$$

(`OrderedWeakening`) $$\frac{A - c_+\widetilde{L} \geqslant 0}{A \geqslant 0}$$

(here $\widetilde{L}$ is maximal in $A - \widetilde{L}$; A may not contain further occurrences of $\widetilde{L}$).

(`Extract`) $$\frac{\sum c_+^{(i)} L_+^{(i)} + \sum d_+^{(i)} L_+^{(i)*} \geqslant a_+}{\sum b_+^{(i)} L_+^{(i)} \geqslant b_+},$$

where the maximum literal of the premise is among the literals $L_+^{(i)}$ and

$$\sum d_+^{(i)} \leqslant a_+ - b_+, \quad \text{(thus the rule is sound)}$$

$$\sum_{c_+^{(i)} > b_+^{(i)}} (c_+^{(i)} - b_+^{(i)}) + \sum d_+^{(i)} > a_+ - b_+, \quad \text{(thus the result is not subsumed by the premise).}$$

We use this rule for $b_+^{(i)} = b_+ = 1$. In fact, it is enough to generate any set of clauses that subsumes the clauses that follow by this rule with this parameters (depending on the version of subsumption).

Remark 2.4. We assume that `RemoveMaxCoefficient` (see below) is not applicable to the premise of `Extract`. (Recall that one-premise simplification rules are always applied before any other rule.)

(`Legalize`) $$\frac{L_+ = X}{X \geqslant 0;\ X \leqslant 1}.$$

Reduction rules.

Recall that only equalities that have `canApply()` set to true can participate in these rules.

Two-premise.

(`SubstRewrite`) $$\frac{S + M_+ M'' \geqslant 0, \qquad M'_+ = S'}{S|_\rho + S'|_\rho M''|_\rho \geqslant 0} \qquad (\rho = \mathrm{mgu}(M_+, M'_+))$$

Simplification rules.

Recall that only equalities that have `canApply()` set to true can participate in these rules.

Two-premise.

(`Rewrite`) $$\frac{S + M_+ M'' \geqslant 0, \qquad M'_+ = S'}{S + S'|_\sigma M'' \geqslant 0} \qquad (M_+ = M'_+|_\sigma)$$

One-premise.

(`RemoveMaxCoefficient`) $$\frac{\widetilde{S}_0 + c\widetilde{M} - d \geqslant 0}{\widetilde{S}_0 + d\widetilde{M} - d \geqslant 0} \qquad (c > d > 0)$$

(`Rounding`) $$\frac{a_+ S \geqslant b}{S \geqslant \lceil \frac{b}{a_+} \rceil} \qquad (a_+ \geqslant 2)$$

(`Condensation`)
Inequality is replaced with its own factor that subsumes the inequality.

Subsumption rules.

Two-premise.

(`Subsumption`) $$\frac{S \geqslant 0, \quad S|_\sigma + \widetilde{S}'_0 + a_+ \geqslant 0}{S|_\sigma + \widetilde{S}'_0 + a_+ \geqslant 0 \text{ is deleted}}$$

Remark 2.5. Semialgebraic subsumption differs from the "classical" clause subsumption by the fact that $S|_\sigma + \widetilde{S}'_0 + a_+ \geqslant 0$ does not necessary contain all the monomials from $S|_\sigma \geqslant 0$ (since $S \geqslant 0$ might be stronger than $S|_\sigma + \widetilde{S}'_0 + a_+ \geqslant 0$). However, in the current implementation, we require that if a monomial from $S|_\sigma \geqslant 0$ occurs in $S|_\sigma + \widetilde{S}'_0 + a_+ \geqslant 0$, it occurs with the same coefficient.

Remark 2.6. We also use several restricted versions of subsumption.

3.2.2 Rules that produce or modify modification objects

Generation rules.

One-premise.

(`OppositeInequalities1`) $$\frac{0 \leqslant S}{S = 0} \qquad (S \geqslant 0 \text{ is trivial})$$

Two-premise.

(`OppositeInequalities2`) $$\frac{0 \leqslant S \qquad 0 \leqslant -S'}{S|_\sigma = 0} \qquad (\sigma = \mathrm{mgu}(S, S'))$$

Simplification rules.

One-premise.

(`GCD`) $$\frac{a_+ S = 0}{S = 0} \qquad (a_+ > 1)$$

Multi-premise.

We have just one rule `NormalizeEq` of this type; it applies set of new modifiers (equalities) against the existing set of equalities. As input we obtain a number d (maximal degree of possible equalities), we are provided with an existing set `Modifiers` and a set of new equalities `NewModifiers`. Equalities always appear in the form $M = P$, where the left-hand side is a single monomial, and every monomial in P is smaller than M (with respect to the current ordering).

Algorithm:

`For all` $Eq : M = P$ `in NewModifiers do`

1. `If the obvious upper bound for the left-hand side of` Eq `is equal to the obvious lower bound for the right-hand side of` Eq`, then for every monomial` N `in` Eq `whose degree is at least` 1`:`
   (a) `If its coefficient is positive and it belongs to the left-hand side of` Eq `or its coefficient is negative and it belongs to the right-hand side of` Eq`, deduce` $N = 1$`.`
   (b) `If its coefficient is negative and it belongs to the left-hand side of` Eq `or its coefficient is positive and it belongs to the right-hand side of` Eq`, deduce` $N = 0$`.`
2. `Similar as Step 1 for the case if the obvious lower bound for the left-hand side matches the obvious upper bound for the right-hand side.`
3. `For all variables` $x \in M$ `do`
   (a) `Deduce the equality` $Eq_1 : P = Px$`.`
   (b) `Simplify` Eq_1 `with NewModifiers`
   (c) `If` Eq_1 `is not trivial, and` $\deg(Eq_1) \leqslant \deg(Eq)$`, add` Eq_1 `to NewModifiers.`
4. `For all elements` $Eq' : M' = P'$ `from Modifiers do`
   (a) `For all monomials` N `in` Eq'`:`
      i. `If` $M \mid N$`, substitute in` Eq' N `with` $\frac{N}{M}P$`.`
   (b) `If any monomial in` Eq' `was simplified during the previous step, simplify` Eq' `with NewModifiers. If the result is not trivial, add` Eq' `to NewModifiers and goto` $4i$`.`
   (c) `For all monomials` N `in` Eq`:`
      i. `If` $M' \mid N$`, substitute in` Eq N `with` $\frac{N}{M'}P$`.`
   (d) `If any monomial in` Eq `was simplified during the previous step, simplify` Eq `with NewModifiers. If the result is not trivial, add` Eq `to NewModifiers and goto` 6`.`
   (e) `If` $\gcd(M', M) \neq 1$ `and the "-B" ("--buchberger") option is enabled, deduce equality of the form`

$$P\frac{M'}{\gcd(M',M)} - P'\frac{M}{\gcd(M',M)} = 0.$$

`We denote it by` $Eq_2 : L = Q$`.`

```
(f) If Eq₂ is a contradiction, return Contradiction.
(g) If deg(Eq₂) > d, goto 4i.
(h) Add Eq₂ to NewModifiers.
(i) End of cycle.
5. Add Eq : M = P to Modifiers.
6. End of cycle.
```

Remark 2.7. Every time we obtain a new equality that we never encountered before, we try to simplify it with `Modifiers`. This is done both for efficiency and to avoid creating equalities similar to the ones already existing in `Modifiers`, that in some cases leads to endless cycles.

Remark 2.8. Ordered simplification with an equality that has two or more maximal monomials is undefined. Therefore, the algorithm above needs to handle equalities with several maximal monomials separately. It creates a set where it puts such equalities and tries to simplify them with new equalities; in case a simplification results in an equality to have one maximal monomial, it is added to `Modifiers`.

Remark 2.9. If there exist monomials M, M' such that $M < M'$ and $M' \mid M$, it should not be possible to simplify with an equality of the kind $M' = \ldots M \ldots$ – this would immediately lead to endless cycles. Therefore, in orderings where this is possible, we check that the degree of an equality's left-hand side is not smaller than the degree of its right-hand side before trying to simplify with it.

Remark 2.10. In the first-order case, proper substitutions should be applied before looking for the gcd of monomials. For example, in step $4a$ we should try to find the minimal substitution that allows to divide N by M and apply it to both equalities before pairing them. However, simplifying an equality like described above, we would lose completeness (since applying a substitution makes an object weaker). On the other hand, simply adding thus created objects to `Modifiers` would create endless cycles, as deeper and deeper terms would change each other in the same equality, presenting new equalities but sending the algorithm into a cycle. After experimenting with these phenomenae, we decided to reject equalities that might be paired with another by means of a two-sided substitution (that is, a substitution that would have to be applied to both equalities in a pairing) from the very beginning. It has no effect on completeness (the inequalities still remain) and avoids endless cycles. We did not add this to the description of the algorithm for clarity.

3.2.3 Orderings

The inference rules above use certain orderings on monomials. These orderings extend known orderings on terms such as *KBO* [KB70], *LPO* [KL80, Der82], and some *ad hoc* orderings in style of Otter[Kal01]. At the moment, the following extensions are considered:

1. Multiset extension [BG01]: let Σ_1 and Σ_2 be two multisets of terms; then $\Sigma_1 >_{mul} \Sigma_2$ iff

   (a) $\Sigma_1 \neq \Sigma_2$

(b) Whenever $\Sigma_2(x) > \Sigma_1(x)$ then $\Sigma_1(y) > \Sigma_2(y)$ for some y such that $y > x$. Here $\Sigma_i(x)$ denotes the number of "occurrences" of x into multiset Σ_i.

2. Monomials of bigger degree are always bigger than monomials of smaller degree, multiset extension else;

3. Monomials of bigger degree are always smaller than monomials of smaller degree, multiset extension else;

4. Monomials of different degree are always incomparable, multiset extension else;

5. Monomials with bigger coefficient is always smaller than monomials with smaller coefficient, multiset extension else.

On ground monomials, orderings 1–3 and 5 are total. For completeness we need the ordering to be a *reduction ordering* [BG01] on *linear* monomials.

For ground instances we can also use a lexicographical extension instead of the multiset extension (indeed, in this case all terms are comparable and it is possible to order terms within every monomial).

3.3 Clause selection

The `Select` heuristic plays an important role in the proof search.

The simplest version of this heuristic implements FIFO (i.e., it takes the oldest deduction object). This guarantees that every derived object will go to `Active` and every pair of objects will have a chance to be processed by a two-premise derivation rule.

However, usually this heuristic is not the most efficient. Most straightforward situations when it is not so good are

- when the prover derives a unit clause of the form $P(x) \geqslant 1$ that will quickly transform into the equality $P(x) = 1$ so that the predicate P will be eliminated from everywhere;

- when the prover derives an inequality that is "stronger" than a Boolean clause (for example, $x + y + z \geqslant 2$);

- when the prover simplifies an inequality (formally, it must then throw it from `Active` into `Passive`).

However, the direct use of these ideas may lead to the case when some object is never selected (because other "good" objects generate "better" and "better" objects). To avoid this situation, we use the concept of "epochs" (an epoch is a sufficiently long period of time; objects that have been derived in an "ancient" epoch have priority over "recent" objects). Some of the heuristics (most notably, the unit clause heuristic) can violate the notion of epochs; others should respect it.

In general, our `Select` procedure sorts `Passive` using a list of heuristics (each heuristic is implemented as a partial order on objects) applied in a certain order and takes the "smallest" object:

```
if A is better than B w.r.t. the unit clause rule, select A;
else if B is better than A w.r.t. the unit clause rule, select B;
else if A is better than B w.r.t. the epoch when it was derived, select A;
...
else select the oldest object (out of A and B);
```

The list of heuristics is different for different values provided as the argument of the `-S` key. Here is the list of our "main" heuristics (we use also variations of these):

unit clause: $P(\text{variables}) = \text{const}$ is better than an inequality containing more than one literal.

origin: Some of our rules produce "more desirable" objects (for example, simplification rules), other ones produce "less desirable" objects (for example, `OrderedWeakening`). The integer parameter "origin" is thus set depending on how the object was derived.

epoch: An inequality derived in an "older" epoch is better. Note that if A subsumes B, it may inherit B's epoch if it is better than A's.

length: An inequality that contains less monomials is better.

degree: An inequality of smaller degree is better.

strength: A "stronger" inequality is better. The strength is the number of monomials that must be set to 0 or 1 without making the inequality trivial. Since checking the inequality for triviality is undecidable, we use an approximation.

lexicographic: A lexicographically smaller inequality is better.

max. monomials: An inequality is better if its list of maximal monomials is shorter.

history: An older inequality is better.

3.4 Prover options and output

In addition to the default (optimized by certain heuristics) and complete (a slower, but more robust) versions, our prover has a variety of options that can be used for experimenting as well as for tuning it towards specific classes of benchmarks. There are also command-line parameters that influence its output and job control.

The minimum output is

- "Contradiction." if a proof is found,
- "Satisfiable." if no more inferences are possible and prover feels it is a complete procedure,
- "Time limit exceeded." if time limit is exceeded,
- "Unknown." otherwise (i.e., prover can neither continue nor say "Satisfiable.").

Various keys may produce the proof (if found) and detailed information on the inference process.

In the description below, all *flags* are off by default. The words `INT` and `STRING` denote any integer or string argument, respectively.

3.4.1 Algorithm options

`--static, -T`. Chooses the experimental *static* mode, see Sect. 3.5. This works only with purely propositional formulas (i.e., without variables) and is included mostly for experimenting with the source code. Note that it forces the prover to ignore most other keys (including all algorithm-related ones).

`--complete`. Use a provably complete (but slower) version. This option implies several other options: `-Won -Dmax -w`. Note that even in this case the prover may become incomplete if it exceeds its resources (for example, if coefficients grow beyond the prescribed limit).

This option is required if you need the answer "Satisfiable." (rather than "Unknown").

`--maxdeg=INT, -dINT`. Maximum degree (0=unlimited, 2=default).

`--maxcoeff=INT, -bINT`. Absolute bound for the coefficients in inequalities (0=unlimited, default is the number of different literals in the input). Note that the prover may temporarily violate this bound in its internal procedures (for example, before an inequality is normalized using one-premise simplification rules).

This bound does not impose restrictions on the constant term.

`--maxeqcoeff=INT`. Absolute bound for the coefficients in equalities (0=unlimited, default=max(`maxcoeff`-1,1)).

`--wisesubs, -w`. An inequality of smaller degree cannot be subsumed.

`--subsumption=STRING, -DSTRING`. The argument (a string) selects subsumption.

- `max` (default) requires that all maximal monomials of the stronger polynomial occur also in the weaker polynomial with the same coefficient;
- `clause` requires that translation of Boolean clauses cannot be subsumed by other inequalities;
- `all` imposes no restrictions on the subsumption (except for those imposed by other keys).

`--equalities=STRING, -ESTRING`. Describes what kind of equalities are inferred by `MMetaGenerationRule`$\{1, 2\}$.

- `onestp` (default) limits the right-hand side to just one literal with an exception for the `SumToProduct` rule;
- `one` limits the right-hand side to just one literal (thus disabling the `SumToProduct` rule);
- `none` disables the use of equalities;
- `all` imposes no restrictions.

Note that these restrictions do not affect `MMetaSimplificationRuleMulti`.

`--ordweak=STRING, -WSTRING.` The argument "on" enables the use of the `OrderedWeakening` and `Extract` rules (default=off). Note that this option almost does not makes sense with `-Dall` argument, because the result of `OrderedWeakening` will be subsumed and `Extract` will apply only eventually (it applies rarely in any case).

`--nlinmax, -M.` Only linear inequalities have maximal monomials. (Thus non-linear inequalities will be used only when simplification or reduction rules make them linear right when they appear or if these rules are applied to `Passive`.)

It can speedup the inference in some cases if all necessary equalities are quickly inferred from the initial instance.

`--widesums.` Removes the ordering restriction for multipliers in the `Sum` rule.

`--select=INT, -SINT.` `Select` heuristic (1=FIFO, 2=select the shortest inequality, other=experimental; default=4).

`--monomial-ordering=INT, -OINT.` Specifies how to extend literal ordering to monomials.

1. lexicographic;
2. degree increasing, then lexicographic (the default);
3. degree decreasing, then lexicographic;
4. lexicographic if of equal degree, incomparable otherwise;
5. coefficient decreasing; then lexicographic;
6. multiset extension;
7. degree increasing, then multiset extension;
8. degree decreasing, then multiset extension;
9. multiset extension if of equal degree, incomparable otherwise;
10. coefficient decreasing; then multiset extension.

`--term-ordering, -F.` Selects terms ordering:

1. KBO — Knuth-Bendix ordering
2. LPO — Lexicographical path ordering
3. lexicographic (default);
4. lexicographic with variables (in this ordering we are allowed to compare variables. This order is total; see note below)
5. lexicographic + weights;
6. comparison by roots (we compare predicate names only)

The use of ordering 4 leads to incompleteness. Other term orderings coupled with a monomial ordering based on multiset extension (monomial orderings numbered 6–10) yield a complete first-order procedure. KBO, LPO, and lexicographic + weights coupled with a monomial ordering based on the lexicographical ordering (monomial orderings numbered 1–5) may lead to incompleteness.

`--buchberger`. Use a version of Buchberger's algorithm in `NormalizeEq`. This is more powerful than the default, but may take too much time.

`--liftandproject, -L`. Use restricted Lift-and-Project rule as described in (`LiftAndProject`). Note that this rule is currently off by default. Its current use is limited to producing new inequalities from inequalities that have more than one pair of opposite literals (thus expressing the parity function).

`--samedegreesums`. Use only same degree inequalities in the `Sum` rule.

`--experiment=INT, -XINT`. Turns on the experimental feature number `INT`. This option is intended for experimenting with the source code.

3.4.2 Output options

`--quiet, -q`. Suppress statistics.

`--print-input`. Print the input clauses.

`--print-sets`. Print Active/Passive/Modifiers after each step.

`--print-steps`. Print inference steps as soon as they are made.

`--print-proof=STRING, -pSTRING`. Selects the part of inference for outputting.

- `off` turns off output;
- `on` selects the proof;
- `all` selects the whole inference (which includes objects unused in the final proof).

`--format=STRING, -fSTRING`. Selects output format (out of `html`, `tex`, and `text`). Please note that not all parts of the inference are available in all formats. If you choose HTML or TEX, the prover will create a subdirectory Outputs and a subdirectory in it; look for the output files there.

3.4.3 Job control

`--history=STRING, -HSTRING`. Selects history (`forgettable` is faster; `keeping` is required for any kind of statistics or proof output).

`--time=INT, -tINT`. Sets the time limit in seconds (0=unlimited).

`--input-format=STRING, -iSTRING`. Input format (`tptp` or `cnf`).

3.5 Static proof search

As a part of the prover, we have implemented a static proof search procedure based on the semidefinite programming. Basically, this procedure tries to find a degree-2 static LS_+ proof of the initial system of inequalities. Please refer to [GHP02] for the description of this proof system, and to [Par03] for the description of the method of translating such proof search into an instance of the semidefinite programming. The only difference here is that we are working in a Boolean case, and, therefore, we add to our input the equalities $x^2 - x = 0$ for every propositional variable x in the input.

For solving the resulting semidefinite problem we used the CSDP library, written by Dr. Brian Borchers (please refer to [Bor99] and `http://www.nmt.edu/~borchers/csdp.html` for details on this library). Currently the static proof search is not integrated into the main algorithm and serves as an alternative procedure his works only with purely propositional formulas (i.e., without variables). It is an incomplete procedure (note that the simplest static proof may have degree 3 or higher), i.e., it outputs “Unknown” rather than “Satisfiable” when no proof is found.

The static proof search procedure is included mostly for experimenting with the source code as it is currently dominated by the main prover on almost all benchmarks.

Chapter 4

Theoretical base

4.1 Underlying proof system

Translate a clause $l_1 \vee \ldots \vee l_k$ of a propositional formula in CNF into the inequality $l_1 + \ldots + l_k - 1 \geqslant 0$ (replacing $\neg x$ by $1 - x$ where applicable); the obtained system of linear inequalities has the same 0/1-solutions as the original set of clauses (where 1 corresponds to True, and 0 corresponds to False). In what follows we describe a propositional proof system (first described in [GHP02]) that allows to derive a contradiction (i.e., the inequality -1) if and only if the original set of inequalities has no 0/1-solutions. The systems extends to the first-order logic in a natural way: propositional literals are replaced by first-order literals and the rule

$$\frac{f \geqslant 0}{f|_\sigma \geqslant 0}$$

allowing to apply a substitution σ is added.

We state the initial inequalities as axioms, and add also the axioms

$$\frac{}{x \geqslant 0}, \qquad \frac{}{1 - x \geqslant 0} \tag{1.1}$$

for every (propositional) variable x. The main derivation rule is

$$\frac{f_1 \geqslant 0, \ldots, f_k \geqslant 0}{\sum_{i=1}^{k} \alpha_i f_i \geqslant 0} \qquad \text{(where } \alpha_i \text{ are positive rational constants).} \tag{1.2}$$

The rules for establishing the fact, that we are interested in 0/1-solutions only, are

$$\frac{f \geqslant 0}{fx \mod (x^2 - x) \geqslant 0,} \qquad \frac{f \geqslant 0}{f(1-x) \mod (x^2 - x) \geqslant 0.} \tag{1.3}$$

Of course, we can (and we do) augment this system with other correct rules.

4.2 Completeness

Although the completeness of the underlying propositional proof system follows from the completeness of the Lovász-Schrijver proof system [LS91] (alternatively, the completeness of a linear fragment of this system extended by the `Rounding` rule follows from the completeness of Cutting Plane proof system [Gom63]), it does not suffice for claiming that our prover is a complete procedure. There are multiple points to settle:

- the actual prover uses subsumption;
- the actual prover uses simplification rules (a simplification rule *replaces* an inequality; hence the inequality cannot be used afterwards);
- the actual prover does not use "trivial" axioms (1.1) explicitly;
- the actual prover restricts (1.2) to linear combinations of two inequalities that cancel at least one monomial;
- the actual prover restricts (1.3);
- our aim is the completeness of the first-order extension of the propositional procedure.

In fact, not all options given to the prover make it complete. First of all, we must choose a particular version of subsumption.

Definition 2.1 (Ordered Subsumption). $X' \geqslant 0$ subsumes $X \geqslant 0$ if

- $X - X'|_\sigma$ is a positive linear combination of literals and a nonnegative constant term, for some substitution σ,
- each maximum literal of $X'|_\sigma$ occurs in X with the same coefficient.

This definition of subsumption corresponds to `-Dmax` option (implied by `--complete`).

Definition 2.2. A proof procedure is semi-complete if it does not say "Satisfiable" on a contradictory formula.

Theorem 2.1. The semi-algebraic prover run with `--complete` option is a semi-complete[1] and sound first-order procedure (as long as the restrictions[2] given in the parameters are not violated).

Proof. We first prove that our prover is complete for the propositional calculus. Then we use the standard *lifting* techniques [BG01] to prove semi-completeness in the first-order case. We split the proof into several lemmas. We deal only with linear inequalities in this proof (the algorithm *is* allowed to derive new linear (in)equalities using objects of arbitrary degree; however, all inequalities that we need for completeness are linear). The coefficients are not bounded; in real conditions, if a prover encounters an overflow, it must output the answer "Unknown" (if proof is not found).

In the following lemmas we argue about propositional clauses.

Note that one can translate a Boolean clause $L_1 \vee \ldots \vee L_k$ into an inequality $\sum_i L_i \geqslant 1$ that has the same set of 0/1-solutions.

Definition 2.3. Inequality α *sequentially subsumes* inequality β, if there is a sequence of inequalities $\alpha_1 = \alpha, \ldots, \alpha_k = \beta$, such that for every i, α_i subsumes α_{i+1}.

Remark 2.1. Note that if α sequentially subsumes β, then $\alpha \models \beta$.

Lemma 2.1. Let α and β be two inequalities that have a common maximal literal l. Then if α sequentially subsumes β, then l has the same coefficient in α and β.

[1] We delay a proof of "full" completeness to future versions of this paper.
[2] E.g., the maximum value of coefficients.

Proof. Follows from the definition of subsumption. Note that if l occurs in α_1, α_1 subsumes α_2 that does not contain l, and α_2 sequentially subsumes β, then β does not contain l. ☐

The completeness of the propositional fragment of our calculus relies on the completeness of the given-clause implementation of ordered resolution proved e.g. in [BG01]. The relative result in [BG01] can be reformulated as follows.

Theorem 2.2 (Theorem 4.13, [BG01]). Let $\mathcal{A}$ be the given-clause implementation of the rules of ordered resolution (including binary resolution, subsumption, and simplification rules). Let S_0 be a set of first-order clauses and $S_0 \to S_1 \to S_2 \to \dots$ be a *fair* derivation, where S_i is the set of *all* current clauses (i.e., both active, passive, and current), $S_i \to S_{i+1}$ means that S_{i+1} is the result of application of one iteration of the given-clause algorithm to S_i, and being fair means that every clause is eventually selected from passive. Then S_0 is unsatisfiable iff S_∞ contains the empty clause.

Lemma 2.2. Assume that algorithm $\mathcal{A}$ removes all linear inequalities from Passive in infinite time (i.e., the limit is $\emptyset$) and makes derivations w.r.t. the proof system consisting of the following rules:

1. `Sum` rule;
2. if $X \geqslant 0 \models Y \geqslant 0$ and $Y \geqslant 0$ is a translation of a Boolean clause using the same propositional variables as X, then we can derive $Y \geqslant 0$ from $X \geqslant 0$.

Let S be an unsolvable (in $\{0,1\}$) system of linear inequalities with integer coefficients. Then $\mathcal{A}$ yields a derivation of $-1 \geqslant 0$ from S. Moreover, it will yield it even if Ordered Subsumption is applied at some steps and new inequalities are added "by a third party".

Proof. We demonstrate that $\mathcal{A}$ simulates a prover based on ordered resolution. Namely, we show that $\mathcal{A}$ is able to play the role of such prover in Theorem 2.2.

First of all, rule 2 ensures that we may assume that the original clause set is a system of *Boolean* inequalities (i.e., translations of Boolean clauses). Note that the set of inequalities derived by rule 2 from $X \geqslant 0$ has the same set of solutions as $X \geqslant 0$.

The induction goes on the number of steps in the ordered resolution derivation. We now show the induction step, namely, if there is an ordered resolvent $\widetilde{C} = \widetilde{C}_1 \vee \widetilde{C}_2$ of (Boolean) clauses $\widetilde{D}_1 = l \vee \widetilde{C}_1$ and $\widetilde{D}_2 = \neg l \vee \widetilde{C}_2$, and our current clause set contains inequalities $\widetilde{D}'_i \geqslant 1$ such that $\widetilde{D}'_i \geqslant 1 \models \widetilde{D}_i$, then $\mathcal{A}$ will derive an inequality $\widetilde{D}' \geqslant 1$ such that $\widetilde{D}' \geqslant 1 \models \widetilde{C}$.

Note that rule 2 derives all clauses that are implied by $\widetilde{D}'_i \geqslant 1$. If $\widetilde{D}_i \geqslant 1$ is not present in the derivation at the moment, then it is sequentially subsumed by an inequality that is still present in the proof; therefore, we may assume that $\widetilde{D}'_i \geqslant 1$ sequentially subsumes $\widetilde{D}_i$.

Case 1: The pair of inequalities $\widetilde{D}'_1 \geqslant 1$, $\widetilde{D}'_2 \geqslant 1$ can be selected from inequalities derived by $\mathcal{A}$ in such a way that

- $\widetilde{D}'_i \geqslant 1$ sequentially subsumes $\widetilde{D}_i$,
- $\widetilde{D}'_1 = l + \widetilde{C}'_1 - c_{+,1}$,
- $\widetilde{D}'_2 = (1-l) + \widetilde{C}'_2 - c_{+,2}$,
- and l is maximal in $\widetilde{D}'_1$ and $\widetilde{D}'_2$.

Then by `Sum` one obtains

$$\widetilde{C}'_1 - c_{+,1} + \widetilde{C}'_2 - c_{+,2} \geqslant 1;$$

let us denote the left-hand side by $\widetilde{D'}$. Suppose that $\widetilde{D'} \geqslant 1 \not\models \widetilde{C}$, i.e., there exists an assignment I (not including the value for the literal l) such that

$$\widetilde{D'}[I] \geqslant 1 \tag{2.1}$$

and $\widetilde{C}[I] = \texttt{false}$. Then both $\widetilde{C}_i[I]$ are false. Now we fix the value of l to be 0 in $l + \widetilde{C}'_1[I] - c_{+,1} \geqslant 1 \models l \vee \widetilde{C}_1[I]$ (recall that I does not include the value for the literal l; let $I' = I \cup \{l = 0\}$), and get $\widetilde{C}'_1[I'] - c_{+,1} \geqslant 1 \models \widetilde{C}_1[I']$. As we assumed that $\widetilde{C}_i[I]$ is false, we also have $\widetilde{C}_i[I']$ is false and $\widetilde{C}'_1[I'] - c_{+,1} \geqslant 1 \models \texttt{false}$. Hence, $\widetilde{C}'_1[I'] < c_{+,1}+1$, that is, $\widetilde{C}'_1[I'] \leq c_{+,1}$. Since $\widetilde{C}'$ does not contain the literal l, we also have $\widetilde{C}'_1[I] \leq c_{+,1}$ By analogy, we have $\widetilde{C}'_2[I] \leq c_{+,2}$. Then both $\widetilde{C}'_i[I] \leqslant c_{+,i}$, which contradicts (2.1).

Case 2: At least one inequality, say $D'_1 \geqslant 1$ (the case for $D'_2 \geqslant 1$ is symmetrical), can be selected from the inequalities derived by $\mathcal{A}$ in such a way that $D'_1 \geqslant 1$ does not contain l at all.

We show that $\widetilde{D}'_1 \models \widetilde{C}$. Suppose that $\widetilde{D}'_1 \not\models \widetilde{C}$, i.e., there exists an assignment I (not including the value for the literal l) such that $\widetilde{D}'_1[I] \geqslant 1$ and $\widetilde{C}[I] = \texttt{false}$ (thus, $\widetilde{C_1}[I] = \texttt{false}$. Consider $I' = I \cup \{l = \texttt{false}\}$. Then $\widetilde{D}_1[I'] = \texttt{false}$ and still $\widetilde{D}'_1[I'] \geqslant 1$, which contradicts the fact that $\widetilde{D}'_1 \models \widetilde{D}_1$.

We now prove that we considered all possible cases. Assume that the conditions of cases 1 and 2 do not hold.

Since the condition of case 2 does not hold, the stronger inequality contains l. By Lemma 2.1 l is maximal in it and has coefficient 1, i.e., the condition of case 1 would hold. □

Lemma 2.3. Let $A[E]$ denote the result of simplifying an inequality A with all equalities in the set E using the `Rewrite` rule. Let E_1 denote the `Modifiers` set on some step, and E_2 denote the same set on the next step (we can, of course, apply induction on the number of steps afterwards and obtain the statement for any later step, not only the next one). Then $(A[E_1])[E_2] = A[E_2]$ (the simplification result does not depend on when the modification objects are applied).

Proof. E_2 is obtained from E_1 by applying the `NormalizeEq` rule to E_1 and some set E'_1 of `NewModifiers`. We proceed by induction on the number of non-trivial steps in the algorithm described in Section 5.2.

First, note that the only steps we have to consider in the `NormalizeEq` algorithm are simplification steps, as being the only ones that may remove an equality from the set of `Modifiers`. Other steps only add new equalities to the set and may not affect the statement of this lemma (if $E_1 \subseteq E_2$, the claim is trivial). Also note that we may apply induction on the number of steps in simplifying an inequality with an equality, so in the remainder we consider only one such step.

Therefore, let us consider the case of simplifying an inequality $A : Q + N \leqslant a$ with an equality $e : M = P$, $M \mid N$ (here M,N denote monomials, Q,P – polynomials). The result will be $A[e] : Q + \frac{N}{M}P \leqslant a$. Now let us see what happens after simplifying e with $e_1 : M_1 = P_1$. We consider two cases:

1. $e : M = P = P' + M'$, $M_1 \mid M'$. Then after simplification we obtain $e_2 : M = P' + \frac{M'}{M_1} P_1$, the left-hand side (that is, the maximal monomial) will not change by the properties of monomial orderings, and

$$A[e][e_1] : Q + \frac{N}{M}(P' + \frac{M'}{M_1} P_1) \leqslant a$$

$$A[e_2] : Q + \frac{N}{M}(P' + \frac{M'}{M_1} P_1) \leqslant a,$$

which proves the statement in this case.

2. $M_1 \mid M$. Then after simplifying e with e_1 we get $e_2 : \frac{M}{M_1} P_1 = P$. In case when the maximal monomial of the new equality comes from $\frac{M}{M_1} P_1$, we proceed similar to the previous case, and it is easy to see that $A[e] = (A[e_1])[e_2]$. In the opposite case, we obtain by similar considerations that $A[e_1] = (A[e])[e_2]$. This proves the claim in this case, because in this case $E_1 = \{e\}, E_2 = \{e_1, e_2\}$, and $(A[E_1])[E_2] = A[E_2]$.

□

Definition 2.4. *Simplification*:

$$\frac{L_+ = X;\ Y \geqslant 0}{Y|_{L_+/X} \geqslant 0;\ X \geqslant 0;\ X \leqslant 1},$$

where $Y|_{L_+/X}$ denotes the result of replacing L_+ by X in Y.

Remark 2.2. Note that the simplification is composed of two actual rules of the prover: `Rewrite` and `Legalize`.

Lemma 2.4. The claim of Lemma 2.2 still holds even if we allow an "adversary" to apply simplification at any moment.

Proof. The proof goes by induction on the number of propositional variables. Note that the simplification applied to the whole proof (by Lemma 2.3 we can always think it is applied to the whole proof) yields an unsolvable system of inequalities. Then we can apply Lemma 2.2 to this system (which has one propositional variable less). □

Lemma 2.5. Let S be an unsolvable (in $\{0, 1\}$) system of linear inequalities with integer coefficients. Then the system that uses the `SumRule`, `OrderedWeakening`, `RemoveMaxCoefficient` and `Extract` rules yields a derivation of $-1 \geqslant 0$ from S even if we allow an "adversary" to apply subsumption and simplification at any moment.

Proof. Rule 2 is simulated by iterative applications of `RemoveMaxCoefficient`, `Extract` and `OrderedWeakening` rules. Note that we need to derive only clauses that do not subsume each other (and are not subsumed by the premise of rule 2). □

Proof of Theorem 2.1. The soundness of the semi-algebraic prover follows from the (trivially checkable) fact that our rules preserve satisfiability. By Lemma 2.5, the semi-algebraic prover is complete for the propositional case.

In order to prove semi-completeness in the first-order case, it suffices to show that, first, the deduction rules of our prover preserve unsatisfiability, and second, that the algorithm can always proceed on an unsatisfiable set.

In order to show that the rules of our prover preserve unsatisfiability, we consider a finite unsatisfiable set of ground instances of the kept first-order objects (the existence of such a finite unsatisfiable set is guaranteed by Herbrand's theorem) and show that all rules of our prover are *liftable*, that is, if an object $\mathbf{O'}$ is obtained from objects $\mathbf{O'_1}, \ldots, \mathbf{O'_k}$ by an inference rule, and $\mathbf{O'_i}$ is an instance of $\mathbf{O_i}$, for every $i = 1, \ldots, k$, then there exists an object $\mathbf{O}$ obtained from $\mathbf{O_1}, \ldots, \mathbf{O_k}$ by the same inference rule such that $\mathbf{O'}$ is an instance of $\mathbf{O}$. Obviously, liftable inference rules preserve unsatisfiability.

The fact that the first-order analog of Lemma 2.2 holds can be verified by a routine check (the proof goes along the same lines literally; note that usual resolution factorization rule is simulated by `Additive Factorization` followed by `RemoveMaxCoefficient`). It implies liftability of the `Sum` rule. Obviously, subsumption preserves unsatisfiability.

Dealing with simplifications is a bit trickier: Let $S' \geq 0$ be a ground instance of a first-order inequality $S \geq 0$, and $L'_+ = X'$ be a ground instance of a first-order equality $L_+ = X$. Let $L_+ = X$ simplifies $S' \geq 0$ to $S'_1 \geq 0$. If in addition *every* ground instance of $S \geq 0$ can be simplified with *some* ground instance of $L_+ = X$, there is no problem in lifting this kind of simplification by introducing a first-order version of the simplification rule (which substitutes $S \geq 0$ with the result of simplification $S_1 \geq 0$ such that $S'_1 \geq 0$ is an instance of $S_1 \geq 0$). Note, however, that if there exists a ground instances of $S \geq 0$ that cannot be simplified with a ground instance of $L_+ = X$, we are not allowed to delete the first-order inequality—we do not know in advance which ground instances of inequalities (as provided by Herbrand theorem) should be taken to dedice a contradiction. Therefore, in addition to first-order simplification, we also consider *reduction* (the rules `Rewrite` and `SubstRewrite`, respectively, of our prover). Apart from the fact that reduction uses a most general unifier and simplification uses an instance, the difference between them is that the reduction rule *adds* a new inequality to the set of inequalities, while the simplification rule *replaces* the inequality. It can be seen that when the reduction rule is applied, all needed ground instances are preserved, and the (ground) simplification rule is, thus, liftable.

Note that, since simplified and reduced objects are moved into `Passive`, it is always possible to find such ground instances of first-order objects in `Active` and `Passive` that the prover deduces a contradiction being run with these ground sets. Therefore, by liftability of our rule, a step of the algorithm can always be made. (Though it is possible that the algorithm starts generating all the same objects in a loop, but it never says "Satisfiable" on an unsatisfiable instance). □

Chapter 5

Empirical data

Our prover is better viewed as a framework for experiments with semialgebraic (and other) derivation procedures. Thus it is not surprising that it does not compete well with other ("optimized" and "finalized") provers on many benchmarks. However, even this implementation shows that the semialgebraic prover outperforms traditional provers on several classes of benchmarks by orders of magnitude.

In what follows, we describe in details the statistics on the classes where our prover performs well and compare it to other provers (namely, Vampire version 6.0 [RV01] and E version 0.8 [Sch02] taken from the CASC-19 competition [SS03] cite[1]). Then we show briefly the statistics of our prover on other examples.

5.1 Deriving strong inequalities from 2-clauses

One of the features of semialgebraic proofs that allow exponential speed-up over resolution proofs is "packing" several clauses into one. For example, a single linear inequality

$$x_1 + \ldots + x_n \geqslant n - 1$$

can be expressed only by $\binom{n}{2}$ 2-clauses. Reasoning with such inequalities (instead of the underlying 2-clauses) leads to provably exponential proofs shortening compared to resolution or bounded-depth Gentzen-style proof systems on the propositional pigeon-hole principle (cf., e.g., [Pud99]). A similar behavior in shown by empirical results on `hgen8` propositional random formulas and the Checkerboard & Dominoes problems from the TPTP library[2]. Our prover behaves well also on a first-order extension of the pigeon-hole formulas.

While other provers work faster on small size benchmarks of this type, they produce huge proofs and their performance deteriorates quickly as the size of benchmarks grows.

5.1.1 The pigeon-hole principle (propositional version)

The experiments were made on an Intel P-III-1000 computer with RAM size restricted to 500 Mbytes. We experimented with the original formulation (see Sect. 6.1 for details), "inverted" formulation (where all literals are replaced by their negations), and "shuffled" formulation

[1] `http://http://www.cs.miami.edu/~tptp/CASC/19/`

[2] `http://www.tptp.org/`

(some of the literals are replaced with their negations, the order of clauses and variables is changed; all this is done at random).

Vampire-6 and E were unable to solve the last benchmarks within the time limit of 3 minutes.

The keys provided to our prover were: `-w -M -Dall`.

Problem name	Time			saprover			Proof size		
	sap.	Vamp.	E	Generated	Deleted	Iterations	sap.	Vamp.	E
php_4_3	0	0	0	102	27	36	58	119	88
php_5_4	1	0	0	213	60	67	125	312	371
php_6_5	6	0	7	386	110	113	231	960	2067
php_7_6	17	30	10	635	180	177	385	4293	8821
php_7_6_inv	22	34	180	761	306	177	385	4287	-
php_7_6_seed1	16	5	48	573	208	177	385	5598	13020
php_7_6_seed2	17	8	64	597	203	177	385	5261	11594
php_7_6_seed3	18	5	43	629	199	177	385	4529	12185
php_7_6_seed4	18	7	100	649	220	177	385	6260	13490
php_7_6_seed5	16	9	142	608	204	177	385	5942	12776
php_7_6_seed6	16	10	20	580	198	177	385	7085	9947
php_7_6_seed7	16	7	57	590	206	177	385	5552	13117
php_7_6_seed8	17	12	177	604	214	177	385	6349	13213
php_7_6_seed9	18	10	64	638	214	177	385	6946	14484
php_8_7	41	180	180	974	273	262	596	-	-
php_8_7_inv	52	180	180	1170	469	262	596	-	-
php_8_7_seed1	39	180	180	936	318	262	596	-	-
php_8_7_seed2	38	180	180	891	317	262	596	-	-
php_8_7_seed3	39	180	180	881	319	262	596	-	-
php_8_7_seed4	41	180	180	918	338	262	596	-	-
php_8_7_seed5	37	180	180	898	305	262	596	-	-
php_8_7_seed6	39	180	180	946	309	262	596	-	-
php_8_7_seed7	39	180	180	906	324	262	596	-	-
php_8_7_seed8	40	180	180	928	330	262	596	-	-
php_8_7_seed9	38	180	180	887	317	262	596	-	-
php_9_8	89	180	180	1417	392	371	873	-	-

We provide an example of such short proof in Sect. 6.1.

5.1.2 The pigeon-hole principle (first-order variation)

Our prover is able to use the same "propositional" technique even if the pigeon-hole principle is moderately hidden by replacing 0-ary predicates with reasonable first-order terms. The resulting benchmarks are harder both for our prover and other two provers used in the experiments. The experiments were made on an Intel P-IV-1800 computer with RAM size restricted to 3 Gbytes (which was its physical RAM size).

Vampire-6 and saprover were unable to solve several benchmarks within the time limit of 2 hours. E exited without answer on two benchmarks before the time limit.

The keys provided to our prover were: `-w -M -Dall -b1 --earlyreductions --samedegreesums`.

Problem name	Time			saprover			Proof size		
	sap.	Vamp.	E	Generated	Deleted	Iterations	sap.	Vamp.	E
php_4_3	49	0	0	218	39	57	64	178	108
php_4_3_inv	47	1	0	152	42	43	64	282	147
php_4_3_seed1	36	0	0	226	57	50	64	175	96
php_4_3_seed2	19	0	0	144	37	44	64	139	97
php_4_3_seed3	77	0	0	249	45	58	64	156	83
php_4_3_seed4	124	0	0	420	129	61	64	174	116
php_4_3_seed5	23	0	0	159	43	46	64	183	128
php_4_3_seed6	17	0	0	129	28	42	64	173	102
php_4_3_seed7	32	0	0	193	41	47	64	168	95
php_4_3_seed8	135	0	0	543	193	66	64	175	128
php_4_3_seed9	62	0	0	276	48	59	64	161	114
php_5_4	6113	244	58	1081	309	300	137	1026	695
php_5_4_inv	7200	7200	7200				-	-	-
php_5_4_seed1	1305	7200	1151	1052	438	120	137	-	1843
php_5_4_seed2	2283	7200	2058	1503	546	141	137	-	2098
php_5_4_seed3	4101	7200	2058	2270	553	218	137	-	2148
php_5_4_seed4	7200	6051	677				-	1964	1462
php_5_4_seed5	1049	7200	7200	1313	416	221	157	-	-
php_5_4_seed6	7200	6192	269				-	1649	1434
php_5_4_seed7	1913	7200	2411	1212	333	153	137	-	3043
php_5_4_seed8	2147	7200	345	1117	272	149	137	-	2520
php_5_4_seed9	7200	7200	430				-	-	1082

We provide an example of such short proof (for a formula with terms of smaller height) in Sect. 6.2.

5.1.3 "Record" `hgen8` benchmarks from SAT Competition 2003

The experiments were made on an Intel P-III-1000 computer with RAM size restricted to 500 Mbytes. The benchmarks (being randomly generated themselves) were "shuffled" before providing to the provers. Also the "record" benchmark of this series (the smallest unsolved unsatisfiable formula of SAT Competition 2003[3]) was included in the experiments.

The number n in the name `hgen8-shuffled.`$n.s$ stands for (an upper bound on) the number of variables. The number s denotes the seed used for randomization.

In the table below, "res" means that the prover resigned before the time limit (answers "Unknown" for saprover, "Cannot prove" for Vampire-6); "600" means that the prover did not answer within the time limit of 10 minutes.

The keys provided to our solver were: `-w -M -Dall`.

Problem name	Time			saprover			Proof size		
	sap.	Vamp.	E	Generated	Deleted	Iterations	sap.	Vamp.	E
hgen8-shuffled.10.01	0	0	0	87	27	30	51	74	49
hgen8-shuffled.10.02	0	0	0	95	50	32	59	59	35
hgen8-shuffled.10.03	0	0	0	83	27	30	51	74	47

[3]See `http://www.satlive.org/SATCompetition/2003/`

hgen8-shuffled.10.04	0	0	0	76	24	30	51	73	48
hgen8-shuffled.10.05	0	0	0	85	27	30	51	77	42
hgen8-shuffled.20.01	2	0	0	199	70	67	125	261	366
hgen8-shuffled.20.02	2	0	0	197	72	67	125	200	252
hgen8-shuffled.20.03	1	0	0	204	65	67	125	193	292
hgen8-shuffled.20.04	2	0	0	220	70	67	125	239	262
hgen8-shuffled.20.05	2	0	0	201	65	67	125	251	302
hgen8-shuffled.30.01	4	0	0	281	101	88	161	339	363
hgen8-shuffled.30.02	3	0	0	274	95	89	163	306	1039
hgen8-shuffled.30.03	3	0	0	272	93	89	163	365	642
hgen8-shuffled.30.04	3	0	0	264	89	89	163	390	749
hgen8-shuffled.30.05	3	0	0	281	95	89	163	492	530
hgen8-shuffled.40.01	6	0	1	361	116	111	201	557	1380
hgen8-shuffled.40.02	6	0	0	366	109	112	204	657	1002
hgen8-shuffled.40.03	6	0	0	357	111	112	204	779	1298
hgen8-shuffled.40.04	6	0	0	359	113	112	204	706	1324
hgen8-shuffled.40.05	5	0	0	339	113	111	201	654	1048
hgen8-shuffled.50.01	9	0	6	431	136	137	255	1261	4567
hgen8-shuffled.50.02	8	0	4	404	126	134	242	1139	3668
hgen8-shuffled.50.03	9	1	2	453	137	134	242	1818	2676
hgen8-shuffled.50.04	9	0	4	443	134	134	242	1172	4099
hgen8-shuffled.50.05	8	0	5	408	124	134	242	1526	4697
hgen8-shuffled.60.01	13	5	112	508	155	157	283	5003	19605
hgen8-shuffled.60.02	11	17	449	463	146	155	277	5068	24228
hgen8-shuffled.60.03	11	5	104	482	151	155	277	4439	9725
hgen8-shuffled.60.04	12	1	19	516	180	158	286	2061	9634
hgen8-shuffled.60.05	11	4	6	456	146	158	286	3095	7527
hgen8-shuffled.70.01	16	130	267	559	174	180	324	12974	48196
hgen8-shuffled.70.02	16	78	600	545	166	178	318	13181	-
hgen8-shuffled.70.03	18	27	600	589	178	182	330	8673	-
hgen8-shuffled.70.04	16	17	312	569	171	179	321	6825	52650
hgen8-shuffled.70.05	16	114	510	551	168	181	327	10177	65320
hgen8-shuffled.80.01	21	126	600	623	209	200	356	9604	-
hgen8-shuffled.80.02	21	600	600	604	186	202	362	-	-
hgen8-shuffled.80.03	20	600	600	624	201	204	368	-	-
hgen8-shuffled.80.04	22	600	600	665	189	203	365	-	-
hgen8-shuffled.80.05	20	res	600	606	190	202	362	-	-
hgen8-shuffled.90.01	26	600	600	691	219	225	407	-	-
hgen8-shuffled.90.02	26	600	600	725	226	228	409	-	-
hgen8-shuffled.90.03	25	600	600	697	223	226	406	-	-
hgen8-shuffled.90.04	25	res	600	697	204	225	403	-	-
hgen8-shuffled.90.05	26	res	600	702	225	225	403	-	-
hgen8-shuffled.100.01	31	res	600	769	237	248	444	-	-
hgen8-shuffled.100.02	31	res	600	797	241	248	444	-	-
hgen8-shuffled.100.03	31	600	600	757	231	247	441	-	-
hgen8-shuffled.100.04	34	600	600	790	241	246	438	-	-
hgen8-shuffled.100.05	32	600	600	757	236	248	440	-	-

hgen8-shuffled.110.01	36	600	600	790	238	268	476	-	-
hgen8-shuffled.110.02	38	600	600	798	252	271	485	-	-
hgen8-shuffled.110.03	40	600	600	858	266	270	482	-	-
hgen8-shuffled.110.04	39	600	600	807	248	269	479	-	-
hgen8-shuffled.110.05	40	600	600	844	260	273	491	-	-
hgen8-shuffled.120.01	47	600	600	910	278	294	526	-	-
hgen8-shuffled.120.02	48	600	600	923	276	294	526	-	-
hgen8-shuffled.120.03	46	600	600	893	272	292	520	-	-
hgen8-shuffled.120.04	51	600	600	920	306	295	533	-	-
hgen8-shuffled.120.05	46	600	600	903	269	292	520	-	-
hgen8-shuffled.130.01	59	600	600	1015	305	313	555	-	-
hgen8-shuffled.130.02	60	600	600	1003	301	317	567	-	-
hgen8-shuffled.130.03	53	600	600	960	300	315	561	-	-
hgen8-shuffled.130.04	55	600	600	1001	309	314	558	-	-
hgen8-shuffled.130.05	58	600	600	983	297	315	561	-	-
hgen8-shuffled.140.01	65	600	600	1080	343	338	602	-	-
hgen8-shuffled.140.02	66	600	600	1064	341	342	614	-	-
hgen8-shuffled.140.03	64	600	600	1062	334	341	611	-	-
hgen8-shuffled.140.04	71	600	600	1060	314	344	620	-	-
hgen8-shuffled.140.05	62	600	600	1026	323	336	596	-	-
hgen8-shuffled.150.01	74	600	600	1142	365	361	643	-	-
hgen8-shuffled.150.02	77	600	600	1143	353	363	650	-	-
hgen8-shuffled.150.03	78	600	600	1165	356	364	652	-	-
hgen8-shuffled.150.04	77	600	600	1156	360	362	650	-	-
hgen8-shuffled.150.05	71	600	600	1102	349	359	637	-	-
hgen8-shuffled.160.01	92	600	600	1200	355	384	684	-	-
hgen8-shuffled.160.02	88	600	600	1193	355	386	690	-	-
hgen8-shuffled.160.03	88	600	600	1158	348	385	687	-	-
hgen8-shuffled.160.04	89	600	600	1218	374	385	687	-	-
hgen8-shuffled.160.05	90	600	600	1197	373	383	681	-	-
hgen8-shuffled.170.01	93	600	600	1272	402	403	713	-	-
hgen8-shuffled.170.02	93	600	600	1226	390	403	713	-	-
hgen8-shuffled.170.03	93	600	600	1252	396	404	716	-	-
hgen8-shuffled.170.04	107	600	600	1264	388	408	728	-	-
hgen8-shuffled.170.05	86	600	600	1233	381	405	719	-	-
hgen8-shuffled.180.01	res	600	600				-	-	-
hgen8-shuffled.180.02	122	600	600	1328	390	426	754	-	-
hgen8-shuffled.180.03	116	600	600	1336	405	428	760	-	-
hgen8-shuffled.180.04	121	600	600	1337	410	434	782	-	-
hgen8-shuffled.180.05	120	600	600	1339	408	430	766	-	-
hgen8-shuffled.190.01	129	600	600	1357	418	450	798	-	-
hgen8-shuffled.190.02	res	600	600				-	-	-
hgen8-shuffled.190.03	127	600	600	1399	434	449	795	-	-
hgen8-shuffled.190.04	119	600	600	1402	425	448	792	-	-
hgen8-shuffled.190.05	122	600	600	1390	418	450	798	-	-
hgen8-shuffled.200.01	155	600	600	1478	449	475	842	-	-
hgen8-shuffled.200.02	146	600	600	1513	453	477	851	-	-

hgen8-shuffled.200.03	156	600	600	1512	462	476	848	-	-
hgen8-shuffled.200.04	156	600	600	1513	453	473	839	-	-
hgen8-shuffled.200.05	154	600	600	1516	464	476	848	-	-
hgen8-shuffled.210.01	170	600	600	1605	492	502	898	-	-
hgen8-shuffled.210.02	155	600	600	1654	486	510	923	-	-
hgen8-shuffled.210.03	158	600	600	1534	467	497	887	-	-
hgen8-shuffled.210.04	146	600	600	1513	473	497	883	-	-
hgen8-shuffled.210.05	res	600	600				-	-	-
hgen8-shuffled.220.01	184	600	600	1647	511	517	915	-	-
hgen8-shuffled.220.02	179	600	600	1662	477	517	915	-	-
hgen8-shuffled.220.03	174	600	600	1578	474	516	916	-	-
hgen8-shuffled.220.04	169	600	600	1600	493	519	921	-	-
hgen8-shuffled.220.05	600	600	600				-	-	-
hgen8-shuffled.230.01	202	600	600	1708	516	545	971	-	-
hgen8-shuffled.230.02	211	600	600	1739	536	541	959	-	-
hgen8-shuffled.230.03	200	600	600	1630	496	540	956	-	-
hgen8-shuffled.230.04	res	600	600				-	-	-
hgen8-shuffled.230.05	res	600	600				-	-	-
hgen8-shuffled.240.01	215	600	600	1724	529	567	1009	-	-
hgen8-shuffled.240.02	220	600	600	1769	545	569	1015	-	-
hgen8-shuffled.240.03	226	600	600	1809	563	568	1012	-	-
hgen8-shuffled.240.04	213	600	600	1707	545	561	991	-	-
hgen8-shuffled.240.05	229	600	600	1779	528	568	1012	-	-
hgen8-shuffled.250.01	251	600	600	1783	538	585	1035	-	-
hgen8-shuffled.250.02	245	600	600	1877	571	589	1047	-	-
hgen8-shuffled.250.03	279	600	600	1918	578	589	1047	-	-
hgen8-shuffled.250.04	228	600	600	1834	555	588	1044	-	-
hgen8-shuffled.250.05	res	600	600				-	-	-
hgen8-shuffled.260.01	264	600	600	1933	588	614	1094	-	-
hgen8-shuffled.260.02	317	600	600	1767	584	806	-	-	-
hgen8-shuffled.260.03	268	600	600	1978	604	613	1095	-	-
hgen8-shuffled.260.04	274	600	600	1931	578	609	1080	-	-
hgen8-shuffled.260.05	269	600	600	1960	582	614	1098	-	-
hgen8.260.record	238	600	600	1851	564	605	1067	-	-
hgen8-shuffled.270.01	313	600	600	2018	594	634	1126	-	-
hgen8-shuffled.270.02	600	600	600				-	-	-
hgen8-shuffled.270.03	260	600	600	1966	576	634	1126	-	-
hgen8-shuffled.270.04	264	600	600	1961	586	631	1117	-	-
hgen8-shuffled.270.05	289	600	600	1960	596	637	1135	-	-
hgen8-shuffled.280.01	340	600	600	2150	629	660	1176	-	-
hgen8-shuffled.280.02	335	600	600	2073	638	654	1162	-	-
hgen8-shuffled.280.03	301	600	600	2043	626	657	1167	-	-
hgen8-shuffled.280.04	304	600	600	2097	660	660	1180	-	-
hgen8-shuffled.280.05	600	600	600				-	-	-
hgen8-shuffled.290.01	365	600	600	2129	641	680	1208	-	-
hgen8-shuffled.290.02	600	600	600				-	-	-
hgen8-shuffled.290.03	332	600	600	2119	628	679	1205	-	-

hgen8-shuffled.290.04	362	600	600	2112	640	683	1217	-	-
hgen8-shuffled.290.05	354	600	600	2121	627	680	1208	-	-
hgen8-shuffled.300.01	341	600	600	2192	683	701	1247	-	-
hgen8-shuffled.300.02	406	600	600	2254	690	705	1255	-	-
hgen8-shuffled.300.03	338	600	600	2173	681	698	1234	-	-
hgen8-shuffled.300.04	600	600	600				-	-	-
hgen8-shuffled.300.05	373	600	600	2190	672	702	1246	-	-

5.1.4 Checkerboard & Dominoes problems from the TPTP library

While our prover is not as fast as others on these problems, it generates substantially shorter proofs. We conjecture that if similar problems of larger size were available, the semialgebraic prover could outperform other provers on these benchmarks. The experiments were made on an Intel P-III-1000 computer with RAM size restricted to 500 Mbytes.

The keys provided to our prover were: `-w -M -Dall`.

Problem name	----Time----			-----------saprover-----------			--Proof size--		
	sap.	Vamp.	E	Generated	Deleted	Iterations	sap.	Vamp.	E
PUZ015-2.006.p	40	0	3	1241	245	284	254	1095	1108
PUZ016-2.005.p	13	0	0	824	186	178	161	314	315

5.2 Reasoning about integers and modular arithmetic

Semialgebraic systems of moderate degree allow to reason about integers and, hence, modular arithmetic (see [GHP02]). We succeeded in quick search for short proofs that use mod 2 arithmetic.

The examples used in the experiments include degree three Tseitin tautologies and verification problems for xor chains (by Lintao Zhang and Sharad Malik); these problems form a special case of degree three Tseitin tautologies. Note that that the xor-chains benchmarks were among *SAT 2002 Competition challenging benchmarks*[4], and various versions of Tseitin tautologies (usually of slightly greater degree) form a common (and historically first) example of benchmarks provably exponentially hard for resolution and other proof systems.

5.2.1 Degree three Tseitin tautologies

The experiments were made on an Intel P-III-1000 computer with RAM size restricted to 500 Mbytes.

The number n in the name `tseitin_t_`n`_03` stands for the number of vertices in a graph.

Vampire-6 and E were unable to solve the last several problems within the time limit of 10 minutes. (Vampire-6 returned the answer "Cannot prove" after running on `tseitin_t_25_03` for 274 seconds.)

The keys provided to our prover were: `-L -S5 -O5 -b2 --samedegreesums`.

Problem name	----Time----			------------saprover----------			--Proof size--		
	sap.	Vamp.	E	Generated	Deleted	Iterations	sap.	Vamp.	E
tseitin_t_03_03	0	0	0	23	8	9	15	27	14

[4]See `http://www.satlive.org/SATCompetition/2002/`

tseitin_t_05_03	0	0	0	65	21	27	47	68	40
tseitin_t_07_03	0	0	0	64	9	51	54	151	135
tseitin_t_09_03	0	0	0	91	12	71	70	258	280
tseitin_t_11_03	0	0	0	146	30	97	89	436	527
tseitin_t_13_03	0	0	1	130	15	99	102	723	1128
tseitin_t_15_03	0	0	6	146	17	115	118	1449	3038
tseitin_t_17_03	1	1	42	162	19	131	134	2343	7059
tseitin_t_19_03	1	5	600	178	21	147	150	4857	-
tseitin_t_21_03	1	17	600	194	23	163	166	8551	-
tseitin_t_23_03	2	126	600	210	25	179	182	18127	-
tseitin_t_25_03	2	600	600	226	27	195	198	-	-
tseitin_t_27_03	3	600	600	242	29	211	214	-	-
tseitin_t_29_03	4	600	600	258	31	227	230	-	-
tseitin_t_39_03	12	600	600	338	41	307	310	-	-
tseitin_t_49_03	29	600	600	418	51	387	390	-	-

We provide an example of such short proof in Sect. 6.5. An example of a short proof for a first-order variation of Tseitin tautology is given in Sect. 6.6.

5.2.2 XOR-chains benchmarks

The experiments were made on an Intel P-III-1000 computer with RAM size restricted to 500 Mbytes. Vampire-6 and E did not solve any of these benchmarks within the time limit of 30 minutes. (On several benchmarks, Vampire-6 terminated in 2–5 minutes with the answer "Cannot prove". E terminated several times with "out of memory" message.) The data collected for the semialgebraic prover follows. The keys provided to the prover were: `-L -S5 -O5 -b2 --samedegreesums`.

Problem name	Time	Generated	Deleted	Iterations	Proof size
x1_36.shuffled	35	689	72	563	566
x1_40.shuffled	42	759	80	627	630
x1_44.shuffled	35	853	88	691	694
x1_48.shuffled	39	942	97	755	758
x1_56.shuffled	91	1087	112	883	886
x1_64.shuffled	132	1245	128	1011	1014
x1_72.shuffled	289	1397	144	1139	1142
x1_80.shuffled	458	1567	160	1267	1270
x1_96.shuffled	622	1859	192	1523	1526
x1_128.shuffled	1746	2497	256	2035	2038
x2_40.shuffled	26	765	80	627	630
x2_44.shuffled	36	871	88	691	694
x2_48.shuffled	84	953	96	755	758
x2_56.shuffled	142	1130	129	896	887
x2_64.shuffled	136	1250	129	1015	1014
x2_72.shuffled	361	1391	144	1139	1142
x2_80.shuffled	271	1561	160	1267	1270
x2_96.shuffled	697	1878	193	1523	1526
x2_128.shuffled	1959	2497	256	2035	2038

5.3 TPTP benchmarks — overall picture

To evaluate behavior of our prover on first-order benchmarks, we selected 827 unsatisfiable problems from the TPTP library[5]. Our prover was able to solve 271 problems (with different key sets; the most successful one was `-w -M -S6 -b2 --earlyreductions --samedegreesums`, which solved 219 problems). Vampire 6.0 proved 754 problems to be unsatisfiable, and E proved 742 to be unsatisfiable; both provers were run in "auto mode". Note, however, that if we only consider problems solved by our prover, the proof size is smaller for our prover. It can be best seen considering 178 problems that can be proved unsatisfiable by both our prover with the key set `-w -M -Dall -b3 --earlyreductions --samedegreesums`, E, and Vampire.

	saprover	Vampire 6.0	E
Average proof size	36.37	45.34	41.49
Average time	3.26	0	0.1

The experiments were made on an AMD Athlon 1.3 GHz computer with RAM size restricted to 500 Mbytes.

[5] `http://www.tptp.org/`

Chapter 6

Examples

In this section we present the examples of actual proofs found by our prover (the TEX output is provided by the `-ftex` option).

6.1 The propositional pigeon-hole principle

The propositional pigeon-hole principle `php_m_n` says that m pigeons cannot sit in $n < m$ holes, at most one per hole. The fact that the pigeon number i sits in the hole number j is encoded by the Boolean variable p_{ij}.

It is known that there are no polynomial-size resolution proofs of `php_(n+1)_n` (see, e.g., [Hak85]). However, there are short Lovász-Schrijver proofs [Pud99] and Cutting Plane proofs [CCT87] of these formulas. Fortunately, a version of Lovász-Schrijver proofs still exists in our (quite restricted) calculus and can be found by our prover.

Command line.

```
./bin/saprover -pon -ftex
tests/propositional/php/php_4_3.tptp
```

Proof.

1. $1 \leqslant +p_{13} + p_{12} + p_{11}$.
   Initial.

2. $1 \leqslant +p_{23} + p_{22} + p_{21}$.
   Initial.

3. $1 \leqslant +p_{33} + p_{32} + p_{31}$.
   Initial.

4. $1 \leqslant +p_{43} + p_{42} + p_{41}$.
   Initial.

5. $-1 \leqslant -p_{21} - p_{11}$.
   Initial.

6. $-1 \leqslant -p_{31} - p_{11}$.
   Initial.

7. $-1 \leqslant -p_{41} - p_{11}$.
Initial.

8. $-1 \leqslant -p_{31} - p_{21}$.
Initial.

9. $-1 \leqslant -p_{41} - p_{21}$.
Initial.

10. $-1 \leqslant -p_{41} - p_{31}$.
Initial.

11. $-1 \leqslant -p_{22} - p_{12}$.
Initial.

12. $-1 \leqslant -p_{32} - p_{12}$.
Initial.

13. $-1 \leqslant -p_{42} - p_{12}$.
Initial.

14. $-1 \leqslant -p_{32} - p_{22}$.
Initial.

15. $-1 \leqslant -p_{42} - p_{22}$.
Initial.

16. $-1 \leqslant -p_{42} - p_{32}$.
Initial.

17. $-1 \leqslant -p_{23} - p_{13}$.
Initial.

18. $-1 \leqslant -p_{33} - p_{13}$.
Initial.

19. $-1 \leqslant -p_{43} - p_{13}$.
Initial.

20. $-1 \leqslant -p_{33} - p_{23}$.
Initial.

21. $-1 \leqslant -p_{43} - p_{23}$.
Initial.

22. $-1 \leqslant -p_{43} - p_{33}$.
Initial.

99. $+p_{31} \cdot p_{11} = +0$.
SumToProduct. Premises: `6`.

148. $+p_{31} \cdot p_{21} = +0$.
SumToProduct. Premises: `8`.

191. $-1 \leqslant +p_{31} \cdot p_{11} - p_{31} - p_{21} - p_{11}$.
MultiplyRewrite/Neg. Monomial $= +p_{31}$. Premises: `5, 148`.

223. $-1 \leqslant -p_{31} - p_{21} - p_{11}$.
Rewrite. Premises: `191, 99`.

263. $+p_{32} \cdot p_{12} = +0$.
SumToProduct. Premises: `12`.

334. $+p_{32} \cdot p_{22} = +0$.
SumToProduct. Premises: `14`.

389. $-1 \leqslant +p_{32} \cdot p_{12} - p_{32} - p_{22} - p_{12}$.
MultiplyRewrite/Neg. Monomial $= +p_{32}$. Premises: `11, 334`.

431. $-1 \leqslant -p_{32} - p_{22} - p_{12}$.
Rewrite. Premises: `389, 263`.

483. $+p_{33} \cdot p_{13} = +0$.
SumToProduct. Premises: `18`.

576. $+p_{33} \cdot p_{23} = +0$.
SumToProduct. Premises: `20`.

645. $-1 \leqslant +p_{33} \cdot p_{13} - p_{33} - p_{23} - p_{13}$.
MultiplyRewrite/Neg. Monomial $= +p_{33}$. Premises: `17, 576`.

695. $-1 \leqslant -p_{33} - p_{23} - p_{13}$.
Rewrite. Premises: `645, 483`.

759. $+p_{41} \cdot p_{11} = +0$.
SumToProduct. Premises: `7`.

874. $+p_{41} \cdot p_{21} = +0$.
SumToProduct. Premises: `9`.

1091. $+p_{41} \cdot p_{31} = +0$.
SumToProduct. Premises: `10`.

1177. $-1 \leqslant +p_{41} \cdot p_{21} + p_{41} \cdot p_{11} - p_{41} - p_{31} - p_{21} - p_{11}$.
MultiplyRewrite/Neg. Monomial $= +p_{41}$. Premises: `223, 1091`.

1243. $-1 \leqslant +p_{41} \cdot p_{21} - p_{41} - p_{31} - p_{21} - p_{11}$.
Rewrite. Premises: `1177, 759`.

1246. $-1 \leqslant -p_{41} - p_{31} - p_{21} - p_{11}$.
Rewrite. Premises: `1243, 874`.

1332. $+p_{42} \cdot p_{12} = +0$.
SumToProduct. Premises: `13`.

1479. $+p_{42} \cdot p_{22} = +0$.
SumToProduct. Premises: `15`.

1746. $+p_{42} \cdot p_{32} = +0$.
SumToProduct. Premises: `16`.

1859. $-1 \leqslant +p_{42} \cdot p_{22} + p_{42} \cdot p_{12} - p_{42} - p_{32} - p_{22} - p_{12}$.
MultiplyRewrite/Neg. Monomial $= +p_{42}$. Premises: `431, 1746`.

1930. $-1 \leqslant +p_{42} \cdot p_{22} - p_{42} - p_{32} - p_{22} - p_{12}$.
Rewrite. Premises: `1859, 1332`.

1933. $-1 \leqslant -p_{42} - p_{32} - p_{22} - p_{12}$.
Rewrite. Premises: `1930, 1479`.

2037. $+p_{43} \cdot p_{13} = +0$.
SumToProduct. Premises: `19`.

2216. $+p_{43} \cdot p_{23} = +0$.
SumToProduct. Premises: `21`.

2533. $+p_{43} \cdot p_{33} = +0$.
SumToProduct. Premises: `22`.

2658. $-1 \leqslant +p_{43} \cdot p_{23} + p_{43} \cdot p_{13} - p_{43} - p_{33} - p_{23} - p_{13}$.
MultiplyRewrite/Neg. Monomial $= +p_{43}$. Premises: `695, 2533`.

2749. $-1 \leqslant +p_{43} \cdot p_{23} - p_{43} - p_{33} - p_{23} - p_{13}$.
Rewrite. Premises: `2658, 2037`.

2752. $-1 \leqslant -p_{43} - p_{33} - p_{23} - p_{13}$.
Rewrite. Premises: `2749, 2216`.

4119. $0 \leqslant +p_{42} + p_{41} - p_{33} - p_{23} - p_{13}$.
Sum/Pos. Premises: `4, 2752`.

4218. $-1 \leqslant +p_{41} - p_{33} - p_{32} - p_{23} - p_{22} - p_{13} - p_{12}$.
Sum/Pos. Premises: `4119, 1933`.

4317. $-2 \leqslant -p_{33} - p_{32} - p_{31} - p_{23} - p_{22} - p_{21} - p_{13} - p_{12} - p_{11}$.
Sum/Pos. Premises: `4218, 1246`.

4422. $-1 \leqslant -p_{23} - p_{22} - p_{21} - p_{13} - p_{12} - p_{11}$.
Sum/Pos. Premises: `4317, 3`.

4573. $0 \leqslant -p_{13} - p_{12} - p_{11}$.
Sum/Pos. Premises: `4422, 2`.

4765. $1 \leqslant +0$.
Sum/Pos. Premises: `4573, 1`.

Total of 58 lines.

6.2 A first-order variation of the propositional pigeon-hole principle

Our prover is able to use the same "propositional" technique even if the pigeon-hole principle is moderately hidden by replacing 0-ary predicates with reasonable first-order terms.

Command line.

```
./bin/saprover -pon -ftex -S3 --earlyreductions
tests/first-order/php_FO/php_4_3.tptp
```

Proof.

1. $1 \leqslant +p_{13}(f(X_1^1, f(X_1^1, X_1^4))) + p_{12}(f(X_1^1, f(X_1^1, X_1^3))) +$
$p_{11}(f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, X_1^2))))))).$
Initial.

2. $1 \leqslant +p_{23}(f(X_2^1, X_2^4)) + p_{22}(f(X_2^1, f(X_2^1, f(X_2^1, f(X_2^1, f(X_2^1, X_2^3)))))) +$
$p_{21}(f(X_2^1, f(X_2^1, f(X_2^1, f(X_2^1, X_2^2))))).$
Initial.

3. $1 \leqslant +p_{33}(f(X_3^1, f(X_3^1, f(X_3^1, f(X_3^1, f(X_3^1, X_3^4)))))) + p_{32}(f(X_3^1, f(X_3^1, f(X_3^1, X_3^3)))) +$
$p_{31}(f(X_3^1, f(X_3^1, X_3^2))).$
Initial.

4. $1 \leqslant +p_{43}(f(X_4^1, X_4^4)) + p_{42}(f(X_4^1, f(X_4^1, X_4^3))) + p_{41}(f(X_4^1, X_4^2)).$
Initial.

5. $-1 \leqslant -p_{21}(f(X_5^1, f(X_5^1, X_5^3))) - p_{11}(f(X_5^1, f(X_5^1, f(X_5^1, X_5^2)))).$
Initial.

6. $-1 \leqslant -p_{31}(f(X_6^1, f(X_6^1, f(X_6^1, f(X_6^1, X_6^3))))) - p_{11}(f(X_6^1, X_6^2)).$
Initial.

7. $-1 \leqslant -p_{41}(f(X_7^1, f(X_7^1, X_7^3))) - p_{11}(f(X_7^1, f(X_7^1, f(X_7^1, X_7^2)))).$
Initial.

8. $-1 \leqslant -p_{31}(f(X_8^1, f(X_8^1, f(X_8^1, X_8^3)))) - p_{21}(f(X_8^1, f(X_8^1, f(X_8^1, X_8^2)))).$
Initial.

9. $-1 \leqslant -p_{41}(f(X_9^1, f(X_9^1, X_9^3))) - p_{21}(f(X_9^1, X_9^2)).$
Initial.

10. $-1 \leqslant -p_{41}(f(X_{10}^1, f(X_{10}^1, f(X_{10}^1, f(X_{10}^1, f(X_{10}^1, X_{10}^3)))))) -$
$p_{31}(f(X_{10}^1, f(X_{10}^1, f(X_{10}^1, f(X_{10}^1, X_{10}^2))))).$
Initial.

11. $-1 \leqslant -p_{22}(f(X_{11}^1, f(X_{11}^1, X_{11}^3))) - p_{12}(f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, X_{11}^2)))))).$
Initial.

12. $-1 \leqslant -p_{32}(f(X_{12}^1, f(X_{12}^1, X_{12}^3))) - p_{12}(f(X_{12}^1, f(X_{12}^1, X_{12}^2))).$
Initial.

13. $-1 \leqslant -p_{42}(f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, X_{13}^3))))))))) -$
$p_{12}(f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, X_{13}^2))))).$
Initial.

14. $-1 \leqslant -p_{32}(f(X_{14}^1, X_{14}^3)) - p_{22}(f(X_{14}^1, X_{14}^2)).$
Initial.

15. $-1 \leqslant -p_{42}(f(X_{15}^1, f(X_{15}^1, f(X_{15}^1, f(X_{15}^1, f(X_{15}^1, X_{15}^3)))))) - p_{22}(f(X_{15}^1, X_{15}^2)).$
Initial.

16. $-1 \leqslant -p_{42}(f(X_{16}^1, f(X_{16}^1, f(X_{16}^1, X_{16}^3)))) -$
$p_{32}(f(X_{16}^1, f(X_{16}^1, f(X_{16}^1, f(X_{16}^1, f(X_{16}^1, f(X_{16}^1, X_{16}^2))))))).$
Initial.

17. $-1 \leqslant -p_{23}(f(X^1_{17}, f(X^1_{17}, f(X^1_{17}, X^3_{17})))) - p_{13}(f(X^1_{17}, f(X^1_{17}, f(X^1_{17}, X^2_{17}))))$.
Initial.

18. $-1 \leqslant -p_{33}(f(X^1_{18}, f(X^1_{18}, X^3_{18}))) - p_{13}(f(X^1_{18}, f(X^1_{18}, X^2_{18})))$.
Initial.

19. $-1 \leqslant -p_{43}(f(X^1_{19}, f(X^1_{19}, X^3_{19}))) - p_{13}(f(X^1_{19}, f(X^1_{19}, f(X^1_{19}, f(X^1_{19}, X^2_{19})))))$.
Initial.

20. $-1 \leqslant -p_{33}(f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, X^3_{20}))))))))) - p_{23}(f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, X^2_{20}))))$.
Initial.

21. $-1 \leqslant -p_{43}(f(X^1_{21}, f(X^1_{21}, f(X^1_{21}, f(X^1_{21}, f(X^1_{21}, f(X^1_{21}, X^3_{21})))))) - p_{23}(f(X^1_{21}, f(X^1_{21}, f(X^1_{21}, X^2_{21}))))$.
Initial.

22. $-1 \leqslant -p_{43}(f(X^1_{22}, f(X^1_{22}, X^3_{22}))) - p_{33}(f(X^1_{22}, f(X^1_{22}, X^2_{22})))$.
Initial.

99. $+p_{31}(f(X^1_{99}, f(X^1_{99}, f(X^1_{99}, f(X^1_{99}, X^3_{99}))))) \cdot p_{11}(f(X^1_{99}, X^2_{99})) = +0$.
SumToProduct. Premises: `6`.

150. $+p_{31}(f(X^1_{150}, f(X^1_{150}, f(X^1_{150}, X^3_{150})))) \cdot p_{21}(f(X^1_{150}, f(X^1_{150}, f(X^1_{150}, X^2_{150})))) = +0$.
SumToProduct. Premises: `8`.

194. $-1 \leqslant +p_{31}(f(X^2_{194}, f(X^2_{194}, f(X^2_{194}, X^4_{194})))) \cdot p_{11}(f(X^2_{194}, f(X^2_{194}, f(X^2_{194}, X^1_{194})))) - p_{31}(f(X^2_{194}, f(X^2_{194}, f(X^2_{194}, X^4_{194})))) - p_{21}(f(X^2_{194}, f(X^2_{194}, f(X^2_{194}, X^3_{194})))) - p_{11}(f(X^2_{194}, f(X^2_{194}, f(X^2_{194}, X^1_{194}))))$.
MultiplyRewrite/Neg. Monomial $= +p_{31}(f(X^1_{150}, f(X^1_{150}, f(X^1_{150}, X^3_{150}))))$. Substitution $= \{X^1_5 \to X^1_{150}, X^3_5 \to f(X^1_{150}, X^2_{150})\}$. Premises: `5, 150`.

236. $-1 \leqslant -p_{31}(f(X^3_{236}, f(X^3_{236}, f(X^3_{236}, f(X^3_{236}, X^1_{236}))))) - p_{21}(f(X^3_{236}, f(X^3_{236}, f(X^3_{236}, X^4_{236})))) - p_{11}(f(X^3_{236}, f(X^3_{236}, f(X^3_{236}, X^2_{236}))))$.
SubstRewrite. Substitution $= \{X^1_{99} \to X^2_{194}, X^2_{99} \to f(X^2_{194}, f(X^2_{194}, X^1_{194})), X^4_{194} \to f(X^2_{194}, X^3_{99})\}$. Premises: `194, 99`.

328. $+p_{32}(f(X^1_{328}, f(X^1_{328}, X^3_{328}))) \cdot p_{12}(f(X^1_{328}, f(X^1_{328}, X^2_{328}))) = +0$.
SumToProduct. Premises: `12`.

404. $+p_{32}(f(X^1_{404}, X^3_{404})) \cdot p_{22}(f(X^1_{404}, X^2_{404})) = +0$.
SumToProduct. Premises: `14`.

464. $-1 \leqslant +p_{32}(f(X^3_{464}, X^4_{464})) \cdot p_{12}(f(X^3_{464}, f(X^3_{464}, f(X^3_{464}, f(X^3_{464}, f(X^3_{464}, X^1_{464})))))) - p_{32}(f(X^3_{464}, X^4_{464})) - p_{22}(f(X^3_{464}, f(X^3_{464}, X^2_{464}))) - p_{12}(f(X^3_{464}, f(X^3_{464}, f(X^3_{464}, f(X^3_{464}, f(X^3_{464}, X^1_{464}))))))$.
MultiplyRewrite/Neg. Monomial $= +p_{32}(f(X^1_{404}, X^3_{404}))$. Substitution $= \{X^1_{11} \to X^1_{404}, X^2_{404} \to f(X^1_{404}, X^3_{11})\}$. Premises: `11, 404`.

522. $-1 \leqslant -p_{32}(f(X^4_{522}, f(X^4_{522}, X^1_{522}))) - p_{22}(f(X^4_{522}, f(X^4_{522}, X^3_{522}))) - p_{12}(f(X^4_{522}, f(X^4_{522}, f(X^4_{522}, f(X^4_{522}, f(X^4_{522}, X^2_{522}))))))$.
SubstRewrite. Substitution $= \{X^1_{328} \to X^3_{464}, X^2_{328} \to f(X^3_{464}, f(X^3_{464}, f(X^3_{464}, X^1_{464}))), X^4_{464} \to f(X^3_{464}, X^3_{328})\}$. Premises: `464, 328`.

635. $+p_{33}(f(X^1_{635}, f(X^1_{635}, X^3_{635}))) \cdot p_{13}(f(X^1_{635}, f(X^1_{635}, X^2_{635}))) = +0.$
SumToProduct. Premises: `18`.

738. $+p_{33}(f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, X^3_{738})))))))))) \cdot$
$p_{23}(f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, X^2_{738})))) = +0.$
SumToProduct. Premises: `20`.

812. $-1 \leqslant$
$+p_{33}(f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, X^4_{812})))))))))) \cdot$
$p_{13}(f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, X^1_{812})))) -$
$p_{33}(f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, X^4_{812})))))))))) -$
$p_{23}(f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, X^3_{812})))) - p_{13}(f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, X^1_{812})))).$
MultiplyRewrite/Neg. Monomial $=$
$+p_{33}(f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, f(X^1_{738}, X^3_{738})))))))))).$
Substitution $= \{X^1_{17} \to X^1_{738}, X^3_{17} \to X^2_{738}\}$. Premises: `17, 738`.

901. $-1 \leqslant$
$-p_{33}(f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, X^4_{901}))))))))))-$
$p_{23}(f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, X^3_{901})))) - p_{13}(f(X^2_{901}, f(X^2_{901}, f(X^2_{901}, X^1_{901})))).$
Rewrite. Substitution $= \{X^1_{635} \to X^2_{812}, X^2_{635} \to f(X^2_{812}, X^1_{812}), X^3_{635} \to$
$f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, f(X^2_{812}, X^4_{812})))))))\}$. Premises: `812, 635`.

983. $+p_{41}(f(X^1_{983}, f(X^1_{983}, X^3_{983}))) \cdot p_{11}(f(X^1_{983}, f(X^1_{983}, f(X^1_{983}, X^2_{983})))) = +0.$
SumToProduct. Premises: `7`.

1110. $+p_{41}(f(X^1_{1110}, f(X^1_{1110}, X^3_{1110}))) \cdot p_{21}(f(X^1_{1110}, X^2_{1110})) = +0.$
SumToProduct. Premises: `9`.

1408. $+p_{41}(f(X^1_{1408}, f(X^1_{1408}, f(X^1_{1408}, f(X^1_{1408}, f(X^1_{1408}, X^3_{1408})))))) \cdot$
$p_{31}(f(X^1_{1408}, f(X^1_{1408}, f(X^1_{1408}, f(X^1_{1408}, X^2_{1408}))))) = +0.$
SumToProduct. Premises: `10`.

1507. $-1 \leqslant +p_{41}(f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^5_{1507})))))) \cdot$
$p_{21}(f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^2_{1507})))) +$
$p_{41}(f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^5_{1507})))))) \cdot$
$p_{11}(f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^1_{1507})))) -$
$p_{41}(f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^5_{1507})))))) -$
$p_{31}(f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^4_{1507}))))) -$
$p_{21}(f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^2_{1507})))) - p_{11}(f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^1_{1507})))).$
MultiplyRewrite/Neg. Monomial $=$
$+p_{41}(f(X^1_{1408}, f(X^1_{1408}, f(X^1_{1408}, f(X^1_{1408}, f(X^1_{1408}, X^3_{1408}))))))$. Substitution $=$
$\{X^1_{236} \to X^2_{1408}, X^3_{236} \to X^1_{1408}\}$. Premises: `236, 1408`.

1758. $-1 \leqslant +p_{41}(f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, X^5_{1758})))))) \cdot$
$p_{21}(f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, X^2_{1758})))) -$
$p_{41}(f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, X^5_{1758})))))) -$
$p_{31}(f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, X^4_{1758}))))) -$
$p_{21}(f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, X^2_{1758})))) - p_{11}(f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, X^1_{1758})))).$
Rewrite. Substitution $=$
$\{X^1_{983} \to X^3_{1507}, X^2_{983} \to X^1_{1507}, X^3_{983} \to f(X^3_{1507}, f(X^3_{1507}, f(X^3_{1507}, X^5_{1507})))\}$. Premises:
`1507, 983`.

1761. $-1 \leqslant -p_{41}(f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, X^5_{1761})))))) - p_{31}(f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, X^4_{1761}))))) - p_{21}(f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, X^2_{1761})))) - p_{11}(f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, X^1_{1761}))))$.
Rewrite. Substitution $= \{X^1_{1110} \to X^3_{1758}, X^2_{1110} \to f(X^3_{1758}, f(X^3_{1758}, X^2_{1758})), X^3_{1110} \to f(X^3_{1758}, f(X^3_{1758}, f(X^3_{1758}, X^5_{1758})))\}$. Premises: `1758, 1110`.

1958.
$+p_{42}(f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, X^3_{1958}))))))))))\cdot p_{12}(f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, f(X^1_{1958}, X^2_{1958}))))) = +0$.
SumToProduct. Premises: `13`.

2120. $+p_{42}(f(X^1_{2120}, f(X^1_{2120}, f(X^1_{2120}, f(X^1_{2120}, f(X^1_{2120}, X^3_{2120})))))) \cdot p_{22}(f(X^1_{2120}, X^2_{2120})) = +0$.
SumToProduct. Premises: `15`.

2534. $+p_{42}(f(X^1_{2534}, f(X^1_{2534}, f(X^1_{2534}, X^3_{2534})))) \cdot p_{32}(f(X^1_{2534}, f(X^1_{2534}, f(X^1_{2534}, f(X^1_{2534}, f(X^1_{2534}, f(X^1_{2534}, X^2_{2534}))))))) = +0$.
SumToProduct. Premises: `16`.

2672. $-1 \leqslant +p_{42}(f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, X^5_{2672})))) \cdot p_{22}(f(X^3_{2672}, f(X^3_{2672}, X^2_{2672}))) + p_{42}(f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, X^5_{2672})))) \cdot p_{12}(f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, X^1_{2672})))))) - p_{42}(f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, X^5_{2672})))) - p_{32}(f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, X^4_{2672}))))))) - p_{22}(f(X^3_{2672}, f(X^3_{2672}, X^2_{2672}))) - p_{12}(f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, X^1_{2672}))))))$.
MultiplyRewrite/Neg. Monomial $= +p_{42}(f(X^1_{2534}, f(X^1_{2534}, f(X^1_{2534}, X^3_{2534}))))$. Substitution $= \{X^1_{522} \to f(X^1_{2534}, f(X^1_{2534}, f(X^1_{2534}, f(X^1_{2534}, X^2_{2534})))), X^4_{522} \to X^1_{2534}\}$. Premises: `522, 2534`.

2918. $-1 \leqslant +p_{42}(f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, X^1_{2918}))))))))))\cdot p_{22}(f(X^4_{2918}, f(X^4_{2918}, X^3_{2918}))) - p_{42}(f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, X^1_{2918}))))))))))- p_{32}(f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, X^5_{2918}))))))) - p_{22}(f(X^4_{2918}, f(X^4_{2918}, X^3_{2918}))) - p_{12}(f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, X^2_{2918}))))))$.
SubstRewrite. Substitution $= \{X^1_{1958} \to X^3_{2672}, X^2_{1958} \to f(X^3_{2672}, X^1_{2672}), X^5_{2672} \to f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, f(X^3_{2672}, X^3_{1958}))))))\}$. Premises: `2672, 1958`.

3309. $-1 \leqslant -p_{42}(f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, X^1_{3309}))))))))))- p_{32}(f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, X^5_{3309}))))))) - p_{22}(f(X^4_{3309}, f(X^4_{3309}, X^3_{3309}))) - p_{12}(f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, X^2_{3309}))))))$.
Rewrite. Substitution $= \{X^1_{2120} \to X^4_{2918}, X^2_{2120} \to f(X^4_{2918}, X^3_{2918}), X^3_{2120} \to f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, f(X^4_{2918}, X^1_{2918}))))\}$. Premises: `2918, 2120`.

3805. $+p_{43}(f(X^1_{3805}, f(X^1_{3805}, X^3_{3805}))) \cdot p_{13}(f(X^1_{3805}, f(X^1_{3805}, f(X^1_{3805}, f(X^1_{3805}, X^2_{3805}))))) = +0$.
SumToProduct. Premises: `19`.

4007. $+p_{43}(f(X^1_{4007}, f(X^1_{4007}, f(X^1_{4007}, f(X^1_{4007}, f(X^1_{4007}, f(X^1_{4007}, X^3_{4007}))))))) \cdot p_{23}(f(X^1_{4007}, f(X^1_{4007}, f(X^1_{4007}, X^2_{4007})))) = +0$.
SumToProduct. Premises: `21`.

4493. $+p_{43}(f(X^1_{4493}, f(X^1_{4493}, X^3_{4493}))) \cdot p_{33}(f(X^1_{4493}, f(X^1_{4493}, X^2_{4493}))) = +0$.
SumToProduct. Premises: `22`.

4633. $-1 \leqslant +p_{43}(f(X^4_{4633}, f(X^4_{4633}, X^5_{4633}))) \cdot p_{23}(f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, X^2_{4633})))) +$
$p_{43}(f(X^4_{4633}, f(X^4_{4633}, X^5_{4633}))) \cdot p_{13}(f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, X^1_{4633})))) -$
$p_{43}(f(X^4_{4633}, f(X^4_{4633}, X^5_{4633}))) -$
$p_{33}(f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, X^3_{4633}))))))))))-$
$p_{23}(f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, X^2_{4633})))) - p_{13}(f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, X^1_{4633}))))$.
MultiplyRewrite/Neg. Monomial $= +p_{43}(f(X^1_{4493}, f(X^1_{4493}, X^3_{4493})))$. Substitution $=$
$\{X^2_{901} \to X^1_{4493}, X^2_{4493} \to$
$f(X^1_{4493}, f(X^1_{4493}, f(X^1_{4493}, f(X^1_{4493}, f(X^1_{4493}, f(X^1_{4493}, f(X^1_{4493}, X^4_{901})))))))\}$. Premises: `901, 4493`.

4874. $-1 \leqslant +p_{43}(f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, X^1_{4874}))))))) \cdot$
$p_{13}(f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, X^2_{4874})))) -$
$p_{43}(f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, X^1_{4874}))))))) -$
$p_{33}(f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, X^4_{4874}))))))))))-$
$p_{23}(f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, X^3_{4874})))) - p_{13}(f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, X^2_{4874}))))$.
SubstRewrite. Substitution $=$
$\{X^1_{4007} \to X^4_{4633}, X^2_{4007} \to X^2_{4633}, X^5_{4633} \to f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, f(X^4_{4633}, X^3_{4007}))))\}$.
Premises: `4633, 4007`.

5056. $-1 \leqslant -p_{43}(f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, X^2_{5056}))))))) -$
$p_{33}(f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, X^4_{5056}))))))))))-$
$p_{23}(f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, X^3_{5056})))) - p_{13}(f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, X^1_{5056})))))$.
SubstRewrite. Substitution $= \{X^1_{3805} \to X^5_{4874}, X^3_{3805} \to$
$f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, f(X^5_{4874}, X^1_{4874})))), X^2_{4874} \to f(X^5_{4874}, X^2_{3805})\}$. Premises: `4874, 3805`.

17288. $0 \leqslant +p_{42}(f(X^6_{17288}, f(X^6_{17288}, X^2_{17288}))) + p_{41}(f(X^6_{17288}, X^1_{17288})) -$
$p_{33}(f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, X^5_{17288}))))))))))-$
$p_{23}(f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, X^4_{17288})))) -$
$p_{13}(f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, f(X^6_{17288}, X^3_{17288})))))$.
Sum/Pos. Substitution $=$
$\{X^1_4 \to X^5_{5056}, X^4_4 \to f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, f(X^5_{5056}, X^2_{5056})))))\}$. Premises: `4, 5056`.

17537. $-1 \leqslant +p_{41}(f(X^3_{17537}, X^5_{17537})) -$
$p_{33}(f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, X^8_{17537}))))))))))-$
$p_{32}(f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, X^4_{17537}))))))) -$
$p_{23}(f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, X^7_{17537})))) - p_{22}(f(X^3_{17537}, f(X^3_{17537}, X^2_{17537}))) -$
$p_{13}(f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, X^6_{17537}))))) -$
$p_{12}(f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, f(X^3_{17537}, X^1_{17537}))))))$.
Sum/Pos. Substitution $= \{X^2_{17288} \to$
$f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, f(X^4_{3309}, X^1_{3309}))))))), X^6_{17288} \to X^4_{3309}\}$.
Premises: `17288, 3309`.

18058. $-2 \leqslant$
$-p_{33}(f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, X^{10}_{18058}))))))))))-$
$p_{32}(f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, X^7_{18058}))))))) -$
$p_{31}(f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, X^4_{18058}))))) -$
$p_{23}(f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, X^9_{18058})))) - p_{22}(f(X^3_{18058}, f(X^3_{18058}, X^6_{18058}))) -$
$p_{21}(f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, X^2_{18058})))) -$
$p_{13}(f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, X^8_{18058}))))) -$

$p_{12}(f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, X^5_{18058})))))) -$
$p_{11}(f(X^3_{18058}, f(X^3_{18058}, f(X^3_{18058}, X^1_{18058})))).$
Sum/Pos. Substitution =
$\{X^3_{17537} \to X^3_{1761}, X^5_{17537} \to f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, f(X^3_{1761}, X^5_{1761}))))\}$. Premises: `17537, 1761`.

18805. $-1 \leqslant -p_{32}(f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, X^9_{18805})))))))+$
$p_{32}(f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, X^3_{18805})))) -$
$p_{31}(f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, X^6_{18805}))))) + p_{31}(f(X^1_{18805}, f(X^1_{18805}, X^2_{18805}))) -$
$p_{23}(f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, X^{11}_{18805})))) - p_{22}(f(X^1_{18805}, f(X^1_{18805}, X^8_{18805}))) -$
$p_{21}(f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, X^5_{18805})))) -$
$p_{13}(f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, X^{10}_{18805}))))) -$
$p_{12}(f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, X^7_{18805})))))) -$
$p_{11}(f(X^1_{18805}, f(X^1_{18805}, f(X^1_{18805}, X^4_{18805})))).$
Sum/Pos. Substitution = $\{X^4_3 \to f(X^1_3, f(X^1_3, f(X^1_3, f(X^1_3, X^{10}_{18058})))), X^3_{18058} \to X^1_3\}$.
Premises: `18058, 3`.

18808. $-1 \leqslant -p_{31}(f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, X^5_{18808}))))) +$
$p_{31}(f(X^1_{18808}, f(X^1_{18808}, X^2_{18808}))) - p_{23}(f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, X^9_{18808})))) -$
$p_{22}(f(X^1_{18808}, f(X^1_{18808}, X^7_{18808}))) - p_{21}(f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, X^4_{18808})))) -$
$p_{13}(f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, X^8_{18808}))))) -$
$p_{12}(f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, X^6_{18808})))))) -$
$p_{11}(f(X^1_{18808}, f(X^1_{18808}, f(X^1_{18808}, X^3_{18808})))).$
Factor/Add. Premises: `18805`.

18924. $-1 \leqslant -p_{23}(f(X^1_{18924}, f(X^1_{18924}, f(X^1_{18924}, X^7_{18924})))) - p_{22}(f(X^1_{18924}, f(X^1_{18924}, X^5_{18924}))) -$
$p_{21}(f(X^1_{18924}, f(X^1_{18924}, f(X^1_{18924}, X^3_{18924})))) -$
$p_{13}(f(X^1_{18924}, f(X^1_{18924}, f(X^1_{18924}, f(X^1_{18924}, X^6_{18924}))))) -$
$p_{12}(f(X^1_{18924}, f(X^1_{18924}, f(X^1_{18924}, f(X^1_{18924}, f(X^1_{18924}, X^4_{18924})))))) -$
$p_{11}(f(X^1_{18924}, f(X^1_{18924}, f(X^1_{18924}, X^2_{18924})))).$
Factor/Add. Premises: `18808`.

21580. $0 \leqslant -p_{22}(f(X^1_{21580}, f(X^1_{21580}, X^7_{21580}))) +$
$p_{22}(f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, X^3_{21580})))))) -$
$p_{21}(f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, X^5_{21580})))) +$
$p_{21}(f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, X^2_{21580}))))) -$
$p_{13}(f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, X^8_{21580}))))) -$
$p_{12}(f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, X^6_{21580})))))) -$
$p_{11}(f(X^1_{21580}, f(X^1_{21580}, f(X^1_{21580}, X^4_{21580})))).$
Sum/Pos. Substitution = $\{X^4_2 \to f(X^1_2, f(X^1_2, X^7_{18924})), X^1_{18924} \to X^1_2\}$. Premises: `18924, 2`.

21594. $0 \leqslant -p_{21}(f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, X^4_{21594})))) +$
$p_{21}(f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, X^2_{21594}))))) -$
$p_{13}(f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, X^6_{21594}))))) -$
$p_{12}(f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, X^5_{21594})))))) -$
$p_{11}(f(X^1_{21594}, f(X^1_{21594}, f(X^1_{21594}, X^3_{21594})))).$
Factor/Add. Premises: `21580`.

21776. $0 \leqslant -p_{13}(f(X^1_{21776}, f(X^1_{21776}, f(X^1_{21776}, f(X^1_{21776}, X^4_{21776}))))) -$
$p_{12}(f(X^1_{21776}, f(X^1_{21776}, f(X^1_{21776}, f(X^1_{21776}, f(X^1_{21776}, X^3_{21776})))))) -$
$p_{11}(f(X^1_{21776}, f(X^1_{21776}, f(X^1_{21776}, X^2_{21776})))).$
Factor/Add. Premises: `21594`.

103335. $1 \leqslant -p_{12}(f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, X^5_{103335})))))) +$
$p_{12}(f(X^1_{103335}, f(X^1_{103335}, X^3_{103335}))) - p_{11}(f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, X^4_{103335})))) +$
$p_{11}(f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, X^2_{103335}))))))).$
Sum/Pos. Substitution $= \{X^4_1 \to f(X^1_1, f(X^1_1, X^4_{21776})), X^1_{21776} \to X^1_1\}$. Premises: `21776`, `1`.

103344. $1 \leqslant -p_{11}(f(X^1_{103344}, f(X^1_{103344}, f(X^1_{103344}, X^3_{103344})))) +$
$p_{11}(f(X^1_{103344}, f(X^1_{103344}, f(X^1_{103344}, f(X^1_{103344}, f(X^1_{103344}, f(X^1_{103344}, X^2_{103344}))))))).$
Factor/Add. Substitution $= \{X^4_{103335} \to f(X^1_{103335}, f(X^1_{103335}, f(X^1_{103335}, X^2_{103335})))\}$.
Premises: `103335`.

103598. $1 \leqslant +0$.
Factor/Add. Substitution $= \{X^3_{103344} \to f(X^1_{103344}, f(X^1_{103344}, f(X^1_{103344}, X^2_{103344})))\}$.
Premises: `103344`.

Total of 64 lines.

6.3 A simple first-order problem

This is a problem taken from the TPTP library[1].

Command line.

```
./bin/saprover -S2 -pon -ftex
tests/first-order/TPTP/PUZ/PUZ002-1.p
```

Proof.

1. $0 \leqslant -\mathit{in\_house}(X^1_1) + \mathit{cat}(X^1_1)$.
   Initial.
2. $0 \leqslant +\mathit{suitable\_pet}(X^1_2) - \mathit{gazer}(X^1_2)$.
   Initial.
3. $0 \leqslant -\mathit{detested}(X^1_3) + \mathit{avoided}(X^1_3)$.
   Initial.
4. $0 \leqslant +\mathit{prowler}(X^1_4) - \mathit{carnivore}(X^1_4)$.
   Initial.
5. $0 \leqslant +\mathit{mouse\_killer}(X^1_5) - \mathit{cat}(X^1_5)$.
   Initial.
6. $0 \leqslant -\mathit{takes\_to\_me}(X^1_6) + \mathit{in\_house}(X^1_6)$.
   Initial.
7. $-1 \leqslant -\mathit{suitable\_pet}(X^1_7) - \mathit{kangaroo}(X^1_7)$.
   Initial.
8. $0 \leqslant -\mathit{mouse\_killer}(X^1_8) + \mathit{carnivore}(X^1_8)$.
   Initial.

[1] `www.tptp.org`

9. $1 \leqslant +takes\_to\_me(X_9^1) + detested(X_9^1)$.
Initial.

10. $0 \leqslant -prowler(X_{10}^1) + gazer(X_{10}^1)$.
Initial.

11. $1 \leqslant +kangaroo(the\_kangaroo)$.
Initial.

12. $0 \leqslant -avoided(the\_kangaroo)$.
Initial.

13. $+kangaroo(the\_kangaroo) = +1$.
Opposite. Premises: `11`.

24. $+avoided(the\_kangaroo) = +0$.
Opposite. Premises: `12`.

131. $0 \leqslant -detested(the\_kangaroo)$.
SubstRewrite. Substitution = $\{X_3^1 \to the\_kangaroo\}$. Premises: `3, 24`.

151. $+detested(the\_kangaroo) = +0$.
Opposite. Premises: `131`.

204. $+prowler(X_{204}^1) \cdot carnivore(X_{204}^1) = +carnivore(X_{204}^1)$.
SumToProduct. Premises: `4`.

271. $+mouse\_killer(X_{271}^1) \cdot cat(X_{271}^1) = +cat(X_{271}^1)$.
SumToProduct. Premises: `5`.

472. $0 \leqslant -suitable\_pet(the\_kangaroo)$.
SubstRewrite. Substitution = $\{X_7^1 \to the\_kangaroo\}$. Premises: `7, 13`.

561. $0 \leqslant -gazer(the\_kangaroo)$.
Sum/Pos. Substitution = $\{X_2^1 \to the\_kangaroo\}$. Premises: `472, 2`.

773. $1 \leqslant +takes\_to\_me(the\_kangaroo)$.
SubstRewrite. Substitution = $\{X_9^1 \to the\_kangaroo\}$. Premises: `9, 151`.

863. $1 \leqslant +in\_house(the\_kangaroo)$.
Sum/Pos. Substitution = $\{X_6^1 \to the\_kangaroo\}$. Premises: `773, 6`.

930. $1 \leqslant +cat(the\_kangaroo)$.
Sum/Pos. Substitution = $\{X_1^1 \to the\_kangaroo\}$. Premises: `863, 1`.

1004. $1 \leqslant +mouse\_killer(the\_kangaroo)$.
MultiplyRewrite/Neg. Monomial = $+mouse\_killer(the\_kangaroo)$. Substitution = $\{X_{271}^1 \to the\_kangaroo\}$. Premises: `930, 271`.

1089. $1 \leqslant +carnivore(the\_kangaroo)$.
Sum/Pos. Substitution = $\{X_8^1 \to the\_kangaroo\}$. Premises: `1004, 8`.

1146. $1 \leqslant +prowler(the\_kangaroo)$.
MultiplyRewrite/Neg. Monomial = $+prowler(the\_kangaroo)$. Substitution = $\{X_{204}^1 \to the\_kangaroo\}$. Premises: `1089, 204`.

1250. $+prowler(X^1_{1250}) \cdot gazer(X^1_{1250}) = +prowler(X^1_{1250})$.
SumToProduct. Premises: `10`.

1340. $0 \leqslant -prowler(the\_kangaroo)$.
MultiplyRewrite/Pos. Monomial $= +prowler(the\_kangaroo)$. Substitution $= \{X^1_{1250} \to the\_kangaroo\}$. Premises: `561, 1250`.

1499. $1 \leqslant +0$.
Sum/Pos. Premises: `1340, 1146`.

Total of 29 lines.

6.4 Another first-order problem

This is a problem taken from a set of examples provided to us by John Harrison.

Command line.

```
./bin/saprover --complete -M --samedegreesums -O5 -S5 -L -b4 -pon -ftex
tests/first-order/John_Harrison/4031.tptp
```

Proof.

5. $1 \leqslant +p_{18}(f_9(X^1_5), f_{14}(X^1_5), f_{14}(X^1_5)) + p_{15}(X^1_5)$.
Initial.

6. $1 \leqslant +p_{15}(c_{21})$.
Initial.

8. $-1 \leqslant -p_{16}(X^1_8, X^4_8, X^3_8) - p_{16}(X^1_8, X^2_8, X^4_8) + p_{16}(X^1_8, X^2_8, X^3_8)$.
Initial.

12. $0 \leqslant -p_{18}(f_9(f_4(f_7(X^3_{12}, X^5_{12}), X^1_{12})), X^2_{12}, X^4_{12}) + p_{18}(f_9(X^1_{12}), X^2_{12}, X^4_{12}) + p_{16}(f_9(X^1_{12}), X^2_{12}, X^3_{12})$.
Initial.

13. $0 \leqslant -p_{18}(f_9(f_4(f_7(X^3_{13}, X^5_{13}), X^1_{13})), X^2_{13}, X^4_{13}) + p_{18}(f_9(X^1_{13}), X^2_{13}, X^4_{13}) + p_{16}(f_9(X^1_{13}), f_{12}(X^1_{13}, X^3_{13}, X^5_{13}, X^2_{13}, X^4_{13}), X^4_{13})$.
Initial.

15. $0 \leqslant -p_{18}(f_9(f_4(f_7(X^3_{15}, X^5_{15}), X^1_{15})), X^2_{15}, X^4_{15}) + p_{18}(f_9(X^1_{15}), X^2_{15}, X^4_{15}) + p_{17}(f_{12}(X^1_{15}, X^3_{15}, X^5_{15}, X^2_{15}, X^4_{15}), f_{11}(X^5_{15}))$.
Initial.

33. $-1 \leqslant -p_{18}(f_9(X^1_{33}), X^2_{33}, X^2_{33}) - p_{15}(X^1_{33})$.
Initial.

34. $0 \leqslant -p_{15}(f_4(f_7(c_{19}, c_{20}), c_{21}))$.
Initial.

35. $-1 \leqslant -p_{17}(X^1_{35}, f_{11}(c_{20})) - p_{16}(f_9(c_{21}), X^1_{35}, c_{19})$.
Initial.

47. $+p_{15}(c_{21}) = +1.$
Opposite. Premises: `6`.

65. $+p_{15}(f_4(f_7(c_{19}, c_{20}), c_{21})) = +0.$
Opposite. Premises: `34`.

321. $+p_{17}(X^1_{321}, f_{11}(c_{20})) \cdot p_{16}(f_9(c_{21}), X^1_{321}, c_{19}) = +0.$
SumToProduct. Premises: `35`.

417. $1 \leqslant +p_{18}(f_9(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))).$
SubstRewrite. Substitution $= \{X^1_5 \to f_4(f_7(c_{19}, c_{20}), c_{21})\}$. Premises: `5, 65`.

460. $+p_{18}(f_9(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))) = +1.$
Opposite. Premises: `417`.

558. $0 \leqslant -p_{18}(f_9(c_{21}), X^1_{558}, X^1_{558}).$
SubstRewrite. Substitution $= \{X^1_{33} \to c_{21}\}$. Premises: `33, 47`.

643. $+p_{18}(f_9(c_{21}), X^1_{643}, X^1_{643}) = +0.$
Opposite. Premises: `558`.

3312. $1 \leqslant +p_{18}(f_9(c_{21}), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))) +$
$p_{16}(f_9(c_{21}), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), c_{19}).$
SubstRewrite. Substitution $= \{X^1_{12} \to c_{21}, X^2_{12} \to f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), X^3_{12} \to c_{19}, X^4_{12} \to$
$f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), X^5_{12} \to c_{20}\}$. Premises: `12, 460`.

3388. $1 \leqslant +p_{16}(f_9(c_{21}), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), c_{19}).$
Rewrite. Substitution $= \{X^1_{643} \to f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))\}$. Premises: `3312, 643`.

3556. $0 \leqslant -p_{16}(f_9(c_{21}), X^1_{3556}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))) + p_{16}(f_9(c_{21}), X^1_{3556}, c_{19}).$
Sum/Pos. Substitution $= \{X^1_8 \to f_9(c_{21}), X^3_8 \to c_{19}, X^4_8 \to f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))\}$.
Premises: `3388, 8`.

5670. $1 \leqslant +p_{18}(f_9(c_{21}), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))) +$
$p_{16}(f_9(c_{21}), f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))).$
SubstRewrite. Substitution $= \{X^1_{13} \to c_{21}, X^2_{13} \to f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), X^3_{13} \to c_{19}, X^4_{13} \to$
$f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), X^5_{13} \to c_{20}\}$. Premises: `13, 460`.

5845. $1 \leqslant$
$+p_{16}(f_9(c_{21}), f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))).$
Rewrite. Substitution $= \{X^1_{643} \to f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))\}$. Premises: `5670, 643`.

6272. $1 \leqslant +p_{16}(f_9(c_{21}), f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))), c_{19}).$
Sum/Pos. Substitution $=$
$\{X^1_{3556} \to f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})))\}$. Premises:
`5845, 3556`.

6529. $0 \leqslant -p_{17}(f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))), f_{11}(c_{20})).$
MultiplyRewrite/Pos. Monomial $=$
$+p_{17}(f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))), f_{11}(c_{20})).$
Substitution $= \{X^1_{321} \to f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})))\}$.
Premises: `6272, 321`.

11011. $1 \leqslant +p_{18}(f_9(c_{21}), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))) + p_{17}(f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))), f_{11}(c_{20}))$.
SubstRewrite. Substitution = $\{X^1_{15} \to c_{21}, X^2_{15} \to f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), X^3_{15} \to c_{19}, X^4_{15} \to f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), X^5_{15} \to c_{20}\}$. Premises: `15, 460`.

11291. $1 \leqslant +p_{17}(f_{12}(c_{21}, c_{19}, c_{20}, f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21})), f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))), f_{11}(c_{20}))$.
Rewrite. Substitution = $\{X^1_{643} \to f_{14}(f_4(f_7(c_{19}, c_{20}), c_{21}))\}$. Premises: `11011, 643`.

12144. $1 \leqslant +0$.
Sum/Pos. Premises: `11291, 6529`.

Total of 26 lines.

6.5 Tseitin formulas

Tseitin [Tse68] was first who proved that resolution has no polynomial-size proofs of certain propositional tautologies. He used formulas that encoded the fact that it is impossible to label edges of a graph with zeroes and ones so that

- the sum of labels around each vertex is odd;
- the number of vertices is odd.

Since then, various versions of these tautologies (for different, most notably bounded-degree, graphs) were used for proving lower bounds for various proof systems.
It is known [GHP02] that the proof system underlying our prover has short proofs of bounded-degree Tseitin tautologies. We present here a proof that the prover has found for a tautology based on a certain degree three graph. (It is not formally proved that the tautologies based on this particular kind of graphs are hard for resolution; however, we believe it is the case.)
Note that the `xor-chains` benchmarks (by Lintao Zhang and Sharad Malik) encoding verification problems of two xor-chains form a special case of degree three Tseitin tautologies. These benchmarks were among *SAT 2002 Competition challenging benchmarks*[2].

Command line.

```
./bin/saprover -pon -ftex -L -S5 -O5 -b2 --samedegreesums
tests/propositional/tseitin/tseitin_t_07_03.tptp
```

Proof.

1. $0 \leqslant +p_7 + p_2 - p_1$.
   Initial.

2. $0 \leqslant +p_7 - p_2 + p_1$.
   Initial.

3. $0 \leqslant -p_7 + p_2 + p_1$.
   Initial.

[2]See `http://satlive.org/SATCompetition/2002/`

4. $-2 \leqslant -p_7 - p_2 - p_1$.
   Initial.

5. $0 \leqslant +p_4 + p_3 - p_1$.
   Initial.

6. $0 \leqslant +p_4 - p_3 + p_1$.
   Initial.

7. $0 \leqslant -p_4 + p_3 + p_1$.
   Initial.

8. $-2 \leqslant -p_4 - p_3 - p_1$.
   Initial.

9. $0 \leqslant +p_6 + p_5 - p_2$.
   Initial.

10. $0 \leqslant +p_6 - p_5 + p_2$.
    Initial.

11. $0 \leqslant -p_6 + p_5 + p_2$.
    Initial.

12. $-2 \leqslant -p_6 - p_5 - p_2$.
    Initial.

13. $0 \leqslant +p_8 - p_7 + p_3$.
    Initial.

14. $0 \leqslant +p_8 + p_7 - p_3$.
    Initial.

15. $0 \leqslant -p_8 + p_7 + p_3$.
    Initial.

16. $-2 \leqslant -p_8 - p_7 - p_3$.
    Initial.

17. $0 \leqslant +p_9 + p_8 - p_4$.
    Initial.

18. $0 \leqslant +p_9 - p_8 + p_4$.
    Initial.

19. $0 \leqslant -p_9 + p_8 + p_4$.
    Initial.

20. $-2 \leqslant -p_9 - p_8 - p_4$.
    Initial.

21. $0 \leqslant +p_9 - p_5 + p_{10}$.
    Initial.

22. $0 \leqslant -p_9 + p_5 + p_{10}$.
    Initial.

23. $0 \leqslant +p_9 + p_5 - p_{10}$.
Initial.

24. $-2 \leqslant -p_9 - p_5 - p_{10}$.
Initial.

25. $1 \leqslant +p_6 + p_{10}$.
Initial.

26. $-1 \leqslant -p_6 - p_{10}$.
Initial.

52. $+p_6 = -p_{10} + 1$.
Opposite. Premises: `25, 26`.

99. $0 \leqslant -p_4 + p_3 - p_1 + 2p_4 \cdot p_1$.
LiftAndProject. Premises: `7, 5`.

119. $0 \leqslant +p_4 - p_3 + p_1 - 2p_4 \cdot p_1$.
LiftAndProject. Premises: `8, 6`.

127. $-1 \leqslant +p_5 - p_2 - p_{10}$.
Rewrite. Premises: `9, 52`.

136. $-1 \leqslant -p_5 + p_2 - p_{10}$.
Rewrite. Premises: `10, 52`.

149. $1 \leqslant +p_5 + p_2 + p_{10}$.
Rewrite. Premises: `11, 52`.

163. $1 \leqslant +p_5 + p_2 + p_{10} - 2p_5 \cdot p_{10}$.
LiftAndProject. Premises: `149, 136`.

169. $-1 \leqslant -p_5 - p_2 + p_{10}$.
Rewrite. Premises: `12, 52`.

181. $-1 \leqslant -p_5 - p_2 - p_{10} + 2p_5 \cdot p_{10}$.
LiftAndProject. Premises: `169, 127`.

211. $0 \leqslant -p_7 + p_2 - p_1 + 2p_7 \cdot p_1$.
LiftAndProject. Premises: `3, 1`.

231. $0 \leqslant +p_7 - p_2 + p_1 - 2p_7 \cdot p_1$.
LiftAndProject. Premises: `4, 2`.

263. $0 \leqslant -p_8 + p_7 - p_3 + 2p_8 \cdot p_3$.
LiftAndProject. Premises: `15, 14`.

279. $0 \leqslant +p_8 - p_7 + p_3 - 2p_8 \cdot p_3$.
LiftAndProject. Premises: `16, 13`.

313. $0 \leqslant -p_9 + p_5 - p_{10} + 2p_9 \cdot p_{10}$.
LiftAndProject. Premises: `23, 22`.

329. $0 \leqslant +p_9 - p_5 + p_{10} - 2p_9 \cdot p_{10}$.
LiftAndProject. Premises: `24, 21`.

407. $0 \leqslant -p_9 + p_8 - p_4 + 2p_9 \cdot p_4$.
LiftAndProject. Premises: `19, 17`.

440. $0 \leqslant +p_9 - p_8 + p_4 - 2p_9 \cdot p_4$.
LiftAndProject. Premises: `20, 18`.

515. $0 \leqslant +p_8 - p_5 - p_4 + p_{10} + 2p_9 \cdot p_4 - 2p_9 \cdot p_{10}$.
Sum/Pos. Premises: `407, 329`.

523. $0 \leqslant +p_7 - p_5 - p_4 - p_3 + p_{10} + 2p_9 \cdot p_4 - 2p_9 \cdot p_{10} + 2p_8 \cdot p_3$.
Sum/Pos. Premises: `515, 263`.

533. $0 \leqslant -p_5 - p_4 - p_3 + p_2 + p_{10} - p_1 + 2p_9 \cdot p_4 - 2p_9 \cdot p_{10} + 2p_8 \cdot p_3 + 2p_7 \cdot p_1$.
Sum/Pos. Premises: `523, 211`.

543. $1 \leqslant -p_4 - p_3 - p_1 + 2p_9 \cdot p_4 - 2p_9 \cdot p_{10} + 2p_8 \cdot p_3 + 2p_7 \cdot p_1 - 2p_5 \cdot p_{10} + 2p_2 + 2p_{10}$.
Sum/Pos. Premises: `533, 163`.

555. $1 \leqslant +p_9 \cdot p_4 - p_9 \cdot p_{10} + p_8 \cdot p_3 + p_7 \cdot p_1 - p_5 \cdot p_{10} - p_4 \cdot p_1 - p_3 + p_2 + p_{10}$.
Sum/Pos. Premises: `543, 119`.

582. $0 \leqslant -p_8 + p_5 + p_4 - p_{10} - 2p_9 \cdot p_4 + 2p_9 \cdot p_{10}$.
Sum/Pos. Premises: `440, 313`.

598. $0 \leqslant -p_7 + p_5 + p_4 + p_3 - p_{10} - 2p_9 \cdot p_4 + 2p_9 \cdot p_{10} - 2p_8 \cdot p_3$.
Sum/Pos. Premises: `582, 279`.

610. $0 \leqslant +p_5 + p_4 + p_3 - p_2 - p_{10} + p_1 - 2p_9 \cdot p_4 + 2p_9 \cdot p_{10} - 2p_8 \cdot p_3 - 2p_7 \cdot p_1$.
Sum/Pos. Premises: `598, 231`.

622. $-1 \leqslant +p_4 + p_3 + p_1 - 2p_9 \cdot p_4 + 2p_9 \cdot p_{10} - 2p_8 \cdot p_3 - 2p_7 \cdot p_1 + 2p_5 \cdot p_{10} - 2p_2 - 2p_{10}$.
Sum/Pos. Premises: `610, 181`.

632. $0 \leqslant -p_9 \cdot p_4 + p_9 \cdot p_{10} - p_8 \cdot p_3 - p_7 \cdot p_1 + p_5 \cdot p_{10} + p_4 \cdot p_1 + p_3 - p_2 - p_{10}$.
Sum/Pos. Premises: `622, 99`.

668. $1 \leqslant +0$.
Sum/Pos. Premises: `632, 555`.

Total of 54 lines.

6.6 A first-order variation of Tseitin formulas

Our prover is able to use the same "propositional" technique even if the Tseitin tautology is moderately hidden by replacing 0-ary predicates with reasonable first-order terms.

Command line.

```
./bin/saprover -L -S5 -O5 -b2 --samedegreesums --earlyreductions -pon -ftex
tests/first-order/tseitin_FO/tseitin_t_07_03.tptp
```

Proof.

1. $0 \leqslant +p_{07}(f(X_1^1, f(X_1^1, X_1^4))) + p_{02}(f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, X_1^3)))))))) - p_{01}(f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, f(X_1^1, X_1^2)))))))).$
   Initial.

2. $0 \leqslant +p_{07}(f(X_2^1, X_2^4)) - p_{02}(f(X_2^1, f(X_2^1, f(X_2^1, f(X_2^1, f(X_2^1, X_2^3)))))) + p_{01}(f(X_2^1, f(X_2^1, f(X_2^1, f(X_2^1, X_2^2))))).$
   Initial.

3. $0 \leqslant -p_{07}(f(X_3^1, f(X_3^1, f(X_3^1, f(X_3^1, f(X_3^1, X_3^4)))))) + p_{02}(f(X_3^1, f(X_3^1, f(X_3^1, X_3^3)))) + p_{01}(f(X_3^1, f(X_3^1, X_3^2))).$
   Initial.

4. $-2 \leqslant -p_{07}(f(X_4^1, X_4^4)) - p_{02}(f(X_4^1, f(X_4^1, X_4^3))) - p_{01}(f(X_4^1, X_4^2)).$
   Initial.

5. $0 \leqslant +p_{04}(f(X_5^1, X_5^4)) + p_{03}(f(X_5^1, f(X_5^1, X_5^3))) - p_{01}(f(X_5^1, f(X_5^1, f(X_5^1, X_5^2)))).$
   Initial.

6. $0 \leqslant +p_{04}(f(X_6^1, f(X_6^1, X_6^4))) - p_{03}(f(X_6^1, f(X_6^1, f(X_6^1, f(X_6^1, X_6^3))))) + p_{01}(f(X_6^1, f(X_6^1, f(X_6^1, X_6^2)))).$
   Initial.

7. $0 \leqslant -p_{04}(f(X_7^1, f(X_7^1, X_7^4))) + p_{03}(f(X_7^1, f(X_7^1, f(X_7^1, X_7^3)))) + p_{01}(f(X_7^1, f(X_7^1, f(X_7^1, X_7^2)))).$
   Initial.

8. $-2 \leqslant -p_{04}(f(X_8^1, f(X_8^1, f(X_8^1, f(X_8^1, f(X_8^1, X_8^4)))))) - p_{03}(f(X_8^1, f(X_8^1, f(X_8^1, f(X_8^1, X_8^3))))) - p_{01}(f(X_8^1, X_8^2)).$
   Initial.

9. $0 \leqslant +p_{06}(f(X_9^1, X_9^4)) + p_{05}(f(X_9^1, f(X_9^1, X_9^3))) - p_{02}(f(X_9^1, f(X_9^1, f(X_9^1, f(X_9^1, X_9^2))))).$
   Initial.

10. $0 \leqslant +p_{06}(f(X_{10}^1, f(X_{10}^1, f(X_{10}^1, f(X_{10}^1, X_{10}^4))))) - p_{05}(f(X_{10}^1, f(X_{10}^1, X_{10}^3))) + p_{02}(f(X_{10}^1, f(X_{10}^1, X_{10}^2))).$
    Initial.

11. $0 \leqslant -p_{06}(f(X_{11}^1, X_{11}^4)) + p_{05}(f(X_{11}^1, f(X_{11}^1, X_{11}^3))) + p_{02}(f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, f(X_{11}^1, X_{11}^2))))))))).$
    Initial.

12. $-2 \leqslant -p_{06}(f(X_{12}^1, f(X_{12}^1, f(X_{12}^1, f(X_{12}^1, f(X_{12}^1, f(X_{12}^1, f(X_{12}^1, X_{12}^4)))))))) - p_{05}(f(X_{12}^1, f(X_{12}^1, f(X_{12}^1, f(X_{12}^1, f(X_{12}^1, X_{12}^3)))))) - p_{02}(f(X_{12}^1, X_{12}^2)).$
    Initial.

13. $0 \leqslant +p_{08}(f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, X_{13}^4)))) - p_{07}(f(X_{13}^1, f(X_{13}^1, X_{13}^2))) + p_{03}(f(X_{13}^1, f(X_{13}^1, f(X_{13}^1, X_{13}^3)))).$
    Initial.

14. $0 \leqslant +p_{08}(f(X_{14}^1, f(X_{14}^1, f(X_{14}^1, f(X_{14}^1, X_{14}^4))))) + p_{07}(f(X_{14}^1, f(X_{14}^1, X_{14}^2))) - p_{03}(f(X_{14}^1, f(X_{14}^1, X_{14}^3))).$
    Initial.

15. $0 \leqslant -p_{08}(f(X^1_{15}, f(X^1_{15}, f(X^1_{15}, f(X^1_{15}, f(X^1_{15}, f(X^1_{15}, f(X^1_{15}, f(X^1_{15}, f(X^1_{15}, X^4_{15}))))))))) + p_{07}(f(X^1_{15}, X^2_{15})) + p_{03}(f(X^1_{15}, f(X^1_{15}, f(X^1_{15}, X^3_{15}))))$.
Initial.

16. $-2 \leqslant -p_{08}(f(X^1_{16}, f(X^1_{16}, X^4_{16}))) - p_{07}(f(X^1_{16}, f(X^1_{16}, f(X^1_{16}, X^2_{16})))) - p_{03}(f(X^1_{16}, f(X^1_{16}, f(X^1_{16}, f(X^1_{16}, f(X^1_{16}, f(X^1_{16}, X^3_{16})))))))$.
Initial.

17. $0 \leqslant +p_{09}(f(X^1_{17}, f(X^1_{17}, f(X^1_{17}, X^4_{17})))) + p_{08}(f(X^1_{17}, f(X^1_{17}, f(X^1_{17}, f(X^1_{17}, f(X^1_{17}, X^3_{17})))))) - p_{04}(f(X^1_{17}, X^2_{17}))$.
Initial.

18. $0 \leqslant +p_{09}(f(X^1_{18}, X^4_{18})) - p_{08}(f(X^1_{18}, f(X^1_{18}, X^3_{18}))) + p_{04}(f(X^1_{18}, f(X^1_{18}, f(X^1_{18}, X^2_{18}))))$.
Initial.

19. $0 \leqslant -p_{09}(f(X^1_{19}, f(X^1_{19}, X^4_{19}))) + p_{08}(f(X^1_{19}, f(X^1_{19}, X^3_{19}))) + p_{04}(f(X^1_{19}, f(X^1_{19}, f(X^1_{19}, f(X^1_{19}, f(X^1_{19}, f(X^1_{19}, X^2_{19})))))))$.
Initial.

20. $-2 \leqslant -p_{09}(f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, X^4_{20})))) - p_{08}(f(X^1_{20}, f(X^1_{20}, X^3_{20}))) - p_{04}(f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, f(X^1_{20}, X^2_{20})))))$.
Initial.

21. $0 \leqslant +p_{10}(f(X^1_{21}, f(X^1_{21}, X^4_{21}))) + p_{09}(f(X^1_{21}, f(X^1_{21}, X^3_{21}))) - p_{05}(f(X^1_{21}, X^2_{21}))$.
Initial.

22. $0 \leqslant +p_{10}(f(X^1_{22}, f(X^1_{22}, X^4_{22}))) - p_{09}(f(X^1_{22}, X^3_{22})) + p_{05}(f(X^1_{22}, f(X^1_{22}, f(X^1_{22}, f(X^1_{22}, f(X^1_{22}, f(X^1_{22}, X^2_{22})))))))$.
Initial.

23. $0 \leqslant -p_{10}(f(X^1_{23}, X^4_{23})) + p_{09}(f(X^1_{23}, X^3_{23})) + p_{05}(f(X^1_{23}, f(X^1_{23}, f(X^1_{23}, f(X^1_{23}, f(X^1_{23}, f(X^1_{23}, X^2_{23})))))))$.
Initial.

24. $-2 \leqslant -p_{10}(f(X^1_{24}, f(X^1_{24}, f(X^1_{24}, f(X^1_{24}, X^4_{24}))))) - p_{09}(f(X^1_{24}, X^3_{24})) - p_{05}(f(X^1_{24}, f(X^1_{24}, f(X^1_{24}, X^2_{24}))))$.
Initial.

25. $1 \leqslant +p_{10}(f(X^1_{25}, f(X^1_{25}, f(X^1_{25}, X^3_{25})))) + p_{06}(f(X^1_{25}, f(X^1_{25}, f(X^1_{25}, f(X^1_{25}, X^2_{25})))))$.
Initial.

26. $-1 \leqslant -p_{10}(f(X^1_{26}, f(X^1_{26}, f(X^1_{26}, X^3_{26})))) - p_{06}(f(X^1_{26}, f(X^1_{26}, X^2_{26})))$.
Initial.

56. $+p_{10}(f(X^1_{56}, f(X^1_{56}, f(X^1_{56}, X^3_{56})))) = -p_{06}(f(X^1_{56}, f(X^1_{56}, f(X^1_{56}, f(X^1_{56}, X^2_{56}))))) + 1$.
Opposite. Premises: `25, 26`.

80. $0 \leqslant -p_{06}(f(X^1_{80}, f(X^1_{80}, X^2_{80}))) + p_{06}(f(X^1_{80}, f(X^1_{80}, f(X^1_{80}, f(X^1_{80}, X^3_{80})))))$.
Rewrite. Substitution $= \{X^1_{56} \to X^1_{26}, X^3_{56} \to X^3_{26}\}$. Premises: `26, 56`.

141. $0 \leqslant +p_{04}(f(X^1_{141}, f(X^1_{141}, X^6_{141}))) \cdot p_{03}(f(X^1_{141}, f(X^1_{141}, f(X^1_{141}, X^5_{141})))) - p_{04}(f(X^1_{141}, f(X^1_{141}, X^6_{141}))) \cdot p_{03}(f(X^1_{141}, f(X^1_{141}, X^3_{141}))) + p_{04}(f(X^1_{141}, f(X^1_{141}, X^6_{141}))) \cdot p_{01}(f(X^1_{141}, f(X^1_{141}, f(X^1_{141}, X^4_{141})))) + p_{04}(f(X^1_{141}, f(X^1_{141}, X^6_{141}))) \cdot p_{01}(f(X^1_{141}, f(X^1_{141}, f(X^1_{141}, X^2_{141})))) - p_{04}(f(X^1_{141}, f(X^1_{141}, X^6_{141}))) + p_{03}(f(X^1_{141}, f(X^1_{141}, X^3_{141}))) - p_{01}(f(X^1_{141}, f(X^1_{141}, f(X^1_{141}, X^2_{141}))))$.
LiftAndProject. Substitution $= \{X^4_5 \to f(X^1_5, X^4_7), X^1_7 \to X^1_5\}$. Premises: `7, 5`.

151. $0 \leqslant +p_{04}(f(X^1_{151}, f(X^1_{151}, X^5_{151}))) \cdot p_{01}(f(X^1_{151}, f(X^1_{151}, f(X^1_{151}, X^3_{151})))) + p_{04}(f(X^1_{151}, f(X^1_{151}, X^5_{151}))) \cdot p_{01}(f(X^1_{151}, f(X^1_{151}, f(X^1_{151}, X^2_{151})))) - p_{04}(f(X^1_{151}, f(X^1_{151}, X^5_{151}))) + p_{03}(f(X^1_{151}, f(X^1_{151}, f(X^1_{151}, X^4_{151})))) - p_{01}(f(X^1_{151}, f(X^1_{151}, f(X^1_{151}, X^2_{151}))))$.
Factor/Add. Premises: `141`.

168. $0 \leqslant -p_{04}(f(X^1_{168}, f(X^1_{168}, X^4_{168}))) + p_{03}(f(X^1_{168}, f(X^1_{168}, f(X^1_{168}, X^3_{168})))) - p_{01}(f(X^1_{168}, f(X^1_{168}, f(X^1_{168}, X^2_{168})))) + 2p_{04}(f(X^1_{168}, f(X^1_{168}, X^4_{168}))) \cdot p_{01}(f(X^1_{168}, f(X^1_{168}, f(X^1_{168}, X^2_{168}))))$.
Factor/Add. Premises: `151`.

270. $0 \leqslant -p_{04}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^6_{270})))))) \cdot p_{03}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^5_{270}))))) + p_{04}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^6_{270})))))) \cdot p_{03}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^3_{270}))))) - p_{04}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^6_{270})))))) \cdot p_{01}(f(X^1_{270}, X^4_{270})) - p_{04}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^6_{270})))))) \cdot p_{01}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^2_{270})))) + p_{04}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^6_{270})))))) - p_{03}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^3_{270}))))) + p_{01}(f(X^1_{270}, f(X^1_{270}, f(X^1_{270}, X^2_{270}))))$.
LiftAndProject. Substitution $= \{X^4_6 \to f(X^1_6, f(X^1_6, f(X^1_6, X^4_8))), X^1_8 \to X^1_6\}$. Premises: `8, 6`.

287. $0 \leqslant -p_{04}(f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, X^5_{287})))))) \cdot p_{01}(f(X^1_{287}, X^4_{287})) - p_{04}(f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, X^5_{287})))))) \cdot p_{01}(f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, X^2_{287})))) + p_{04}(f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, X^5_{287})))))) - p_{03}(f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, X^3_{287}))))) + p_{01}(f(X^1_{287}, f(X^1_{287}, f(X^1_{287}, X^2_{287}))))$.
Factor/Add. Premises: `270`.

307. $0 \leqslant +p_{04}(f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, X^4_{307})))))) - p_{03}(f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, X^3_{307}))))) + p_{01}(f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, X^2_{307})))) - 2p_{04}(f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, X^4_{307})))))) \cdot p_{01}(f(X^1_{307}, f(X^1_{307}, f(X^1_{307}, X^2_{307}))))$.
Factor/Add. Premises: `287`.

632. $0 \leqslant +p_{06}(f(X^1_{632}, X^4_{632})) \cdot p_{05}(f(X^1_{632}, f(X^1_{632}, X^6_{632}))) - p_{06}(f(X^1_{632}, X^4_{632})) \cdot p_{05}(f(X^1_{632}, f(X^1_{632}, X^3_{632}))) + p_{06}(f(X^1_{632}, X^4_{632})) \cdot p_{02}(f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, X^5_{632})))))))) + p_{06}(f(X^1_{632}, X^4_{632})) \cdot p_{02}(f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, X^2_{632}))))) - p_{06}(f(X^1_{632}, X^4_{632})) + p_{05}(f(X^1_{632}, f(X^1_{632}, X^3_{632}))) - p_{02}(f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, f(X^1_{632}, X^2_{632})))))$.
LiftAndProject. Substitution $= \{X^1_{11} \to X^1_9, X^4_{11} \to X^4_9\}$. Premises: `11, 9`.

650. $0 \leqslant +p_{06}(f(X^1_{650}, X^4_{650})) \cdot p_{02}(f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, X^5_{650})))))))) + p_{06}(f(X^1_{650}, X^4_{650})) \cdot p_{02}(f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, X^2_{650}))))) - p_{06}(f(X^1_{650}, X^4_{650})) + p_{05}(f(X^1_{650}, f(X^1_{650}, X^3_{650}))) - p_{02}(f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, f(X^1_{650}, X^2_{650})))))$.
Factor/Add. Premises: `632`.

671. $0 \leqslant -p_{06}(f(X^1_{671}, X^3_{671})) + p_{05}(f(X^1_{671}, f(X^1_{671}, X^2_{671}))) - p_{02}(f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, X^4_{671})))))))) + 2p_{06}(f(X^1_{671}, X^3_{671})) \cdot p_{02}(f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, X^4_{671}))))))))$.
Factor/Add. Premises: `650`.

817. $-2 \leqslant -p_{06}(f(X^3_{817}, f(X^3_{817}, X^4_{817}))) - p_{05}(f(X^3_{817}, f(X^3_{817}, f(X^3_{817}, f(X^3_{817}, f(X^3_{817}, X^2_{817})))))) - p_{02}(f(X^3_{817}, X^1_{817}))$.

Sum/Pos. Substitution = $\{X_{12}^1 \to X_{80}^1, X_{80}^3 \to f(X_{80}^1, f(X_{80}^1, f(X_{80}^1, X_{12}^4)))\}$. Premises: `12, 80`.

1166. $0 \leqslant +p_{07}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^6)))))) \cdot p_{02}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^5)))) - p_{07}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^6)))))) \cdot p_{02}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^3)))))))) + p_{07}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^6)))))) \cdot p_{01}(f(X_{1166}^1, f(X_{1166}^1, X_{1166}^4))) + p_{07}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^6)))))) \cdot p_{01}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^2)))))))) - p_{07}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^6)))))) + p_{02}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^3)))))))) - p_{01}(f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, f(X_{1166}^1, X_{1166}^2))))))))$.
LiftAndProject. Substitution = $\{X_1^4 \to f(X_1^1, f(X_1^1, f(X_1^1, X_3^4))), X_3^1 \to X_1^1\}$. Premises: `3, 1`.

1181. $0 \leqslant +p_{07}(f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, X_{1181}^5)))))) \cdot p_{01}(f(X_{1181}^1, f(X_{1181}^1, X_{1181}^4))) + p_{07}(f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, X_{1181}^5)))))) \cdot p_{01}(f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, X_{1181}^2)))))))) - p_{07}(f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, X_{1181}^5)))))) + p_{02}(f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, X_{1181}^3)))))))) - p_{01}(f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, f(X_{1181}^1, X_{1181}^2))))))))$.
Factor/Add. Premises: `1166`.

1198. $0 \leqslant -p_{07}(f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, X_{1198}^4)))))) + p_{02}(f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, X_{1198}^3)))))))) - p_{01}(f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, X_{1198}^2)))))))) + 2p_{07}(f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, X_{1198}^4)))))) \cdot p_{01}(f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, f(X_{1198}^1, X_{1198}^2))))))))$.
Factor/Add. Premises: `1181`.

1325. $0 \leqslant -p_{07}(f(X_{1325}^1, X_{1325}^4)) \cdot p_{02}(f(X_{1325}^1, f(X_{1325}^1, X_{1325}^6))) + p_{07}(f(X_{1325}^1, X_{1325}^4)) \cdot p_{02}(f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, X_{1325}^3)))))) - p_{07}(f(X_{1325}^1, X_{1325}^4)) \cdot p_{01}(f(X_{1325}^1, X_{1325}^5)) - p_{07}(f(X_{1325}^1, X_{1325}^4)) \cdot p_{01}(f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, X_{1325}^2))))) + p_{07}(f(X_{1325}^1, X_{1325}^4)) - p_{02}(f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, X_{1325}^3)))))) + p_{01}(f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, f(X_{1325}^1, X_{1325}^2)))))$.
LiftAndProject. Substitution = $\{X_4^1 \to X_2^1, X_4^4 \to X_2^4\}$. Premises: `4, 2`.

1340. $0 \leqslant -p_{07}(f(X_{1340}^1, X_{1340}^4)) \cdot p_{01}(f(X_{1340}^1, X_{1340}^5)) - p_{07}(f(X_{1340}^1, X_{1340}^4)) \cdot p_{01}(f(X_{1340}^1, f(X_{1340}^1, f(X_{1340}^1, f(X_{1340}^1, X_{1340}^2))))) + p_{07}(f(X_{1340}^1, X_{1340}^4)) - p_{02}(f(X_{1340}^1, f(X_{1340}^1, f(X_{1340}^1, f(X_{1340}^1, f(X_{1340}^1, X_{1340}^3)))))) + p_{01}(f(X_{1340}^1, f(X_{1340}^1, f(X_{1340}^1, f(X_{1340}^1, X_{1340}^2)))))$.
Factor/Add. Premises: `1325`.

1359. $0 \leqslant +p_{07}(f(X_{1359}^1, X_{1359}^4)) - p_{02}(f(X_{1359}^1, f(X_{1359}^1, f(X_{1359}^1, f(X_{1359}^1, f(X_{1359}^1, X_{1359}^3)))))) + p_{01}(f(X_{1359}^1, f(X_{1359}^1, f(X_{1359}^1, f(X_{1359}^1, X_{1359}^2))))) - 2p_{07}(f(X_{1359}^1, X_{1359}^4)) \cdot p_{01}(f(X_{1359}^1, f(X_{1359}^1, f(X_{1359}^1, f(X_{1359}^1, X_{1359}^2)))))$.
Factor/Add. Premises: `1340`.

1575. $0 \leqslant +p_{08}(f(X_{1575}^1, f(X_{1575}^1, f(X_{1575}^1, f(X_{1575}^1, f(X_{1575}^1, f(X_{1575}^1, f(X_{1575}^1, f(X_{1575}^1, f(X_{1575}^1, X_{1575}^6)))))))))) \cdot p_{07}(f(X_{1575}^1, X_{1575}^4)) -$

$p_{08}(f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, X^6_{1575}))))))))))\cdot$
$p_{07}(f(X^1_{1575}, f(X^1_{1575}, X^2_{1575}))) +$
$p_{08}(f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, X^6_{1575}))))))))))\cdot$
$p_{03}(f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, X^5_{1575})))) +$
$p_{08}(f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, X^6_{1575}))))))))))\cdot$
$p_{03}(f(X^1_{1575}, f(X^1_{1575}, X^3_{1575}))) -$
$p_{08}(f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, f(X^1_{1575}, X^6_{1575}))))))))))+$
$p_{07}(f(X^1_{1575}, f(X^1_{1575}, X^2_{1575}))) - p_{03}(f(X^1_{1575}, f(X^1_{1575}, X^3_{1575})))$.
LiftAndProject. Substitution =
$\{X^4_{14} \to f(X^1_{14}, f(X^1_{14}, f(X^1_{14}, f(X^1_{14}, f(X^1_{14}, X^4_{15}))))), X^1_{15} \to X^1_{14}\}$. Premises: `15, 14`.

1588. $0 \leqslant$
$+p_{08}(f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, X^5_{1588}))))))))))\cdot$
$p_{03}(f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, X^4_{1588})))) +$
$p_{08}(f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, X^5_{1588}))))))))))\cdot$
$p_{03}(f(X^1_{1588}, f(X^1_{1588}, X^3_{1588}))) -$
$p_{08}(f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, f(X^1_{1588}, X^5_{1588}))))))))))+$
$p_{07}(f(X^1_{1588}, f(X^1_{1588}, X^2_{1588}))) - p_{03}(f(X^1_{1588}, f(X^1_{1588}, X^3_{1588})))$.
Factor/Add. Premises: `1575`.

1605. $0 \leqslant$
$-p_{08}(f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, X^4_{1605}))))))))))+$
$p_{07}(f(X^1_{1605}, f(X^1_{1605}, X^2_{1605}))) - p_{03}(f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, X^3_{1605})))) +$
$2p_{08}(f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, X^4_{1605}))))))))))\cdot$
$p_{03}(f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, X^3_{1605}))))$.
Factor/Add. Premises: `1588`.

1735. $0 \leqslant -p_{08}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^4_{1735})))) \cdot p_{07}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^5_{1735})))) +$
$p_{08}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^4_{1735})))) \cdot p_{07}(f(X^1_{1735}, f(X^1_{1735}, X^2_{1735}))) -$
$p_{08}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^4_{1735})))) \cdot$
$p_{03}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^6_{1735}))))))) -$
$p_{08}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^4_{1735})))) \cdot p_{03}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^3_{1735})))) +$
$p_{08}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^4_{1735})))) - p_{07}(f(X^1_{1735}, f(X^1_{1735}, X^2_{1735}))) +$
$p_{03}(f(X^1_{1735}, f(X^1_{1735}, f(X^1_{1735}, X^3_{1735}))))$.
LiftAndProject. Substitution = $\{X^1_{16} \to X^1_{13}, X^4_{16} \to f(X^1_{13}, X^4_{13})\}$. Premises: `16, 13`.

1749. $0 \leqslant -p_{08}(f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, X^3_{1749})))) \cdot$
$p_{03}(f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, X^5_{1749}))))))) -$
$p_{08}(f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, X^3_{1749})))) \cdot p_{03}(f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, X^2_{1749})))) +$
$p_{08}(f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, X^3_{1749})))) - p_{07}(f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, X^4_{1749})))) +$
$p_{03}(f(X^1_{1749}, f(X^1_{1749}, f(X^1_{1749}, X^2_{1749}))))$.
Factor/Add. Premises: `1735`.

1768. $0 \leqslant +p_{08}(f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, X^2_{1768})))) - p_{07}(f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, X^3_{1768})))) +$
$p_{03}(f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, X^4_{1768}))))))) -$
$2p_{08}(f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, X^2_{1768})))) \cdot$
$p_{03}(f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, f(X^1_{1768}, X^4_{1768})))))))$.
Factor/Add. Premises: `1749`.

1995. $0 \leqslant +p_{09}(f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, X^4_{1995})))) \cdot p_{08}(f(X^1_{1995}, f(X^1_{1995}, X^6_{1995}))) -$
$p_{09}(f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, X^4_{1995})))) \cdot$
$p_{08}(f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, X^3_{1995})))))) +$

$p_{09}(f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, X^4_{1995})))) \cdot$
$p_{04}(f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, X^5_{1995})))))))+$
$p_{09}(f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, X^4_{1995})))) \cdot p_{04}(f(X^1_{1995}, X^2_{1995})) -$
$p_{09}(f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, X^4_{1995})))) +$
$p_{08}(f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, f(X^1_{1995}, X^3_{1995})))))) - p_{04}(f(X^1_{1995}, X^2_{1995}))$.
LiftAndProject. Substitution = $\{X^1_{19} \to X^1_{17}, X^4_{19} \to f(X^1_{17}, X^4_{17})\}$. Premises: 19, 17.

2010. $0 \leqslant +p_{09}(f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, X^4_{2010})))) \cdot$
$p_{04}(f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, X^5_{2010}))))))) +$
$p_{09}(f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, X^4_{2010})))) \cdot p_{04}(f(X^1_{2010}, X^2_{2010})) -$
$p_{09}(f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, X^4_{2010})))) +$
$p_{08}(f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, f(X^1_{2010}, X^3_{2010})))))) - p_{04}(f(X^1_{2010}, X^2_{2010}))$.
Factor/Add. Premises: 1995.

2027. $0 \leqslant -p_{09}(f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, X^3_{2027})))) +$
$p_{08}(f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, X^2_{2027})))))) -$
$p_{04}(f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, X^4_{2027}))))))) +$
$2p_{09}(f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, X^3_{2027})))) \cdot$
$p_{04}(f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, f(X^1_{2027}, X^4_{2027})))))))$.
Factor/Add. Premises: 2010.

2192. $0 \leqslant -p_{09}(f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, X^6_{2192})))) \cdot p_{08}(f(X^1_{2192}, f(X^1_{2192}, X^5_{2192}))) +$
$p_{09}(f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, X^6_{2192})))) \cdot p_{08}(f(X^1_{2192}, f(X^1_{2192}, X^3_{2192}))) -$
$p_{09}(f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, X^6_{2192})))) \cdot p_{04}(f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, X^4_{2192}))))) -$
$p_{09}(f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, X^6_{2192})))) \cdot p_{04}(f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, X^2_{2192})))) +$
$p_{09}(f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, X^6_{2192})))) - p_{08}(f(X^1_{2192}, f(X^1_{2192}, X^3_{2192}))) +$
$p_{04}(f(X^1_{2192}, f(X^1_{2192}, f(X^1_{2192}, X^2_{2192}))))$.
LiftAndProject. Substitution = $\{X^4_{18} \to f(X^1_{18}, f(X^1_{18}, X^4_{20})), X^1_{20} \to X^1_{18}\}$. Premises: 20, 18.

2205. $0 \leqslant -p_{09}(f(X^1_{2205}, f(X^1_{2205}, f(X^1_{2205}, X^5_{2205})))) \cdot$
$p_{04}(f(X^1_{2205}, f(X^1_{2205}, f(X^1_{2205}, f(X^1_{2205}, X^4_{2205}))))) - p_{09}(f(X^1_{2205}, f(X^1_{2205}, f(X^1_{2205}, X^5_{2205})))) \cdot$
$p_{04}(f(X^1_{2205}, f(X^1_{2205}, f(X^1_{2205}, X^2_{2205})))) + p_{09}(f(X^1_{2205}, f(X^1_{2205}, f(X^1_{2205}, X^5_{2205})))) -$
$p_{08}(f(X^1_{2205}, f(X^1_{2205}, X^3_{2205}))) + p_{04}(f(X^1_{2205}, f(X^1_{2205}, f(X^1_{2205}, X^2_{2205}))))$.
Factor/Add. Premises: 2192.

2223. $0 \leqslant +p_{09}(f(X^1_{2223}, f(X^1_{2223}, f(X^1_{2223}, X^4_{2223})))) - p_{08}(f(X^1_{2223}, f(X^1_{2223}, X^2_{2223}))) +$
$p_{04}(f(X^1_{2223}, f(X^1_{2223}, f(X^1_{2223}, f(X^1_{2223}, X^3_{2223}))))) -$
$2p_{09}(f(X^1_{2223}, f(X^1_{2223}, f(X^1_{2223}, X^4_{2223})))) \cdot p_{04}(f(X^1_{2223}, f(X^1_{2223}, f(X^1_{2223}, f(X^1_{2223}, X^3_{2223})))))$.
Factor/Add. Premises: 2205.

2449. $-1 \leqslant +p_{09}(f(X^1_{2449}, f(X^1_{2449}, X^3_{2449}))) - p_{06}(f(X^1_{2449}, f(X^1_{2449}, f(X^1_{2449}, f(X^1_{2449}, X^4_{2449}))))) -$
$p_{05}(f(X^1_{2449}, X^2_{2449}))$.
SubstRewrite. Substitution = $\{X^4_{21} \to f(X^1_{21}, X^3_{56}), X^1_{56} \to X^1_{21}\}$. Premises: 21, 56.

2480. $-1 \leqslant -p_{09}(f(X^1_{2480}, X^3_{2480})) - p_{06}(f(X^1_{2480}, f(X^1_{2480}, f(X^1_{2480}, f(X^1_{2480}, X^4_{2480}))))) +$
$p_{05}(f(X^1_{2480}, f(X^1_{2480}, f(X^1_{2480}, f(X^1_{2480}, f(X^1_{2480}, f(X^1_{2480}, X^2_{2480})))))))$.
SubstRewrite. Substitution = $\{X^4_{22} \to f(X^1_{22}, X^3_{56}), X^1_{56} \to X^1_{22}\}$. Premises: 22, 56.

2517. $1 \leqslant +p_{09}(f(X^1_{2517}, X^3_{2517})) + p_{06}(f(X^1_{2517}, f(X^1_{2517}, f(X^1_{2517}, f(X^1_{2517}, X^4_{2517}))))) +$
$p_{05}(f(X^1_{2517}, f(X^1_{2517}, f(X^1_{2517}, f(X^1_{2517}, f(X^1_{2517}, f(X^1_{2517}, X^2_{2517})))))))$.
SubstRewrite. Substitution = $\{X^4_{23} \to f(X^1_{23}, f(X^1_{23}, X^3_{56})), X^1_{56} \to X^1_{23}\}$. Premises: 23, 56.

3654. $-1 \leqslant -p_{09}(f(X^1_{3654}, X^3_{3654})) + p_{06}(f(X^1_{3654}, f(X^1_{3654}, f(X^1_{3654}, f(X^1_{3654}, X^4_{3654}))))) - p_{05}(f(X^1_{3654}, f(X^1_{3654}, f(X^1_{3654}, X^2_{3654}))))$.
Rewrite. Substitution $= \{X^1_{56} \to X^1_{24}, X^3_{56} \to f(X^1_{24}, X^4_{24})\}$. Premises: `24, 56`.

3775. $0 \leqslant -p_{06}(f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, X^4_{3775}))))) \cdot p_{05}(f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, X^6_{3775})))))) + p_{06}(f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, X^4_{3775}))))) \cdot p_{05}(f(X^1_{3775}, f(X^1_{3775}, X^3_{3775}))) - p_{06}(f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, X^4_{3775}))))) \cdot p_{02}(f(X^1_{3775}, X^5_{3775})) - p_{06}(f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, X^4_{3775}))))) \cdot p_{02}(f(X^1_{3775}, f(X^1_{3775}, X^2_{3775}))) + p_{06}(f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, f(X^1_{3775}, X^4_{3775}))))) - p_{05}(f(X^1_{3775}, f(X^1_{3775}, X^3_{3775}))) + p_{02}(f(X^1_{3775}, f(X^1_{3775}, X^2_{3775})))$.
LiftAndProject. Substitution $= \{X^3_{817} \to X^1_{10}, X^4_{817} \to f(X^1_{10}, f(X^1_{10}, X^4_{10}))\}$. Premises: `817, 10`.

3802. $0 \leqslant -p_{06}(f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, X^3_{3802}))))) \cdot p_{02}(f(X^1_{3802}, X^4_{3802})) - p_{06}(f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, X^3_{3802}))))) \cdot p_{02}(f(X^1_{3802}, f(X^1_{3802}, X^2_{3802}))) + p_{06}(f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, X^3_{3802}))))) - p_{05}(f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, f(X^1_{3802}, X^5_{3802})))))) + p_{02}(f(X^1_{3802}, f(X^1_{3802}, X^2_{3802})))$.
Factor/Add. Premises: `3775`.

3836. $0 \leqslant +p_{06}(f(X^1_{3836}, f(X^1_{3836}, f(X^1_{3836}, f(X^1_{3836}, X^3_{3836}))))) - p_{05}(f(X^1_{3836}, f(X^1_{3836}, f(X^1_{3836}, f(X^1_{3836}, f(X^1_{3836}, X^4_{3836})))))) + p_{02}(f(X^1_{3836}, f(X^1_{3836}, X^2_{3836}))) - 2p_{06}(f(X^1_{3836}, f(X^1_{3836}, f(X^1_{3836}, f(X^1_{3836}, X^3_{3836}))))) \cdot p_{02}(f(X^1_{3836}, f(X^1_{3836}, X^2_{3836})))$.
Factor/Add. Premises: `3802`.

4305. $-1 \leqslant -p_{09}(f(X^1_{4305}, f(X^1_{4305}, X^3_{4305}))) \cdot p_{06}(f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, X^6_{4305}))))) + p_{09}(f(X^1_{4305}, f(X^1_{4305}, X^3_{4305}))) \cdot p_{06}(f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, X^4_{4305}))))) + p_{09}(f(X^1_{4305}, f(X^1_{4305}, X^3_{4305}))) \cdot p_{05}(f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, X^5_{4305}))))))) + p_{09}(f(X^1_{4305}, f(X^1_{4305}, X^3_{4305}))) \cdot p_{05}(f(X^1_{4305}, X^2_{4305})) - p_{09}(f(X^1_{4305}, f(X^1_{4305}, X^3_{4305}))) - p_{06}(f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, f(X^1_{4305}, X^4_{4305}))))) - p_{05}(f(X^1_{4305}, X^2_{4305}))$.
LiftAndProject. Substitution $= \{X^1_{2480} \to X^1_{2449}, X^3_{2480} \to f(X^1_{2449}, X^3_{2449})\}$. Premises: `2480, 2449`.

4316. $-1 \leqslant +p_{09}(f(X^1_{4316}, f(X^1_{4316}, X^3_{4316}))) \cdot p_{05}(f(X^1_{4316}, f(X^1_{4316}, f(X^1_{4316}, f(X^1_{4316}, f(X^1_{4316}, f(X^1_{4316}, X^5_{4316}))))))) + p_{09}(f(X^1_{4316}, f(X^1_{4316}, X^3_{4316}))) \cdot p_{05}(f(X^1_{4316}, X^2_{4316})) - p_{09}(f(X^1_{4316}, f(X^1_{4316}, X^3_{4316}))) - p_{06}(f(X^1_{4316}, f(X^1_{4316}, f(X^1_{4316}, f(X^1_{4316}, X^4_{4316}))))) - p_{05}(f(X^1_{4316}, X^2_{4316}))$.
Factor/Add. Premises: `4305`.

4339. $-1 \leqslant -p_{09}(f(X^1_{4339}, f(X^1_{4339}, X^2_{4339}))) - p_{06}(f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, X^3_{4339}))))) - p_{05}(f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, X^4_{4339}))))))) + 2p_{09}(f(X^1_{4339}, f(X^1_{4339}, X^2_{4339}))) \cdot p_{05}(f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, f(X^1_{4339}, X^4_{4339})))))))$.
Factor/Add. Premises: `4316`.

4514. $-1 \leqslant -p_{08}(f(X^1_{4514}, f(X^1_{4514}, X^2_{4514}))) - p_{06}(f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, X^5_{4514}))))) - p_{05}(f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, X^6_{4514}))))))) + p_{04}(f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, X^3_{4514}))))) + 2p_{09}(f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, X^4_{4514})))) \cdot p_{05}(f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, X^6_{4514}))))))) - 2p_{09}(f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, X^4_{4514})))) \cdot p_{04}(f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, f(X^1_{4514}, X^3_{4514})))))$.

Sum/Pos. Substitution = $\{X^1_{4339} \to X^1_{2223}, X^2_{4339} \to f(X^1_{2223}, X^4_{2223})\}$. Premises: `4339, 2223`.

4573. $-1 \leqslant -p_{07}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^3_{4573})))) -$
$p_{06}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^7_{4573}))))) -$
$p_{05}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^8_{4573})))))))+$
$p_{04}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^5_{4573})))))+$
$p_{03}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^4_{4573})))))))+$
$2p_{09}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^6_{4573})))) \cdot$
$p_{05}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^8_{4573})))))))-$
$2p_{09}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^6_{4573})))) \cdot$
$p_{04}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^5_{4573}))))) -$
$2p_{08}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^2_{4573})))) \cdot$
$p_{03}(f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, f(X^1_{4573}, X^4_{4573}))))))).$
Sum/Pos. Substitution = $\{X^1_{4514} \to X^1_{1768}, X^2_{4514} \to f(X^1_{1768}, X^2_{1768})\}$. Premises: `4514, 1768`.

4650. $-1 \leqslant -p_{06}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^9_{4650}))))) -$
$p_{05}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^{10}_{4650})))))))+$
$p_{04}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^7_{4650})))))+$
$p_{03}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^6_{4650}))))))) -$
$p_{02}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^3_{4650}))))))+$
$p_{01}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^2_{4650})))))+$
$2p_{09}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^8_{4650})))) \cdot$
$p_{05}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^{10}_{4650}))))))) -$
$2p_{09}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^8_{4650})))) \cdot$
$p_{04}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^7_{4650}))))) -$
$2p_{08}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^4_{4650})))) \cdot$
$p_{03}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^6_{4650}))))))) -$
$2p_{07}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^5_{4650})))) \cdot p_{01}(f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, f(X^1_{4650}, X^2_{4650}))))).$
Sum/Pos. Substitution = $\{X^4_{1359} \to f(X^1_{1359}, f(X^1_{1359}, X^3_{4573})), X^1_{4573} \to X^1_{1359}\}$. Premises: `4573, 1359`.

4783. $-1 \leqslant -p_{05}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^{12}_{4783}))))))) -$
$p_{05}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^4_{4783}))))))+$
$p_{04}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^{10}_{4783})))))+$
$p_{03}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^9_{4783}))))))) -$
$p_{02}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^6_{4783}))))))+$
$p_{02}(f(X^1_{4783}, f(X^1_{4783}, X^2_{4783}))) + p_{01}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^5_{4783}))))) +$
$2p_{09}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^{11}_{4783})))) \cdot$
$p_{05}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^{12}_{4783}))))))) -$
$2p_{09}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^{11}_{4783})))) \cdot$
$p_{04}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^{10}_{4783}))))) -$
$2p_{08}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^7_{4783})))) \cdot$
$p_{03}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^9_{4783}))))))) -$
$2p_{07}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^8_{4783})))) \cdot$
$p_{01}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^5_{4783}))))) -$
$2p_{06}(f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, f(X^1_{4783}, X^3_{4783}))))) \cdot p_{02}(f(X^1_{4783}, f(X^1_{4783}, X^2_{4783}))).$
Sum/Pos. Substitution = $\{X^1_{4650} \to X^1_{3836}, X^9_{4650} \to X^3_{3836}\}$. Premises: `4650, 3836`.

4787. $-1 \leqslant +p_{04}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^9_{4787})))))+$
$p_{03}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^8_{4787}))))))) -$

$p_{02}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^5_{4787})))))) +$
$p_{02}(f(X^1_{4787}, f(X^1_{4787}, X^2_{4787}))) + p_{01}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^4_{4787}))))) +$
$2p_{09}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^{10}_{4787})))) \cdot$
$p_{05}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^{11}_{4787}))))))) -$
$2p_{09}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^{10}_{4787})))) \cdot$
$p_{04}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^9_{4787}))))) -$
$2p_{08}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^6_{4787})))) \cdot$
$p_{03}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^8_{4787}))))))) -$
$2p_{07}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^7_{4787})))) \cdot$
$p_{01}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^4_{4787}))))) -$
$2p_{06}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^3_{4787}))))) \cdot p_{02}(f(X^1_{4787}, f(X^1_{4787}, X^2_{4787}))) -$
$2p_{05}(f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, f(X^1_{4787}, X^{11}_{4787}))))))).$
Factor/Add. Premises: `4783`.

4796. $-1 \leqslant +p_{04}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^8_{4796}))))) +$
$p_{03}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^7_{4796}))))))) +$
$p_{01}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^3_{4796}))))) +$
$2p_{09}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^9_{4796})))) \cdot$
$p_{05}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^{10}_{4796}))))))) -$
$2p_{09}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^9_{4796})))) \cdot$
$p_{04}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^8_{4796}))))) -$
$2p_{08}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^5_{4796})))) \cdot$
$p_{03}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^7_{4796}))))))) -$
$2p_{07}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^6_{4796})))) \cdot$
$p_{01}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^3_{4796}))))) -$
$2p_{06}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^2_{4796}))))) \cdot$
$p_{02}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^4_{4796})))))) -$
$2p_{05}(f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, f(X^1_{4796}, X^{10}_{4796}))))))).$
Factor/Add. Premises: `4787`.

5240. $-1 \leqslant +p_{03}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^9_{5240}))))))) +$
$p_{03}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^3_{5240})))) +$
$p_{01}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^5_{5240}))))) -$
$p_{01}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^2_{5240})))) + 2p_{09}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^{11}_{5240})))) \cdot$
$p_{05}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^{12}_{5240}))))))) -$
$2p_{09}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^{11}_{5240})))) \cdot$
$p_{04}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^{10}_{5240}))))) -$
$2p_{08}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^7_{5240})))) \cdot$
$p_{03}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^9_{5240}))))))) -$
$2p_{07}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^8_{5240})))) \cdot$
$p_{01}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^5_{5240}))))) -$
$2p_{06}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^4_{5240}))))) \cdot$
$p_{02}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^6_{5240})))))) -$
$2p_{05}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^{12}_{5240}))))))) +$
$2p_{04}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^{10}_{5240}))))) \cdot p_{01}(f(X^1_{5240}, f(X^1_{5240}, f(X^1_{5240}, X^2_{5240})))).$
Sum/Pos. Substitution = $\{X^4_{168} \to f(X^1_{168}, f(X^1_{168}, X^8_{4796})), X^1_{4796} \to X^1_{168}\}$. Premises:
`4796, 168`.

5244. $-1 \leqslant +p_{01}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^4_{5244}))))) -$
$p_{01}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^2_{5244})))) + 2p_{09}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^{10}_{5244})))) \cdot$
$p_{05}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^{11}_{5244}))))))) -$

$2p_{09}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^{10}_{5244})))) \cdot$
$p_{04}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^9_{5244}))))) -$
$2p_{08}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^6_{5244})))) \cdot$
$p_{03}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^8_{5244}))))))) -$
$2p_{07}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^7_{5244})))) \cdot$
$p_{01}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^4_{5244}))))) -$
$2p_{06}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^3_{5244}))))) \cdot$
$p_{02}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^5_{5244})))))) -$
$2p_{05}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^{11}_{5244}))))))) +$
$2p_{04}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^9_{5244}))))) \cdot$
$p_{01}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^2_{5244})))) +$
$2p_{03}(f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, f(X^1_{5244}, X^8_{5244}))))))).$
Factor/Add. Premises: `5240`.

5250. $0 \leqslant +p_{09}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^9_{5250})))) \cdot$
$p_{05}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^{10}_{5250}))))))) -$
$p_{09}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^9_{5250})))) \cdot p_{04}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^8_{5250}))))) -$
$p_{08}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^5_{5250})))) \cdot$
$p_{03}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^7_{5250}))))))) -$
$p_{07}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^6_{5250})))) \cdot p_{01}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^3_{5250}))))) -$
$p_{06}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^2_{5250}))))) \cdot$
$p_{02}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^4_{5250})))))) -$
$p_{05}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^{10}_{5250}))))))) +$
$p_{04}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^8_{5250}))))) \cdot$
$p_{01}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^3_{5250}))))) +$
$p_{03}(f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, f(X^1_{5250}, X^7_{5250}))))))).$
Factor/Add. Premises: `5244`.

5487. $1 \leqslant -p_{09}(f(X^3_{5487}, X^5_{5487})) \cdot p_{06}(f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, X^2_{5487}))))) +$
$p_{09}(f(X^3_{5487}, X^5_{5487})) \cdot p_{06}(f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, X^6_{5487}))))) -$
$p_{09}(f(X^3_{5487}, X^5_{5487})) \cdot p_{05}(f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, X^1_{5487}))))))) -$
$p_{09}(f(X^3_{5487}, X^5_{5487})) \cdot p_{05}(f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, X^4_{5487})))) + p_{09}(f(X^3_{5487}, X^5_{5487})) +$
$p_{06}(f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, X^2_{5487}))))) +$
$p_{05}(f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, f(X^3_{5487}, X^1_{5487}))))))).$
LiftAndProject. Substitution $= \{X^1_{2517} \to X^1_{3654}, X^3_{2517} \to X^3_{3654}\}$. Premises: `2517, 3654`.

5522. $1 \leqslant -p_{09}(f(X^2_{5522}, X^4_{5522})) \cdot$
$p_{05}(f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, X^1_{5522}))))))) -$
$p_{09}(f(X^2_{5522}, X^4_{5522})) \cdot p_{05}(f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, X^3_{5522})))) + p_{09}(f(X^2_{5522}, X^4_{5522})) +$
$p_{06}(f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, X^5_{5522}))))) +$
$p_{05}(f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, f(X^2_{5522}, X^1_{5522}))))))).$
Factor/Add. Premises: `5487`.

5557. $1 \leqslant +p_{09}(f(X^2_{5557}, X^3_{5557})) + p_{06}(f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, X^4_{5557}))))) +$
$p_{05}(f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, X^1_{5557}))))))) -$
$2p_{09}(f(X^2_{5557}, X^3_{5557})) \cdot p_{05}(f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, f(X^2_{5557}, X^1_{5557}))))))).$
Factor/Add. Premises: `5522`.

5887. $1 \leqslant +p_{08}(f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, X^2_{5887})))))) +$
$p_{06}(f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, X^6_{5887}))))) +$
$p_{05}(f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, X^5_{5887}))))))) -$
$p_{04}(f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, X^4_{5887}))))))) -$

$2p_{09}(f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, X^3_{5887})))) \cdot$
$p_{05}(f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, X^5_{5887})))))))+$
$2p_{09}(f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, X^3_{5887})))) \cdot$
$p_{04}(f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, f(X^1_{5887}, X^4_{5887}))))))).$
Sum/Pos. Substitution $= \{X^2_{5557} \to X^1_{2027}, X^3_{5557} \to f(X^1_{2027}, f(X^1_{2027}, X^3_{2027}))\}$. Premises: `5557, 2027`.

6046. $1 \leqslant +p_{07}(f(X^1_{6046}, f(X^1_{6046}, X^2_{6046}))) + p_{06}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^8_{6046}))))) +$
$p_{05}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^7_{6046}))))))) -$
$p_{04}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^6_{6046}))))))) -$
$p_{03}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^3_{6046})))) - 2p_{09}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^5_{6046})))) \cdot$
$p_{05}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^7_{6046}))))))) +$
$2p_{09}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^5_{6046})))) \cdot$
$p_{04}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^6_{6046}))))))) +$
$2p_{08}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^4_{6046})))))))))) \cdot$
$p_{03}(f(X^1_{6046}, f(X^1_{6046}, f(X^1_{6046}, X^3_{6046})))).$
Sum/Pos. Substitution $=$
$\{X^1_{5887} \to X^1_{1605}, X^2_{5887} \to f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, f(X^1_{1605}, X^4_{1605}))))\}$. Premises: `5887, 1605`.

7488. $1 \leqslant +p_{06}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^{10}_{7488}))))) +$
$p_{05}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^9_{7488}))))))) -$
$p_{04}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^8_{7488}))))))) -$
$p_{03}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^5_{7488})))) +$
$p_{02}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^3_{7488})))))))) -$
$p_{01}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^2_{7488})))))))) -$
$2p_{09}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^7_{7488})))) \cdot$
$p_{05}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^9_{7488}))))))) +$
$2p_{09}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^7_{7488})))) \cdot$
$p_{04}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^8_{7488}))))))) +$
$2p_{08}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^6_{7488})))))))))) \cdot$
$p_{03}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^5_{7488})))) +$
$2p_{07}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^4_{7488})))))) \cdot$
$p_{01}(f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, f(X^1_{7488}, X^2_{7488})))))))).$
Sum/Pos. Substitution $= \{X^1_{6046} \to X^1_{1198}, X^2_{6046} \to f(X^1_{1198}, f(X^1_{1198}, f(X^1_{1198}, X^4_{1198})))\}$.
Premises: `6046, 1198`.

28694. $1 \leqslant +p_{05}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^{11}_{28694}))))))) +$
$p_{05}(f(X^1_{28694}, f(X^1_{28694}, X^2_{28694}))) -$
$p_{04}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^{10}_{28694}))))))) -$
$p_{03}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^7_{28694})))) +$
$p_{02}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^5_{28694})))))))) -$
$p_{02}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^3_{28694}))))))))) -$
$p_{01}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^4_{28694})))))))) -$
$2p_{09}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^9_{28694})))) \cdot$
$p_{05}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^{11}_{28694}))))))) +$
$2p_{09}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^9_{28694})))) \cdot$
$p_{04}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^{10}_{28694}))))))) +$
$2p_{08}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^8_{28694})))))))))) \cdot$
$p_{03}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^7_{28694})))) +$
$2p_{07}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^6_{28694})))))) \cdot$

$p_{01}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^4_{28694})))))))) +$
$2p_{06}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^{12}_{28694})))))\cdot$
$p_{02}(f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, f(X^1_{28694}, X^3_{28694}))))))))).$
Sum/Pos. Substitution = $\{X^3_{671} \to f(X^1_{671}, f(X^1_{671}, f(X^1_{671}, X^{10}_{7488}))), X^1_{7488} \to X^1_{671}\}$.
Premises: `7488, 671`.

28804. $1 \leqslant -p_{04}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^9_{28804})))))))-$
$p_{03}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^6_{28804})))) +$
$p_{02}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^4_{28804})))))))) -$
$p_{02}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^2_{28804}))))))))) -$
$p_{01}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^3_{28804})))))))) -$
$2p_{09}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^8_{28804}))))\cdot$
$p_{05}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^{10}_{28804}))))))) +$
$2p_{09}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^8_{28804}))))\cdot$
$p_{04}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^9_{28804}))))))) +$
$2p_{08}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^7_{28804}))))))))))\cdot$
$p_{03}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^6_{28804})))) +$
$2p_{07}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^5_{28804}))))))\cdot$
$p_{01}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^3_{28804})))))))) +$
$2p_{06}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^{11}_{28804})))))\cdot$
$p_{02}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^2_{28804}))))))))) +$
$2p_{05}(f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, f(X^1_{28804}, X^{10}_{28804}))))))).$
Factor/Add. Premises: `28694`.

30422. $1 \leqslant -p_{04}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^8_{30422}))))))) -$
$p_{03}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^5_{30422})))) -$
$p_{01}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^3_{30422})))))))) -$
$2p_{09}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^7_{30422}))))\cdot$
$p_{05}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^9_{30422}))))))) +$
$2p_{09}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^7_{30422}))))\cdot$
$p_{04}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^8_{30422}))))))) +$
$2p_{08}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^6_{30422}))))))))))\cdot$
$p_{03}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^5_{30422})))) +$
$2p_{07}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^4_{30422}))))))\cdot$
$p_{01}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^3_{30422})))))))) +$
$2p_{06}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^{10}_{30422})))))\cdot$
$p_{02}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^2_{30422}))))))))) +$
$2p_{05}(f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, f(X^1_{30422}, X^9_{30422}))))))).$
Factor/Add. Premises: `28804`.

339914. $1 \leqslant -p_{03}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^7_{339914})))) -$
$p_{03}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^3_{339914}))))) -$
$p_{01}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^5_{339914})))))))) +$
$p_{01}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^2_{339914})))) -$
$2p_{09}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^9_{339914}))))\cdot$
$p_{05}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^{11}_{339914}))))))) +$
$2p_{09}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^9_{339914}))))\cdot$
$p_{04}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^{10}_{339914}))))))) +$
$2p_{08}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^8_{339914}))))))))))\cdot$
$p_{03}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^7_{339914})))) +$
$2p_{07}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^6_{339914}))))))\cdot$

$p_{01}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^5_{339914}))))))))+$
$2p_{06}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^{12}_{339914})))))\ \cdot$
$p_{02}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^4_{339914})))))))))+$
$2p_{05}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^{11}_{339914})))))))\ -$
$2p_{04}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^{10}_{339914})))))))\ \cdot$
$p_{01}(f(X^1_{339914}, f(X^1_{339914}, f(X^1_{339914}, X^2_{339914})))).$
Sum/Pos. Substitution = $\{X^4_{307} \to f(X^1_{307}, X^8_{30422}), X^1_{30422} \to X^1_{307}\}$. Premises: `30422`, `307`.

339944. $1 \leqslant$
$-p_{01}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^5_{339944}))))))))+$
$p_{01}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^2_{339944}))))\ -$
$2p_{09}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^8_{339944}))))\ \cdot$
$p_{05}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^{10}_{339944})))))))\ +$
$2p_{09}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^8_{339944}))))\ \cdot$
$p_{04}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^9_{339944})))))))\ +$
$2p_{08}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^7_{339944}))))))))))\cdot$
$p_{03}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^3_{339944})))))\ +$
$2p_{07}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^6_{339944}))))))\ \cdot$
$p_{01}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^5_{339944}))))))))+$
$2p_{06}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^{11}_{339944})))))\ \cdot$
$p_{02}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^4_{339944})))))))))+$
$2p_{05}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^{10}_{339944})))))))\ -$
$2p_{04}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^9_{339944})))))))\ \cdot$
$p_{01}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^2_{339944}))))\ -$
$2p_{03}(f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, f(X^1_{339944}, X^3_{339944}))))).$
Factor/Add. Premises: `339914`.

340481. $1 \leqslant -p_{09}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^7_{340481}))))\ \cdot$
$p_{05}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^9_{340481})))))))\ +$
$p_{09}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^7_{340481}))))\ \cdot$
$p_{04}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^8_{340481})))))))\ +$
$p_{08}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^6_{340481}))))))))))\cdot$
$p_{03}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^2_{340481})))))\ +$
$p_{07}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^5_{340481}))))))\ \cdot$
$p_{01}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^4_{340481}))))))))+$
$p_{06}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^{10}_{340481})))))\ \cdot$
$p_{02}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^3_{340481})))))))))+$
$p_{05}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^9_{340481})))))))\ -$
$p_{04}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^8_{340481})))))))\ \cdot$
$p_{01}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^4_{340481}))))))))-$
$p_{03}(f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, f(X^1_{340481}, X^2_{340481}))))).$
Factor/Add. Premises: `339944`.

397221. $1 \leqslant +p_{09}(f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, X^9_{397221}))))\ \cdot$
$p_{04}(f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, X^{15}_{397221})))))))\ -$
$p_{09}(f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, X^9_{397221}))))\ \cdot$
$p_{04}(f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, X^8_{397221})))))\ +$
$p_{08}(f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, X^{14}_{397221}))))))))))\cdot$
$p_{03}(f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, X^{10}_{397221})))))\ -$
$p_{08}(f(X^1_{397221}, f(X^1_{397221}, f(X^1_{397221}, X^5_{397221}))))\ \cdot$

$p_{03}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{7}_{397221}))))))) +$
$p_{07}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{13}_{397221})))))) \cdot$
$p_{01}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{12}_{397221}))))))))-$
$p_{07}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{6}_{397221})))) \cdot$
$p_{01}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{3}_{397221})))))+$
$p_{06}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{16}_{397221}))))) \cdot$
$p_{02}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{11}_{397221})))))))))-$
$p_{06}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{2}_{397221}))))) \cdot$
$p_{02}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{4}_{397221}))))))-$
$p_{04}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{15}_{397221}))))))) \cdot$
$p_{01}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{12}_{397221}))))))))+$
$p_{04}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{8}_{397221}))))) \cdot$
$p_{01}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{3}_{397221}))))) -$
$p_{03}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{10}_{397221}))))) +$
$p_{03}(f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},f(X^{1}_{397221},X^{7}_{397221}))))))).$
Sum/Pos. Substitution $= \{X^{1}_{340481} \to X^{1}_{5250}, X^{7}_{340481} \to X^{9}_{5250}, X^{9}_{340481} \to X^{10}_{5250}\}$. Premises:
`340481, 5250`.

397225. $1 \leqslant$
$+p_{08}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{12}_{397225})))))))))) \cdot$
$p_{03}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{8}_{397225})))))-$
$p_{08}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{5}_{397225})))) \cdot$
$p_{03}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{7}_{397225})))))))+$
$p_{07}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{11}_{397225})))))) \cdot$
$p_{01}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{10}_{397225}))))))))-$
$p_{07}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{6}_{397225})))) \cdot$
$p_{01}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{3}_{397225}))))) +$
$p_{06}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{14}_{397225}))))) \cdot$
$p_{02}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{9}_{397225})))))))))-$
$p_{06}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{2}_{397225}))))) \cdot$
$p_{02}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{4}_{397225}))))))-$
$p_{04}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{13}_{397225}))))))) \cdot$
$p_{01}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{10}_{397225}))))))))+$
$p_{04}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{13}_{397225}))))))) \cdot$
$p_{01}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{3}_{397225})))))-$
$p_{03}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{8}_{397225}))))) +$
$p_{03}(f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},f(X^{1}_{397225},X^{7}_{397225}))))))).$
Factor/Add. Substitution $= \{X^{10}_{397221} \to f(X^{1}_{397221}, f(X^{1}_{397221}, X^{7}_{397221}))\}$. Premises:
`397221`.

397330. $1 \leqslant +p_{07}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{8}_{397330})))))) \cdot$
$p_{01}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{7}_{397330}))))))))-$
$p_{07}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{5}_{397330})))) \cdot$
$p_{01}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{3}_{397330})))))+$
$p_{06}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{10}_{397330})))))\cdot$
$p_{02}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{6}_{397330})))))))))-$
$p_{06}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{2}_{397330})))))\cdot$
$p_{02}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{4}_{397330}))))))-$
$p_{04}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{9}_{397330})))))))\cdot$
$p_{01}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{7}_{397330}))))))))+$
$p_{04}(f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},f(X^{1}_{397330},X^{9}_{397330})))))))\cdot$

$p_{01}(f(X^1_{397330}, f(X^1_{397330}, f(X^1_{397330}, f(X^1_{397330}, X^3_{397330}))))).$
Factor/Add. Substitution = $\{X^8_{397225} \to f(X^1_{397225}, f(X^1_{397225}, X^7_{397225}))\}$. Premises: 397225.

398096. $1 \leqslant +p_{06}(f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, X^5_{398096}))))) \cdot$
$p_{02}(f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, X^4_{398096}))))))))) -$
$p_{06}(f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, X^2_{398096}))))) \cdot$
$p_{02}(f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, X^3_{398096})))))).$
Factor/Add. Substitution = $\{X^3_{397330} \to f(X^1_{397330}, f(X^1_{397330}, f(X^1_{397330}, X^7_{397330})))\}$.
Premises: 397330.

398974. $1 \leqslant +0.$
Factor/Add. Substitution =
$\{X^3_{398096} \to f(X^1_{398096}, f(X^1_{398096}, f(X^1_{398096}, X^4_{398096}))), X^5_{398096} \to X^2_{398096}\}$. Premises: 398096.

Total of 92 lines.

6.7 Random formulas

To make the picture more diverse, we present a proof that our prover has found for a propositional formula generated at random. We generated this formula using a slightly modified version of `hgen3` generator used at SAT Competition 2002 (this generator typically generates harder formulas than the generator that takes clauses at random independently; the reason is that it tries to avoid the variables to be too dependent on each other and/or to occur too rarely/frequently).

Command line.

```
./bin/saprover -pon -ftex -L -b4 --samedegreesums -Dall -w -O5 -S5
tests/propositional/hgen3lazy/hgen3-14-14.tptp
```

Proof.

3. $-1 \leqslant -p_{14} - p_{12} + p_{05}.$
Initial.

4. $-1 \leqslant -p_{10} + p_{04} - p_{01}.$
Initial.

6. $0 \leqslant +p_{11} - p_{08} + p_{06}.$
Initial.

7. $-1 \leqslant +p_{07} - p_{04} - p_{03}.$
Initial.

8. $0 \leqslant -p_{07} + p_{06} + p_{01}.$
Initial.

9. $0 \leqslant +p_{07} - p_{03} + p_{01}.$
Initial.

10. $1 \leqslant +p_{11} + p_{06} + p_{03}$.
    Initial.

11. $-1 \leqslant +p_{13} - p_{10} - p_{08}$.
    Initial.

12. $0 \leqslant -p_{14} + p_{06} + p_{04}$.
    Initial.

13. $0 \leqslant +p_{14} + p_{05} - p_{02}$.
    Initial.

14. $1 \leqslant +p_{11} + p_{09} + p_{03}$.
    Initial.

15. $-2 \leqslant -p_{13} - p_{11} - p_{07}$.
    Initial.

16. $0 \leqslant -p_{14} + p_{07} + p_{06}$.
    Initial.

18. $0 \leqslant +p_{12} + p_{09} - p_{04}$.
    Initial.

20. $0 \leqslant -p_{09} + p_{08} + p_{02}$.
    Initial.

23. $-1 \leqslant -p_{06} - p_{05} + p_{02}$.
    Initial.

24. $-1 \leqslant +p_{14} - p_{11} - p_{01}$.
    Initial.

25. $0 \leqslant -p_{14} + p_{05} + p_{01}$.
    Initial.

26. $0 \leqslant +p_{11} - p_{09} + p_{02}$.
    Initial.

27. $0 \leqslant +p_{12} + p_{09} - p_{03}$.
    Initial.

29. $-2 \leqslant -p_{13} - p_{09} - p_{08}$.
    Initial.

30. $0 \leqslant +p_{12} + p_{05} - p_{03}$.
    Initial.

31. $-1 \leqslant -p_{12} + p_{09} - p_{05}$.
    Initial.

32. $-1 \leqslant -p_{12} - p_{10} + p_{01}$.
    Initial.

34. $0 \leqslant +p_{10} - p_{06} + p_{04}$.
    Initial.

35. $0 \leqslant -p_{05} + p_{03} + p_{01}$.
Initial.

37. $-2 \leqslant -p_{12} - p_{07} - p_{03}$.
Initial.

38. $-1 \leqslant -p_{14} + p_{08} - p_{02}$.
Initial.

39. $1 \leqslant +p_{14} + p_{09} + p_{04}$.
Initial.

42. $-1 \leqslant -p_{11} + p_{10} - p_{08}$.
Initial.

44. $-1 \leqslant +p_{10} - p_{08} - p_{04}$.
Initial.

45. $-1 \leqslant -p_{11} - p_{05} + p_{02}$.
Initial.

46. $0 \leqslant +p_{14} + p_{08} - p_{02}$.
Initial.

47. $-1 \leqslant +p_{12} - p_{06} - p_{02}$.
Initial.

48. $1 \leqslant +p_{11} + p_{10} + p_{04}$.
Initial.

49. $-1 \leqslant -p_{12} + p_{10} - p_{04}$.
Initial.

70. $0 \leqslant +p_{06} - p_{03} + p_{01}$.
Sum/Pos. Premises: `8, 9`.

95. $-1 \leqslant -p_{06} - p_{01} + p_{04}$.
Sum/Pos. Premises: `34, 4`.

147. $-1 \leqslant -p_{09} - p_{05} + p_{02}$.
Sum/Pos. Premises: `26, 45`.

216. $0 \leqslant +p_{10} + p_{06} - p_{08}$.
Sum/Pos. Premises: `42, 6`.

218. $-1 \leqslant +p_{10} - p_{09} - p_{08} + p_{02}$.
Sum/Pos. Premises: `42, 26`.

222. $0 \leqslant -p_{08} + p_{04} + p_{10}$.
Sum/Pos. Premises: `42, 48`.

257. $-1 \leqslant -p_{07} + p_{05} - p_{03}$.
Sum/Pos. Premises: `37, 30`.

302. $-1 \leqslant -p_{05} - p_{03} + p_{09}$.
Sum/Pos. Premises: `31, 27`.

304. $-1 \leqslant -p_{05} - p_{04} + p_{09}$.
Sum/Pos. Premises: `31, 18`.

344. $-2 \leqslant -p_{10} - p_{06} - p_{02} + p_{01}$.
Sum/Pos. Premises: `32, 47`.

350. $-1 \leqslant -p_{10} + p_{09} - p_{04} + p_{01}$.
Sum/Pos. Premises: `32, 18`.

389. $0 \leqslant +p_{10} + p_{09} - p_{04}$.
Sum/Pos. Premises: `49, 18`.

537. $-2 \leqslant -p_{10} - p_{09} - p_{08}$.
Sum/Pos. Premises: `11, 29`.

601. $-3 \leqslant -p_{11} - p_{10} - p_{08} - p_{07}$.
Sum/Pos. Premises: `15, 11`.

647. $0 \leqslant +p_{07} + p_{06} + p_{05} - p_{02}$.
Sum/Pos. Premises: `16, 13`.

685. $0 \leqslant +p_{08} - p_{02}$.
Sum/Pos. Premises: `46, 38`.

822. $1 \leqslant +p_{09} + p_{06} + p_{04}$.
Sum/Pos. Premises: `39, 12`.

877. $-1 \leqslant -p_{12} - p_{02} + p_{05}$.
Sum/Pos. Premises: `3, 13`.

992. $-1 \leqslant -p_{05} - p_{03} + p_{02} + p_{01}$.
Sum/Pos. Premises: `70, 23`.

1008. $0 \leqslant +p_{05} + p_{01} - p_{03}$.
Sum/Pos. Premises: `257, 9`.

1010. $-1 \leqslant +p_{05} - p_{04} - p_{03}$.
Sum/Pos. Premises: `257, 7`.

1037. $0 \leqslant +p_{10} + p_{04} - p_{02}$.
Sum/Pos. Premises: `222, 685`.

1136. $-2 \leqslant -p_{09} - p_{04} - p_{08}$.
Sum/Pos. Premises: `537, 44`.

1138. $-1 \leqslant -p_{09} + p_{06} - p_{08}$.
Sum/Pos. Premises: `537, 216`.

1235. $-1 \leqslant +p_{09} - p_{04} - p_{03}$.
Sum/Pos. Premises: `1010, 302`.

1276. $1 \leqslant +p_{08} + p_{06} + p_{04} + p_{02}$.
Sum/Pos. Premises: `822, 20`.

1309. $0 \leqslant -p_{08} + p_{04} + p_{06}$.
Sum/Pos. Premises: `1138, 822`.

1330. $-1 \leqslant -p_{02} - p_{01} + p_{04}$.
Sum/Pos. Premises: `1037, 4`.

1367. $-1 \leqslant -p_{06} + p_{05} - p_{02}$.
Sum/Pos. Premises: `877, 47`.

1462. $0 \leqslant +p_{06} + p_{04} - p_{02}$.
Sum/Pos. Premises: `1309, 685`.

1517. $-2 \leqslant -p_{05} - p_{04} - p_{03} + p_{02}$.
Sum/Pos. Premises: `1235, 147`.

1564. $0 \leqslant +p_{05} + p_{04} - p_{02}$.
Sum/Pos. Premises: `1462, 1367`.

1614. $0 \leqslant +p_{04} + p_{03} - p_{02} + p_{01}$.
Sum/Pos. Premises: `1564, 35`.

2055. $-1 \leqslant +p_{04} - p_{02} + p_{01} - p_{06}$.
Sum/Pos. Premises: `344, 34`.

2367. $0 \leqslant +p_{01} + p_{09} - p_{04}$.
Sum/Pos. Premises: `350, 389`.

2431. $0 \leqslant +p_{08} - p_{04} + p_{02} + p_{01}$.
Sum/Pos. Premises: `2367, 20`.

2437. $-1 \leqslant -p_{08} + p_{01} - p_{04}$.
Sum/Pos. Premises: `2367, 1136`.

2489. $-1 \leqslant -p_{04} - p_{02} + p_{01}$.
Sum/Pos. Premises: `2437, 685`.

2696. $-1 \leqslant +p_{02} - p_{09} - p_{08}$.
Sum/Pos. Premises: `218, 537`.

3593. $-2 \leqslant -p_{10} - p_{07} + p_{06} - p_{08}$.
Sum/Pos. Premises: `601, 6`.

3691. $0 \leqslant +p_{02} - p_{03} + p_{01}$.
Sum/Pos. Premises: `992, 1008`.

4718. $1 \leqslant +p_{02} + p_{06} + p_{04}$.
Sum/Pos. Premises: `1276, 1309`.

4770. $0 \leqslant +p_{02} - p_{01} + p_{04}$.
Sum/Pos. Premises: `4718, 95`.

4821. $0 \leqslant +p_{04} - p_{01}$.
Sum/Pos. Premises: `4770, 1330`.

4848. $+p_{04} \cdot p_{01} = +p_{01}$.
SumToProduct. Premises: `4821`.

5663. $-1 \leqslant +p_{02} - p_{04} - p_{03}$.
Sum/Pos. Premises: `1517, 1010`.

5730. $-1 \leqslant -p_{03} + p_{02} - p_{01}$.
Sum/Pos. Premises: `5663, 4821`.

5914. $0 \leqslant +p_{03} - p_{02} + p_{01}$.
Sum/Pos. Premises: `1614, 2489`.

5967. $0 \leqslant -p_{02} + p_{01} + 2p_{03} \cdot p_{02} - 2p_{03} \cdot p_{01}$.
LiftAndProject. Premises: `5914, 5730`.

6218. $-1 \leqslant -p_{06} - p_{02} + p_{01}$.
Sum/Pos. Premises: `2055, 2489`.

6253. $-1 \leqslant -p_{03} - p_{02} + p_{01}$.
Sum/Pos. Premises: `6218, 70`.

6296. $0 \leqslant -p_{02} + p_{01}$.
Sum/Pos. Premises: `6253, 5914`.

6317. $+p_{02} \cdot p_{01} = +p_{02}$.
SumToProduct. Premises: `6296`.

6691. $0 \leqslant -p_{03} + p_{01}$.
Sum/Pos. Premises: `6296, 3691`.

6741. $+p_{03} \cdot p_{01} = +p_{03}$.
SumToProduct. Premises: `6691`.

6777. $0 \leqslant -p_{02} + p_{01} + 2p_{03} \cdot p_{02} - 2p_{03}$.
Rewrite. Premises: `5967, 6741`.

8567. $0 \leqslant +p_{02} - p_{04} + p_{01}$.
Sum/Pos. Premises: `2431, 2437`.

12507. $-1 \leqslant -p_{07} - p_{08} + p_{06}$.
Sum/Pos. Premises: `3593, 216`.

12677. $-1 \leqslant -p_{07} + p_{06} - p_{02}$.
Sum/Pos. Premises: `12507, 685`.

12789. $0 \leqslant +p_{05} + p_{06} - p_{02}$.
Sum/Pos. Premises: `12677, 647`.

12880. $0 \leqslant +p_{05} - p_{02}$.
Sum/Pos. Premises: `12789, 1367`.

13486. $-1 \leqslant +p_{09} - p_{04} - p_{02}$.
Sum/Pos. Premises: `12880, 304`.

13814. $-2 \leqslant -p_{08} - p_{02} - p_{04}$.
Sum/Pos. Premises: `13486, 1136`.

14052. $-1 \leqslant -p_{04} - p_{02}$.
Sum/Pos. Premises: `13814, 685`.

14113. $+p_{04} \cdot p_{02} = +0$.
SumToProduct. Premises: `14052`.

14526. $0 \leqslant +p_{04} \cdot p_{01} - p_{04}$.
MultiplyRewrite/Pos. Monomial $= +p_{04}$. Premises: `8567, 14113`.

14608. $-1 \leqslant -p_{02} - p_{01}$.
Sum/Pos. Premises: `14052, 4821`.

14686. $0 \leqslant -p_{04} + p_{01}$.
Rewrite. Premises: `14526, 4848`.

14753. $+p_{02} \cdot p_{01} = +0$.
SumToProduct. Premises: `14608`.

14788. $+p_{02} = +0$.
EqSimplified. Premises: `14753, 6317`.

14858. $0 \leqslant +p_{01} + 2p_{03} \cdot p_{02} - 2p_{03}$.
Rewrite. Premises: `6777, 14788`.

14870. $-1 \leqslant -p_{09} - p_{08}$.
Rewrite. Premises: `2696, 14788`.

14911. $-1 \leqslant -p_{06} - p_{05}$.
Rewrite. Premises: `23, 14788`.

14980. $0 \leqslant -p_{09} + p_{08}$.
Rewrite. Premises: `20, 14788`.

15024. $-1 \leqslant -p_{11} - p_{05}$.
Rewrite. Premises: `45, 14788`.

15044. $0 \leqslant +p_{11} - p_{09}$.
Rewrite. Premises: `26, 14788`.

15263. $0 \leqslant +p_{01} - 2p_{03}$.
Rewrite. Monomial $= +p_{03}$. Premises: `14858, 14788`.

15436. $0 \leqslant -p_{03}$.
MultiplyRewrite/Pos. Monomial $= +p_{03}$. Premises: `15263, 6741`.

15484. $+p_{03} = +0$.
Opposite. Premises: `15436`.

15562. $1 \leqslant +p_{11} + p_{06}$.
Rewrite. Premises: `10, 15484`.

15569. $1 \leqslant +p_{11} + p_{09}$.
Rewrite. Premises: `14, 15484`.

16220. $+p_{09} \cdot p_{08} = +p_{09}$.
SumToProduct. Premises: `14980`.

16993. $+p_{11} \cdot p_{05} = +0$.
SumToProduct. Premises: `15024`.

17544. $+p_{11} \cdot p_{06} = +p_{11} + p_{06} - 1$.
SumToProduct. Premises: `15562`.

17695. $0 \leqslant +p_{11} \cdot p_{05} - p_{05}$.
MultiplyRewrite/Neg. Monomial $= +p_{11}$. Premises: `14911, 17544`.

17969. $0 \leqslant -p_{05}$.
Rewrite. Premises: `17695, 16993`.

18064. $+p_{05} = +0$.
Opposite. Premises: `17969`.

18174. $0 \leqslant -p_{14} + p_{01}$.
Rewrite. Premises: `25, 18064`.

18178. $-1 \leqslant -p_{14} - p_{12}$.
Rewrite. Premises: `3, 18064`.

19113. $+p_{11} \cdot p_{09} = +p_{09}$.
SumToProduct. Premises: `15044`.

19755. $+p_{11} \cdot p_{09} = +p_{11} + p_{09} - 1$.
SumToProduct. Premises: `15569`.

19806. $+p_{11} = +1$.
EqSimplified. Premises: `19755, 19113`.

19913. $0 \leqslant +p_{14} - p_{01}$.
Rewrite. Premises: `24, 19806`.

22302. $+p_{14} = +p_{01}$.
Opposite. Premises: `18174, 19913`.

22461. $1 \leqslant +p_{09} + p_{04} + p_{01}$.
Rewrite. Premises: `39, 22302`.

22623. $-1 \leqslant -p_{12} - p_{01}$.
Rewrite. Premises: `18178, 22302`.

22652. $+p_{12} \cdot p_{01} = +0$.
SumToProduct. Premises: `22623`.

22973. $+p_{09} \cdot p_{08} = +0$.
SumToProduct. Premises: `14870`.

22990. $+p_{09} = +0$.
EqSimplified. Premises: `22973, 16220`.

23130. $0 \leqslant +p_{12} - p_{04}$.
Rewrite. Premises: `18, 22990`.

23286. $+p_{12} \cdot p_{04} = +p_{04}$.
SumToProduct. Premises: `23130`.

23487. $0 \leqslant +p_{12} \cdot p_{01} - p_{01}$.
MultiplyRewrite/Neg. Monomial $= +p_{12}$. Premises: `4821, 23286`.

23679. $0 \leqslant -p_{01}$.
Rewrite. Premises: `23487, 22652`.

23770. $+p_{01} = +0$.
Opposite. Premises: `23679`.

26208. $0 \leqslant -p_{04}$.
Rewrite. Premises: `14686, 23770`.

26239. $+p_{04} = +0$.
Opposite. Premises: `26208`.

26920. $1 \leqslant +p_{04} + p_{01}$.
Rewrite. Premises: `22461, 22990`.

26923. $1 \leqslant +p_{04}$.
Rewrite. Premises: `26920, 23770`.

26930. $1 \leqslant +0$.
Rewrite. Premises: `26923, 26239`.

Total of 145 lines.

Bibliography

[BG01] L. Bachmair and H. Ganzinger. Resolution theorem proving. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, chapter 2, pages 19–99. Elsevier, 2001.

[Bor99] B. Borchers. CSDP, a C library for semidefinite programming. *Optimization Methods and Software*, 11-2(1-4):613–623, 1999.

[CCT87] W. Cook, C. R. Coullard, and G. Turán. On the complexity of cutting-plane proofs. *Discrete Applied Mathematics*, 18(1):25–38, 1987.

[Der82] N. Dershowitz. Orderings for term-rewriting systems. *Theoretical computer science*, 17(3):279–307, 1982.

[GHP02] Dima Grigoriev, Edward A. Hirsch, and Dmitrii V. Pasechnik. Complexity of semi-algebraic proofs. *Moscow Mathematical Journal*, 2(4):647–679, 2002. `http://www.ams.org/distribution/mmj/`.

[Gom63] R. E. Gomory. An algorithm for integer solutions of linear programs. In R. L. Graves and P. Wolfe, editors, *Recent Advances in Mathematical Programming*, pages 269–302. McGraw-Hill, 1963.

[Hak85] A. Haken. The intractability of resolution. *TCS*, 39:297–308, 1985.

[Kal01] J. A. Kalman. *Automated Reasoning with Otter*. Rinton Press, 2001.

[KB70] D. Knuth and P. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational problems in abstract algebra*, pages 263–297. Pergamon Press, Oxford, 1970.

[KL80] S. Kamin and J.-J. Levy. Two generalizations of the recursive path ordering. Unpublished note, Dept. of Computer Science, Univ. of Illinois, Urbana, IL, 1980.

[LS91] L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM J. Optimization*, 1(2):166–190, 1991.

[Par03] P.A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming Ser. B*, 96(2):293–320, 2003.

[Pud99] Pavel Pudlák. On the complexity of the propositional calculus. In *Sets and Proofs: Invited papers from Logic Colloquium'97*, pages 197–218. Cambridge University Press, 1999.

[RV01] A. Riazanov and A. Voronkov. Vampire 1.1 (System description). In Rajeev Goré, Alexander Leitsch, and Tobias Nipkow, editors, *Proc. Intl. Joint Conf. on Automated Reasoning, Siena, Italy*, number 2083 in LNCS. Springer-Verlag, 2001.

[Sch02] S. Schulz. E – A Brainiac Theorem Prover. *Journal of AI Communications*, 15(2/3):111–126, 2002.

[SS03] G. Sutcliffe and C. B. Suttner. The CADE-19 ATP System Competition. Submitted, 2003.

[Tse68] G. S. Tseitin. On the complexity of derivation in the propositional calculus. *Zapiski nauchnykh seminarov LOMI*, 8:234–259, 1968. English translation of this volume: Consultants Bureau, N.Y., 1970, pp. 115–125.