# Trichotomy in the Complexity of Minimal Inference


Arnaud Durand
ELM (CNRS UMR 7056)
Univ. Denis-Diderot Paris 7
75251 Paris cedex 05, France
durand@logique.jussieu.fr

Miki Hermann
LIX (CNRS UMR 7161)
École Polytechnique
91128 Palaiseau, France
hermann@lix.polytechnique.fr

Gustav Nordh
Dept. of Comp. and Inf. Sciences
Linköpings Universitet
581 83 Linköping, Sweden
gusno@ida.liu.se



***Abstract*—We study the complexity of the propositional minimal inference problem. Its complexity has been extensively studied before because of its fundamental importance in artificial intelligence and nonmonotonic logics. We prove that the complexity of the minimal inference problem with unbounded queries has a trichotomy (between $\mathrm{P}$, $\mathrm{coNP}$-complete, and $\Pi_2\mathrm{P}$-complete). This result finally settles with a positive answer the trichotomy conjecture of Kirousis and Kolaitis [A dichotomy in the complexity of propositional circumscription, LICS'01] in the unbounded case. We also present simple and efficiently computable criteria separating the different cases.**


## I. Introduction and Summary of Results

Reasoning with minimal models of a theory is a general idea widely used in artificial intelligence, especially for capturing various aspects of common sense and nonmonotonic reasoning. In particular, it is the main idea behind circumscription [16], [17], diagnosis [7], default logic [22], and logic programming under stable model semantics [10]. Minimal inference has been shown by Gelfond *et al.* [11] to coincide with reasoning under the extended closed world assumption, which is one of the main formalisms for reasoning with incomplete information. We focus in this paper on the important basic case where the theory is propositional, which is also the most investigated case from a complexity point of view. In this case, the minimality of models is defined with respect to the *pointwise partial order*, extending the order $0 < 1$ on truth values.

The complexity of several basic algorithmic problems has been studied in connection with minimal models of propositional formulas: among them are the *model selection* [5], [20], *model checking* [3], [14], and *inference* [4], [8], [9], [15] problems. Given a propositional formula $\varphi$, the minimal model selection problem requires to compute a minimal model of $\varphi$. Similarly, given a propositional formula $\varphi$ and a truth assignment $m$, the minimal model checking problem asks whether $m$ is a minimal model of $\varphi$. Given two propositional formulas $\varphi$ and $\psi$, the minimal inference problem asks whether $\psi$ (also called the query) is true in every minimal model of $\varphi$. In propositional theories these problems are identical to the corresponding problems for propositional circumscription. Hence, they are often called the model selection, model checking, and inference problems for propositional circumscription, respectively.

The topic of this paper is to classify the complexity of the minimal inference problem for all restrictions on the types of clauses allowed in the theory $\varphi$, presented by a formula in conjunctive normal form. Note that the corresponding classification problem for satisfiability was solved in a seminal paper by Schaefer [24].

The inference problem was proved $\Pi_2\mathrm{P}$-complete by Eiter and Gottlob in [9] if no restrictions are imposed on the propositional theory $\varphi$. Cadoli and Lenzerini proved in [4] that the inference problem becomes $\mathrm{coNP}$-complete if $\varphi$ is a bijunctive or a dual Horn formula. Durand and Hermann proved in [8] that also the inference problem for affine formulas $\varphi$ is $\mathrm{coNP}$-complete. Finally, Kirousis and Kolaitis [15] proved a dichotomy theorem separating the $\Pi_2\mathrm{P}$-complete cases from the cases in $\mathrm{coNP}$. Moreover, they conjectured that the cases in $\mathrm{coNP}$ could be separated into $\mathrm{coNP}$-complete and polynomial-time decidable cases.

The proof of the dichotomy for minimal inference by Kirousis and Kolaitis [15] turned out to be difficult for several reasons. One of them was the impossibility to apply the well-known approach through the theory of clones and Post's lattice [2], [12], [21] to obtain a complexity classification. The culprit is the existential quantification which does not combine well with minimality. This implies that the co-clones, which are the sets of relations closed under variable identification, variable permutation, conjunction, and existential quantification, are not invariant with respect to the complexity of minimal inference. This situation precludes the use of the Galois correspondence, which allows first to prove a completeness result for a subset of relations and subsequently to extend it to the whole co-clone.

However, if the pointwise partial order on models is relaxed, which is the consequence of studying generalized forms of propositional circumscription, then it has been observed that the Galois correspondence can be restored [19] and the approach via clone theory works. With the help of these powerful tools it is relatively easy to classify the complexity of these generalized forms of minimal inference for every restriction on the types of clauses in the theory [18].

Following the results of Kirousis and Kolaitis [15], it is clear


This research was partially supported by the grant ANR-07-BLAN-0327.
Gustav Nordh was partially supported by the Swedish Research Council (VR) under grant 2008-4675 and the Swedish French foundation. This work was finished during his post-doctoral scientific stay in the LIX laboratory of Ecole Polytechnique.

that we cannot use the algebraic approach for attacking the complexity of the minimal inference problem. Our approach is instead based on refinements of Schaefer's approach to classify the complexity of the satisfiability problem in propositional logic [6], [24]. The main difficulty of Schaefer's dichotomy proof for the satisfiability problem was not to identify the tractable classes. These where, in his own words, either trivial (0-valid, and 1-valid) or well-known (Horn, dual-Horn, bijunctive, and affine). The difficulty resided in proving that all other cases were hard.

The situation for the minimal inference problem is similar. We identify only one new additional significant tractable class, namely, when $(\neg x \vee \neg y)$ and $(x \not\equiv y)$ are the types of clauses allowed in the theory. The main difficulty is to prove that every other set of allowed clauses (which is not known to be tractable) is coNP-hard. We deal with this challenge in the same way as Schaefer. We prove that every such set of allowed clauses expresses (implements) at least one special case already proved to be coNP-hard. We prove a number of new coNP-hard cases for the minimal inference problem and use them together with some already known hard cases as our targets. There is a significant difference though between Schaefer's implementations and ours. We can only use conjunction, variable identification, and variable permutation (*without* existential quantifiers) in our implementations. This is due to the fact, as already explained, that existential quantification does not preserve the minimal models, nor the complexity of the problem. Nevertheless, we manage to separate the coNP-hard cases from the tractable ones by using this approach through a more fine-grained analysis, thus, finishing the complexity classification of minimal inference with restrictions on the types of allowed clauses. Note that we address the minimal inference problem for unbounded queries. Some remarks on the bounded case can be found in the conclusion.

This result finally settles with a positive answer the trichotomy conjecture of Kirousis and Kolaitis [15]. We also present simple and efficiently computable criteria separating the different cases. In the process we also strengthen and give a much simplified proof of the coNP-completeness result by Durand and Hermann [8]. We also believe that the implementations we present might be interesting in their own right, since they could be useful during the study of other problems, which are not invariant under existential quantification.

## II. Preliminaries

Throughout the paper we use the standard correspondence between constraints and relations. We use the same symbol for a constraint and its corresponding relation, since the meaning will always be clear from the context, and we say that the constraint *represents* the relation.

An $n$-ary *logical relation* $R$ is a Boolean relation of arity $n$. Each element of a logical relation $R$ is an $n$-ary Boolean vector (also called a tuple) $m = (m_1, \ldots, m_n) \in \{0,1\}^n$. To save space, we will often write the vector $(m_1, \ldots, m_n)$ in the form $m_1 \cdots m_n$. Let $V$ be a set of variables. A *constraint* is an application of $R$ to an $n$-tuple of variables from $V$, i.e., $R(x_1, \ldots, x_n)$.

Consider a relation $R$ represented by a Boolean matrix, i.e., the vectors of $R$ constitute the rows of the matrix. We say that a relation $R$ is *irredundant* if it does not contain two identical columns and it cannot be transformed by column permutation to a relation of the form $Q \times \{0,1\}^k$ for a $k \geq 1$, where $Q$ is another relation. If $R$ is redundant then the corresponding *irredundant reduction* $R^\circ$ is formed by identifying identical columns and removing all columns of the form $\{0,1\}^k$. A set of relations $S$ is irredundant if every relation in $S$ is irredundant. Given a set of relations $S$, we form the irredundant reduction $S^\circ$ by replacing all redundant relations $R$ in $S$ by their corresponding irredundant reductions $R^\circ$. We prove, in Proposition 5, that the minimal inference problems of $S$ and $S^\circ$ are equivalent under polynomial many-one reductions. Hence, we assume throughout the paper that the set of relations $S$ is irredundant.

An *assignment* is a mapping $m\colon V \to \{0,1\}$ assigning a Boolean value $m(x)$ to each variable $x \in V$. If we arrange the variables in some arbitrary but fixed order, say as a vector $(x_1, \ldots, x_n)$, then the assignments can be identified with vectors from $\{0,1\}^n$. The $i$-th component of a vector $m$, denoted by $m[i]$, corresponds to the value of the $i$-th variable, i.e., $m(x_i) = m[i]$. An assignment $m$ satisfies the constraint $R(x_1, \ldots, x_n)$ if $(m(x_1), \ldots, m(x_n)) \in R$ holds. An assignment $m$ satisfying a constraint $R(x_1, \ldots, x_n)$ is called a *model* of $R(x_1, \ldots, x_n)$.

Throughout the text we refer to different types of Boolean constraint relations following Schaefer's terminology [24] (see also the monograph [6]). A Boolean relation $R$ is

- 1-*valid* if $1 \cdots 1 \in R$ and it is 0-*valid* if $0 \cdots 0 \in R$,
- *Horn* (*dual Horn*) if $R$ can be represented by a conjunctive normal form (CNF) formula having at most one unnegated (negated) variable in each clause,
- *bijunctive* if it can be represented by a CNF formula having at most two variables in each clause,
- *affine* if it can be represented by an affine system of equations $S\colon Ax = b$ over $\mathbb{Z}_2$,
- *complementive* if for each $(m_1, \ldots, m_n) \in R$ also $(\neg m_1, \ldots, \neg m_n) \in R$.

A set $S$ of Boolean relations is called 0-valid (1-valid, Horn, dual Horn, affine, bijunctive, complementive) if *every* relation in $S$ is 0-valid (1-valid, Horn, dual Horn, affine, bijunctive, complementive). A relation $R$ (a set of relations $S$) is called *Schaefer* if it belongs to one of the classes Horn, dual Horn, bijunctive, or affine. In the sequel we need to determine the properties of relations. This will be done by means of closure operations (see [2] for a survey). We say that a relation $R$ is *closed* under a Boolean function $f\colon \{0,1\}^k \to \{0,1\}$, or that $f$ is a *polymorphism* of $R$, if for any choice of $k$ vectors $m_1, \ldots, m_k \in R$, we have that

$$\Big(f\big(m_1[1], \ldots, m_k[1]\big),\ \ldots,\ f\big(m_1[n], \ldots, m_k[n]\big)\Big) \in R,$$

i.e., that the new vector constructed coordinate-wise from

| | | |
|---|---|---|
| $R$ is Horn | $\Leftrightarrow$ | $m, m' \in R$ implies $m \wedge m' \in R$ |
| $R$ is dual Horn | $\Leftrightarrow$ | $m, m' \in R$ implies $m \vee m' \in R$ |
| $R$ is bijunctive | $\Leftrightarrow$ | $m, m', m'' \in R$ implies $\mathrm{maj}(m, m', m'') \in R$ |
| $R$ is affine | $\Leftrightarrow$ | $m, m', m'' \in R$ implies $m + m' + m'' \in R$ |
| $R$ is complementive | $\Leftrightarrow$ | $m \in R$ implies $\neg m \in R$ |

TABLE I
CORRESPONDENCE WITH CLOSURE OPERATIONS

$m_1, \ldots, m_k$ by means of $f$ belongs to $R$. In particular, we study the closure under the Boolean functions called conjunction ($\wedge$), disjunction ($\vee$), majority (maj), addition over $\mathbb{Z}_2$ called exclusive-or ($+$), and negation ($\neg$), where the majority operation is defined by the identity $\mathrm{maj}(m, m', m'') = (m \vee m') \wedge (m' \vee m'') \wedge (m'' \vee m)$. The correspondence between the type of relation and the closure operation is specified in Table I.

Let $S$ be a non-empty finite set of Boolean relations, also called a *constraint language*. An *S-formula* is a finite conjunction of clauses $\varphi = c_1 \wedge \cdots \wedge c_k$, where each clause $c_i$ is a constraint application of some logical relation $R \in S$. An assignment $m$ satisfies the formula $\varphi$ if it satisfies all clauses $c_i$. Hence the notion of models naturally extends from constraints to formulas. We denote by $[\varphi]$ the set of models of a formula $\varphi$. It is clear that each $[\varphi]$ denotes a Boolean relation. We also denote by $\langle S \rangle$ the set of all relations that can be expressed using relations from $S$, conjunction (Cartesian product), and variable identification. Notice that we do *not* require closure under existential quantification. This means that

- if $R \in S$ then $R(\vec{x}) \in \langle S \rangle$,
- if $R_1(\vec{x}), R_2(\vec{y}) \in \langle S \rangle$ then $(R_1 \times R_2)(\vec{x}, \vec{y}) = R_1(\vec{x}) \wedge R_2(\vec{y}) \in \langle S \rangle$,
- if $R(x_1, x_2, \vec{y}) \in \langle S \rangle$ then $R(x, x, \vec{y}) \in \langle S \rangle$, where $x$ is a fresh variable.

Let $m = (m[1], \ldots, m[n])$ and $m' = (m'[1], \ldots, m'[n])$ be two Boolean vectors from $\{0,1\}^n$. We write $m \leq m'$ to denote that $m[i] \leq m'[i]$ holds for every $i \leq n$, as well as $m < m'$ for $m \leq m'$ and $m \neq m'$. The relation $\leq$ is called the *pointwise partial order* on models. Let $\varphi(x_1, \ldots, x_n)$ be a Boolean formula having $x_1, \ldots, x_n$ as its variables and let $m \in \{0,1\}^n$ be a truth assignment. We say that $m$ is a *minimal model* of $\varphi$ if $m$ is a model of $\varphi$ and there is no other model $m'$ of $\varphi$ that satisfies the relation $m' < m$. Two vectors $a, b \in R$ satisfying the relation $a < b$ are called *comparable*. We say that a relation $R$ is *incomparable* if it does not contain comparable vectors. We say that a set of relations $S$ is incomparable if each relation $R \in S$ is incomparable.

Let $\varphi$ and $\psi$ be two propositional formulas in conjunctive normal form. We say that the *query* $\psi$ follows from $\varphi$ in propositional circumscription, denoted by $\varphi \models_{\min} \psi$, if the query $\psi$ is true in every minimal model of $\varphi$. Since the query $\psi$ is a conjunction $c_1 \wedge \cdots \wedge c_k$ of clauses $c_i$, then $\varphi \models_{\min} \psi$ if and only if $\varphi \models_{\min} c_i$ for each $i$. Hence we can restrict ourselves to consider only a single query clause instead of a CNF query $\psi$ at the right-hand side of the propositional inference problem $\varphi \models_{\min} c$.

## III. EXPRESSIVITY TECHNIQUES

In this section we review two very useful techniques. The first is due to Schaefer [24], for showing that certain relations belong to $\langle S \rangle$ based on information about the polymorphisms of $S$. The second is due to Kirousis and Kolaitis [15]. We begin by stating some basic but useful facts about the set $\langle S \rangle$.

**Lemma 1** *If $S$ is not closed under the functions $f_1, \ldots, f_k$ then there exists a single relation $R \in \langle S \rangle$ which is not closed under any of the functions $f_1, \ldots, f_k$.*

**Lemma 2** *If a relation $R$ is not closed under the functions $f_1, \ldots, f_k$ of arities $a_1, \ldots, a_k$ where $a$ denotes the maximum arity of the functions, then there is a relation $R' \in \langle R \rangle$ containing tuples $t_1, \ldots t_a$, such that for any function $f_i$ and any $a_i$ tuples from $t_1, \ldots, t_a$, the tuple resulting from applying $f_i$ to these $a_i$ tuples (in any order), is not in $R'$.*

The notation $R[x/V]$ means that in the relation $R$ the variables $V$ are replaced with the new variable $x$. As a shorthand, we also write $R[V]$ instead of $R[x/V]$ when the new variable $x$ is implicitly clear or when $V$ represents both the set of variables to be replaced and the fresh variable. This means $V$ is a set of variables, but also the identifier of a new variable, by which all variables in $V$ are replaced in $R[V]$.

The following is the typical situation we are faced with when proving our hardness results. We have a set of relations $S$ for which we want to prove a hardness result. We also know a hardness result for a relation $R$. The goal is to show that $R \in \langle S \rangle$. For this purpose we have some information on $S$, namely we know that $S$ is closed under the operations $f_1, \ldots, f_j$, but it is *not* closed under the operations $g_1, \ldots, g_k$. The first step is to use Lemmas 1 and 2 to produce a single relation $R' \in \langle S \rangle$ that has the properties stated in Lemma 2 and is not closed under any of the operations $g_1, \ldots, g_k$.

The next (crucial) step is to prove that $R$ can be implemented by $R'$ using variable identification. We first show that $R'$ must contain a number of vectors $a_1, \ldots, a_p$, usually by the argument that $R'$ would otherwise be closed under one of the functions $g_1, \ldots, g_k$. For all Boolean vectors $m$ of length $p$ we construct the variable sets $V_m = \{x \mid a_1(x) = m[1], \ldots, a_p(x) = m[p]\}$. We then construct the constraint $Q(V_0, \ldots, V_{2^p-1}) = R'[V_i \mid i = 0, \ldots, 2^p - 1]$. For instance, when we know the existence of the vectors $a, b, c \in R'$, we create the variable sets $V_{ijk} = \{x \mid a(x) = i, b(x) =$

$j, c(x) = k\}$ and construct the constraint $Q(V_0, \ldots, V_7) = R'[V_{000}, \ldots, V_{111}]$.

Finally, we must show that the constructed constraint $R'[V_i \mid i = 0, \ldots, 2^p - 1]$, possibly after additional variable identification, really generates the relation $R$. For more information on this technique, see the monograph [6] or Schaefer's original exposition [24]. Note again that our implementations differ from Schaefer's since we are not allowed to use existential quantification.

We will also need a well-known concept from coding theory which was already used for circumscription [14] and minimal inference [15].

**Definition 3** Let $R \subseteq \{0,1\}^k$ be a $k$-ary Boolean relation. We say that a relation $R'$ is a **direct 0-section** of $R$ if there exists an index $i \in \{1, \ldots, k\}$, such that

$$R' = \{(m[1], \ldots, m[i-1], m[i+1], \ldots, m[n]) \mid m \in R \text{ and } m[i] = 0\}.$$

We say that a relation $R''$ is a **0-section** of $R$ if there exists a finite sequence of Boolean relations $R_0$, $R_1$, …, $R_n$, such that $R_0 = R$, $R'' = R_n$, and $R_{j+1}$ is a direct 0-section of $R_j$ for each $j = 0, \ldots, n-1$.

Let $S$ be a finite set of Boolean relations. We say that $S^*$ is a 1-valid **restriction** of $S$ if it contains all relations $R^*$ which are both 1-valid and a 0-section of a relation from $S$.

Note that starting from an arbitrary relation $R \in S$, we always arrive at a 1-valid 0-section $R^*$ by iterating the 0-section operation long enough, unless $R = \{0 \cdots 0\}$. It is easy to see that the property of $S$ being Schaefer implies the restriction $S^*$ to be Schaefer, since the classes of Horn, dual Horn, bijunctive, and affine constraints are stable by constant substitution. Note also that a restriction $S^*$ is 0-valid if and only if $S$ is 0-valid.

## IV. Complexity of Minimal Inference and Extension

We are interested in the complexity of the following problem, which is exactly the minimal inference problem studied for example in [4], [9], [15].

**Problem:** MININF$(S)$
*Input:* A conjunction $\varphi$ of constraints from $S$ and a clause $\psi$.
*Question:* Is $\psi$ satisfiable in every minimal model of $\varphi$, i.e., does $\varphi \models_{\min} \psi$ hold?

It was proved by Eiter and Gottlob [9] that MININF is $\Pi_2$P-complete. Cadoli and Lenzerini [4] showed that the MININF problem for dual-Horn or bijunctive formulas $\varphi$ is coNP-complete. Durand and Hermann [8] showed that the MININF problem for affine formulas $\varphi$ is coNP-complete. Kirousis and Kolaitis [15] showed that there exists a dichotomy between the $\Pi_2$P-complete general case of MININF and the special cases included in coNP. The MININF problem for Horn formulas $\varphi$ is trivially known to be polynomial-time decidable.

It is sometimes more convenient to investigate the dual problem MINEXT of generalized minimal extension, defined as follows.

**Problem:** MINEXT$(S)$
*Input:* A conjunction $\varphi(\vec{x}, \vec{y})$ of constraints from $S$, and a partial assignment $m$ for the variables $\vec{x}$.
*Question:* Can $m$ be extended to a minimal model $\bar{m}$ of $\varphi$?

The relationship between MININF and MINEXT can be easily established through the following construction. Let $S$ be a finite set of Boolean relations and $m$ an assignment to the variables $x_1$, …, $x_n$. Let $c_m$ be the largest clause falsified by the assignment $m$, i.e., $c_m = l_1 \vee \cdots \vee l_n$, where $l_i = x_i$ if $m(x_i) = 0$, and $l_i = \neg x_i$ otherwise. It is clear that the clause $c_m$ is *not* satisfiable in every minimal model of the formula $\varphi$ if and only if the assignment $m$ can be extended to a minimal model $\bar{m}$ of $\varphi$. From this follows that for each set of relations $S$ the problem MININF$(S)$ is $\Pi_2$P-complete, coNP-complete, or polynomial-time decidable if and only if MINEXT$(S)$ is $\Sigma_2$P-complete, NP-complete, or polynomial-time decidable, respectively.

To be able to perform the required complexity analysis, we need reduction theorems between minimal inference problems parametrized by different sets of relations. Since our sets of relations $\langle S \rangle$ are not closed under existential quantification, we cannot have the usual reduction theorem based on inclusion of polymorphisms.

**Proposition 4** *Let $R$ be a Boolean relation and $S$ a set of relations. If $R \in \langle S \rangle$ then there exists a polynomial many-one reduction from* MININF$(R)$ *to* MININF$(S)$.

**Proposition 5** *If $S^\circ$ is the irredundant reduction of a set of relations $S$, then* MININF$(S)$ *and* MININF$(S^\circ)$ *are equivalent under polynomial many-one reductions.*

We proceed with a sharpening of a result from Kirousis and Kolaitis [15].

**Proposition 6** *Let $S$ be a non-Schaefer and non-0-valid set of Boolean relations, with $S^*$ being the corresponding 1-valid restriction. If $S^*$ is Schaefer, then* MININF$(S)$ *is* coNP-*complete, otherwise it is* $\Pi_2$P-*complete.*

There exists one case which does not enter into the usual Schaefer classification of relations.

**Proposition 7** *If a set of relations $S$ is Schaefer and incomparable then* MININF$(S)$ *is in* P.

*Proof:* Let $\varphi \models_{\min} \psi$ be an instance of MININF with $S$ incomparable. It can be shown that $[\varphi]$ is an incomparable relation. This follows from the fact that incomparable relations are preserved under conjunction and variable identification. Given two formulas $\varphi_1$ and $\varphi_2$, each having only incomparable models, their conjunction $\varphi_1 \wedge \varphi_2$ have only incomparable models since a model of $\varphi_1$ can never be extended to two comparable models of $\varphi_1 \wedge \varphi_2$ (because $\varphi_2$ has only incomparable models). Variable identification only reduces the set of models of a formula. If the original set of models were all incomparable, then of course the reduced set of models will also be incomparable.

Since $\varphi$ is expressed by conjunction and variable identification over an incomparable set of relations $S$ it follows that all models of $\varphi$ are incomparable and hence also minimal. Thus, $\varphi \models_{\min} \psi$ if and only if $\varphi \models \psi$, where the latter can be checked in polynomial time since $S$ is Schaefer. ■

Consider now $S$ to be Schaefer or 0-valid. Tractability of MININF($S$) for $S$ being 0-valid is trivial and the polynomial-time decidability of MININF($S$) for $S$ being Horn follows from the fact that a satisfiable Horn formula has a unique minimal model computable in polynomial time. In the rest of the paper we investigate the other cases for which MININF($S$) is in coNP, namely when $S$ is dual Horn, bijunctive, or affine.

## V. Affine Relations

In this section we separate the coNP-complete cases from the tractable cases for MININF($S$) with an affine set of relations $S$. This result is a significant strengthening of the previous result due to Durand and Hermann [8], which states that there exists a set of affine relations $S$ for which MININF($S$) is coNP-complete. Moreover, their proof is rather involved in comparison to the proofs we present here.

**Lemma 8** *Let $R$ be a Boolean relation which is affine and 1-valid, but neither 0-valid, nor other Schaefer. Then we can construct the relation $[(x+y+z=1) \wedge (w=1)]$ from $R$ by conjunction and variable identification.*

*Proof:* Since $R$ is not Horn, there exist two vectors $a, b \in R$, such that $a \wedge b \notin R$. Construct the variable sets $V_{ij} = \{x \in V \mid a(x) = i, b(x) = j\}$. Identify the variables in each set $V_{ij}$, i.e., construct the relation $R' = R[V_{00}, V_{01}, V_{10}, V_{11}]$. It can be easily seen that the vectors 0011, 0101, and 1111 belong to $R'$, whereas $0000 \notin R'$ (not 0-valid) and $0001 \notin R'$ (not Horn). Since $R'$ is affine, we have $0011 + 0101 + 1111 = 1001 \in R'$.

Let $m, m' \in R'$ and $m'' \notin R'$. Then also $m + m' + m'' \notin R'$, since otherwise from the membership of $m$, $m'$, and $m + m' + m''$ in $R'$ follows that $(m + m' + m'') + m + m' = m'' \in R'$, because $R'$ is affine. Using this result, from $0011, 0101 \in R'$ and $0001 \notin R'$ follows $0110 \notin R'$. From $0101, 1111 \in R'$ and $0001 \notin R'$ follows $1011 \notin R'$. Finally from $0011, 1111 \in R'$ and $0001 \notin R'$ follows $1101 \notin R'$. We can force the variable $V_{11}$ to take the value 1 by the constraint $R'(V_{11}, V_{11}, V_{11}, V_{11})$, since $R$ as well as $R'$ are both 1-valid but not 0-valid. Hence, the constraint $R'(x, y, z, w) \wedge R'(w, w, w, w)$ generates the relation $[(x+y+z=1) \wedge (w=1)] = \{0011, 0101, 1001, 1111\}$. ■

**Lemma 9** MINEXT($R$) *for $R = [(x+y+z=1) \wedge (x+w=1)]$ and $R = [(x+y+z=0) \wedge (x+w=1)]$ is* NP*-complete.*

*Proof:* The membership in NP is clear, we focus on the NP-hardness proof by means of a reduction from 3SAT. Let $\varphi(x_1, \ldots, x_n) = c_1 \wedge \cdots \wedge c_k$ be a 3SAT formula. Associate the variables $y_i$ with the clauses $c_i$ for $i = 1, \ldots, k$. Construct the system in the following way.

For each clause $c_i = l_i^1 \vee l_i^2 \vee l_i^3$ add the following three formulas $(z_{3i-2} + v_i^1 + y_i = 1) \wedge (v_i^1 + u_i^1 = 1)$, $(z_{3i-1} + v_i^2 + y_i = 1) \wedge (v_i^2 + u_i^2 = 1)$, $(z_{3i} + v_i^3 + y_i = 1) \wedge (v_i^3 + u_i^3 = 1)$ to the formula $\varphi'$, where

$$v_i^j = \begin{cases} x_p & \text{if } l_i^j = \neg x_p, \\ x'_p & \text{if } l_i^j = x_p. \end{cases}$$

and

$$u_i^j = \begin{cases} x'_p & \text{if } l_i^j = \neg x_p, \\ x_p & \text{if } l_i^j = x_p. \end{cases}$$

The variable $v_i^j$ is a placeholder for the negation $\neg l_i^j$ of the literal $l_i^j$ and vice versa for $u_i^j$.

*Claim:* The formula $\varphi$ is satisfiable if and only if the assignment $s(y_i) = 1$ for all $i = 1, \ldots, k$ of $\varphi'$ has a minimal extension.

*a) Let $\varphi$ be satisfiable:* Let $m$ be a satisfying assignment of $\varphi$. Then in each clause $c_i$ there must be a literal $l_i^j$ such that $m(l_i^j) = 1$. Hence, for each extension $\bar{m}$ of $m$ that satisfies $\varphi'$ we must have $\bar{m}(v_i^j) = 0$ following the definition of $v_i^j$, where the equation $x + x' = 1$ enforces the variables $x$ and $x'$ to take opposite values. The value $\bar{m}(v_i^j) = 0$ implies two different incomparable assignments for the variables $y_i$ and $z_{3i+3-j}$. The first one is $\bar{m}(z_{3i+3-j}) = 1$ and $\bar{m}(y_i) = 0$, which is not interesting for us. The second is $\bar{m}(z_{3i+3-j}) = 0$ and $\bar{m}(y_i) = 1$, which is the desired minimal assignment. It is clear from the construction of the formula $\varphi'$ that this assignment is minimal. It is also clear that $\bar{m}$ is an extension of the assignment $s(y_i) = 1$ for all $i = 1, \ldots, k$. Hence, the assignment $s(y_i) = 1$ for all $i$ can be extended to a minimal one.

*b) Let $\varphi$ be unsatisfiable:* Then for all assignments $m$ of the variables $V$ there exists a falsified clause $c_i$. This implies that $m(l_i^j) = 0$ for all $j = 1, 2, 3$. Let $\bar{m}$ be an extension of $m$ that satisfies the formula $\varphi'$. Then the structure of the formula $\varphi'$ implies that we have $\bar{m}(v_i^1) = \bar{m}(v_i^2) = \bar{m}(v_i^3) = 1$. This implies the existence of a minimal assignment with $y_i = 0$ and $\bar{m}(z_{3i+3-j}) = 0$ for all $j$. Hence, the assignment $s(y_i) = 1$ for all $i$ cannot be extended to a minimal one.

For the latter relation, by swapping variables we get $[(x + y + z = 0) \wedge (x + w = 1)] = [(w + y + z = 1) \wedge (x + w = 1)]$ and the result follows from the previous relation. ■

**Lemma 10** MINEXT($R$) *for $R = [x+y+z=1]$ and $R = [(x+y+z=1) \wedge (w=1)]$ is* NP*-complete.*

**Proposition 11** MININF($S$) *is* coNP*-complete for each set of relations $S$ which is affine and 1-valid, but neither 0-valid, nor other Schaefer.*

*Proof:* We can construct a relation $R$ which is affine and 1-valid, but not 0-valid, not Horn, not dual Horn, nor bijunctive, by Cartesian product of relations from $S$. The rest follows from Lemma 8, Lemma 10, and from the relationship between MINEXT and MININF. ∎

**Lemma 12** *If $S$ is neither complementive nor 0-valid nor 1-valid, then the relation $R_{01} = \{01\}$ can be constructed from $S$ by conjunction and variable identification.*

**Lemma 13** *Given an irredundant affine relation $R$ which is not incomparable, then any two comparable tuples $a < b$ in $R$ must differ in at least two positions. Moreover, there must be a third tuple $c$ which is not constant on the positions where $a$ and $b$ differ.*

*Proof:* We claim that the existence two comparable tuples $a < b$ differing in just one position implies the relation $R$ to be redundant. Denote by $i$ the coordinate on which $a$ and $b$ differ. For any tuple $c$ in $R$ we construct the tuple $c' = a+b+c$, which is identical to $c$ except that $c'[i] = \neg c[i]$. Since $R$ is affine, the tuple $c'$ must be in the relation. Hence, $R$ is redundant since it is of the form $Q \times \{0,1\}$. Thus in any irredundant relation any two comparable tuples must differ in at least two positions.

Now let $a$ and $b$ differ on at least two positions, say $i$ and $j$. If all tuples $c \in R$ are constant on the positions $i$ and $j$ where $a$ and $b$ differ, i.e., $c[i] = c[j]$ for all $c \in R$, then $R$ is again redundant because in particular the columns $i$ and $j$ are identical. ∎

**Proposition 14** MININF$(S)$ *is* coNP*-hard for each set of relations $S$, which is affine, but neither incomparable, nor other Schaefer, nor 0-valid, nor 1-valid, nor complementive.*

*Proof:* By Lemma 12 we know that we have access to the relation $R_{01}(x,y) = \{01\}$ which allows us to impose variables to take constant values. Since $S$ is neither Horn nor dual Horn, by taking Cartesian products we can assume that there is a relation $N \in \langle S \rangle$ and two tuples $a, b \in N$, such that $a \wedge b \notin N$ and $a \vee b \notin N$. Use Schaefer's variable identification on $a$ and $b$ to get the relation $N[V_{00}, V_{01}, V_{10}, V_{11}]$. Now we need to impose the constant 0 on $V_{00}$ and 1 on $V_{11}$, using the relation $R_{01}$. We construct the constraint $N(z,x,y,w) \wedge R_{01}(z,w) = (x \not\equiv y) \wedge (z = 0) \wedge (w = 1)$ by conjunction and variable identification.

By taking Cartesian products of relations in $S$, there exists a single relation $R$ in $\langle S \rangle$ which is affine, but neither incomparable, nor bijunctive, nor Horn, nor dual-Horn. Since $R$ is not incomparable there exist two tuples $a$ and $b$ satisfying the condition $a < b$. Without loss of generality we can assume that $a$ and $b$ are closest possible, i.e., there is no tuple $t$ where $a < t < b$ holds. By Lemma 13 we know that $a$ and $b$ differ in at least two positions, and that there is a third tuple $c$ not constant on the coordinates where $a$ and $b$ differ.

Since $R$ is neither Horn nor dual-Horn, we can assume by taking a Cartesian product of $R$ with itself, that $c$ satisfies the conditions $a \wedge c \notin R$, $b \wedge c \notin R$, $a \vee c \notin R$, and $b \vee c \notin R$. Form the Schaefer-style variable identification on $R$ based on the tuples $a, b, c$. Note that since $a < b$ holds, the variables $V_{100}$ and $V_{101}$ will not appear. Thus we get the relation $R[V_{000}, V_{001}, V_{010}, V_{011}, V_{110}, V_{111}]$. It can be checked that the variables $V_{001}, V_{010}, V_{011}, V_{110}$ must all appear. Observe that if $V_{001}$ does not appear, then $b \vee c = b \in R$ which is a contradiction. Similarly, if $V_{110}$ does not appear, then $a \wedge c = a \in R$ which is a contradiction. Moreover, $V_{010}$ and $V_{011}$ must appear since $c$ is not constant on the coordinates where $a$ and $b$ differ.

Since $R$ is affine, it also contains the tuple $d = a + b + c = 011001$. Moreover, $R$ does not contain the tuple $t = \mathrm{maj}(a,b,c) = 000111$ since this tuple satisfies the condition $a < t < b$. We add the constraint $(V_{110} \not\equiv V_{001}) \wedge (V_{000} = 0) \wedge (V_{111} = 1)$ to $R[V_{000}, V_{001}, V_{010}, V_{011}, V_{110}, V_{111}]$. The resulting constraint contains the tuples 000011, 001111, 010101, 011001, but it does not contain the tuple 000111. There are only three undetermined variables so there can be at most 8 tuples satisfying the affine constraint. Moreover, since the constraint is affine the number of tuples satisfying the constraint is a power of 2. Therefore, since 000111 is not satisfying the constraint, we have that $R[V_{001}, V_{010}, V_{011}, V_{110}] \wedge (V_{110} \not\equiv V_{001}) \wedge (V_{000} = 0) \wedge (V_{111} = 1) = (V_{001} + V_{010} + V_{011} = 0) \wedge (V_{110} + V_{001} = 1) \wedge (V_{000} = 0) \wedge (V_{111} = 1)$. This relation is coNP-hard by Lemma 9. ∎

**Lemma 15** *If $S$ is complementive and neither 0-valid nor 1-valid, then we can construct $[x \not\equiv y]$ from $S$ by conjunction and variable identification.*

**Lemma 16** MINEXT$(R)$ *for* $R = [(x+y+z+w = 0) \wedge (x + u = 1) \wedge (y + v) = 1]$ *is* NP*-complete.*

*Proof:* NP-membership is clear, we focus on the NP-hardness proof by a reduction from NAE3SAT (not-all-equal 3SAT). Let $\varphi = c_1 \wedge \cdots \wedge c_k$ be a not-all-equal 3sat formula with variables $x_1, \ldots, x_n$ and clauses $c_i = l_i^1 \vee l_i^2 \vee l_i^3$. We construct a formula $\varphi'$ as the conjunction of the following equations for each clause $c_i = l_i^1 \vee l_i^2 \vee l_i^3$:

$$\begin{aligned}
(z_{3i-2} + v_i^1 + v_i^2 + y_i = 0) \quad &\wedge \quad (v_i^1 + u_i^1 = 1) \\
&\wedge \quad (v_i^2 + u_i^2 = 1), \\
(z_{3i-1} + v_i^2 + v_i^3 + y_i = 0) \quad &\wedge \quad (v_i^2 + u_i^2 = 1) \\
&\wedge \quad (v_i^3 + u_i^3 = 1), \\
(z_{3i} + v_i^3 + v_i^1 + y_i = 0) \quad &\wedge \quad (v_i^3 + u_i^3 = 1) \\
&\wedge \quad (v_i^1 + u_i^1 = 1),
\end{aligned}$$

where

$$v_i^j = \begin{cases} x_p & \text{if } l_i^j = x_p, \\ x'_p & \text{if } l_i^j = \neg x_p. \end{cases}$$

and

$$u_i^j = \begin{cases} x'_p & \text{if } v_i^j = x_p, \\ x_p & \text{if } v_i^j = x'_p. \end{cases}$$

The variable $v_i^j$ is a placeholder for the literal $l_i^j$, whereas $u_i^j$ is the negation of $v_i^j$.

*Claim:* The formula $\varphi$ is satisfiable if and only if the assignment $s(y_i) = 1$ for all $i = 1, \ldots, k$ of $\varphi'$ has a minimal extension.

Let $\varphi$ be not-all-equal satisfiable. Then there exists a not-all-equal satisfying assignment $m$ of the formula $\varphi$. Since every clause $c_i$ is not-all-equal satisfied, for each $i$ there must be $a, b \in \{1, 2, 3\}$ such that $m(l_i^a) \neq m(l_i^b)$. Let $\bar{m}$ be an extension of $m$ that satisfies $\varphi'$. Following the construction of $\varphi'$, we must have $\bar{m}(v_i^a) + \bar{m}(v_i^b) = 1$. Then there are two possibilities to get a minimal assignment $\bar{m}$:

1) When we set $\bar{m}(z_{3i-p}) = 1$ and $\bar{m}(y_i) = 0$, we get an uninteresting assignment.
2) When we set $\bar{m}(z_{3i-p} = 0$ and $\bar{m}(y_i) = 1$, we get an assignment which is an extension of the assignment $s(y_i) = 1$ for all $i = 1, \ldots, k$.

These two possible assignments are clearly incomparable and no value can be changed from 1 to 0 to get another satisfying assignment of $\varphi'$. Therefore the assignment $\bar{m}$ from the second case is minimal.

Let $\varphi$ be not-all-equal unsatisfiable. Then for each assignment $m$ there must always be a clause $c_i$ which literals are assigned the same values, i.e., $m(l_i^1) = m(l_i^2) = m(l_i^3) = 0$ or $m(l_i^1) = m(l_i^2) = m(l_i^3) = 1$. Let $\bar{m}$ be an extension of $m$ that satisfies the formula $\varphi'$. Following the construction of $\varphi'$, we have $\bar{m}(v_i^1)+\bar{m}(v_i^2) = \bar{m}(v_i^2)+\bar{m}(v_i^3) = \bar{m}(v_i^3)+\bar{m}(v_i^1) = 0$. Then we can set $\bar{m}(z_{3i-2}) = \bar{m}(z_{3i-1}) = \bar{m}(z_{3i}) = \bar{m}(y_i) = 0$ to produce a minimal assignment. This implies that the assignment $s(y_i) = 1$ for all $i$ cannot be extended to a minimal one. ■

**Proposition 17** MININF$(S)$ *is* coNP-*hard for each set of relations $S$, which is affine and complementive, but neither incomparable, nor other Schaefer, nor* 0-*valid, nor* 1-*valid.*

*Proof:* By taking Cartesian products of relations in $S$ there exists a relation $R$ in $\langle S \rangle$ which is affine, but neither incomparable, nor bijunctive, nor Horn, nor dual-Horn. Since $R$ is not incomparable there are two tuples $a$ and $b$ satisfying the condition $a < b$. Without loss of generality we can assume that $a$ and $b$ are closest, i.e., there is no tuple $t$ satisfying $a < t < b$. By Lemma 13 we know that $a$ and $b$ differ in at least two positions and that there exists a third tuple $c$ which is not constant on the coordinates where $a$ and $b$ differ.

Since $R$ is neither Horn, nor dual-Horn we can assume, by taking a Cartesian products of $R$ with itself, that $c$ satisfies the conditions $a \wedge c \notin R$, $b \wedge c \notin R$, $a \vee c \notin R$, and $b \vee c \notin R$. Form the Schaefer style implementation on $R$ based on the tuples $a, b, c$. Note that since $a < b$ holds, the variables $V_{100}$ and $V_{101}$ will not appear. Hence we get the relation $R[V_{000}, V_{001}, V_{010}, V_{011}, V_{110}, V_{111}]$. It can be checked that the variables $V_{001}, V_{010}, V_{011}, V_{110}$ must all appear. If $V_{001}$ does not appear, then $b \vee c = b \in R$ which is a contradiction. Similarly, if $V_{110}$ does not appear, then $a \wedge c = a \in R$ which is a contradiction. Moreover, $V_{010}$ and $V_{011}$ must appear since $c$ is not constant on the coordinates where $a$ and $b$ differ.

Since $R$ is affine, it must also contain the tuple $d = a + b + c = 011001$. Moreover, $R$ does not contain the tuple $t = \mathrm{maj}(a, b, c) = 000111$ since this tuple satisfies $a < t < b$. Furthermore, since $R$ is complementive it also contains the tuples 100110, 111100 101010, 110000. Now, adding the constraints $V_{000} \not\equiv V_{111}$ and $V_{110} \not\equiv V_{001}$, to which we have access according to Lemma 15, we get the relation $R[V_{000}, V_{001}, V_{010}, V_{011}, V_{110}, V_{111}] \wedge (V_{000} \not\equiv V_{111}) \wedge (V_{110} \not\equiv V_{001})$. Since the value of $V_{110}$ is determined by $V_{001}$ and vice versa, and the same for $V_{000}$ and $V_{111}$, there can be at most 16 tuples satisfying the affine constraint. We already have 8 tuples that satisfy the constraint and we know that the tuple 000111 does not satisfy the constraint. Since the constraint is affine, the number of tuples satisfying the constraint must be a power of 2. The constraint $R[V_{000}, V_{001}, V_{010}, V_{011}, V_{110}, V_{111}] \wedge (V_{000} \not\equiv V_{111}) \wedge (V_{110} \not\equiv V_{001})$ generates the relation {000011, 001111, 010101, 011001, 100110, 111100, 101010, 110000}. It is equivalent to the constraint $(V_{000} + V_{001} + V_{010} + V_{011} = 0) \wedge (V_{000} + V_{111} = 1) \wedge (V_{001} + V_{110} = 1)$, for which the minimal inference problem is coNP-complete according to Lemma 16. ■

## VI. Dual Horn Relations

**Proposition 18** MININF$(S)$ *is* coNP-*hard for each set of relations S, which is dual Horn, but neither Horn, nor* 0-*valid, nor* 1-*valid.*

*Proof:* Construct a relation $R$ from $S$ which is neither 0-valid nor 1-valid. Such a relation exists by taking Cartesian products. Take a tuple $a \in R$ and construct the relation $R[V_0, V_1]$ on $a$. Both variables $V_0$ and $V_1$ must appears since $R$ is neither 0-valid nor 1-valid. Hence $01 \in R[V_0, V_1]$. Neither 00 nor 11 is in $R[V_0, V_1]$ since $R$ is neither 0-valid nor 1-valid. Moreover, 10 is not in $R$, since if it were then the fact that $R$ is dual-Horn together with $01 \in R$ implies $11 \in R$ which we have already ruled out. Thus $R[V_0, V_1]$ represents the relation {01}.

Now construct from $S$ a relation $Q$ which is dual Horn, but not Horn. There must be two tuples $a, b \in Q$, such that $a \wedge b \notin Q$ and $a \vee b \in Q$. Construct the relation $Q[V_{00}, V_{01}, V_{10}, V_{11}]$ on $a$ and $b$. Form the conjunction $Q[V_{00}, V_{01}, V_{10}, V_{11}] \wedge R[V_{00}, V_{11}]$ forcing the variables $V_{00}$ and $V_{11}$ to take the values 0 and 1, respectively. The constraint $Q'(V_{00}, V_{01}, V_{10}, V_{11}) = Q[V_{00}, V_{01}, V_{10}, V_{11}] \wedge R[V_{00}, V_{11}]$ is equal to $(V_{01} \vee V_{10}) \wedge (V_{00} = 0) \wedge (V_{11} = 1)$. Cadoli and Lenzerini [4] proved that MININF$([x \vee y])$ is coNP-complete, therefore also MININF$(Q')$ is coNP-complete for $Q' = [(x \vee y) \wedge (z = 0) \wedge (w = 1)]$. ■

**Lemma 19** MININF$(R)$ *for* $R = [(x \vee y) \wedge (\neg z \vee x) \wedge (\neg z \vee y)]$ *and* $R = [(x \vee y) \wedge (\neg z \vee x)]$ *is* coNP-*complete.*

**Proposition 20** MININF$(S)$ *is* coNP-*hard for each set of relations S, which is dual-Horn and* 1-*valid, but neither* 0-*valid nor Horn.*

*Proof:* We know that there is a relation $R$ in $\langle S\rangle$ which is dual-Horn and 1-valid but not 0-valid, constructed by taking Cartesian products. Now, do Schaefer's construction on two tuples $a, b \in R$ such that $a \wedge b \notin R$ and $a \vee b \in R$. Such tuples exist, since $R$ is dual Horn, but not Horn. This construction gives us the relation $Q(V_{00}, V_{01}, V_{10}, V_{11}) = R[V_{00}, V_{01}, V_{10}, V_{11}]$. Since $Q$ is 1-valid but not 0-valid, we have access to the constraint $(x = 1)$. Construct the conjunction $Q'(V_{00}, V_{01}, V_{10}, V_{11}) = Q(V_{00}, V_{01}, V_{10}, V_{11}) \wedge Q(V_{11}, V_{11}, V_{11}, V_{11})$. Now, $V_{01}$ and $V_{10}$ both exists, otherwise we would have $a \wedge b \in R$. Moreover, $0001 \notin Q'$ since $a \wedge b = 0001$. If $V_{00}$ does not exist then $Q' = [(x \vee y) \wedge (z = 1)]$. Cadoli and Lenzerini [4] that MININF$([x \vee y])$ is coNP-complete, hence also MININF$(Q')$ is coNP-complete.

Assume now that $V_{00}$ exists We know that $Q'$ contains the tuples 1111, 0011, 0101, and 0111. We now have to consider 5 cases according to the presence of the tuples 1001, 1011, and 1101:

**Case 1:** $Q' = \{1111, 0011, 0101, 0111\}$ which is equal to $[(y \vee z) \wedge (\neg x \vee y) \wedge (\neg x \vee z) \wedge (w = 1)]$ for which MININF is coNP-complete according to Lemma 19.

**Case 2:** $Q'$ contains the tuple 1001. Because $Q'$ is dual Horn, it must also contain the tuples 1011 and 1101. Since 0001 is not in the relation, we have constructed $[(x \vee y \vee z) \wedge (w = 1)]$. By variable identification we can get the relation $[(x \vee y) \wedge (w = 1)]$ for which we already know MININF to be coNP-complete. hence we can assume further on that $Q'$ does not contain the tuple 1001.

**Case 3:** $Q'$ contains 1011. Then it is equal to $[(y \vee z) \wedge (\neg x \vee z) \wedge (w = 1)]$ for which MININF is coNP-complete according to Lemma 19.

**Case 4:** $Q'$ contains 1101. Then it is equal to $[(y \vee z) \wedge (\neg x \vee y) \wedge (w = 1)]$ for which MININF is coNP-complete according to Lemma 19.

**Case 5:** $Q'$ contains both tuples 1101 and 1011. Then we can construct the new constraint $Q''(x, y, z, w) = Q'(x, y, z, w) \wedge Q(x, x, x, x)$ which is equivalent to $(y \vee z) \wedge (x = 1) \wedge (w = 1)$. Since MININF$([x \vee y])$ is coNP-complete, we know that MININF$(Q'')$ is coNP-complete.

There are no more cases to consider. ∎

## VII. Bijunctive Relations

**Proposition 21** *MININF$(S)$ for $S = \{[\neg x \vee \neg y], [x \not\equiv y]\}$ is in* P.

*Proof:* Let $\varphi \models_{\min} \psi$ be an instance of MININF$(S)$ for $S = \{[\neg x \vee \neg y], [x \not\equiv y]\}$. Check first that $\varphi$ is satisfiable, what can be done in polynomial time since $S$ is bijunctive. If $\varphi$ is NOT satisfiable, then $\varphi \models_{\min} \psi$ is trivially satisfied.

If the formula $\varphi$ is only produced from the relation $[\neg x \vee \neg y]$ then it is 0-valid and therefore $\varphi \models_{\min} \psi$ is in P.

The formula $\varphi$ is a conjunction of binary constraints of the type $(\neg x \vee \neg y)$ and $(x \not\equiv y)$. Let us consider the structure of $\varphi$. If a variable $v$ occurs only in negative constraints, i.e., only in constraints of the form $(\neg x \vee \neg y)$, then $v$ has the value 0 in all minimal models of $\varphi$. Hence, we can drop all clauses $(\neg v \vee \neg y)$ from $\varphi$ and remove the variable $v$ from $\psi$. Repeat this replacement until all variables occur in a $\not\equiv$ constraint.

Let $\varphi' \models_{\min} \psi'$ be the resulting instance. Assume with the aim of reaching a contradiction that there exist two comparable models $m_0 < m_1$ of $\varphi'$. Then there exists a variable $x$, such that $m_0(x) = 0$ and $m_1(x) = 1$. Let $y$ be a variable occurring together with $x$ in a $(x \not\equiv y)$ constraint. Then we have $m_0(y) = 1$ and $m_1(y) = 0$, which implies that $m_0 \not< m_1$, constituting a contradiction. Hence, all models of $\varphi'$ are incomparable. Therefore $\varphi' \models_{\min} \psi'$ holds if and only if $\varphi' \models \psi'$ which is decidable in polynomial time since $\varphi'$ is bijunctive. ∎

The following result shows that in the case of bijunctive constraints in the minimal inference problem we can impose variables to take constant values. The unary relations $F = \{0\}$ and $T = \{1\}$ can be used to represent the logical values 0 and 1, respectively. Similarly, the binary relation $R_{01} = \{01\}$ can be used to represent both logical values at once. Hence, $T(x)$ implies $x = 1$, $F(x)$ implies $x = 0$, and $R_{01}(x, y)$ implies $x = 0$, $y = 1$.

**Lemma 22** *If $S$ is bijunctive but not other Schaefer, then we can construct by conjunction and variable identification the binary relation $R_{01} = \{01\}$.*

**Lemma 23** *MINEXT$(R)$ for $R = [(x \vee y) \wedge (x \not\equiv z)]$ is* NP-*complete.*

*Proof:* The membership in NP is clear, we focus on the NP-hardness proof by means of a reduction from 3SAT. Let $\varphi(x_1, \ldots, x_n) = c_1 \wedge \cdots \wedge c_k$ be a 3SAT formula. Associate the variable $y_i$ with the clause $c_i$ for $i = 1, \ldots, k$. For each clause $c_i = l_i^1 \vee l_i^2 \vee l_i^3$ we add the following three formulas $(y_i \vee v_i^1) \wedge (v_i^1 \not\equiv \bar{v}_i^1)$, $(y_i \vee v_i^2) \wedge (v_i^2 \not\equiv \bar{v}_i^2)$, and $(y_i \vee v_i^3) \wedge (v_i^3 \not\equiv \bar{v}_i^3)$ to $\varphi'$, where

$$v_i^j = \begin{cases} x & \text{if } l_i^j = \neg x, \\ x' & \text{if } l_i^j = x. \end{cases}$$

and

$$\bar{v}_i^j = \begin{cases} x' & \text{if } l_i^j = \neg x, \\ x & \text{if } l_i^j = x. \end{cases}$$

*Claim:* The partial assignment $s(y_i) = 1$ for each $i = 1, \ldots, k$ can be extended to a minimal assignment of $\varphi'$ if and only if $\varphi$ is satisfiable.

Let $\varphi$ be satisfiable. Every clause $c_i$ evaluates to 1. For each clause $c_i$ there exists a $j$, such that $l_i^j = 1$. Then $v_i^j = 0$ which implies $y_i = 1$. Moreover, since $v_i^j \not\equiv \bar{v}_i^j$ holds, every satisfying assignment to $\varphi'$ is incomparable, hence minimal.

Let $\varphi$ be unsatisfiable. Then there exists a falsified clause $c_i$, i.e., $l_i^1 = l_i^2 = l_i^3 = 0$, which implies $v_i^1 = v_i^2 = v_i^3 = 1$. Therefore there exists a satisfying assignment $m$ of $\varphi'$ with $m(y_i) = 0$. hence $s(y_i) = 1$ for $i = 1, \ldots, k$ cannot be extended to a minimal solution. ∎

We need the following result from [12], based on a previous algebraic result from [1], stating that it is sufficient to consider binary relations when we consider bijunctive constraint languages.

**Proposition 24 (Jeavons *et al.* [12])** *Given a $n$-ary bijunctive constraint $R(x_1, \ldots, x_n)$ then it is equivalent to $\bigwedge_{1 \leq i \leq j} R_{ij}(x_i, x_j)$ where $R_{ij}$ is the projection of the relation $R$ to the coordinates $i$ and $j$.*

**Proposition 25** *Let $N = \{[\neg x \vee \neg y], [x \not\equiv y]\}$. MININF$(S)$ for each set of relations $S$, which is bijunctive, but neither other Schaefer, nor a subset of $\langle N \rangle$, is coNP-complete.*

*Proof:* Let $R$ be the Cartesian product of all relations in $S$. Obviously, $R$ is bijunctive but neither Horn, nor dual-Horn, nor affine, nor a subset of $\langle N \rangle$. Let $\varphi$ be the conjunction of binary constraints representing the relation $R$, produced according to Proposition 24.

There must be a clause $(\ell_p \vee \ell) \in \varphi$ with at least one positive literal, say $\ell_p$, otherwise we would have $[\varphi] \in \langle N \rangle$. If $\ell$ is a negative literal, then $\ell_p$ must not occur in a $\not\equiv$ constraint, otherwise we would again have $[\varphi] \in \langle N \rangle$. If both $\ell_p$ and $\ell$ are positive, then $\ell_p$ or $\ell$ must not occur in a $\not\equiv$ constraint, otherwise we would again have $[\varphi] \in \langle N \rangle$. In all cases, this is because $(\ell_p \vee \ell) \wedge (\ell_p \not\equiv x) = (\neg x \vee \ell) \wedge (\ell_p \not\equiv x)$. Moreover, the literals $\ell_p$ and $\ell$ cannot be assigned constant values. Indeed, if $\varphi = \varphi' \wedge (\ell_p \vee \ell) \models \neg \ell_p$ for a formula $\varphi'$ with $[\varphi'] \in \langle N \rangle$, then $\varphi$ is equivalent to $\varphi' \wedge (\neg \ell_p \vee \neg \ell_p) \wedge (\ell_p \not\equiv \ell)$, implying $[\varphi] \in \langle N \rangle$. If $\varphi = \varphi' \wedge (\ell_p \vee \ell) \models \ell_p$ then $\varphi$ is equivalent to $\varphi'$, implying again $[\varphi] \in \langle N \rangle$.

Hence, we are in the situation $(\ell_p \vee \ell) \in \varphi$ and $\ell_p$ is positive and it does not occur in any $\not\equiv$ constraint. Two cases emerge depending on whether $\ell$ appears in a $\not\equiv$ constraint. For the rest of the proof we focus on the case where $\ell$ appears in a $\not\equiv$ constraint. The other case can be handled in the same way with only minor and obvious modifications. Assume that $\ell_p = x$, $\mathrm{Var}(\ell) = y$, and $y$ occurs in a $\not\equiv$ constraint together with $z$, i.e., that $(x \vee y) \wedge (y \not\equiv z) \in \varphi$, or $(x \vee \neg y) \wedge (y \not\equiv z) \in \varphi$. Without loss of generality we assume the former case since $(x \vee \neg y) \wedge (y \not\equiv z) = (x \vee z) \wedge (y \not\equiv z)$.

Our goal is to produce the constraint $(x \vee y) \wedge (y \not\equiv z) \wedge R_{01}(v, w)$ from $\varphi$ by conjunction and variable identification. It is clear that both aforementioned constraints belong to $\langle S \rangle$, since we can produce from $S$ the constraint $R_{01}(x_0, x_1)$ following Lemma 22.

Simplify $\varphi$ by identifying all variables $x_1$, $x_2$ such that $(x_1 = x_2) \in \varphi$, then remove all equality constraints from $\varphi$. Form the sets of variables $V_0 = \{v \mid \varphi \models \neg v\}$ and $V_1 = \{w \mid \varphi \models w\}$. Remove from $\varphi$ all clauses containing variables in $V_0 \cup V_1$, then add the constraint $R_{01}(V_0, V_1)$ to $\varphi$. Obviously, this transformation of $\varphi$ does not change the fact that the projection of $\varphi$ onto $\{x, y\}$ is equivalent to $(x \vee y)$, whereas the projection of $\varphi$ onto $\{y, z\}$ is equivalent to $(y \not\equiv z)$. Note that there can be no $(x_1 = x_2) \in \varphi$ where $\{x_1, x_2\} \subseteq \{x, y, z\}$. In the same manner, we must have $\{x, y, z\} \cap V_0 = \emptyset$ and $\{x, y, z\} \cap V_1 = \emptyset$.

Continue simplifying the formula $\varphi$ by eliminating all $(x_1 \not\equiv x_2)$ constraints, except the one $(y \not\equiv z)$, by replacing $x_1$ by $\neg x_2$, as well as $\neg x_1$ by $x_2$, throughout the formula. Denote the resulting formula by $\varphi'$. Since $x$ does not occur in any $\not\equiv$ constraint and neither $y$ nor $z$ occur in any other $\not\equiv$ constraint except in $(y \not\equiv z)$, otherwise there would have to be some $(x_1 = x_2)$ constraint still left, we have $(x \vee y) \wedge (y \not\equiv z) \in \varphi'$.

Recall that resolution between two clauses $(c \vee v)$ and $(\neg v \vee c')$ produces the new clause $(c \vee c')$ and discards the two previous clauses. Note that resolution on binary clauses produces a binary clause. For any variable $x_i \notin \{x, y, z\}$ occurring both positively and negatively in $\varphi'$, apply resolution to get a formula, where every variable not in $\{x, y, z\}$ only occurs either positively or negatively, but not both. Denote the resulting formula by $\varphi''$. Form the variable sets $V_0'' = \{v \mid v \text{ only occurs negatively in } \varphi''\}$ and $V_1'' = \{w \mid w \text{ only occurs positively in } \varphi''\}$. Discard from $\varphi''$ the clauses containing variables from $V_0'' \cup V_1''$ and add the constraint $R_{01}(V_0'', V_1'')$ to the formula.

Hence using only conjunction and variable identification, we can implement $(x \vee y) \wedge (y \not\equiv z) \wedge R_{01}(v, w)$ from $S$. Using Lemma 23 we obtain the desired coNP-completeness result. ■

## VIII. Main Result

**Theorem 26 (Trichotomy of Minimal Inference)** *Let $S$ be a finite nonempty set of Boolean relations and $S^*$ the corresponding 1-valid restriction. If every relation in $S$ is Horn, or 0-valid, or both Schaefer and incomparable, or a subset of $\langle N \rangle$, where $N = \{[\neg x \vee \neg y], [x \not\equiv y]\}$, then MININF$(S)$ is decidable in polynomial time. Else if $S^*$ is Schaefer then MININF$(S)$ is coNP-complete. Otherwise MININF$(S)$ is $\Pi_2$P-complete.*

*Proof:* The parts concerning $\Pi_2$P-completeness and membership in coNP follow from the dichotomy theorem due to Kirousis and Kolaitis [15]. As for tractability, this is, as already explained, trivial for $S$ being Horn or 0-valid. Tractability of MININF$(S)$ for $S$ being Schaefer and incomparable, or a subset of $\langle [\neg x \vee \neg y], [x \not\equiv y] \rangle$, is proved in Propositions 7, and 21, respectively. The coNP-hardness for MININF$(S)$ when $S$ is neither Schaefer, nor 0-valid, and $S^*$ is Schaefer, is proved in Proposition 6.

Hence, what remains to be done is to prove coNP-hardness for all sets of relations $S$ that do not fall into one of the tractable classes when $S$ is affine, dual Horn, or bijunctive. This is done in Sections V, VI, and VII, respectively. For $S$ being affine, the analysis is divided into three cases, depending on whether $S$ is 1-valid (Proposition 11), complementive (Proposition 17), or neither 1-valid nor complementive (Proposition 14). Similarly, the analysis for the dual Horn case is divided into two parts depending on whether $S$ is 1-valid (Proposition 20) or not (Proposition 18). Finally, the case where $S$ is bijunctive is treated in Proposition 25. ■

Checking whether a relation $R$ is Schaefer can be done in polynomial time by testing for closure under corresponding operations (see Table I). Checking whether a relation is 0-valid or incomparable can be done in polynomial time by inspection. Checking whether $R$ is a subset of $\langle N \rangle$ is more subtle, since we cannot use the Galois correspondence with polymorphisms, because $\langle N \rangle$ is *not* closed under existential quantification. Recall that $\langle N \rangle$ is the set of all relations that can be expressed using relations from $N$, conjunction, and variable identification. However, we can apply Proposition 24, since the inclusion $R \subseteq \langle N \rangle$ implies that $R$ must be bijunctive. It is then sufficient to check that $R$ is closed under majority and that all projections $R_{ij}$ of $R$ are equal to one of the relations $[\neg x \vee \neg y]$ or $[x \not\equiv y]$. This test can be performed in polynomial time.

## IX. Concluding Remarks

We have proved a trichotomy for the complexity of the minimal inference problem with unbounded queries. It is natural to ask if such a result can be obtained in the case of bounded queries. Gottlob and Eiter [9] proved the $\Pi_2\mathrm{P}$-completeness of the minimal inference problem in general already for queries with a single literal which was propagated by Kirousis and Kolaitis [15] to obtain their dichotomy theorem. However, Cadoli and Lenzerini [4] showed for the bijunctive case that the minimal inference problem is $\mathrm{coNP}$-complete for unbounded queries, but it becomes polynomial-time decidable when the query is restricted to a single literal. The same effect was observed by Durand and Hermann in [8] for the affine case. Our $\mathrm{coNP}$-hardness proofs for the bijunctive, affine, and dual Horn cases are not valid for bounded queries. In fact, the bijunctive case is easily seen to be tractable for bounded queries. The result of [4], showing that there are $\mathrm{coNP}$-complete dual Horn minimal inference problems even for single literal queries, together with the fact that there exist tractable bounded query dual Horn minimal inference problems, indicate that the classification for bounded queries in the dual Horn case is more intricate.

The affine case might be more complicated to solve. It is easy to see that it corresponds to the following problem over representable matroids: given a set $S$ of $t$ elements, find a circuit passing through $S$. When $t \geq 2$ is bounded, the complexity of this problem is widely open [13]. Notice that the (simpler) problem restricted to graphs can be reduced to the $t$-disjoint path problem, showed to be polynomial-time decidable after considerable effort [23]. Therefore the final answer to complexity classification of the minimal inference problem with bounded queries still remains a challenging open question.


## Acknowledgment

We thank Phokion Kolaitis for valuable contributions and discussions in the initial stages of this work.


## References


[1] K. A. Baker and A. F. Pixley. Polynomial interpolation and the Chinese Remainder Theorem for algebraic systems. *Mathematische Zeitschrift*, 143(2):165–174, 1975.

[2] E. Böhler, N. Creignou, S. Reith, and H. Vollmer. Playing with Boolean blocks, part II: Constraint satisfaction problems. *SIGACT News, Complexity Theory Column 43*, 35(1):22–35, 2004.

[3] M. Cadoli. The complexity of model checking for circumscriptive formulae. *Information Processing Letters*, 44(3):113–118, 1992.

[4] M. Cadoli and M. Lenzerini. The complexity of propositional closed world reasoning and circumscription. *Journal of Computer and System Sciences*, 48(2):255–310, 1994.

[5] Z.-Z. Chen and S. Toda. The complexity of selecting maximal solutions. *Information and Computation*, 119(2):231–239, 1995.

[6] N. Creignou, S. Khanna, and M. Sudan. *Complexity Classifications of Boolean Constraint Satisfaction Problems*, volume 7 of *SIAM Monographs on Discrete Mathematics and Applications*. SIAM, Philadelphia (PA), 2001.

[7] J. de Kleer, A. K. Mackworth, and R. Reiter. Characterizing diagnoses and systems. *Artificial Intelligence*, 56(2-3):197–222, 1992.

[8] A. Durand and M. Hermann. The inference problem for propositional circumscription of affine formulas is coNP-complete. In H. Alt and M. Habib, editors, *Proceedings 20th Symposium on Theoretical Aspects of Computer Science (STACS 2003), Berlin (Germany)*, volume 2607 of *Lecture Notes in Computer Science*, pages 451–462. Springer-Verlag, February 2003.

[9] T. Eiter and G. Gottlob. Propositional circumscription and extended closed-world reasoning are $\Pi_2^p$-complete. *Theoretical Computer Science*, 114(2):231–245, 1993.

[10] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In R. A. Kowalski and K. A. Bowen, editors, *Proceeding 5th International Conference and Symposium on Logic Programming ICLP/SLP, Seatle (Washington)*, pages 1070–1080. MIT Press, 1988.

[11] M. Gelfond, H. Przymusinska, and T. C. Przymusinski. On the relationship between circumscription and negation as failure. *Artificial Intelligence*, 38(1):75–94, 1989.

[12] P. Jeavons, D. Cohen, and M. Gyssens. Closure properties of constraints. *Journal of the Association for Computing Machinery*, 44(4):527–548, 1997.

[13] L. Khachiyan, E. Boros, K. Elbassioni, V. Gurvich, and K. Makino. On the complexity of some enumeration problems for matroids. *SIAM Journal on Discrete Mathematics*, 19(4):966–984, 2005.

[14] L. M. Kirousis and P. G. Kolaitis. The complexity of minimal satisfiability problems. *Information and Computation*, 187(1):20–39, 2003.

[15] L. M. Kirousis and P. G. Kolaitis. A dichotomy in the complexity of propositional circumscription. *Theory of Computing Systems*, 37(6):695–715, 2004.

[16] J. McCarthy. Circumscription — A form of non-monotonic reasoning. *Artificial Intelligence*, 13(1-2):27–39, 1980.

[17] J. McCarthy. Applications of circumscription to formalizing commonsense knowledge. *Artificial Intelligence*, 28(1):89–116, 1986.

[18] G. Nordh. A trichotomy in the complexity of propositional circumscription. In F. Baader and A. Voronkov, editors, *Proceedings 11th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR 2004), Montevideo (Uruguay)*, volume 3452 of *Lecture Notes in Computer Science*, pages 257–269. Springer-Verlag, March 2005.

[19] G. Nordh and P. Jonsson. An algebraic approach to the complexity of propositional circumscription. In *Proceedings 19th IEEE Symposium on Logic in Computer Science (LICS 2004), Turku (Finland)*, pages 367–376, July 2004.

[20] C. H. Papadimitriou. On selecting a satisfying truth assignment (extended abstract). In *Proceedings 32nd IEEE Symposium on Foundations of Computer Science (FOCS'91), San Juan (Puerto Rico)*, pages 163–169. IEEE Computer Society, October 1991.

[21] E. L. Post. The two-valued iterative systems of mathematical logic. *Annals of Mathematical Studies*, 5:1–122, 1941.

[22] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1-2):81–132, 1980.

[23] N. Robertson and P. D. Seymour. Graph minors XIII: The disjoint paths problem. *Journal of Combinatorial Theory, Series B*, 63(1):65–110, 1995.

[24] T. J. Schaefer. The complexity of satisfiability problems. In *Proceedings 10th Symposium on Theory of Computing (STOC'78), San Diego (California, USA)*, pages 216–226, 1978.