The Complexity of Theory Revision*

Russell Greiner†
Department of Computing Science
University of Alberta
Edmonton, AB T6G 2H1 Canada
greiner@cs.ualberta.ca http://www.cs.ualberta.ca/~greiner

November 16, 1998

Abstract

A knowledge-based system uses its database (a.k.a. its "theory") to produce answers to the queries it receives. Unfortunately, these answers may be incorrect if the underlying theory is faulty. Standard "theory revision" systems use a given set of "labeled queries" (each a query paired with its correct answer) to transform the given theory, by adding and/or deleting either rules and/or antecedents, into a related theory that is as accurate as possible. After formally defining the theory revision task, this paper provides both sample and computational complexity bounds for this process. It first specifies the number of labeled queries necessary to identify a revised theory whose error is close to minimal with high probability. It then considers the computational complexity of finding this best theory, and proves that, unless $P = NP$, no polynomial time algorithm can identify this near-optimal revision, even given the exact distribution of queries, except in certain simple situation. It also shows that, except in such simple situations, no polynomial-time algorithm can produce a theory whose error is even close to (*i.e.*, within a particular polynomial factor of) optimal. The first (sample-complexity) results suggest reasons why theory revision can be more effective than learning from scratch, while the second (computational complexity) results explain many aspects of the standard theory revision systems, including the practice of hill-climbing to a locally-optimal theory, based on a given set of labeled queries.

Keywords theory revision, computational learning theory,
inductive logic programming, agnostic learning

---

*This paper extends the short article that appeared in the *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI95)*, Montreal, August 1995.

†Much of this work was done while I worked at Siemens Corporate Research, in Princeton, NJ. I gratefully acknowledge receiving helpful comments from Edoardo Amaldi, Mukesh Dalal, George Drastal, Adam Grove, Tom Hancock, Sheila McIlraith, Roni Khardon, Dan Roth and especially the very thorough comments from the anonymous referees.

1 Introduction

There are many fielded knowledge-based systems, ranging from expert systems and logic programs to production systems and database management systems [HRJ94]. Each such system uses its database of general task-related information (a.k.a. its "theory") to produce an answer to each given query; this can correspond to retrieving information from a database or to providing the diagnosis or repair appropriate for a given set of symptoms. Unfortunately, these responses may be incorrect if the underlying theory includes erroneous information. If we observe that some answers are incorrect (*e.g.*, if the patient does not get better, or the proposed repair does not correct the device's faults), we can then ask a human expert to supply the correct answer. We would like to use the set of these correctly-answered queries to produce a new theory that is more accurate; *i.e.*, which will make fewer mistakes, on these and other queries drawn from the same distribution.

Standard learning algorithms use *only these queries* to learn a good theory. This is wasteful in the common situation where the initial theory was already very accurate, as such learning algorithms would, in effect, have to re-learn most of the initial theory. Instead, it is often more efficient to correct that initial theory. *Theory revision* is the process of using these correctly-answered queries to *modify* the given initial theory, to produce a new, more accurate theory.

Most theory revision algorithms use a set of transformations to hill-climb through successive theories, until reaching a theory whose empirical error is (locally) optimal, based on a set of correctly-answered queries; *cf.*, [Pol85, MB88, Coh90, OM94, WP93, CS90, LDRG94]. This report addresses the obvious questions about this approach: When is theory revision a good idea — and in particular, when should it work more effectively than learning from scratch? How many correctly-answered training queries are required? And when is it possible to efficiently compute the *globally* optimal revised theory?

Section 2 first states the theory revision objective more precisely: as finding the theory with the lowest expected error from the space of theories formed by applying a sequence of transformations to a given initial theory, where each transform involves either adding or deleting either a rule or an antecedent. The next sections address two challenges to finding this best revised theory. First, as the error of a theory depends on the distribution of queries addressed, the theory that is best for one distribution may not be best for another. We therefore need to know information about the distribution to decide which theory is optimal. While such information is usually not known *a priori*, relevant information can be estimated by sampling. Section 3 considers the *sample complexity* — *i.e.*, given any values of $\epsilon, \delta > 0$, how many samples (each a query/answer pair) are required to find a theory whose error is within ϵ of the optimum (in the specified space of theories), with probability at least $1 - \delta$. We also argue that this theory revision process will often require many fewer samples than would be required to learn a good theory from scratch, and further compare the relative difficulties of *deleting* arbitrary portions of a theory, versus *adding* new parts (either new antecedents or new rules).

The second issue in finding the optimal (or even near-optimal) revised theory is the *computational* complexity of this task, once given these samples. Section 4 first observes that finding a good theory is easy if such a good theory is syntactically very close to the initial theory — which appears to often be the case, in practice. We then prove that, in general, the task of computing the optimal theory in many obvious spaces of theories is intractable, even in very simple contexts — *e.g.*, even when dealing with propositional Horn theories, or when considering with only atomic queries, or when considering only a bounded number of transformations, etc.[1] These results hold both in situations where there is a perfect Horn theory (*i.e.*, there is a Horn theory that correctly labels *all* of the instances), as well as the "agnostic" setting [KSS92], where there need not be any such theory. We then show that the "agnostic task" cannot even be approximated: *i.e.*, that no efficient algorithm can find a theory whose error is even close to (*i.e.*, within a particular small polynomial of) the optimum! We also prove that these negative results apply even when we are only generalizing, or only specializing, the initial theory. By providing efficient algorithms for other restricted variants of theory revision, we provide sharp boundaries that describe exactly when this task is guaranteed to be tractable.

These results provide several insights into the theory revision process: The sample complexity results argue that theory revision can be better than *tabula rasa* learning, as theory revision can require many fewer samples. The computational complexity results show first that theory revision can be performed efficiently if the initial theory is syntactically close to a highly-accurate theory; but then that no tractable algorithm will be able to find such a *globally* optimal theory if it is "syntactically far away" from the initial theory. Such results may help motivate the standard practice of hill-climbing to a local optimum, within the space formed using specified transformations — as this will usually find an acceptable theory, even when it is intractable to find an optimal one.

Our negative results may inspire future researchers and developers to look for other techniques to modify existing theories, perhaps by changing the underlying representation [KKS93, KR94b] or by exploiting other information that may be available, such as the assumption (if true) that each training example includes only the information required to classify that instance [GGK97].

The appendix supplies the relevant proofs. We close this section by describing related research.

Related Results: Our underlying task, of producing a theory that is as correct as possible, is the main objective of most research in **inductive learning**, including as notable instances CART [BFOS84], C4.5 [Qui92] and connectionist learning algorithms [Hin89]. While many of these systems learn descriptions based on bit vectors or simple hierarchies, our work deals with logical descriptions. Here too there is a history, dating back (at least) to Plotkin [Plo71] and Shapiro [Sha83], and including the more contemporary FOIL [Qui90] and the body of work on **inductive logic programming** (ILP) [Mug92]. However, while most of these projects begin with an "empty theory" and attempt to learn a target logic program by adding new clauses, theory revision processes work by modifying a given initial theory (which can involve both adding and deleting clauses), attempting to approximate a more general target function, which here need not even correspond to a logical theory. (See also the comparison in Section 4.4.)

[1] Throughout, we will assume that $P \neq NP$ [GJ79], which implies that any NP-hard problem is intractable. This also implies certain approximation claims, presented below.

There are several **implemented theory revision systems**. Most use essentially the same set of transformations described here — *e.g.*, AUDREY [WP93], FONTE [MB88], EITHER [OM94] and DELTA [LDRG94] each consider adding or deleting antecedents or rules. Our analysis, and results, can easily be applied to many other types of modifications — *e.g.*, specializing or generalizing antecedents [OM94], using "n-of-m rules" [BM93], or merging rules and removing chains of rules that produced incorrect results [Coh90, Coh92].[2] While these projects provide empirical evidence for the effectiveness of their specific algorithms, and deal with classification (*i.e.*, determining whether a given element or tuple is a member of some target class) rather than general derivation, our work formally addresses the complexities inherent in finding the best theory, for handling arbitrary queries.

There are several related **complexity results**: First, Cohen [Coh90] observed that the challenge of computing the *smallest* modification was intractable in a particular context; this relates to our Corollary 4.1. Second, Wilkins and Ma [WM94] show the intractability of determining the best set of rules to *delete* in the context of "weighted" rules, where a conclusion is believed if a particular function of the weights of the supporting rules exceeds a threshold. Our results show that this problem remains intractable (and is in fact, not even approximatable) even in the propositional case, when all rules have unit weight and a single rule is sufficient to establish a conclusion. Third, Valtorta and Ling [LV91, LV95] also considered the computational complexity of modifying a theory. Their analysis, however, dealt with a different type of modifications: *viz.*, adjusting various numeric weights within a given network (*e.g.*, altering the certainty factors associated with the rules), but not changing the structure by adding or deleting rules. Fourth, Mooney [Moo94] addressed the sample complexity of certain types of theory revision systems. His analysis assumes that a completely correct theory can be reached by some sequence of K transformations; our sample complexity bounds extend his by considering various specified sets of possible transformations, and by not requiring that a perfect theory be within K transformations of the starting theory. (In fact, our analysis does not even require the *existence* of a perfect theory.) We also consider the computational complexity of such processes. Finally, there are a number of results on the complexity of "pac-learning" logic programs *from scratch* (*i.e.*, of inductive logic programming, ILP); *cf.*, [Coh95b, Coh95a, Coh96, DMR92]. As mentioned above, this framework is different, as ILP systems can return any Horn theory (rather than just the theories that are syntactically close to an initial theory), and many ILP systems assume there is a Horn theory that is perfect.

There are many other frameworks that **use new observations to improve a given description of the world**. For example, many Bayesian systems use such observations to update their representations, often by adjusting the (continuous) parameters in a Dirichlet distribution within a given belief net structure [Hec95]. We, however, are making discrete changes to the structure of the Horn theory.

Similarly, **belief revision** systems [AGM85, Dal88, Gar88, KM91] take as input an

[2](1) However, we make no claims concerning the applicability of our techniques to systems like KBANN [Tow91], which use a completely different means of modifying a theory. (2) The companion paper [Gre99] considers yet other ways of modifying a theory, *viz.*, by rearranging the order of its component rules or antecedents.

initial theory T_0 and a new assertion $\langle q, +\rangle$ (resp., a new retraction $\langle r, -\rangle$) and return a new consistent theory T' that entails q (resp., does not entail r) but otherwise is "close" to T_0 [Dal88]. In general, the resulting revised theory will not depend on the syntactic structure of the initial theory — *i.e.*, if $T_1 \equiv T_2$, then the theory obtained by revising T_1 with the assertion $\langle q, +\rangle$ is equivalent to the theory obtained by revising T_2 with $\langle q, +\rangle$.

Most belief revision formalisms use only a *single labeled query* (either assertion or retraction) to modify an initial theory T_0, seeking a theory *semantically close to* T_0 that correctly does/does-not entail that query.[3] By contrast, theory revision uses a *set of labeled queries* when modifying T_0, searching within the space of theories that are *syntactically close* to T_0 for a theory with *optimal accuracy*, with respect to those queries. Notice a theory revision system (1) does not require that the revised theory be correct for any specific labeled query, and (2) may produce semantically different theories from semantically equivalent initial theories (as it may search different spaces of theories). As a final distinction, our results show that the theory revision task is difficult even if both the initial and final theories, as well as the queries, are Horn; by contrast, many belief revision frameworks deal with arbitrary CNF formulae. (Of course, the standard belief revision tasks — *e.g.*, the "counterfactual problem" — are complete for higher levels in polynomial-time hierarchy [EG92].)

Notice theory revision seeks a theory, from within the syntactically defined class of "all theories produced by applying certain syntactical modifications to an initial theory", whose performance is optimal on the semantically-defined task of "either entailing, or not entailing, certain queries". Below we present two other research corpora that similarly seek the "semantically-best" theory from within some "syntactically-defined" class.

First, there may be no class member that exhibits *perfect* performance on the task; here, for example, no Horn theory may be able to correctly classify all of the labeled queries. We still want to find the optimal member of the class. This corresponds exactly to the "**agnostic learning**" model; Kearns, Schapire and Sellie [KSS92] have shown that this task is often intractable. Our framework differs by dealing with a different class of "samples" (arbitrary queries, not bit vectors), and by having a different class of hypotheses (predicate calculus Horn theories, rather than propositional conjunctions). More significantly, we present situations where the computational task is not just intractable, but is not even approximatable.

Second, many works on "**approximations**" [BE89, SK91, DE92, GS92] and "**structural identification**" [DP92] seek a theory, of a specified syntactic form, that is semantically close to an explicitly given theory T_{target} (*i.e.*, which entails essentially the same set of propositions that T_{target} entails). As two representative results: Dechter and Pearl [DP92] agnostically seek a theory W_{opt}, of a specified syntactic form (*e.g.*, Horn or k-Horn) that is a "strongest weakening" of a given extension T_{target};[4] and Kautz, Kearns and Selman [KKS95] provide

[3]While the work on "iterated revision" [Bou93, GPS94, FL94, DP94] also considers more than a single assertion, it usually deals with a *sequence* of assertions, where each new assertion must be incorporated, as it arrives. Afterwards, it is no longer distinguished from any other information in the current theory (but see [FH96]). We, however, consider the assertions as a *set*, which is seen at once, and whose elements need not all be incorporated.

[4](i) A k-Horn theory is a Horn theory, defined below, whose clauses each contain at most k literals. (ii) A theory W_{opt} is a "strongest weakening" of the theory T_{target} if $T_{target} \models W_{opt}$ and there are no other theories W' of this syntactic form strictly between T_{target} and W_{opt}; *i.e.*, $T_{target} \models W' \models W_{opt}$ implies

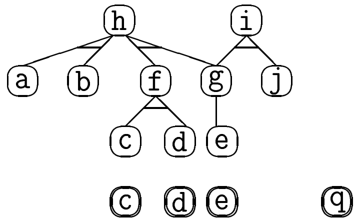
an efficient randomized algorithm that, given an extension T_{target}, agnostically produces a Horn theory W that is usually a strong weakening of T_{target} (*i.e.*, with high probability, W's models include all models of the original T_{target}, and at most a small number of others). Our results differ as (1) our semantic task involves accommodating a set of both positively- and negatively- labeled queries, which loosely resembles a conjunction of (Horn) disjunctions, rather than a complete extension (*i.e.*, a CNF rather than a DNF formula); (2) we seek the theory that minimizes the two-sided error (*i.e.*, our set of positively-labeled queries does not necessarily entail our revised theory W); and (3) we consider only (Horn) theories within a specified space of theories, which is implicitly defined by the syntactic transformations applied to a given theory. (Hence, our space is typically smaller than the space of all Horn theories.)

2 Framework

We define a "(Horn) theory" as a conjunction of (propositional or first order) Horn clauses, where each clause is a disjunction of literals, at most one of which is positive. Borrowing from [Lev84, DP91], we also view a theory T as a function that maps each query to its proposed answer; hence, T: $\mathcal{Q} \mapsto \mathcal{A}$, where $\mathcal{Q}$ is a (possibly infinite) set of Horn queries, and $\mathcal{A} = \{$ `Yes`, `No` $\}$ is the set of possible answers.[5] Hence, given

T_1 =

```
h :- a, b.
h :- f, g.
i :- g, j.
f :- c, d.
g :- e.
c.  d.  e.  q.
```



(1)

T_1(`h`) = `Yes`, T_1(`i`) = `No` and T_1(`i :- e,j.`) = `Yes`. We will later use T_2, the theory that differs from T_1 only by excluding the "`g :- e`" rule.

While the non-atomic queries may seem unusual at first, they are actually quite common. For example, a medical expert system typically collects relevant data $\{\, \mathtt{f}_1\mathtt{(p)}, \ldots, \mathtt{f}_n\mathtt{(p)} \,\}$ about an individual patient `p`, then determines whether `p` has some specific disease $\mathtt{disease}_i$; *i.e.*, if $T \cup \{\, \mathtt{f}_1\mathtt{(p)}, \ldots, \mathtt{f}_n\mathtt{(p)} \,\} \models \mathtt{disease}_i\mathtt{(p)}$, where T is the expert system's initial theory that contains general information about diseases, etc. Notice this entailment condition holds iff $T \models \neg\mathtt{f}_1\mathtt{(p)} \vee \ldots \vee \neg\mathtt{f}_n\mathtt{(p)} \vee \mathtt{disease}_i\mathtt{(p)}$; *i.e.*, iff the Horn query "$\mathtt{disease}_i\mathtt{(p)}$:- $\mathtt{f}_1\mathtt{(p)}$, ..., $\mathtt{f}_n\mathtt{(p)}$" follows from the initial theory. Such queries also clearly connect to the standard classification task used within Machine Learning: given a complete assignment of the attributes, determine whether class membership is entailed. Here, how-

$W' \equiv W_{opt}$. (*iii*) An "extension" is a DNF formula, whose conjuncts are each a complete assignment to the variables.

[5](1) The "`No`" answer actually means the theory did not find an answer. (2) To simplify our presentation, the main body of this paper will deal only with propositional logic; the end of this section discusses the extensions needed to deal with predicate calculus.

ever, we do not necessary deal with a single complete assignment — *e.g.*, a theory entails $f_1\&f_2 \Rightarrow d$ only if all 2^{n-2} instances $\langle 1, 1,\ 0, \ldots, 0\rangle$ through $\langle 1, 1,\ 1, \ldots, 1\rangle$ are all positive instances of d (*i.e.*, if each of $\langle f_1 = 1,\ f_2 = 1,\ f_3 = 0,\ \ldots,\ f_n = 0,\ d = 1\rangle$ through $\langle f_1 = 1,\ f_2 = 1,\ f_3 = 1,\ \ldots,\ f_n = 1,\ d = 1\rangle$ is a model). Moreover, our model can allow many different classes (*e.g.*, both $f_1\&f_2 \Rightarrow d$ specifying positive instances of d, and $f_7\&f_{19} \Rightarrow e$ specifying positive instances of e, etc.). Finally, these "classes" can be interrelated (via "chaining"); *e.g.*, we can have $f_1\&f_2 \Rightarrow d$, and also $f_7\&d \Rightarrow e$, etc. See also "entailment queries" [FP93, KR94a].

For now, we will assume there is a single correct answer to each question, and represent it using the "target function" (or "real-world oracle") $O\colon \mathcal{Q} \mapsto \mathcal{A}$. Here, perhaps, $O(\,\mathtt{h}\,) = \mathtt{No}$, meaning that "h" should not hold. We will consider two classes of target functions: each member of $\mathcal{O}_{Horn}$ corresponds to a Horn theory, and each member of $\mathcal{O}_{Det}$ corresponds to a deterministic mapping of queries to answers (*e.g.*, perhaps $O(\,\mathtt{a}\,) = \mathtt{Yes}$, $O(\,\mathtt{b\ :-\ a}\,) = \mathtt{Yes}$, and $O(\,\mathtt{b}\,) = \mathtt{No}$). While the first class of target function is more standard in the Inductive Logic Programming literature (as it guarantees there is a Horn theory capable of correctly classifying all of the training data), it is not as realistic for the real-world task of finding the best possible theory to explain some observed data, as real-world data may in fact be noisy, or correspond to a situation where there is no perfect theory. This is the same motivation that gave rise to the study of "agnostic learning" [KSS92].

In general, our goal is to find a theory that is as close to the target function $O(\,\cdot\,)$ as possible. To quantify this, we first define the "error function" $\mathrm{err}(\cdot,\,\cdot)$ where $\mathrm{err}(\mathrm{T},\,q)$ is the error of the answer the theory T returned for the query q:

$$\mathrm{err}(\mathrm{T},\,q) \quad \stackrel{def}{=} \quad \begin{cases} 0 & \text{if } \mathrm{T}(q) = O(\,q\,) \\ 1 & \text{otherwise} \end{cases}$$

(Notice $\mathrm{err}(\mathrm{T},\,\cdot)$ implicitly depends on the target function $O(\,\cdot\,)$.) Hence, as $O(\,\mathtt{h}\,) = \mathtt{No}$, $\mathrm{err}(\mathrm{T}_2,\ \text{“}\mathtt{h}\text{”}) = 0$ as T_2 provides the correct answer while $\mathrm{err}(\mathrm{T}_1,\ \text{“}\mathtt{h}\text{”}) = 1$ as T_1 returns the wrong answer.

This $\mathrm{err}(\mathrm{T},\,\cdot)$ function measures T's error for a single query. In general, our theories must deal with a range of queries. We model this using a stationary, but unknown, probability function $Pr : \mathcal{Q} \mapsto [0, 1]$, where $Pr(q)$ is the probability that the query q will be posed. Given this distribution, we can compute the "expected error" of a theory, T:

$$\textsc{Err}(\,\mathrm{T}\,) \quad = \quad E[\,\mathrm{err}(\mathrm{T},\,q)\,] \quad = \quad \sum_{q\in\mathcal{Q}} Pr(q) \times \mathrm{err}(\mathrm{T},\,q)\ .$$

We will consider various sets of possible theories, $\mathcal{T} = \{\mathrm{T}_i\}$, where each such $\mathcal{T}$ contains the set of theories formed by applying various sequences of transformations to a given initial theory; see Section 2.1 below. Our challenge is to identify the theory $\mathrm{T}_{opt} \in \mathcal{T}$ whose expected error is minimal; *i.e.*,

$$\forall\,\mathrm{T} \in \mathcal{T}\colon\ \textsc{Err}(\,\mathrm{T}_{opt}\,)\ \leq\ \textsc{Err}(\,\mathrm{T}\,)\ . \tag{2}$$

The next two sections address two challenges in finding such optimal theories: First, the optimal theory depends on the distribution of queries. While this is not known initially, relevant information can be estimated by observing a set of samples (each a query/answer pair), drawn from that distribution. Section 3 quantifies how the number of samples required to obtain the information needed to identify a good $\mathrm{T}^* \in \mathcal{T}$ (with high probability) depends on the space of theories $\mathcal{T}$ being searched; it then provides the sample complexity for various spaces.

We are then left with the challenge of computing the best theory, once given these samples. Section 4 addresses the computational complexity of this process, showing that the task is not just intractable,[6] but it is also not approximatable — *i.e.*, no efficient algorithm can even find a theory whose expected error is even close (in a sense defined below) to the optimal value.

The rest of this section describes the transformations used to define the various spaces of theories, and then discusses the extensions needed to handle stochastic oracles, predicate calculus theories and queries, and non-categorical responses.

2.1 Standard Transformations

Standard theory revision algorithms modify the given initial theory by applying a sequence of zero or more transformations. We consider four classes of transformations:

$$\begin{aligned}
\Upsilon_{DR} &= \{\, \tau^{DR}\colon \mathcal{T} \mapsto \mathcal{T} \mid \tau^{DR}(\mathrm{T}) \text{ deletes an existing rule from T } \} \\
\Upsilon_{AR} &= \{\, \tau^{AR}\colon \mathcal{T} \mapsto \mathcal{T} \mid \tau^{AR}(\mathrm{T}) \text{ adds a new rule to T } \} \\
\Upsilon_{DA} &= \{\, \tau^{DA}\colon \mathcal{T} \mapsto \mathcal{T} \mid \tau^{DA}(\mathrm{T}) \text{ deletes an existing antecedent from an existing rule in T } \} \\
\Upsilon_{AA} &= \{\, \tau^{AA}\colon \mathcal{T} \mapsto \mathcal{T} \mid \tau^{AA}(\mathrm{T}) \text{ adds a new antecedent to an existing rule in T } \}
\end{aligned}$$

We let $\Upsilon = \Upsilon_{DR} \cup \Upsilon_{AR} \cup \Upsilon_{DA} \cup \Upsilon_{AA}$ be the set of all transformations, and let $\Upsilon^\infty[\mathrm{T}_0] = \{\Upsilon(\mathrm{T}_0) \mid \upsilon \in \Upsilon^\infty\}$ be the theories formed by applying some sequence of theory-to-theory transformations $\upsilon = \tau_1 \circ \tau_2 \circ \ldots \circ \tau_\ell \in \Upsilon^\infty$ to the given initial theory T_0. (Table 1 provides a concise reference for the notation used in this paper.)

The cost function $c : \Upsilon \mapsto \mathcal{N}$ maps each transformation to the number of symbols it adds to, or deletes from, T to form $\tau(\mathrm{T})$; we further let $c(\upsilon) = c(\tau_1) + c(\tau_2) + \ldots + c(\tau_\ell)$ be the cost of the sequence of transformations $\upsilon = \tau_1 \circ \tau_2 \circ \ldots \circ \tau_\ell$. In the propositional case, $c(\tau_{AA}) = c(\tau_{DA}) = 1$ for each transformation that either adds or deletes an antecedent; and $c(\tau_\rho^{AR}) = c(\tau_\rho^{DR}) = |\rho|$ for each add-rule (resp., delete-rule) transformation that adds (resp., deletes) the rule ρ, which has 1 conclusion and $|\rho| - 1$ antecedent literals. In predicate calculus, these costs are more complicated, as they depend on the number of symbols used in all of the affected literals.

[6]A naïve way of evaluating $\mathrm{err}(\mathrm{T}, q)$ would require computing $\mathrm{T}(q)$. As this could require proving an arbitrary theorem, this computation alone can be computationally intractable, if not undecidable. Our results show that the task of finding the optimal theory is intractable *even given a polynomial-time oracle that performs these arbitrary derivations*. Of course, as we are considering only Horn theories, these computations are guaranteed to be polynomial-time in the propositional case [BCH90].

T	= a theory; *i.e.*, a set of Horn clauses
$\mathcal{L}$	= the language used
Set of transformations Υ_χ *that map a theory* T *to a set of new theories* $\Upsilon_\chi(\mathrm{T})$	
$\Upsilon_{AR}(\mathrm{T})$	$= \{\ \tau^{AR} \mid \tau^{AR}$ adds a new clause to a theory $\mathrm{T}\ \}$
$\Upsilon_{DR}(\mathrm{T})$	$= \{\ \tau^{DR} \mid \tau^{DR}$ deletes an existing clauses from a theory $\mathrm{T}\ \}$
$\Upsilon_{AA}(\mathrm{T})$	$= \{\ \tau^{AA} \mid \tau^{AA}$ adds a new antecedent to an existing rule in $\mathrm{T}\ \}$
$\Upsilon_{DA}(\mathrm{T})$	$= \{\ \tau^{DA} \mid \tau^{DA}$ deletes an existing antecedent from an existing rule in $\mathrm{T}\ \}$
Sequences of transformations:	
$\Upsilon^{+A=k_1,\ +R=k_2,\ -A=k_3,\ -R=k_4}(\mathrm{T})$	= theories formed by $\left\{\begin{array}{l}\text{adding} \le k_1 \text{ new antecedents to existing rules in T}\\ \text{adding} \le k_2 \text{ new rules to T}\\ \text{deleting} \le k_3 \text{ existing antecedents from existing rules in T}\\ \text{deleting} \le k_4 \text{ existing rules from T}\end{array}\right\}$
Notes	$*$ each $k_i = k_i(\lvert\mathrm{T}\rvert)$ may be a function of (the size of) the theory considered T
	$*\ \Upsilon^\infty \equiv \Upsilon^{+A=\infty,\ +R=\infty,\ -A=\infty,\ -R=\infty}$
	$*\ \Upsilon^{+R} \equiv \Upsilon^{+A=0,\ +R=\infty,\ -A=0,\ -R=0}$, etc.
Decision Problem, for any $\Upsilon^\dagger = \Upsilon^{+A=k_1,\ +R=k_2,\ -A=k_3,\ -R=k_4}$ *that maps a theory to a set of theories:*	
$\textsc{ThRev}[\Upsilon^\dagger]$	= Decision problem defined in Definition 1
$\textsc{ThRev}_{Perf}[\Upsilon^\dagger]$	$= \textsc{ThRev}[\Upsilon]$ with $p = 1$
Gen'l: $\textsc{ThRev}_{Opt}[\Upsilon^\dagger]$	allows arbitrary p
$\textsc{ThRev}_{Prop}[\Upsilon^\dagger]$	$= \textsc{ThRev}[\Upsilon^\dagger]$ with propositional theories
Gen'l: $\textsc{ThRev}_{PC}[\Upsilon^\dagger]$	allows predicate calculus
$\textsc{ThRev}_{Atom}[\Upsilon^\dagger]$	$= \textsc{ThRev}_{Atom}[\Upsilon^\dagger]$ with atomic queries
Gen'l$_1$: $\textsc{ThRev}_{Horn}[\Upsilon^\dagger]$	allows Horn queries
Gen'l$_2$: $\textsc{ThRev}_{Disj}[\Upsilon^\dagger]$	allows arbitrary disjunctive queries
Optimization Problem, for any $\Upsilon^\dagger$ *that maps a theory to a set of theories:*	
$\textsc{MinThRev}_\rho[\Upsilon^\dagger]$	= minimization problem, with "constraints" $\rho \subset \{\text{Perf, Prop, Atom, } \ldots\}$ (see above)
MinPerf$[\textsc{MinThRev}_\rho[\Upsilon^\dagger]](\,B, x\,)$	= error score of algorithm B on instance x (see Equation 5)

Table 1: Definitions and Notation

We use this cost function to define "K-bounded sequences"

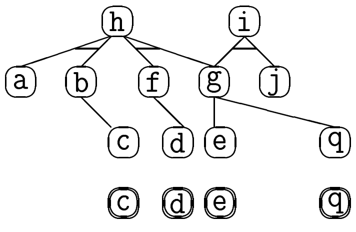
$$\Upsilon^K \quad = \quad \{\, \upsilon = \tau_1 \circ \tau_2 \circ \ldots \circ \tau_\ell \mid \tau_i \in \Upsilon \;\&\; c(\upsilon) \le K \,\}$$

whose members $\upsilon = \tau_1 \circ \tau_2 \circ \ldots \circ \tau_\ell \ \in \ \Upsilon^K$ are sequences of transformations whose total cost $c(\upsilon)$ is at most K. In some situations, we will allow the number of transformations to grow with the size of the theory; here, we will abuse notation by viewing K as a function $K : \mathcal{T} \mapsto \mathcal{N}$, which returns an integer value as a function of the input (size of the) initial theory.

To illustrate these transformations, consider the T_1 theory from Equation 1. The $\tau^{DR}_{g:-e}$ delete-rule transformation will remove the "`g :- e.`" rule, reducing T_1 to a new theory with only 8 clauses (4 rules and 4 atomic literals), called T_2 above. Another delete-rule, τ^{DR}_{d} removes the atomic "`d.`" clause. The $\tau^{DA}_{h:-f,g;\ -g}$ delete-antecedent transformation removes the "`g`" antecedent from the "`h :- f, g.`" rule; an alternative delete-antecedent transformation, $\tau^{DA}_{h:-f,g;\ -f}$, removes the "`f`" from that rule. Of course, yet other delete-antecedent transformations modify other rules. The add-antecedent transformation $\tau^{AA}_{g:-e;\ +q}$ adds the literal "`q`" to the "`g :- e.`" rule, forming "`g :- e, q.`", at cost $c(\tau^{AA}_{g:-e;\ +q}) = 1$.[7] A second add-antecedent transformation $\tau^{AA}_{g:-e,q;\ +d}$ could then add the literal "`d`" to this rule, forming the "`g :- e, q, d.`"; yet another $\tau^{AA}_{i:-g,j;\ +a}$ adds the literal "`a`" to "`i :- g, j.`" to form "`i :- g, j, a.`", etc. Finally, the add-rule transformations add in new clauses: $\tau^{AR}_{b:-f}$ adds "`b :- f.`", leading to the 10-element theory $T_3 = T_1 \cup \{$ `b :- f.`$\}$. Its cost is $c(\tau^{AR}_{b:-f}) = 2$. A different add-rule τ^{AR}_{j} adds the atomic clause "`j.`" (at cost 1), etc.

As expected, a "transformation sequence" is a sequence of transformations; so applying the 3-element sequence $\upsilon = \tau^{AR}_{b:-c} \circ \tau^{AA}_{g:-e;\ +q} \circ \tau^{DA}_{f:-c,d;\ -c}$ with total cost $c(\upsilon) = c(\tau^{AR}_{b:-c}) + c(\tau^{AA}_{g:-e;\ +q}) + c(\tau^{DA}_{f:-c,d;\ -c}) = 2 + 1 + 1 = 4$, will transform T_1 into $\mathrm{T}_4 = \upsilon(\mathrm{T}_1) = \tau^{AR}_{b:-c}(\,\tau^{AA}_{g:-e;\ +q}(\,\tau^{DA}_{f:-c,d;\ -c}(\,\mathrm{T}_1\,)\,)\,)$ which is a theory with 10 clauses that differs from T_1 by including the clause "`f:-d`" rather than "`f:-c,d`", including the clause "`g:-e,q`" rather than "`g:-e`", and by including an extra clause "`b:-c`":

$$\mathrm{T}_4 \quad = \quad \boxed{\begin{array}{l} \texttt{h :- a, b.} \\ \texttt{h :- f, g.} \\ \texttt{i :- g, j.} \\ \texttt{f :- d.} \\ \texttt{g :- e, q.} \\ \texttt{b :- c.} \\ \texttt{c.\ \ d.\ \ e.\ \ q.} \end{array}}$$



Of course, one transformation in a sequence can modify the clause affected by an earlier transformation in the same sequence; *e.g.*, $\upsilon_2 = \tau^{DR}_{b:-f} \circ \tau^{AR}_{b:-f}$ is a no-op, in that $\upsilon_2(\mathrm{T}) \equiv \mathrm{T}$, (provided "`b :- f`" $\notin \mathrm{T}$), albeit at a cost of $c(\tau^{DR}_{b:-f} \circ \tau^{AR}_{b:-f}) = 6$.

[7] As we are dealing with a pure version of logic programs, and seeking all answers to each query, the order of these antecedents will not matter. Similarly, the order of rules is also irrelevant in this model. The companion paper [Gre99] considers alternative models in which these orders can matter.

Finally, we will also consider various other spaces of transformations, of the form

$$\Upsilon^{+A=k_1,\ +R=k_2,\ -A=k_3,\ -R=k_4} \quad = \quad \left\{ \upsilon = \tau_1 \circ \tau_2 \circ \ldots \circ \tau_\ell \;\middle|\; \tau_i \in \Upsilon \;\&\; \begin{array}{l} \sum_{\tau \in \Upsilon_{AA}} c(\tau) \leq k_1 \;\& \\ \sum_{\tau \in \Upsilon_{AR}} c(\tau) \leq k_2 \;\& \\ \sum_{\tau \in \Upsilon_{DA}} c(\tau) \leq k_3 \;\& \\ \sum_{\tau \in \Upsilon_{DR}} c(\tau) \leq k_4 \end{array} \right\}$$

where each integer $k_i \in \mathcal{N}$ or $k_i = \infty$ is a bound on the sum of the costs of the transformations of type Υ_i. We will also abbreviate the superscripts by omitting each term of the form "$+A = 0$", and replacing each "$+R = \infty$" by simply "$+R$"; hence $\Upsilon^{+A=0,\ +R=\infty,\ -A=7,\ -R=\infty}$ can be written $\Upsilon^{+R,\ -A=7,\ -R}$. As above, we will sometimes let these k_i values be functions of (the size of) the given theory.

2.2 Extensions

All of the following theorems will hold even if we use a stochastic real-world oracle, encoded as $O' : \mathcal{Q} \times \mathcal{A} \mapsto [0, 1]$, where the correct answer to the query q is a with probability $O'(q, a)$. This allows us to model the situation where, for a particular set of observations, different repairs are appropriate at different times; this could happen, for example, if the correct repair depends on some unobserved variables as well as the observations; see [KS90]. Notice here that $\mathrm{err}(\mathrm{T},\, q) \;=\; 1 - O'(q, \mathrm{T}(q))$; and that our deterministic oracle is a special case of this, where $O'(q, a_q) = 1$ for a single $a_q \in \mathcal{A}$ and $O'(q, a) = 0$ for all $a \neq a_q$.

To handle predicate calculus expressions, we may have to consider answers of the form $\{\texttt{Yes}[\ X_i/\texttt{v}_i\ ]\}$, where the expression within each $\texttt{Yes}[\cdot]$ is a binding list of the free variables, corresponds to a single answer to the query. For example, given the theory[8]

$$\mathrm{T}_{pc} \quad = \quad \left\{ \begin{array}{ll} \texttt{tall(john).} & \texttt{short(fred).} \\ \texttt{rich(john).} & \texttt{rich(fred).} \\ \multicolumn{2}{l}{\texttt{eligible(X) :- tall(X), rich(X).}} \end{array} \right\}$$

the query `short(Y)` will return $\mathrm{T}_{pc}(\texttt{short(Y)}) = \{\,\texttt{Yes}[\texttt{Y/fred}]\,\}$, the query `rich(Z)` will return the pair of answers $\mathrm{T}_{pc}(\texttt{rich(Z)}) = \{\texttt{Yes}[\,\texttt{Z/john}\,],\ \texttt{Yes}[\,\texttt{Z/fred}\,]\}$, and $\mathrm{T}_{pc}(\texttt{eligible(A)}) = \{\ \texttt{Yes}[\texttt{A/john}]\ \}$. As $O(\,\cdot\,)$ and $\mathrm{T}(\cdot)$ may each return a *set* of answers to each query, we therefore define T's accuracy score (which is $1 - \textsc{Err}(\,\mathrm{T}\,)$) as the ratio of the number of correct answers, to all answers from both $O(\,q\,)$ and $T(q)$: $\mathrm{err}(\mathrm{T},\, q) \;=\; 1 - \frac{|O(\,q\,) \cap T(q)|}{|O(\,q\,) \cup T(q)|} \;\in\; [0, 1]$. We will use $\texttt{Yes}[X/\,?]$ to indicate that there is an instantiation that is satisfied, but the particular value of that instantiation is not important. (This corresponds to an "existential question" [RBK88].) All of the results in this paper hold even when considering only non-recursive theories; and all computational results hold even for Datalog (*i.e.*, "function-free") theories.

As a related extension, we can also allow our theories to return $\mathrm{T}(q) \;= \texttt{IDK}$, which stands for the non-categorical answer "I Don't Know"; here perhaps $\mathrm{err}(\mathrm{T},\, q) = 1/2$. Finally, there are obvious ways of extending our analysis to allow a more comprehensive error function

[8] Following Prolog's conventions, we will capitalize each variable, as in the "X" above.

err(T, $\cdot$) that could apply different rewards and penalties for different queries (*e.g.*, to permit different penalties for incorrectly identifying the location of a salt-shaker, versus the location of a stalking tiger). As these extensions lead to strictly more general situations, our underlying task (of identifying the optimal theory) remains as difficult; *e.g.*, it remains computationally intractable in general.

3 Sample Complexity

As mentioned above, a theory revision process seeks a revision of the initial theory (from the allowed set of revisions) with the minimum possible expected error, *over the distribution of queries*. While this distribution is unknown, we can use a set of labeled samples $S = \{\langle q_i, O(q_i)\rangle\}$ to (implicitly) obtain the "empirical error" of each of the theories $\mathrm{T}_j \in \mathcal{T}$, written

$$\overline{\mathrm{E{\scriptsize RR}}_S}(\mathrm{T}_j) \quad = \quad \frac{1}{|S|} \sum_{\langle q_i, O(q_i)\rangle \in S} \mathrm{err}(\mathrm{T}_j,\, q_i) \tag{3}$$

and then select the theory whose empirical error is smallest; *i.e.*, the T^* in $\mathcal{T}$ such that $\forall \mathrm{T}_i \in \mathcal{T}, \overline{\mathrm{E{\scriptsize RR}}_S}(\mathrm{T}^*) \leq \overline{\mathrm{E{\scriptsize RR}}_S}(\mathrm{T}_i)$.

While this theory T^* does have the least error on the training samples S, it may not be the one which has the least error *over the entire distribution of queries*; *i.e.*, we do not know that $\mathrm{T}^* = \mathrm{T}_{opt}$, or even that $\mathrm{E{\scriptsize RR}}(\mathrm{T}^*) \approx \mathrm{E{\scriptsize RR}}(\mathrm{T}_{opt})$, using the T_{opt} defined in Equation 2. Basically, this is because we do not know that $\overline{\mathrm{E{\scriptsize RR}}_S}(\mathrm{T}^*)$ will be close to $\mathrm{E{\scriptsize RR}}(\mathrm{T}^*)$, nor that $\overline{\mathrm{E{\scriptsize RR}}_S}(\mathrm{T}_{opt})$ will be close to $\mathrm{E{\scriptsize RR}}(\mathrm{T}_{opt})$.

We can however use statistical methods to quantify our confidence in the closeness of these estimates, as a function of the number of samples used $|S|$ and the size of the space of possible theories, $|\mathcal{T}|$. The following theorem provides an upper bound on the number of samples required to be at least $1 - \delta$ confident that the true error of empirically-optimal theory T^* will be within ϵ of the truly best theory of $\mathcal{T}$, T_{opt}:

Theorem 1 (from [Vap82, Theorem 6.2]) *Given a class of theories $\mathcal{T}$, and $\epsilon, \delta > 0$, let $\mathrm{T}^* \in \mathcal{T}$ be the theory with the smallest empirical error after*

$$m_{upper}(\mathcal{T}, \epsilon, \delta) \quad = \quad \left\lceil \frac{2}{\epsilon^2} \ln\left(\frac{|\mathcal{T}|}{\delta}\right) \right\rceil$$

labeled queries, drawn independently from a stationary distribution. Then, with probability at least $1 - \delta$, the expected error of T^* will be within ϵ of the optimal theory in $\mathcal{T}$; i.e., $Pr[\,\mathrm{E{\scriptsize RR}}(\mathrm{T}^*) \leq \mathrm{E{\scriptsize RR}}(\mathrm{T}_{opt}) + \epsilon\,] \geq 1 - \delta$, using the T_{opt} from Equation 2.

Notice this means a polynomial number of samples is sufficient to identify an ϵ-good theory from $\mathcal{T}$ with probability at least $1 - \delta$, whenever $\ln(|\mathcal{T}|)$ is polynomial in the relevant parameters.[9] Of course, this bound will also depend on $|\mathcal{L}|$, the number of symbols in the

[9] Note that even fewer samples are required to reliably determine whether there is a theory in the given space of theories $\mathcal{T}$ whose error is within ϵ of a *given* quantity, say 0%; see [Vap82, Theorem 6.1].

language of the theories, $\mathcal{L}$. (We are not considering new symbols; *i.e.*, this set $\mathcal{L}$ is fixed.) This boundedness property is true for $\mathcal{T} = \Upsilon^K[\mathrm{T}_0]$:

Observation 1 $\ln(|\Upsilon^K[\mathrm{T}_0]|) \le K \times [\ln(|\mathcal{L}|) + 2\ln(|\mathrm{T}_0| + K)]$, *where $\mathcal{L}$ is the set of symbols in the language of the theories.*

This observation gives some insights into why theory revision may be useful. An ILP (or *tabula rasa*) learning system, which starts with no "approximation" of the target theory, may require a great many samples to collect the information required to identify the optimal theory T_{opt}; even in the propositional case, $\Omega(M)$ labeled queries are required to reliably build a M-clause theory from scratch (see Theorem 2 below). A theory revision system, however, can exploit the initial theory T_0. In many situations, this T_0 will be syntactically close to the optimal T_{opt} (or at least to a theory T_* whose error is nearly optimal), in the sense that T_{opt} (or T_*) will be in $\Upsilon^K[\mathrm{T}_0]$ for some small K. In particular, when $K \ll M = |\mathrm{T}_{opt}|$, the number of samples required to "transform" T_0 to T_{opt} will be *much less* than would be required to learn T_{opt} from scratch.

(As another way to look at this: A small number of samples is usually sufficient to identify the best theory within a small set of theories. In the theory revision framework, this set corresponds to the theories that are syntactically close to the initial theory, which (in practice) tends to be fairly accurate. As syntactically similar theories often tend to have similar accuracies,[10] this space may include many very accurate theories, and so perhaps the optimal theory. By contrast, an ILP system is biased to find the best small theory, as it prefers theories that are syntactically close to the empty theory. Unfortunately, even the best such theory may not be very accurate.)

We close this section by describing alternative spaces of transformations, and then providing lower bounds on the required number of samples. These comments provide a theoretical justification for the intuition that it takes more evidence to justify *adding* a new part to a theory, than is required to *delete* an existing part. Note that several theory revision systems, including KRUST [CS90], incorporate this bias.

Alternative Spaces: The set $\Upsilon^{+A=K,+R=K,-A,-R}$ strictly extends Υ^K by including transformation-sequences that can delete an *unrestricted* number of symbols, as well as add up to $K+K$ symbols. Observe that $\ln(|\Upsilon^{+A=K,\ +R=K,\ -A,\ -R}[\mathrm{T}]|)$ is still polynomial is $|\mathcal{L}|$ and $|\mathrm{T}|$, meaning it can potentially be learned using a polynomial number of samples.

By contrast, consider $\Upsilon^{+A,\ +R,\ -A=K,\ -R=K}$, whose transformation-sequences can delete only a bounded number ($2K$) of symbols, but can add an unrestricted number. Here, if $\mathcal{L}$ is non-trivial (*i.e.*, includes at least one constant c, one function f and one relation symbol r), then $\Upsilon^{+A,\ +R,\ -A=K,\ -R=K}[\mathrm{T}]$ (and hence $\ln(|\Upsilon^{+A,\ +R,\ -A=K,\ -R=K}[\mathrm{T}]|)$) is infinite. (To see this: Let $\mathrm{T}_0 = \{\}$ be the empty theory and observe $\Upsilon^{+A,\ +R,\ -A=K,\ -R=K}[\mathrm{T}_0] \supset \Upsilon^{+R}[\mathrm{T}_0]$, and so includes all 2^ω subsets of the countably infinite $\omega = \{\ r(c),\ r(f(c)),\ r(f(f(c))),\ \ldots\ \}$.)

The following comment provides a stronger claim, showing that we cannot supply an *a priori* bound on the number of samples required to learn the best theory in the $\Upsilon^{+R}[\mathrm{T}_0]$ set, much less $\Upsilon^{+A,\ +R,\ -A=K,\ -R=K}[\mathrm{T}_0]$ or $\Upsilon^\infty[\mathrm{T}_0]$.

[10]Of course, this is just a heuristic that does not always hold.

Lower Bounds: To obtain a *lower bound* on the number of samples required to be at least $1-\delta$ confident of finding a theory within ϵ of optimal, we can use

Theorem 2 (Sample Complexity [BEHW89, EH89]) *Given a class of theories $\mathcal{T}$ and values $\epsilon, \delta > 0$, let $\mathrm{T}^* \in \mathcal{T}$ be any theory with empirical error of $\overline{\mathrm{Err}_S}(\mathrm{T}^*) = 0$ based on m samples drawn independently from a stationary distribution over the query class $\mathcal{Q}$. To be at least $1-\delta$ confident that $\mathrm{Err}(\mathrm{T}^*)$ is at most ϵ (i.e., that $Pr[\,\mathrm{Err}(\mathrm{T}^*) \le \epsilon\,] \ge 1-\delta$, where this distribution is the product distribution over sets of samples drawn by any revision algorithm), we need at least*

$$m = m_{lower}(\mathcal{T}, \epsilon, \delta) \quad \ge \quad \max\left\{ \frac{1-\epsilon}{\epsilon}\log\frac{1}{\delta},\ \frac{VCdim_{\mathcal{Q}}(\mathcal{T})-1}{2e\epsilon} \right\} \tag{4}$$

samples, where $VCdim_{\mathcal{Q}}(\mathcal{T})$ is the Vapnik-Chervonenkis Dimension of the set $\mathcal{T}$, with respect to the query set $\mathcal{Q}$ (defined below).

(Notice this lower bound assumes there is a theory in $\mathcal{T}$ whose error is 0; if not, then we will require yet more samples to find $\mathcal{T}$'s optimal theory.)

Here, $\mathrm{VCdim}_{\mathcal{Q}}(\mathcal{T})$ is the largest number of queries from $\mathcal{Q}$ that can "shatter" a subset of $\mathcal{T}$ — *i.e.*, the largest number of queries $\{q_1, \ldots, q_n\} \subseteq \mathcal{Q}$ such that, for each of the 2^n possible answer-lists $\langle a_1, \ldots, a_n\rangle \in \{\texttt{Yes}, \texttt{No}\}^n$, there is a theory in $\mathcal{T}$ that produces exactly the answers, $\mathrm{T}(q_i) = a_i$. That is, $\mathcal{T}$ must include a theory $\mathrm{T}_{N\ldots N}$ that returns `No` to each query (*i.e.*, $\mathrm{T}_{N\ldots N}(q_i) = \texttt{No}$ for $i = 1..n$), another $\mathrm{T}_{N\ldots N,Y} \in \mathcal{T}$ that returns `No` to all but the final q_n, a third $\mathrm{T}_{N\ldots Y,N} \in \mathcal{T}$ that returns `No` to all but q_{n-1}, a fourth that $\ldots$, and a 2^nth $\mathrm{T}_{Y\ldots Y} \in \mathcal{T}$ that returns `Yes` to all n queries. If there is no largest such n, we say that $\mathrm{VCdim}_{\mathcal{Q}}(\mathcal{T})$ is infinite.[11]

Clearly the set of theories $\Upsilon^{+R}[\{\}]$ has infinite VC-dimension (provided $|\mathcal{L}|$ is non-trivial) as it can shatter a set of queries of size n, for any n: Consider the n propositions

$$Q_n = \{\ r(c),\ r(f(c)),\ r(f(f(c))),\ \ldots,\ r(\underbrace{f(\ldots(f}_{n-1}(c))\ldots))\ \},$$

and note that $\Upsilon^{+R}[\{\}]$ includes a theory that contains, and hence entails exactly, each subset of Q_n. This means, for each of the 2^n possible answer-lists $\langle a_1, \ldots, a_n\rangle \in \{\texttt{Yes}, \texttt{No}\}^n$, $\Upsilon^{+R}[\{\}]$ includes a theory that is perfect for

$$\{\ \langle r(c), a_1\rangle,\ \langle r(f(c)), a_2\rangle,\ \langle r(f(f(c))), a_3\rangle,\ \ldots,\ \langle r(\underbrace{f(\ldots(f}_{n-1}(c))\ldots)), a_n\rangle\ \}\ .$$

We can also produce a set of theories with an exponentially large VC-dimension by simply adding new antecedents:

Observation 2 *There is a class of theories $\{\mathrm{T}_n\}$ where each $|\mathrm{T}_n| = O(n)$, such that the VC-dimension of the theory set $\Upsilon^{+A}[\mathrm{T}_n]$, formed by applying add-antecedent transformations, is*

[11] Readers wishing to learn yet more about "Vapnik-Chervonenkis Dimension" are referred to [Hau88].

exponential in n; i.e., where VCdim$_{\mathcal{Q}}(\,\Upsilon^{+A}[\mathrm{T}_n]\,) \geq 2^n$. *This holds even if all of the queries are atomic, they all correspond to simple instantiations of the same relation, and there is a Horn theory that labels this set perfectly.*

By contrast, using the observations that $|\Upsilon^{-R}[\mathrm{T}]| \leq 2^{|T|}$ and $\mathrm{VCdim}_{\mathcal{Q}}(\,\mathcal{T}\,) \leq \ln(|\mathcal{T}|)$, we see that $\mathrm{VCdim}_{\mathcal{Q}}(\,\Upsilon^{-R}[\mathrm{T}]\,) \leq |\mathrm{T}|$. Similarly, $|\Upsilon^{-A}[\mathrm{T}]| \leq 2^{|T|}$ holds, which immediately implies $\mathrm{VCdim}_{\mathcal{Q}}(\,\Upsilon^{-A}[\mathrm{T}]\,) \leq |\mathrm{T}|$. Hence, for these types of transformations $\chi \subseteq \{-R,\ -A\}$,

$$m_{upper}(\Upsilon^{\chi}[\mathrm{T}], \epsilon, \delta) \quad \leq \quad \frac{2}{\epsilon^2}\left[|\mathrm{T}| + \ln\frac{1}{\delta}\right]$$

which shows the sample size is (at worst) linear in the size of the initial theory.

The earlier worst-case results for Υ^{+R} and Υ^{+A} cases each require predicate calculus, as they rely on using function symbols. In the context of a propositional logic system with $2n+1$ variables $\{\,\mathtt{y},\, \mathtt{x}_1^+,\, \mathtt{x}_1^-,\, \ldots,\, \mathtt{x}_n^+,\, \mathtt{x}_n^-\,\}$, we can easily get $\mathrm{VCdim}_{\mathcal{Q}}(\,\Upsilon^{+R}[\{\}]\,) \geq 2^n$: Here, use the 2^n queries "$\mathtt{y}\ \ \mathtt{:-}\ \ \mathtt{x}_1^{\pm}\mathtt{,}\ \ \ldots\mathtt{,}\ \ \mathtt{x}_n^{\pm}$" where each $\mathtt{x}_i^{\pm}$ is either $\mathtt{x}_i^+$ or $\mathtt{x}_i^-$, and observe that there is a theory in $\Upsilon^{+R}[\{\}]$, of size $O(2^n)$, which corresponds to each of the 2^{2^n} possible deterministic oracles, where each such oracle maps some subset of these 2^n queries to `Yes`, and the rest to `No`. To see that each of these oracles leads to a distinct theory, note that each corresponds to a distinct Boolean formula — *i.e.*, here `y` holds iff the disjunction of the rules' respective antecedents holds, which corresponds to an arbitrary DNF formula (identifying each $\mathtt{x}_i^+$ with $\mathtt{x}_i$ and $\mathtt{x}_i^-$ with $\overline{\mathtt{x}}_i$) and there are 2^{2^n} such formulae.

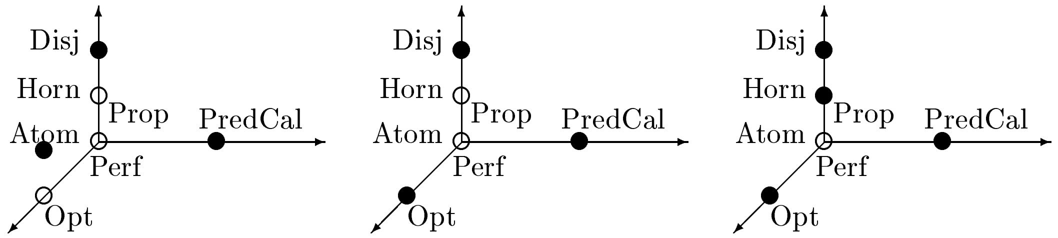
However, if we are only allowed to ask atomic queries, then there are only n queries we can pose (n is number of variables), and so only 2^n possible responses, meaning the VCdim of *any* set of propositional theories can be at most n when considering only atomic queries.

4 Computational Complexity

Our basic challenge is to identify which theory T_{opt} (from a set of revisions) has the smallest possible error. The previous section supplied the number of samples needed to guarantee, with high probability, that the expected error of the theory whose empirical error is smallest, T^*, will be within ϵ of the expected error of this T_{opt}. This section discusses the computational challenge of determining this T^*, given these samples. We show first that this task is tractable in some simple situations: when considering (1) only *atomic* queries posed to a (2) *propositional* theory and being allowed (3) an *arbitrarily large number of modifications* to the initial theory, to produce (4) a *perfect* theory (*i.e.*, one that returns the correct answer to *every* query). This task becomes intractable, however, if we remove (essentially) any of these restrictions: *e.g.*, if we seek optimal (rather than only seeking "perfect") propositional theories and are allowed to pose Horn queries, or if we consider *predicate calculus* theories, etc. (In fact, it is NP-hard for 21 of the $3 \times 2 \times 2 \times 2 = 24$ theory revision situations shown on the left-side of Figure 1.) We see, in particular, that revising a theory using a *bounded* number of modifications is always difficult (*i.e.*, in all $3 \times 2 \times 2$ situations; *e.g.*, even if considering only atomic queries and seeking a perfect propositional theory). This implies that the task of determining the *smallest* number of modifications required to find a perfect

Bounded (All NP-Hard ●) (All NP-Hard ●) (All NP-Hard ●)

Unbounded



Arbitrary $(\Upsilon^{+R,\ -A, -R,\ +A})$ **Generalization** $(\mathcal{G} \approx \Upsilon^{+R,\ -A})$ **Specialization** $(\mathcal{S} \approx \Upsilon^{-R,\ +A})$

<u>Legend:</u>

○ = Easy to solve ● = NP-hard

Bounded = $K < \infty$ **Unbounded** = $K = \infty$

Arbitrary Υ^K = $\Upsilon^{+R,\ -A, -R,\ +A\ (K)}$

Generalization $\mathcal{G}$ = any of $\{\ \Upsilon^{+R,\ -A\ (K)},\ \Upsilon^{+R\ (K)},\ \Upsilon^{-A\ (K)}\ \}$

Specialization $\mathcal{S}$ = any of $\{\ \Upsilon^{-R,\ +A\ (K)},\ \Upsilon^{-R\ (K)},\ \Upsilon^{+A\ (K)}\ \}$

Any task that "projects" down to an NP-hard task, along any axis, is NP-hard. Here, this means all of the "cross terms" are NP-hard. (For example $\mathrm{ThRev}_{PredCal,Horn,Perf}[\Upsilon^\infty]$ is NP-hard, as its projection to the "Prop–PredCal × Perf–Opt" plane, $\mathrm{ThRev}_{PredCal,Atom,Perf}[\Upsilon^\infty]$, is NP-hard.) The $\mathrm{ThRev}_{Prop,Horn,Opt}[\Upsilon^\infty]$ case is shown explicitly as each of its projections is easy; the figures omit all other cross-terms.

Figure 1: Tractability of Theory Revision Tasks

theory is intractable. We also show that many of these tasks are not just intractable but worse, they are not even approximatable, except in very simple situations.

We also consider two restricted subtasks, which allow only transformation that specialize (respectively, only generalize) the initial theory. We show that these tasks, also, are intractable and non-approximatable in essentially all situations; *i.e.*, except when all four of the above conditions hold.[12] Figures 1 and 2 summarize the various cases.

---

[12]Actually, there is one other tractable case in the generalization situation; see Figure 1. Note that the hardness of these restricted situations (say when we are only generalizing the theory) does *not* follow from the hardness of the earlier general case (when we consider both generalization and specializing the theory) in the "agnostic case".

4.1 Basic Complexity Results

To formally state the problem: Let $\Upsilon^{\dagger}[\cdot]$ be a function that maps a theory to a set of candidate revised theories; here, it refers to some $\Upsilon^{k_{aa},k_{ar},k_{dr},k_{da}}$ transformation set.

Definition 1 ($\textsc{ThRev}[\Upsilon^{\dagger}]$ **Decision Problem)**
INSTANCE:
- *Initial theory* T*;*
- *Labeled training sample* $S = \{ \langle q_i,\ O(q_i) \rangle \}$ *containing a set of Horn queries and the correct answers; and*
- *Error value* $p \in [0,1]$.

QUESTION: *Is there a theory* $\mathrm{T}' \in \Upsilon^{\dagger}[\mathrm{T}]$ *such that*
$$\overline{\textsc{Err}_S}(\mathrm{T}') = \frac{1}{|S|}\sum_{\langle q_i, O(q_i)\rangle \in S} err(\mathrm{T}',\ q_i) \ \leq\ p\ ?$$

To simplify our notation, we will henceforth write $\textsc{Err}(\ \mathrm{T}\ )$ for $\overline{\textsc{Err}_S}(\ \mathrm{T}\ )$.

We will also consider the following special cases:

- $\textsc{ThRev}_{Perf}[\Upsilon^{\dagger}]$ requires that $p = 0$ (*i.e.*, seeking perfect theories), rather than "optimal" theories $\textsc{ThRev}_{Opt}[\Upsilon^{\dagger}]$;
- $\textsc{ThRev}_{Prop}[\Upsilon^{\dagger}]$ deals with propositional logic, rather than predicate calculus $\textsc{ThRev}_{PredCal}[\Upsilon^{\dagger}]$; and
- $\textsc{ThRev}_{Atom}[\Upsilon^{\dagger}]$ deals with only atomic queries, as opposed to Horn queries $\textsc{ThRev}_{Horn}[\Upsilon^{\dagger}]$. We will also use $\textsc{ThRev}_{Disj}[\Upsilon^{\dagger}]$ to refer to the task when the queries can be arbitrary disjunctions, which need not be Horn. (While the other subscripts are *restrictions* on $\textsc{ThRev}[\Upsilon^{\dagger}]$, this $Disj$ case is more permissive.)

We will also combine subscripts, with the obvious meanings; hence in general we will write $\textsc{ThRev}_{A,B,C}[\Upsilon^{\dagger}]$ where $A \in \{\text{Prop},\ \text{PredCal}\}$, $B \in \{\text{Atom},\ \text{Horn},\ \text{Disj}\}$ and $C \in \{\text{Perf},\ \text{Opt}\}$. Our default is $\textsc{ThRev}_{PredCal,Horn,Opt}[\Upsilon^{\dagger}]$.

When $\textsc{ThRev}_{\chi}[\Upsilon^{\dagger}]$ is a special case of $\textsc{ThRev}_{\psi}[\Upsilon^{\dagger}]$, finding that $\textsc{ThRev}_{\chi}[\Upsilon^{\dagger}]$ is hard (and later, non-approximatable) immediately implies that $\textsc{ThRev}_{\psi}[\Upsilon^{\dagger}]$ is hard/nonapproximatable. Similarly, seeing that $\textsc{ThRev}_{\psi}[\Upsilon^{\dagger}]$ is easy immediately implies that each special case of $\textsc{ThRev}_{\psi}[\Upsilon^{\dagger}]$ is easy. As a final note: all of the hardness results presented in this paper hold even if we only consider "3-Horn theories" — *i.e.*, rules whose antecedents contain at most 2 literals.

It is easy to find the optimal theory in certain degenerate cases, where either the individual queries can be decoupled (*e.g.*, when using atomic propositional queries) or when our actions are forced (*e.g.*, when seeking perfect propositional theories): just throw away the original theory, then add in propositions corresponding to the "`Yes`-labeled queries". In every other case, however, the task is intractable:

Theorem 3 *(a) The* $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^{\infty}]$ *and* $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^{\infty}]$ *decision problems (and hence* $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{\infty}]$*) are easy; each other problem — in particular,*
(b) $\textsc{ThRev}_{Prop,Horn,Opt}[\Upsilon^{\infty}]$,

(c) $\textsc{ThRev}_{PredCal,Atom,Perf}[\Upsilon^\infty]$ *and*
(d) $\textsc{ThRev}_{Prop,Disj,Perf}[\Upsilon^\infty]$,
and each of their generalizations — is NP-hard.

This information is summarized in lower left "Unbounded, Arbitrary" graph of Figure 1.

Each of these negative results (parts (b), (c) and (d) above) requires that the training data is produced by a $\mathcal{O}_{Det}$ oracle, which supplies a (deterministic) mapping from queries to answers, but does not guarantee that implied target theory is necessarily consistent. In the following theorems, we will explicitly state whether the results hold even if the reviser knows that the oracle is in $\mathcal{O}_{Horn}$.

The above theorem describes the complexity of computing the best theory when we are allowed to use an *arbitrarily expensive* sequence of transformations. (*N.b.*, this permits the theory revision system to throw away the entire initial theory, and generate an arbitrary new theory!) In many cases, however, we may want to consider only short sequences of transformations — *i.e.*, only consider members of $\Upsilon^K[\mathrm{T}]$ for small K. If K is constant, then $\Upsilon^K[\mathrm{T}]$ contains only a polynomial number of theories, which means we can efficiently simply enumerate and test all of these theories. Hence, the associated decision problem is easy:

Observation 3 *For constant K, the $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^K]$ decision problem can be solved in polynomial time.*

This small-K assumption seems implicit to many theory revision systems. Notice, in particular, that this renders theory revision solvable, as this means we will need to see only a small number of samples (see Observation 1), and then perform a simple computation.

However, for some non-constant values of K, the task again becomes intractable:

Theorem 4 *For $K = \Omega(\sqrt{|\mathrm{T}_0|})$, the $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^K]$ decision problem is NP-hard. This is true even if we consider only labeled queries produced by an $\mathcal{O}_{Horn}$ oracle (i.e., even when we know there is a Horn theory that correctly labels all of the queries).*

The observation that determining such "K-step perfect theories" is NP-hard leads immediately to:

Corollary 4.1 *It is NP-hard to compute the minimal-cost transformation sequence required to produce a perfect theory (i.e., to compute the smallest K for which there is a $\mathrm{T}_{perfect} \in \Upsilon^K[T]$ such that $\textsc{Err}(\mathrm{T}_{perfect}) = 0$), even in the propositional case when considering only atomic queries, and when the labeled queries are produced by an $\mathcal{O}_{Horn}$ oracle. Here, it is also NP-hard to compute the "minimal-length" transformation, where the length of the transformation sequence $\tau_1 \circ \tau_2 \circ \ldots \circ \tau_k$ is simply k — i.e., when each transformation has "unit cost".*

(This is the obvious minimization problem corresponding to Theorem 4's decision problem.)

This negative result shows the intractability of the obvious proposal of using a breath-first transversal of the space of all possible theory revisions: First test the initial theory T_0 against the labeled queries, and return T_0 if it has 0% error. If not, then consider all theories

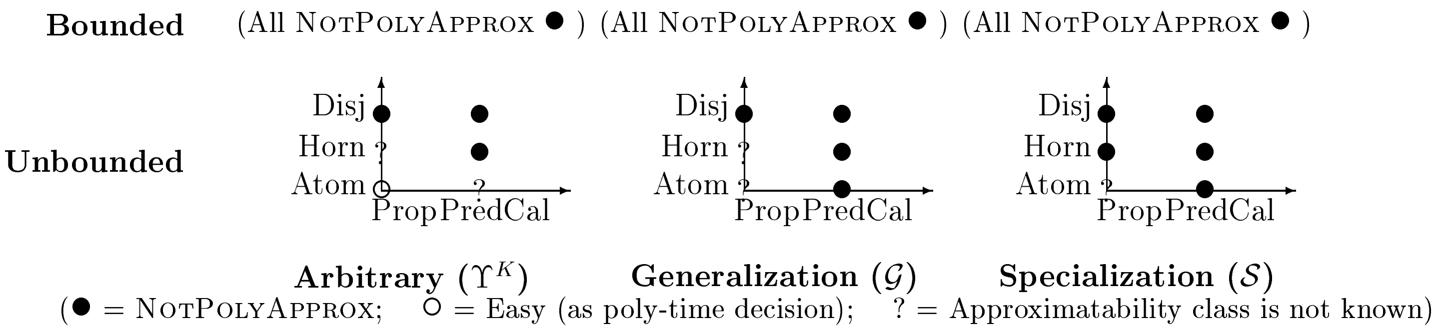


Figure 2: Approximatability of Theory Revision Tasks

formed by applying a single (unit-cost) transformation, and return any perfect $\mathrm{T}_1 \in \Upsilon^1[\mathrm{T}_0]$; and if not, consider all theories in $\Upsilon^2[\mathrm{T}_0]$ (formed by applying sequences of transformations with cost at most two), and return any perfect $\mathrm{T}_2 \in \Upsilon^2[\mathrm{T}_0]$; and so forth. (Notice this may involve using successively more samples on each iteration, *à la* [LMR88].)

4.2 Approximatability

Many decision problems correspond immediately to optimization problems; for example, the MinGraphColor decision problem

> Given a graph $G = \langle N, E\rangle$ and a positive integer K, can each node be labeled by one of K colors in such a way that no edge connects two nodes of the same color; see [GJ79, p191(Chromatic Number)]

corresponds to the minimization problem: Find the minimal coloring of the given graph G. We can similarly view the $\textsc{ThRev}_\chi[\Upsilon^\dagger]$ decision problem as either the minimization problem: "Find the $\mathrm{T}' \in \Upsilon^\dagger[\mathrm{T}]$ whose error is minimal", or the maximization problem: "Find the $\mathrm{T}' \in \Upsilon^\dagger[\mathrm{T}]$ whose *accuracy* is maximal", where a theory's accuracy is $1 - \textsc{Err}(\,\mathrm{T}\,)$. (While the maximally accurate theory also has minimal error, these two formulations can lead to different approximatability results.) For notation, let "$\textsc{MinThRev}_\chi[\Upsilon^\dagger]$" (resp., "$\textsc{MaxThRev}_\chi[\Upsilon^\dagger]$") refer to the minimization (resp., maximization) problem.

Now consider any algorithm B that, given any $\textsc{MinThRev}_\chi[\Upsilon^\dagger]$ instance $x = \langle T, S\rangle$ with initial theory T and labeled training sample S, computes a syntactically legal, but not necessarily optimal, revision $B(x) \in \Upsilon^\dagger[\mathrm{T}]$. Then B's "performance ratio for the instance x" is defined as

$$\mathit{MinPerf}[\textsc{MinThRev}_\chi[\Upsilon^\dagger]](\,B, x\,) \quad = \quad \begin{cases} \frac{\textsc{Err}(\,B(x)\,)}{\textsc{Err}(\,opt(x)\,)} & \text{if } \textsc{Err}(\,opt(x)\,) \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where $opt(x) = opt_{MinThRev_\chi(\Upsilon^\dagger)}(x)$ is the optimal solution for this instance; *i.e.*, $opt(\langle \mathrm{T}, S\rangle)$ is the theory $\mathrm{T}_{opt} \in \Upsilon^\dagger[\mathrm{T}]$ with minimal error over S.

We say a function $g(\cdot)$ "bounds B's performance ratio (over $\textsc{MinThRev}_\chi[\Upsilon^\dagger]$)" iff

$$\forall \text{ instances } x \in \textsc{MinThRev}_\chi[\Upsilon^\dagger], \quad \mathit{MinPerf}[\textsc{MinThRev}_\chi[\Upsilon^\dagger]](\, B, x\,) \;\leq\; g(|x|)$$

where $|x|$ is the size of the instance $x = \langle \mathrm{T},\, S\,\rangle$, which we define to be the number of symbols in T plus the number of symbols used in S. Intuitively, this $g(\cdot)$ function indicates how closely the B algorithm comes to returning the best answer for x, in the worst case over all $\textsc{MinThRev}_\chi[\Upsilon^\dagger]$ instances x.

Now let *Poly*($\textsc{MinThRev}_\chi[\Upsilon^\dagger]$) be the collection of all polytime algorithms that return legal (but not necessarily optimal) answers to $\textsc{MinThRev}_\chi[\Upsilon^\dagger]$ instances. It is natural to ask for the algorithm in *Poly*($\textsc{MinThRev}_\chi[\Upsilon^\dagger]$) with the best performance ratio; this would indicate how close we can come to the optimal solution, using only a feasible computational time. For example, if this function was the constant $1(x) \equiv 1$ for $\textsc{MinThRev}_{Prop}[\Upsilon^\infty]$, then a polynomial-time algorithm could produce the optimal solution to any $\textsc{MinThRev}_{Prop}[\Upsilon^\infty]$ instance; as $\textsc{ThRev}_{Prop}[\Upsilon^\infty]$ is NP-complete,[13] this would mean $P = NP$, which is why we do not expect to obtain this result. Or if this bound was some constant $c(x) \equiv c \;\in \Re^+$, then we could efficiently obtain a solution within a factor of c of optimal, which may be good enough for some applications.[14]

However, not all problems can be so approximated. Following [CP91, Kan92], we define

Definition 2 *A minimization problem* MinP *is* PolyApprox *if*
$$\forall \gamma \in \Re^+,\ \exists B_\gamma \in \mathrm{Poly}(\,\textsc{MinP}\,),\ \forall x \in \textsc{MinP}, \quad \mathit{MinPerf}[\textsc{MinP}](\, B_\gamma,\, x\,) \;\leq\; |x|^\gamma \,.$$

Lund and Yannakakis [LY93] prove that (unless $P = NP$) the "MinGraphColor minimization problem" is not PolyApprox — *i.e.*, there is some $\gamma \in \Re^+$ such that no polynomial-time algorithm can always find a solution within $|x|^\gamma$ of optimal. We use that result to prove:

Theorem 5 *Unless $P = NP$, none of*
$\textsc{MinThRev}_{Prop,Disj}[\Upsilon^\infty]$, $\textsc{MinThRev}_{PredCal,Horn}[\Upsilon^\infty]$ *and* $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^K]$ *is* PolyApprox.

While these results may at first seem immediate, given that it is NP-hard to determine if a perfect theory exists, notice from Equation 5 that $\mathit{MinPerf}[\textsc{MinThRev}[\Upsilon^\infty]](\,\cdot\,)$ essentially ignores such perfect theories. Note also that this result holds in the context based on an "inconsistent" $\mathcal{O}_{Det}$ oracle; in such situations, no theory can be perfect.

As $|x|$ can get arbitrary large, this result means that these $\textsc{MinThRev}_\chi[\Upsilon^\dagger]$ tasks cannot be approximated by any constant, nor even by any logarithmic factor nor any sufficiently small polynomial, etc.

[13] While Theorem 3 only proves $\textsc{ThRev}_{Prop}[\Upsilon^\infty]$ to be NP-hard, this problem is clearly in NP.

[14] There are such constants for some other NP-hard minimization problems. For example, there is a polynomial-time algorithm that computes a solution whose cost is within a factor of 1.5 for any TravelingSalesman-with-Triangle-Inequality problem; see [GJ79, Theorem 6.5].

4.3 Special Cases

If the theory is too general (*i.e.*, returns `Yes` too often), then we may want to consider "specializing" it by applying only the "delete rule" and "add antecedent" transformations. In particular, recall that $\Upsilon^{+A,-R}[\mathrm{T}]$ is the set of theories obtained using *an arbitrary number* of such transformations, and $\Upsilon^{-R}[\mathrm{T}]$ (resp., $\Upsilon^{+A}[\mathrm{T}]$), is the set of theories obtained by applying an arbitrary number of "delete rule" (respectively, "add antecedent") transformations. Similarly, if the theory is too specific (*i.e.*, returns `No` too often), then we may want to consider "generalizing" it by applying only the "add rule" and "delete antecedent" transformations; here, we consider $\Upsilon^{+R,-A}[\mathrm{T}]$, $\Upsilon^{+R}[\mathrm{T}]$ and $\Upsilon^{-A}[\mathrm{T}]$, which are the set of theories obtained by applying an arbitrary number of such transformations.

Even using only these transformations, almost all of these tasks remain intractable:

Theorem 6 *For each* $\mathcal{S} \in \{\, \Upsilon^{-R,+A},\ \Upsilon^{-R},\ \Upsilon^{+A} \,\}$, $\mathcal{S}^K \in \{\, \Upsilon^{-R=K,\ +A=K},\ \Upsilon^{-R=K},\ \Upsilon^{+A=K} \,\}$, $\mathcal{G} \in \{\, \Upsilon^{+R,-A},\ \Upsilon^{+R},\ \Upsilon^{-A} \,\}$, $\mathcal{G}^K \in \{\, \Upsilon^{+R=K,\ -A=K},\ \Upsilon^{+R=K},\ \Upsilon^{-A=K} \,\}$:

1. *It is easy to solve*
   (a) $\textsc{ThRev}_{Prop,Atom,Perf}[\mathcal{S}]$, *and* **(b)** $\textsc{ThRev}_{Prop,Horn,Perf}[\mathcal{G}]$,
2. *Each of the following is NP-hard:*
   (a*) $\textsc{ThRev}_{Prop,Atom,Opt}[\mathcal{S}]$, **(b)** $\textsc{ThRev}_{Prop,Horn,Perf}[\mathcal{S}]$,
   (c*) $\textsc{ThRev}_{PredCal,Atom,Perf}[\mathcal{S}]$, **(d*)** $\textsc{ThRev}_{Prop,Atom,Perf}[\mathcal{S}^K]$
3. *Each of the following is NP-hard:*
   (a*) $\textsc{ThRev}_{Prop,Atom,Opt}[\mathcal{G}]$, **(b)** $\textsc{ThRev}_{Prop,Disj,Perf}[\mathcal{G}]$,
   (c) $\textsc{ThRev}_{PredCal,Atom,Perf}[\mathcal{G}]$, **(d*)** $\textsc{ThRev}_{Prop,Atom,Perf}[\mathcal{G}^K]$.

(The ""s above indicate that the problem is hard even if the target function is constrained to be in* $\mathcal{O}_{Horn}$*.)* □

Worse,

Theorem 7 *Unless* $P = NP$*, none of the following is* PolyApprox*:*
1. $\textsc{MinThRev}_{PredCal,Atom}[\mathcal{S}]$ *and* $\textsc{MinThRev}_{Prop,Horn}[\mathcal{S}]$
   for $\mathcal{S} \in \{\Upsilon^{-R,+A},\ \Upsilon^{-R},\ \Upsilon^{+A}\ \}$
2. $\textsc{MinThRev}_{PredCal,Atom}[\mathcal{G}]$ *and* $\textsc{MinThRev}_{Prop,Disj}[\mathcal{G}]$
   for $\mathcal{G} \in \{\Upsilon^{+R,-A},\ \Upsilon^{+R},\ \Upsilon^{-A}\ \}$
3. $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^{\dagger}]$
   for $\Upsilon^{\dagger} \in \left\{\ \Upsilon^{+A=K,-R=K)},\ \Upsilon^{-R=K},\ \Upsilon^{+A=K},\ \Upsilon^{-A=K,+R=K},\ \Upsilon^{+R=K},\ \Upsilon^{-A=K}\ \right\}$.

In each of these cases, however, there is a straight-forward polynomial-time algorithm that can produce a theory whose *accuracy* (*n.b.*, not inaccuracy) is within a factor of 2 of optimal. Here, we use the ratio of an algorithm's *accuracy* to the optimal value

$$MaxPerf[\textsc{MaxThRev}_{\chi}[\Upsilon^{\dagger}]](\,B, x\,) \quad = \quad \frac{1-Err(\,opt(x)\,)}{1-Err(\,B(x)\,)}$$

Theorem 8 *For each* $\Upsilon^{\dagger} \in \{\ \Upsilon^{-R,+A},\ \Upsilon^{-R},\ \Upsilon^{+A},\ \Upsilon^{+R,-A},\ \Upsilon^{+R},\ \Upsilon^{-A}\ \}$,
$\exists B_{\dagger} \in \mathrm{Poly}(\,\textsc{MaxThRev}[\Upsilon^{\dagger}]\,),\quad MaxPerf[\textsc{MaxThRev}[\Upsilon^{\dagger}]](\,B_{\dagger}, x\,) \ \leq\ 2$

The companion paper [Gre99] considers other related cases, including the above special cases in the context where our underlying theories can use the `not(·)` operator to return `Yes` if the specified goal cannot be proven; *i.e.*, using Negation-as-Failure [Cla78]. It also considers the effect of re-ordering the rules and the antecedents, in the context where such shufflings can affect the answers returned. In most of these cases, we show that the corresponding maximization problem is not in PolyApprox — *i.e.*, is not approximatable within a particular polynomial.

4.4 Comments

Asymmetry: There is an interesting asymmetry between the complexities of addressing $\text{ThRev}_{Prop,Horn,Perf}[\Upsilon^{+R}]$ versus $\text{ThRev}_{Prop,Horn,Perf}[\Upsilon^{-R}]$, as the first is easy to compute, while the second is intractable. Towards explaining this, notice the actions of an "Add-rule" revision system Rev^{+R} are forced: on encountering each positively-labeled query $\langle \rho$ `:-` φ; `Yes`$\rangle$, it should simply add ρ `:-` φ if the initial theory does not already entail "ρ `:-` φ"; and on encountering a negatively-labeled query $\langle \rho$ `:-` $\varphi_1, \ldots, \varphi_n$; `No`$\rangle$, it should add each unentailed φ_i. Clearly there is a perfect theory in $\Upsilon^{+R}[\mathrm{T}]$ iff the resulting theory is perfect.

The actions of a "Delete-rule" revision system Rev^{-R} are not as obvious: Given the pair of labeled queries $\langle \rho$ `:-` $\varphi_1, \ldots, \varphi_n$; `Yes`$\rangle$ and $\langle \rho$; `No`$\rangle$, Rev^{-R} must now make $\bigwedge_i \varphi_i$ unentailed, which happens if at least one of the φ_i is deleted; here, however, Rev^{-R} can select which one. As shown in the proof for Theorem 6, it can be NP-hard to find the appropriate such φ_i, given the other labeled queries.

Notice, by contrast, that the *sample complexity* of deleting rules is easily bounded, whereas the *sample complexity* of adding rules, in the predicate calculus case, has no such bound. This suggests the opposite conclusion: that adding rules should be harder.

Need only *Positive* Non-Horn Queries: While several of the proofs do use non-atomic queries, these queries are always *positive*; *i.e.*, of the form $\langle \rho$`:-`φ; `Yes`$\rangle$. Hence, all of theorems that deal with $\text{MinThRev}_{\ldots,Horn,\ldots}[\cdot]$ continue to hold even if the Horn queries are restricted to be labeled positively. The proofs do, however, require both atomic queries that are labeled positively, and other atomic queries that are labeled negatively.

Relation to Inductive Logic Programming (ILP): While several of our proofs involve adding new clauses to an initially empty theory (see Theorems 3(b,c,d), 5(a,b), 6(3b,3c) and 7(2b)), notice the target function $O(\cdot)$ being approximated does not necessarily correspond to a Horn theory (*i.e.*, $O(\cdot)$ is not always in $\mathcal{O}_{Horn}$); hence, these results deal with a situation that differs from the standard ILP task. In fact, many of these tasks become easy if we consider only target functions that correspond to Horn theories. Frazier and Pitt [FP93], however, prove that learning a perfect Horn theory from Horn queries (which corresponds to $\text{ThRev}_{Prop,Horn,Perf}[\Upsilon^{\infty}]$ when the target oracle is in $\mathcal{O}_{Horn}$) is as hard as learning arbitrary CNFs from examples in this "PAC" framework; *n.b.*, the latter is an open problem in the Computational Learning Theory community.

As a final comment on this theme: It is tempting to view theory revision as simply ILP, where the initial theory is non-empty. If this were so, we could then "lift" the ILP results to this theory revision context, after simply "dividing through" by the initial theory. However, typical ILP results deal only with *adding in* new facts and rules. As our theory revision systems must also consider *removing* parts of the given theory (*e.g.*, deleting existing rules and antecedents of rules), we cannot directly apply those ILP results.

5 Conclusion

A knowledge-based system can produce incorrect answers to queries if its underlying theory is faulty. A "theory revision" system transforms a given theory into a related one that is as accurate as possible, based on a given set of correctly-answered "training queries". This paper analyses this task in an attempt to obtain a better understanding of the underlying process. The positive results (especially Observations 1 and 3) show that a theory revision system can work effectively if the initial theory T_0 is "close to" a theory T^* with low error (*i.e.*, if such a T^* is in $\Upsilon^K(\mathrm{T}_0)$ for some small K), as this guarantees that (1) the required number of samples will be small (and often considerably less than are required to learn an effective theory from scratch) and more importantly, (2) even a naïve exhaustive algorithm will be able to identify this good theory efficiently. Notice this condition is true in the typical situation, when the initial theory T_0 corresponds to a deployed system, and hence itself has low error. (Of course, the revision process will usually find a yet better theory.)

Our negative results, however, show that this is essentially the only situation where theory revision is guaranteed to be computationally feasible: We prove that finding a theory whose error is even close to optimal cannot be done efficiently if we are forced to consider more expensive revisions, which involve extensive modifications. Moreover, these negative results hold even if we consider the obvious restricted sets of possible modifications: *e.g.*, "only generalization transformations" or "only specification transformations".

We view these results as partially explaining several standard theory-revision practices. First, the standard justification for theory revision, in general, is the intuition that a relatively small number of samples should be sufficient to transform a nearly-perfect theory into an even better theory; note this intuition has been borne out empirically [LDRG94]. Our sample complexity results prove this in general: showing that it can take fewer samples to produce a very good theory T^* by *revising* an already good theory, than are required to learn this T^* from scratch. Moreover, the further observation that fewer samples are required to justify deleting parts of a theory, rather than adding new parts, motivates theory revision algorithms that focus on the first task [CS90]. We next examined the computational challenge of producing such T^* theories, and saw this is intractable if T^* is syntactically far from the initial theory T_0. As we do not *a priori* know that T_0 will be close to a theory with minimal error, seeking the globally optimal theory is problematic. It therefore makes sense to instead accept a locally optimal revised theory; this in turn resonates with the standard theory-revision practice of hill-climbing.

Finally, as noted in the Introduction, we hope these results will help push researchers and developers to consider other approaches to revising a sub-optimal theory — perhaps by finding useful special cases, employing alternative approaches (possibly stochastic, or like KBANN [Tow91]), changing representations, or exploiting other types of information present, in either the labeled queries, or the reviser's prior knowledge.

A Proofs

Theorem 1 (from [Vap82, Theorem 6.2]) *Given a class of theories $\mathcal{T}$, and $\epsilon, \delta > 0$, let $\mathrm{T}^* \in \mathcal{T}$ be the theory with the smallest empirical error after*

$$m_{upper}(\mathcal{T}, \epsilon, \delta) \quad = \quad \left\lceil \frac{2}{\epsilon^2} \ln\left(\frac{|\mathcal{T}|}{\delta}\right) \right\rceil$$

labeled queries, drawn independently from a stationary distribution. Then, with probability at least $1-\delta$, the expected error of T^* will be within ϵ of the optimal theory in $\mathcal{T}$; i.e., $Pr[\,\mathrm{Err}(\,\mathrm{T}^*\,) \geq \mathrm{Err}(\,\mathrm{T}_{opt}\,) - \epsilon\,] \geq 1-\delta$.

Proof: As the queries are generated by a stationary distribution, we can view the values of $\{\mathrm{err}(\mathrm{T},\, q_j)\}_j$ as independent, identically-distributed random values with common population mean $\mathrm{Err}(\,\mathrm{T}\,)$. Let $\overline{\mathrm{Err}_S}(\,\mathrm{T}\,)$ be the sample mean after taking $m = m_{upper}(\mathcal{T}, \epsilon, \delta)$ samples, S. Hoeffding-Chernoff bounds [Che52, Bol85] bound the confidence that $\overline{\mathrm{Err}_S}(\,\mathrm{T}\,)$ will be close to $\mathrm{Err}(\,\mathrm{T}\,)$:

$$Pr[\,|\overline{\mathrm{Err}_S}(\,\mathrm{T}\,) - \mathrm{Err}(\,\mathrm{T}\,)| > \lambda\,] \quad < \quad e^{-2m\lambda^2}$$

Using the above value for m, this means $Pr[\,|\overline{\mathrm{Err}_S}(\,\mathrm{T}_i\,) - \mathrm{Err}(\,\mathrm{T}_i\,)| > \frac{\epsilon}{2}\,] < \frac{\delta}{|\mathcal{T}|}$ holds for each $\mathrm{T}_i \in \mathcal{T}$; which implies that the probability that $|\overline{\mathrm{Err}_S}(\,\mathrm{T}_i\,) - \mathrm{Err}(\,\mathrm{T}_i\,)| > \frac{\epsilon}{2}$ holds for *any* i is at most $Pr[\,\exists i\, |\overline{\mathrm{Err}_S}(\,\mathrm{T}_i\,) - \mathrm{Err}(\,\mathrm{T}_i\,)| > \frac{\epsilon}{2}] \;\leq\; |\mathcal{T}| \frac{\delta}{|\mathcal{T}|}$. In particular, this means that the empirical accuracy of both the T^* and T_{opt} theories mentioned above with be within $\epsilon/2$ of their respective expected accuracy, with probability at least $1-\delta$. Hence, with probability at least $1-\delta$,

$$\begin{array}{ccccccc}
\mathrm{Err}(\,\mathrm{T}^*\,) & - & \mathrm{Err}(\,\mathrm{T}_{opt}\,) & & & & \\
= & & (\mathrm{Err}(\,\mathrm{T}^*\,) - \overline{\mathrm{Err}_S}(\,\mathrm{T}^*\,)) & + & (\overline{\mathrm{Err}_S}(\,\mathrm{T}^*\,) - \overline{\mathrm{Err}_S}(\,\mathrm{T}_{opt}\,)) & + & (\overline{\mathrm{Err}_S}(\,\mathrm{T}_{opt}\,) - \mathrm{Err}(\,\mathrm{T}_{opt}\,)) \\
\leq & & \epsilon/2 & + & 0 & + & \epsilon/2 \\
= & & \epsilon & & & &
\end{array}$$

as desired. □ (Theorem 1)

Observation 1 $\ln(|\Upsilon^K[\mathrm{T}_0]|) \;\leq\; K \times [\ln(|\mathcal{L}|) + 2\ln(|\mathrm{T}_0| + K)]$, *where $\mathcal{L}$ is the set of symbols in the language of the theories.*

Proof: To get a quick upper bound: Given $d = |\mathcal{L}|$ possible symbols, we can add in only d^K possible symbols scattered among the existing $n = |\mathrm{T}_0|$ symbols of T_0, leading to at most $d^K\binom{n+K}{K}$ new theories. For each of these theories, we can then remove at most K symbols from the (at most) $n+K$ symbols, which leads to a total of (at most) $|\Upsilon^K[\mathrm{T}_0]| \leq (d^K\binom{n+K}{K}) \times \binom{n+K}{K}) \;\leq\; d^K(n+K)^K(n+K)^K$, whose logarithm is given above. □ (Observation 1)

Observation 2 *There is a class of theories* $\{\mathrm{T}_n\}$*, where each* $|\mathrm{T}_n| = O(n)$*, such that the VC-dimension of the theory set* $\Upsilon^{+A}[\mathrm{T}_n]$*, formed by applying add-antecedent transformations, is exponential in n; i.e., where* $VCdim_{\mathcal{Q}}(\Upsilon^{+A}[\mathrm{T}_n]) \geq 2^n$*. This holds even if all of the queries are atomic, they all correspond to simple instantiations of the same relation, and there is a Horn theory that labels this set perfectly.*

Proof: For each n, use the theory

$$\mathrm{T}_n \quad = \quad \left\{ \begin{array}{l} \texttt{c(X}_1\texttt{,...,X}_n\texttt{) :- } \ell_{true}\texttt{.} \\ \ell_{true}\texttt{.} \\ \texttt{index([], 1).} \\ \texttt{index([0 | Rest], [A}_0\texttt{, A}_1\texttt{] :- index(Rest, A}_0\texttt{).} \\ \texttt{index([1 | Rest], [A}_0\texttt{, A}_1\texttt{] :- index(Rest, A}_1\texttt{).} \end{array} \right\}$$

of size $O(n)$. Notice the `index` relation basically uses the first argument as an index into the n-dimensional second argument, and then succeeds only if the indexed value (of the second argument) is `1`. Hence, the query `index( [1,0,1], [[[1,0],[0,1]], [[1,1],[0,0]]] )` will subgoal to `index( [0,1], [[1,1], [0,0]] )` then to `index( [1], [1, 1] )` and finally to `index( [], 1 )`, which succeeds. However, `index([1,1,0], [[[1,0],[0,1]], [[1,1],[0,0]]])` will reach the subgoal `index( [], 0 )` and so will fail. Now consider the 2^n possible literals of the form ρ_r = `index( [X`$_1$`, ..., X`$_n$`], `$\langle r\rangle$`)`, each formed by storing either `0` or `1` in each of $\langle r\rangle$'s 2^n "locations", and note that one $\tau^{AA}_{\rho_r} \in \Upsilon_{AA}$ could add each such literal to the "`c( X`$_1$`,...,X`$_n$`) :- `ℓ_{true}`.`" rule, forming `c( X`$_1$`,...,X`$_n$`) :-` ℓ_{true}`, index( [X`$_1$`, ..., X`$_n$`], `$\langle r\rangle$`)`. (Notice this requires $\langle r\rangle$ to be exponentially large.) The $\Upsilon^{+A}[\mathrm{T}_n]$ space therefore includes theories that can return `Yes` to any subset of the 2^n $\{\, \texttt{c(x}_1\texttt{,...,x}_n\texttt{)} \mid \texttt{x}_i \in \{\texttt{0},\texttt{1}\}\,\}$ queries, meaning $\mathrm{VCdim}_{\mathcal{Q}}(\Upsilon^{+A}[\mathrm{T}_n]) \geq 2^n$. □ (Observation 2)

Theorem 3 (*a*) *The* $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^\infty]$ *and* $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^\infty]$ *decision problems (and hence* $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^\infty]$*) are easy; each other problem — in particular,*

(*b*) $\textsc{ThRev}_{Prop,Horn,Opt}[\Upsilon^\infty]$,

(*c*) $\textsc{ThRev}_{PredCal,Atom,Perf}[\Upsilon^\infty]$ *and*

(*d*) $\textsc{ThRev}_{Prop,Disj,Perf}[\Upsilon^\infty]$,

and each of their generalizations — is NP-hard.

Proof: (a) The obvious algorithm for both $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^\infty]$ and $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^\infty]$ takes $\langle \mathrm{T}, S, p\rangle$ as its argument and first removes all of the initial theory T, then adds in each "yes-labeled" queries (or in the stochastic case, adds in φ whenever S includes more instances of $\langle\varphi;$ `Yes`$\rangle$ than $\langle\varphi;$ `No`$\rangle$) and finally returns Yes iff the resulting new theory is sufficiently accurate.

(b): We show $\textsc{ThRev}_{Prop,Horn,Opt}[\Upsilon^\infty]$ is NP-hard by reducing to it the NP-complete decision problem:

> **Definition 3 (**$\textsc{MaxIndSet}$ **Decision Problem,** from ([GJ79, p194]):**)** *Given any graph* $G = \langle N, E\rangle$*, with nodes* $N = \{n_i\}$ *and edges* $E \subset N \times N$*, and a positive integer* $k \in \mathcal{Z}^+$*, is there an independent set of size k; i.e., a subset* $S \subset N$ *such that* $|S| = k$ *and* $\forall s_1, s_2 \in S, \langle s_1, s_2\rangle \notin E$*.*

Given any graph $G = \langle N, E\rangle$ and specified size of the independent set k, let $\mathrm{T}_G = \{\}$ be the empty theory, and let S_G be the following $(|N| \times 1) + (|E| \times |N|) + (1 \times |N|)$ queries

$$S_G \;=\; \left\{ \begin{array}{lll} \langle \texttt{n; Yes}\rangle & \text{for } n \in N & \text{(Ask each of these } |N| \text{ queries 1 time)} \\ \langle \texttt{b :- n, m; Yes}\rangle & \text{for } \langle n, m\rangle \in E & \text{(Ask each of these } |E| \text{ queries } |N| \text{ times)} \\ \langle \texttt{b; No}\rangle & & \text{(Ask this query } |N| \text{ times)} \end{array} \right\}$$

Now observe that G has an independent set of size k iff there is a theory $\mathrm{T}_{opt} \in \Upsilon^\infty[\mathrm{T}_G]$ formed by adding new rules to $\mathrm{T}_G = \{\}$,[15] whose error is $p = \frac{|E|-k}{|N|(2+|E|)}$:

$\Longrightarrow$**:** Suppose G has an independent set of size k; call this independent set $U = \{n_i\}_{i=1}^k \subset N$. Let T_U be the theory obtained by adding to $\mathrm{T}_G = \{\}$ the corresponding $\texttt{n}_i$ atomic clauses, $i = 1..k$, as well as the $|E|$ rules "`b :- n, m`", for each $\langle \texttt{n}, \texttt{m}\rangle \in E$. Hence T_U is correct for all $|N|$ copies of the $|E|$ different $\langle$`b :- n, m; Yes`$\rangle$ queries. As U is independent, it contains at most one of any $\langle n,\ m\rangle \in E$ pair, which means T_U can contain at most one of any such $\{\texttt{n}, \texttt{m}\}$ pair, which means T_U will not entail the `b` literal. Hence T_U is correct for all $|N|$ copies of the $\langle$`b; No`$\rangle$ query. As T_U also entails k of the $\texttt{n}_j$ literals, as well as all $|E|$ of the "`b :- n, m`" rules, its error is $\frac{|E|-k}{|N|(2+|E|)}$, as desired.

$\Longleftarrow$**:** Suppose we can add a set of clauses to T_G to form a theory T' whose error is $p = \frac{|E|-k}{|N|(2+|E|)}$. Notice first that the obvious clauses to add are of the form "`b :- n, m`" and "$\texttt{n}_i$"; adding in any other clause can only increase our error. We can assume that T' includes all $|E|$ of the "`b :- n, m`" clauses, as otherwise its error will be strictly over p. Let $\texttt{U} = \{\texttt{n}_i\}$ be the set of $\texttt{n}_i$s added. If this `U` includes both the literals `n` and `m` corresponding to any "`b :- n, m`" rule, then T' would entail `b`, which alone prevents T's error from equaling p. We can therefore assume that `U` includes at most one of any $\{\texttt{n}, \texttt{m}\}$ pair, which means that `U` corresponds to an independent set. As $\textsc{Err}(\,\mathrm{T}'\,) = p$, this set must contain k elements, as desired.

(c): We show that $\textsc{ThRev}_{PredCal,Atom,Perf}[\Upsilon^\infty]$ is NP-hard by reducing to it the (canonical) NP-complete problem:

> **Definition 4 (**3SAT **Decision Problem,** from [GJ79, p259]**:)** *Given a set $U = \{u_1, \ldots, u_n\}$ of variables and formula $\varphi = \{c_1, \ldots, c_m\}$ (a conjunction of clauses over U) such that each clause $c \in C$ is a disjunction of 3 (positive or negative) literals, is there a satisfying truth assignment for φ?*

Given any 3SAT formula φ, let $\mathrm{T}_\varphi = \{\}$ be the empty theory. To define the query/answer pairs, for each $c = \{\tilde{u}_{j1}, \tilde{u}_{j2}, \tilde{u}_{j3}\}$ clause, let $\texttt{v[|c|] = v(X}_1\texttt{, ..., X}_n\texttt{)[\{ X}_{j1}\texttt{/sgn(}\tilde{u}_{j1}\texttt{), X}_{j2}\texttt{/sgn(}\tilde{u}_{j2}\texttt{), X}_{j3}\texttt{/sgn(}\tilde{u}_{j3}\texttt{) \}]}$, where $\texttt{sgn}(u_j) = 0$ and $\texttt{sgn}(\bar{u}_j) = 1$. As an example, $\texttt{v[| }\{u_3, u_5, \bar{u}_8\}\texttt{ |] = v(X}_1\texttt{, X}_2\texttt{, 0, X}_4\texttt{, 0, X}_6\texttt{, X}_7\texttt{, 1, X}_9\texttt{)}$, when there are 9 variables.

[15] Given that T_G is empty, there is no reason to consider any other type of transformation. Also, while this proof considers adding *atomic clauses* (a.k.a. "literals"), it is trivial to consider a variant that adds "non-degenerate" clauses by replacing each $\texttt{n}_j$ literal with the rule "$\texttt{n}_j$ `:-` ℓ_{true}.", and assuming the initial theory $\mathrm{T}_{G'}$ includes the literal ℓ_{true}.

Now for any 3SAT formula $\varphi = \{c_1, c_2, \cdots c_m\}$ let

$$S_\varphi \quad = \quad \left\{ \begin{array}{ll} \langle \texttt{v(X}_1\texttt{, X}_2\texttt{,..., X}_n\texttt{)};\ \texttt{Yes}[\{\texttt{X}_1/?,\ \texttt{X}_2/?,\ \ldots,\ \texttt{X}_n/?\}]\rangle & \\ \langle \texttt{v[|c}_i\texttt{|]};\ \texttt{No}\rangle & \text{for each } i = 1..m \end{array} \right\}$$

For now, assume also require that the language for this theory include only the two constant symbols `0` and `1`, and no function symbols, as well as the relation symbol `v`.

We now show that there is a theory $\mathrm{T}_{opt} \in \Upsilon^\infty[\{\}]$ whose error is $\mathrm{ERR}(\mathrm{T}_{opt}) = 0$ iff there is a satisfying assignment of φ.

$\Longleftarrow$**:** Let $f : U \mapsto \{1, 0\}$ be an assignment that satisfies φ, and let $\mathrm{T}' \in \Upsilon[\{\}]$ be the theory formed by adding to $\mathrm{T}_\varphi = \{\}$ the unit clause $\texttt{v(}\ f(u_1)\texttt{,}\ f(u_2)\texttt{, ...,}\ f(u_n)\ \texttt{)}$. (*E.g.*, if $f = \{\langle u_1, 1\rangle, \langle u_2, 0\rangle, \langle u_3, 0\rangle, \langle u_4, 1\rangle\}$, then $\mathrm{T}' = \{\texttt{v(1, 0, 0, 1)}\}$.) Observe immediately that, as T' entails an instance of $\texttt{v(X}_1\texttt{, X}_2\texttt{,..., X}_n\texttt{)}$, it satisfies the first query, and that $\texttt{v(}\ f(u_1)\texttt{,}\ f(u_2)\texttt{, ...,}\ f(u_n)\ \texttt{)}$ will not match any of the $\texttt{v[|c}_i\texttt{|]}$ literals: *E.g.*, consider $c_i = \{u_3, u_5, \bar{u}_8\}$. As f satisfies φ, it must satisfy this c_i, which means $f(u_3) = 1$ or $f(u_5) = 1$ or $f(u_8) = 0$, which means $\texttt{v(}\ f(u_1)\texttt{,}\ f(u_2)\texttt{, ...,}\ f(u_n)\ \texttt{)}$ will not match $\texttt{v(X}_1\texttt{, X}_2\texttt{, 0, X}_4\texttt{, 0, X}_6\texttt{, X}_7\texttt{, 1, X}_9\texttt{)}$. Hence, T' will produce the correct answers to all of the S_φ queries, and so its error is 0.

$\Longrightarrow$**:** Suppose we can form a perfect theory T_{opt} by adding some clauses to $\{\}$. To satisfy the first query, T_{opt} must include some instance of `v(...)`. Let $\texttt{v(a}_1\texttt{, ..., a}_n\texttt{)}$ be any such literal. We need only show that the mapping $f(u_i) = \texttt{a}_i$ is a satisfying assignment. First, recall the only constant symbols are $\{\texttt{0}, \texttt{1}\}$, which means f's range is appropriate. Second, towards a contradiction, assume f does not satisfy some clause, say $c_i = \{u_3, u_5, \bar{u}_8\}$, which means $f(u_3) = 0$, $f(u_5) = 0$ and $f(u_8) = 1$. This means a literal in T_{opt} will match the $\texttt{v[|c}_i\texttt{|]}$ literal, which means T_{opt} is not perfect; contradiction.

To remove the restriction on the language: If the language includes other constant symbols, say $\{\texttt{s}_1, \ldots, \texttt{s}_m\}$, just include $m \times n$ additional labeled queries, each of the form $\langle \texttt{v(X}_1\texttt{, ..., X}_{j-1}\texttt{, s}_i\texttt{, X}_{j+1}\texttt{, ..., X}_n\texttt{)};\ \texttt{No}\rangle$. We can similarly deal with any function symbols, say $\{\texttt{f}_1, \ldots, \texttt{f}_k\}$, by including the $k \times n$ additional labeled queries of the form $\langle \texttt{v(X}_1\texttt{, ..., X}_{j-1}\texttt{, f}_i\texttt{(Y}_1\texttt{, ..., Y}_{m_i}\texttt{), X}_{j+1}\texttt{, ..., X}_n\texttt{)};\ \texttt{No}\rangle$. (Of course, each $\texttt{f}_i$ is of arity m_i.)

(d): We also use 3SAT to show that $\mathrm{THREV}_{Prop,Disj,Perf}[\Upsilon^\infty]$ is NP-hard: Once again let the initial theory be empty $\{\}$, and let

$$S_\varphi \quad = \quad \left\{ \begin{array}{ll} \langle \tilde{\texttt{u}}_{i1} \vee \tilde{\texttt{u}}_{i2} \vee \tilde{\texttt{u}}_{i3};\ \texttt{Yes}\rangle & \text{for } c_i = \{\tilde{u}_{i1}, \tilde{u}_{i2}, \tilde{u}_{i3}\},\ i = 1..m \\ \langle \texttt{b :- u}_i\texttt{,}\ \bar{\texttt{u}}_i;\ \texttt{Yes}\rangle & \text{for } i = 1..n \\ \langle \texttt{b};\ \texttt{No}\rangle & \text{for } i = 1..n \end{array} \right\}$$

To explain the notation: the query corresponding to $c_1 = \{u_3, u_5, \bar{u}_8\}$ is "$\texttt{u}_3 \vee \texttt{u}_5 \vee \bar{\texttt{u}}_8$", and the correct answer to this query is `Yes`.

We now show that there is a theory $\mathrm{T}_{opt} \in \Upsilon^\infty[\{\}]$ whose error is $\mathrm{ERR}(\mathrm{T}_{opt}) = 0$ iff there is a satisfying assignment for φ.

$\Longleftarrow$**:** Let $f : U \mapsto \{1, 0\}$ be an assignment that satisfies φ, and let $\mathrm{T}' \in \Upsilon^\infty[\{\}]$ be the theory formed by adding to $\{\}$ the unit clause $\mathtt{u}_i$ if $f(u_i)=1$, and $\bar{\mathtt{u}}_i$ if $f(u_i)=0$, as well as the n rules "$\mathtt{b} \;\texttt{:-}\; \mathtt{u}_i,\; \bar{\mathtt{u}}_i$" for $i=1..n$. To see that $\mathrm{ERR}(\,\mathrm{T}'\,)=0$, observe first that T' answers all m "$\mathtt{b} \;\texttt{:-}\; \mathtt{u}_i,\; \bar{\mathtt{u}}_i$" queries correctly. Secondly, as T' includes exactly one of each $\{\mathtt{u}_i, \bar{\mathtt{u}}_i\}$ pair, its answer to the $\mathtt{b}$ query is $\mathrm{T}'(\,\mathtt{b}\,)=\mathtt{No}$. As f is a satisfying assignment, for each j, either $f(u_i)=1$ for some $u_i \in c_j$, or $f(u_{i'})=0$ for some $\bar{u}_{i'} \in c_j$. This means T' includes some $\tilde{u}_{ij}$ corresponding to an element in c_j, which means $\mathrm{T}'(\,\mathtt{c}_j\,)=\mathtt{Yes}$.

$\Longrightarrow$: Suppose we can form a perfect theory T_{opt} by adding some set of rules to $\{\}$. First, T_{opt} must entail each "$\mathtt{b} \;\texttt{:-}\; \mathtt{u}_i,\; \bar{\mathtt{u}}_i$" rule. If T_{opt} also entails both of $\{\mathtt{u}_i,\; \bar{\mathtt{u}}_i\}$ for any i, then it will return the wrong answer to the $\mathtt{b}$ query. We can therefore assume that T_{opt} entails at most one from any pair $\{\mathtt{u}_i,\; \bar{\mathtt{u}}_i\}$. We can further assume that T_{opt} includes (at least) one of the literals from each $c_j=\{\tilde{u}_{j1}, \tilde{u}_{j2}, \tilde{u}_{j3}\}$ clause, as otherwise T_{opt} would return the incorrect answer to the $\tilde{\mathtt{u}}_{j1} \vee \tilde{\mathtt{u}}_{j2} \vee \tilde{\mathtt{u}}_{j3}$ query. Now define the assignment $f: U \mapsto \{0,1\}$ by $f(u_i)=1$ iff $\mathrm{T}_{opt} \models \mathtt{u}_i$, and $f(u_i)=0$ otherwise; and observe (immediately) that f satisfies φ. □ (Theorem 3)

Theorem 4 *For $K=\Omega(\sqrt{|\mathrm{T}_0|})$, the* $\mathrm{THREV}_{Prop,Atom,Perf}[\Upsilon^K]$ *decision problem is NP-hard. This is true even if we consider only labeled queries produced by an $\mathcal{O}_{Horn}$ oracle.*

Proof: We reduce 3SAT (Definition 4) to this problem: Given any 3SAT formula $\varphi = \{c_1, c_2, \cdots, c_m\}$ over the variables $U=\{u_1,\ldots,u_n\}$, use the following $(n+1)(n+3m)$-clause theory

$$\mathrm{T}_\varphi \quad = \quad \left\{ \begin{array}{lll} \mathtt{b}_i^k & \texttt{:-}\; \mathtt{u}_i,\; \bar{\mathtt{u}}_i. & \text{for } i=1..n, \text{ for } k=0..n \\ \mathtt{c}_j^k & \texttt{:-}\; \mathtt{u}_i. & \text{whenever } u_i \in c_j, \text{ for } k=0..n \\ \mathtt{c}_j^k & \texttt{:-}\; \bar{\mathtt{u}}_i. & \text{whenever } \bar{u}_i \in c_j, \text{ for } k=0..n \end{array} \right\}$$

and let S_φ be the following $(n+m)(n+1)$ query/answer pairs:

$$S_\varphi \quad = \quad \left\{ \begin{array}{ll} \langle \mathtt{b}_i^k;\; \mathtt{No} \rangle & \text{for } i=1..n,\; k=0..n \\ \langle \mathtt{c}_j^k;\; \mathtt{Yes} \rangle & \text{for } j=1..m,\; k=0..n \end{array} \right\}$$

Finally, let $K=K(\mathrm{T}_\varphi)=n=\Omega(\sqrt{|\mathrm{T}_\varphi|}\,)$.

We need only show that there is a theory $\mathrm{T}_{opt} \in \Upsilon^K[\mathrm{T}_\varphi]$ whose error is $\mathrm{ERR}(\,\mathrm{T}_{opt}\,)=0$ iff there is a satisfying assignment of φ.

This proof differs from the proof of Theorem 3(c) only by using the fact that there are $n+1$ "copies" of each query to eliminate degenerate solutions: As we can modify at most n rules (using any of the transformations), we cannot simply delete the $n+1$ "$\mathtt{b}_i^k \;\texttt{:-}\; \mathtt{u}_i,\; \bar{\mathtt{u}}_i$" rules for any i; nor can we avoid the effect of these rules by simply adding a new antecedent to each. We must therefore assume that some "$\mathtt{b}_i^k \;\texttt{:-}\; \mathtt{u}_i,\; \bar{\mathtt{u}}_i$" rule will appear in the final T_{opt}, for each i, which means (as $\mathrm{T}_{opt}(\mathtt{b}_i^k)=\mathtt{No}$) that T_{opt} will not contain both $\mathtt{u}_i$ and $\bar{\mathtt{u}}_i$. By a similar counting argument, we cannot simply add $n+1$ new $\mathtt{c}_j^k$ atomic clauses to T_{opt}. For each j, therefore, the only way to insure $\mathrm{T}_{opt}(\mathtt{c}_j^k)=\mathtt{Yes}$ for all k is if T_{opt} includes a literal corresponds to some element of c_j (e.g., some $\tilde{\mathtt{u}}_{ji}$).

To show that these labeled queries are from some function in $\mathcal{O}_{Horn}$, notice they are satisfied by the theory that contains exactly the $m \times (n+1)$ singleton clauses $\mathtt{c}_j^k$, for $j=1..m$, $k=0..n$. □ (Theorem 4)

Theorem 5 *Unless $P = NP$, none of*
$\textsc{MinThRev}_{Prop,Disj}[\Upsilon^\infty]$, $\textsc{MinThRev}_{PredCal,Horn}[\Upsilon^\infty]$ *and* $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^K]$ *is* PolyApprox.

Proof: All three proofs use the following result:

> **Definition 5 (**MinColor **Minimization Problem,** from[GJ79, p191]**:)** *Find the minimal k such that G is k-colorable, where a graph $G = \langle N, E\rangle$ is k-colorable if there is a function $c\colon N \mapsto \{1, \ldots, k\}$ such that $\forall\langle n_1, n_2\rangle \in E,\ c(n_1) \neq c(n_2)$.*
>
> **Theorem 9** *([LY93]) Unless $P = NP$, there is a $\delta \in \Re^+$ such that no polynomial time algorithm can find a coloring for arbitrary* MinColor *graphs $G = \langle N, E\rangle$ within a factor of $|N|^\delta$ of optimal.*
> *(That is,* MinColor *is not* PolyApprox.*)*

(a): We use the following reduction to show that $\textsc{MinThRev}_{Prop,Disj}[\Upsilon^\infty]$ is not PolyApprox: Given any graph $G = \langle N, E\rangle$, let $\mathrm{T}_G = \{\}$ be the empty theory, and let S_G be the following $M = |N| + |N|^2 + |E| \times |N|^2 + |N|$ query/answer pairs (requiring $|N|^2 + |N|^3 + |E| \times |N|^3 + |N|^2$ symbols):

$$\left\{\begin{array}{lll} \langle \mathtt{c}_{\mathtt{n}_1,j} \vee \mathtt{c}_{\mathtt{n}_2,j} \vee \ldots \vee \mathtt{c}_{\mathtt{n}_{|N|},j};\ \mathtt{No}\rangle & \text{for } j = 1..|N| & \\ \langle \mathtt{c}_{\mathtt{n},1} \vee \mathtt{c}_{\mathtt{n},2} \vee \ldots \vee \mathtt{c}_{\mathtt{n},|N|};\ \mathtt{Yes}\rangle & \text{for } n \in N & \text{(Ask each query } |N| \text{ times)} \\ \langle \mathtt{viol\ :-\ c}_{\mathtt{n},j}\mathtt{,\ c}_{\mathtt{m},j};\ \mathtt{Yes}\rangle & \text{for } \langle \mathtt{n}, \mathtt{m}\rangle \in E,\ j = 1..|N| & \text{(Ask each query } |N| \text{ times)} \\ \langle \mathtt{viol};\ \mathtt{No}\rangle & & \text{(Ask this query } |N| \text{ times)} \end{array}\right\}$$

To understand the connection between these propositions and the MinColor problem, think of $\mathtt{c}_{\mathtt{n},j}$ as meaning that the node $\mathtt{n}$ should be colored with the color j; *i.e.*, $c(\mathtt{n}) = j$ for the coloring $c\colon N \mapsto \{1, \ldots, |N|\}$. The first set of queries seeks to minimize the number of distinct colors in c's range; the second set of queries attempts to insure that c is complete: if they are all satisfied, then each node has at least one color; the third and fourth sets attempt to insure that c is a legal coloring: if they are all satisfied, then no pair of nodes connected by an edge will have the same color.

We now show that there is a theory $\mathrm{T}_C \in \Upsilon^\infty[\mathrm{T}_G]$ whose error is $\textsc{Err}(\,\mathrm{T}_C\,) = C/M$, iff there is a solution to the MinColor problem G using C colors.[16]

$\Longleftarrow$**:** Given any legal coloring function $c\colon N \mapsto \{1, \ldots, |N|\}$ whose range has C values, form a new T_C theory by adding to T_G the singleton literal $\mathtt{c}_{\mathtt{n},c(n)}$ for each $n \in N$, as well as the clause $\mathtt{viol\ :-\ c}_{\mathtt{n},j}\mathtt{,\ c}_{\mathtt{m},j}$ for each $\langle \mathtt{n}, \mathtt{m}\rangle \in E$ and each $j = 1..|N|$. Notice this T_C will satisfy each of the final three sets of queries, and fail to satisfy exactly C of the first set; hence $\textsc{Err}(\,\mathrm{T}_C\,) = \frac{C}{M}$.

$\Longrightarrow$**:** Suppose there is a theory $\mathrm{T}_C \in \Upsilon^\infty[\mathrm{T}_G]$ whose error is $\textsc{Err}(\,\mathrm{T}_C\,) = C/M$. Observe first that T_C cannot violate *any* of final 3 sets of queries, as that alone would produce an error of $|N|/M$, which is more than C/M. We can therefore assume that T_C entails the second

[16] To simplify the presentation, we will assume that $C < |N|$.

set of queries which means, for each $n \in N$, there is (at least) one j such that T_C entails $\mathtt{c}_{\mathtt{n},j}$. We can therefore define $c\colon N \mapsto \{1,\ldots,|N|\}$ by $c(n) = \min_j\{\, j \mid \mathrm{T}_C(\mathtt{c}_{\mathtt{n},j}) = 1\,\}$. As T_C entails each "`viol :- c`$_{\mathtt{n},j}$`, c`$_{\mathtt{m},j}$" rule but does not entail `viol`, it cannot entail both $\mathtt{c}_{\mathtt{n},j}$ and $\mathtt{c}_{\mathtt{m},j}$ for any $\langle \mathtt{n},\mathtt{m}\rangle \in E$ and any j, which means c defines a legal coloring. As T_C's error, C/M, is all due to violations of the first set of queries, the c function can use at most C colors.

Now suppose, for every $\delta \in \Re^+$, there is a poly-time algorithm B_δ such that, for any theory + labeled-query-set $x = \langle \mathrm{T}, S\rangle$, $B_\delta(\langle \mathrm{T}, S\rangle)$ returns a theory $\mathrm{T}_\delta \in \Upsilon^\infty[\mathrm{T}]$ whose error is within a factor of $|x|^\delta$ of the error of the optimal $\mathrm{T}_{opt} \in \Upsilon^\infty[\mathrm{T}]$; *i.e.*, such that $\textsc{Err}(\,B_\delta(x)\,)/\textsc{Err}(\,T_{opt}\,) \le |x|^\delta$. We could then use these algorithms to find approximately optimal solutions to any MinColor problem:

Given any MinColor problem $G = \langle N, E\rangle$ (with $|N| \ge 2$), use the above transformation to form $x_G = \langle \mathrm{T}_G, S_G\rangle$. Let $C^* \in \mathcal{Z}^+$ be the optimal solution to G (*i.e.*, the minimal number of colors); this corresponds to the optimal solution for x_G, call it $\mathrm{T}_{G,opt}$, whose error is $\textsc{Err}(\,\mathrm{T}_{G,opt}\,) = \frac{C^*}{M}$. Now use the $B_{\delta/6}$ algorithm to produce a theory $\mathrm{T}_{G,\delta/6}$ with performance ratio $\textsc{Err}(\,T_{G,\delta/6}\,)/\textsc{Err}(\,T_{G,opt}\,) = \frac{C_{\delta/6}}{M}/\frac{C^*}{M} = \frac{C_{\delta/6}}{C^*} \le |\langle \mathrm{T}_G, S_G\rangle|^{\delta/6} \le (|N|^6)^{\delta/6} = |N|^\delta$ (recall that $|\mathrm{T}_G| = 0$ and $|S_G| = |N|^2 + |N|^3 + |E| \times |N|^3 + |N|^2 < |N|^6$ symbols for $|N| > 1$). Notice this corresponds to a feasible MinColor solution to G using $C_{\delta/6}$ colors, meaning we would have produced a solution to G with a performance ratio of under $|N|^\delta$ in polynomial time. As this δ is arbitrary, this contradicts Theorem 9, assuming $P \ne NP$.

(b): To prove that $\textsc{MinThRev}_{PredCal,Horn}[\Upsilon^\infty]$ is not PolyApprox: Given any graph $G = \langle N, E\rangle$, let $\mathrm{T}_G = \{\}$ and S_G be

$$\left\{\begin{array}{lll}
\langle \mathtt{c(X,\ j);\ No}\rangle & \text{for } j = 1..|N| & \text{(Ask each of these } |N| \text{ queries 1 time)}\\
\langle \mathtt{c(n,\ Y);\ Yes}\rangle & \text{for } n \in N & \text{(Ask each of these } |N| \text{ queries } |N| \text{ times)}\\
\langle \mathtt{viol(X,\ Y)\ :\!-\ c(X,\ Z),\ c(Y,\ Z);\ Yes}\rangle & & \text{(Ask this single query } |N| \text{ times)}\\
\langle \mathtt{viol(n,\ m);\ No}\rangle & \text{for } \langle \mathtt{n},\mathtt{m}\rangle \in E & \text{(Ask each of these } |E| \text{ queries } |N| \text{ times)}
\end{array}\right\}$$

Here `c(n, j)` means the node `n` should be colored with the color `j`.

We can use essentially the same arguments used above to show that there is a theory $\mathrm{T}_C \in \Upsilon^\infty[\mathrm{T}_G]$ whose error is $\textsc{Err}(\,\mathrm{T}_C\,) = C/M$, iff there is a solution to the MinColor problem G using C colors; and then show that this correspondence is sufficient to show that $\textsc{MinThRev}_{PredCal,Horn}[\Upsilon^\infty]$ is not PolyApprox, unless $P = NP$.

(c): To show that $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^K]$ is not PolyApprox, we identify the graph $G = \langle N, E\rangle$, with

$$\mathrm{T}_G \quad = \quad \left\{\begin{array}{ll}
\mathtt{use_color}_j\ \mathtt{:\!-}\ \mathtt{c}_{\mathtt{n},j}\,. & \text{for } \mathtt{n} \in N,\ j = 1..|N|\\
\mathtt{viol}^k\ \mathtt{:\!-}\ \mathtt{c}_{\mathtt{n},j},\ \mathtt{c}_{\mathtt{m},j}\,. & \text{for } \langle \mathtt{n},\mathtt{m}\rangle \in E,\ j = 1..|N|, \text{ and } k = 0..|N|\\
\mathtt{colored}^k_{\mathtt{n}}\ \mathtt{:\!-}\ \mathtt{c}_{\mathtt{n},j}\,. & \text{for } \mathtt{n} \in N,\ j = 1..|N|, \text{ and } k = 0..|N|
\end{array}\right\}$$

and S_G with the $M = |N| + |N|(|N|+1) + |N|^2(|N|+1)$ query/answer pairs:

$$\left\{ \begin{array}{lll} \langle \texttt{use\_color}_j;\ \texttt{No} \rangle & \text{for } j = 1..|N| & \\ \langle \texttt{viol}^k;\ \texttt{No} \rangle & \text{for } k = 0..|N| & \text{(Ask each of these } |N|+1 \text{ queries } |N| \text{ times)} \\ \langle \texttt{colored}^k_{\texttt{n}};\ \texttt{Yes} \rangle & \text{for } n \in N \text{ and } k = 0..|N| & \text{(Ask each of these } |N| \times (|N|+1) \text{ queries } |N| \text{ times)} \end{array} \right\}$$

and finally, let $K = K(\mathrm{T}_G) = |N|$. The trick here is use the multiple copies of the literals to avoid degenerate solutions (see proof of Theorem 4).

To show there is a theory $\mathrm{T}_C \in \Upsilon^K[\mathrm{T}_G]$ whose error is $\mathrm{E{\scriptsize RR}}(\mathrm{T}_C) = C/M$, iff there is a solution to the MinColor problem G using C colors:

$\Longleftarrow$: Given any legal coloring function $c : N \mapsto \{1, \ldots, |N|\}$ whose range has C values, form a new T_C theory by adding, for each $\texttt{n}$, the single literal $\texttt{c}_{\texttt{n},c(n)}$. Notice this T_C will satisfy each of the final two sets of queries, and fail to satisfy exactly C of the first set; hence $\mathrm{E{\scriptsize RR}}(\mathrm{T}_C) = \frac{C}{M}$.

$\Longrightarrow$: Suppose there is a theory $\mathrm{T}_C \in \Upsilon^K[\mathrm{T}_G]$ whose error is $\mathrm{E{\scriptsize RR}}(\mathrm{T}_C) = C/M$. As Υ^K transformations can modify at most K of the rules, notice $\mathrm{T}_C \in \Upsilon^K(\mathrm{T}_G)$ must include at least one of the "$\texttt{viol}^k$:- $\texttt{c}_{\texttt{n},j}, c_{\texttt{m},j}$." rules (for each $\langle n, m\rangle \in E$). Hence, if T_C included both $\{\texttt{c}_{\texttt{n},j}, \texttt{c}_{\texttt{m},j}\}$ for any arc $\langle n, m\rangle \in E$, it would answer the corresponding "$\texttt{viol}^k$" query incorrectly, producing an error of at least $(|N|+1)/M$, which strictly exceeds the assumed error of C/M. We can therefore assume T_C includes at most one of each $\{\texttt{c}_{\texttt{n},j}, \texttt{c}_{\texttt{m},j}\}$ pair. By a similar argument, T_C must include at least one $\texttt{c}_{\texttt{n},j}$ for each $\texttt{n}$; otherwise some $\texttt{colored}^k_{\texttt{n}}$ query would be answered incorrectly, which would force $\mathrm{E{\scriptsize RR}}(\mathrm{T}_C) \geq (|N|+1)/M$.

Our quota of $|N|$ symbols is just the number needed to add exactly one $\{\texttt{c}_{n,j}\}_j$ for each node $n \in N$, as required to satisfy the $\texttt{colored}^k_{\texttt{n}}$ queries. We can, therefore, define a coloring $c\colon N \mapsto \{1, \ldots, |N|\}$ by letting $c(n) = \lambda(j)\{\texttt{c}_{\texttt{n},j} \in \mathrm{T}_C\}$ be the single j for which T_C includes the literal $\texttt{c}_{\texttt{n},j}$. Now observe that c is a feasible solution to MinColor, as every node has a color, and no arc connects two nodes of the same color. Notice finally that T_C satisfies the final two sets of queries, meaning it is only inaccurate on some set of exactly C $\langle \texttt{use\_color}_j;\ \texttt{No}\rangle$ labeled queries, meaning the associated coloring c requires exactly C colors.

(The rest of this proof is isomorphic to final piece of part (a), shown above.) $\Box$ (Theorem 5)

Theorem 6 *For each* $\mathcal{S} \in \{\, \Upsilon^{-R,+A},\ \Upsilon^{-R},\ \Upsilon^{+A} \,\}$, $\mathcal{S}^K \in \{\, \Upsilon^{-R=K,\ +A=K},\ \Upsilon^{-R=K},\ \Upsilon^{+A=K} \,\}$, $\mathcal{G} \in \{\, \Upsilon^{+R,-A},\ \Upsilon^{+R},\ \Upsilon^{-A} \,\}$, $\mathcal{G}^K \in \{\, \Upsilon^{+R=K,\ -A=K},\ \Upsilon^{+R=K},\ \Upsilon^{-A=K} \,\}$*:*

1. *It is easy to solve*
   (a) $\textsc{ThRev}_{Prop,Atom,Perf}[\mathcal{S}]$, *and* **(b)** $\textsc{ThRev}_{Prop,Horn,Perf}[\mathcal{G}]$,

2. *Each of the following is NP-hard:*
   (a*) $\textsc{ThRev}_{Prop,Atom,Opt}[\mathcal{S}]$, **(b)** $\textsc{ThRev}_{Prop,Horn,Perf}[\mathcal{S}]$,
   (c*) $\textsc{ThRev}_{PredCal,Atom,Perf}[\mathcal{S}]$, **(d*)** $\textsc{ThRev}_{Prop,Atom,Perf}[\mathcal{S}^K]$

3. *Each of the following is NP-hard:*
   (a*) $\textsc{ThRev}_{Prop,Atom,Opt}[\mathcal{G}]$, **(b)** $\textsc{ThRev}_{Prop,Disj,Perf}[\mathcal{G}]$,
   (c) $\textsc{ThRev}_{PredCal,Atom,Perf}[\mathcal{G}]$, **(d*)** $\textsc{ThRev}_{Prop,Atom,Perf}[\mathcal{G}^K]$.

(The ""s above indicate that the problem is hard even if the target function is constrained to be in $\mathcal{O}_{Horn}$.)*

Proof: (1a): To deal with $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{-R,+A}]$ and $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{-R}]$: For each labeled query $\langle \nu, \texttt{No}\rangle$, use the $\tau^{DR}_{\nu:-\ldots}$ transformation to delete each rule whose head matches ν; then check if the resulting theory T' is perfect. To handle $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{+A}]$: For each labeled query $\langle \varphi, \texttt{No}\rangle$ where $\mathrm{T}(\varphi) = \texttt{Yes}$, note that T must include at least one rule of the form "φ `:-` $\texttt{a}_1$`, ...,` $\texttt{a}_k$" (where $k \geq 0$). To each such rule, add in a new unsatisfied literal ℓ_{false}. In all cases, after performing the appropriate transformations, return Yes iff the resulting theory is perfect.

(1b): To deal with $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^{+R}]$ and $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^{+R,-A}]$: For each positively-labeled query $\langle \phi, \texttt{Yes}\rangle$, use the τ^{+R}_{ϕ} transformation to add in the (possibly new) clause ϕ; then return Yes iff the resulting theory is perfect. We can use a similar approach to handle $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^{-A}]$.

(2a*): We reduce the following NP-complete problem to $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^{-R}]$:

> **Definition:** $\textsc{MinHitSet}$ **Decision Problem,** from [GJ79, p222]**:** Given set of elements $X = \{x_1, \ldots, x_k\}$, collection $C = \{c_i\}$ of subsets of X where each $c_i \subseteq X$, and integer $k \in \mathcal{Z}^+$, is there a subset of X of size k that intersects each subset c_i; *i.e.*, a set $S \subset X$ such that $|S| = k$ and $S \cap c_i \neq \{\}$ for all $c_i \in C$.

Given an arbitrary instance of $\textsc{MinHitSet}$ $\langle X, C, k\rangle$, let

$$\mathrm{T}_{DR} \quad = \quad \left\{ \begin{array}{ll} \texttt{x}_j & \text{for } x_j \in X \\ \texttt{c}_i \;\; \texttt{:-} \;\; \texttt{x}_j\,\texttt{.} & \text{whenever } x_j \in c_i \end{array} \right\}$$

and

$$S_{DR} \quad = \quad \left\{ \begin{array}{lll} \langle \texttt{x}_j;\ \texttt{No}\rangle & \text{for } x_j \in X & \text{(Ask each of these } |X| \text{ queries 1 time)} \\ \langle \texttt{c}_i;\ \texttt{Yes}\rangle & \text{for } c_i \in C & \text{(Ask each of these } |C| \text{ queries } |X| \text{ time)} \end{array} \right\}$$

Now observe that there is a hitting set of size k iff there is a theory $\mathrm{T}' \in \Upsilon^{-R}[\mathrm{T}_{DR}]$ formed by deleting clause from T_{DR}, whose error is $p = \frac{k}{|X| \times (|C|+1)}$:

$\Longrightarrow$**:** Suppose $\langle X, C\rangle$ has an hitting set of size k; call this set $S = \{x_i\}_{i=1}^{k} \subset X$. Let $\bar{S} = X - S = \{x_i\}_{i=k+1}^{n}$ be the complement of this set, and let $\mathrm{T}_S \in \Upsilon^{-R}[\mathrm{T}_{DR}]$ be the theory obtained by deleting the corresponding $\mathtt{x}_i$ literals, $i = k+1..|X|$; hence the only $\mathtt{x}_i$s in T_S's theory correspond to elements of S. As S is a hitting set, T_S will contain at least one $\mathtt{x}_i$ for each $\mathtt{c}_j$, and so it will still be able to derive each $\mathtt{c}_j$. As it is also *able* to derive each of the k literals in $\bar{S}$, its expected error is $\frac{k}{|X|\times(|C|+1)}$.

$\Longleftarrow$: Suppose we can delete a set of rules from T_{DR} to form a theory $\mathrm{T}' \in \Upsilon^{-R}[\mathrm{T}_{DR}]$ whose error is p. As deleting any "$\mathtt{c}_i$ `:-` $\mathtt{x}_j$" rule can only be detrimental, we will only consider deleting some of the $\mathtt{x}_i$ atomic clauses; let $\bar{\mathtt{S}} = \{\mathtt{x}_i\}$ be the set removed, leaving only the set $\mathtt{S}$. Now observe that this $\mathtt{S}$ corresponds to a hitting set $S \subset X$ of size k: First, if $\mathtt{S}$ was not a hitting set, then T' would be unable to derive some $\mathtt{c}_j$, which would prevent it from obtaining the needed accuracy. Second, if $\mathtt{S}$ had more than k elements, then T's error would again be over p.

The proof for $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^{-R,+A}]$ is identical to the one shown above, given the observation that adding antecedents to any "$\mathtt{c}_i$ `:-` $\mathtt{x}_j$" rule is detrimental, and adding any antecedents to a "$\mathtt{x}_j$" clause has the same effect as deleting it. This second observation is used to handle $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^{-A}]$: Simply repeat the above proof, just substituting the operation of "adding the ℓ_{false} antecedent to the '$\mathtt{x}_j$.' atomic clause (forming $\mathtt{x}_j$ `:-` ℓ_{false})" for the "deleting the $\mathtt{x}_i$ clause" used above. (Notice that both operations have the same effect: of preventing $\mathtt{x}_i$ from being entailed.)

(2b): We use 3SAT (Definition 4) to show that $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^{-R}]$ is NP-hard. Given any 3-CNF formula φ, let

$$\mathrm{T}_\varphi = \left\{ \begin{array}{ll} \mathtt{u}_i.\quad \bar{\mathtt{u}}_i. & \text{for } i = 1..n \\ \mathtt{b} \text{ :- } \mathtt{u}_i,\ \bar{\mathtt{u}}_i. & \text{for } i = 1..n \\ \mathtt{c}_j \text{ :- } \mathtt{u}_i. & \text{whenever } u_i \in c_j \\ \mathtt{c}_j \text{ :- } \bar{\mathtt{u}}_i. & \text{whenever } \bar{u}_i \in c_j \end{array} \right\}$$

and

$$S_\varphi = \left\{ \begin{array}{ll} \langle \mathtt{c}_j;\ \mathtt{Yes} \rangle & \text{for } i = 1..m \\ \langle \mathtt{b} \text{ :- } \mathtt{u}_i,\ \bar{\mathtt{u}}_i;\ \mathtt{Yes} \rangle & \text{for } i = 1..n \\ \langle \mathtt{b};\ \mathtt{No} \rangle & \end{array} \right\}$$

Notice deleting any "$\mathtt{c}_j$ `:-` $\mathtt{u}_i$" or "$\mathtt{c}_j$ `:-` $\bar{\mathtt{u}}_i$" rule can only be detrimental for the $\mathtt{c}_j$ queries, and deleting any "$\mathtt{b}$ `:-` $\mathtt{u}_i$, $\bar{\mathtt{u}}_i$" rule can only hurt the corresponding non-atomic queries. Hence, the only way we can form a perfect $\mathrm{T}_{perf} \in \Upsilon^{-R}[\mathrm{T}_\varphi]$ is by deleting some subset of the $\mathtt{u}_i$ or $\bar{\mathtt{u}}_i$ atomic clauses. Now just re-use the same arguments used to prove Theorem 3: We must remove (at least) one of each $\{\mathtt{u}_i,\ \bar{\mathtt{u}}_i\}$ pair to satisfy the first set of queries, suggesting an assignment $f\colon U \mapsto \{0,1\}$ by $f(u_i) = 1$ iff $\mathtt{u}_i \in \mathrm{T}_{perf}$; and then observe that f satisfies φ as it satisfies each clause $\mathtt{c}_j$, as $\mathrm{T}_{perf}(\mathtt{c}_j) = \mathtt{Yes}$.

To show that $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^{-R,+A}]$ and $\textsc{ThRev}_{Prop,Horn,Perf}[\Upsilon^{+A}]$ are also NP-hard, just observe that adding any antecedents to any of the non-atomic clauses is counterproductive; and adding an unsatisfied ℓ_{false} to any $\mathtt{u}_i$ has the same effect as deleting this atomic clause.

(2c*): To handle $\textsc{ThRev}_{PredCal,Atom,Perf}[\Upsilon^{-R}]$, use

$$\mathrm{T}_\varphi{}' \quad = \quad \left\{ \begin{array}{lll} \mathtt{u}_i\mathtt{(0)}. \quad \bar{\mathtt{u}}_i\mathtt{(0)}. \quad \mathtt{u}_i\mathtt{(1)}. \quad \bar{\mathtt{u}}_i\mathtt{(1)}. & & \text{for } i = 1..n \\ \mathtt{b}_i\mathtt{(X)} \ \mathtt{:-}\ \mathtt{u}_i\mathtt{(X)},\ \bar{\mathtt{u}}_i\mathtt{(X)}. & & \text{for } i = 1..n \\ \mathtt{c}_j\mathtt{(X)} \ \mathtt{:-}\ \mathtt{u}_i\mathtt{(X)}. & & \text{whenever } u_i \in c_j \\ \mathtt{c}_j\mathtt{(X)} \ \mathtt{:-}\ \bar{\mathtt{u}}_i\mathtt{(X)}. & & \text{whenever } \bar{u}_i \in c_j \end{array} \right\}$$

and

$$S'_\varphi \quad = \quad \left\{ \begin{array}{ll} \langle \mathtt{b}_i\mathtt{(1)};\ \mathtt{No} \rangle & \text{for } i = 1..n \\ \langle \mathtt{c}_j\mathtt{(1)};\ \mathtt{Yes} \rangle & \text{for } i = 1..m \\ \langle \mathtt{b}_i\mathtt{(0)};\ \mathtt{Yes} \rangle & \text{for } i = 1..n \end{array} \right\}$$

Here, we identify $\mathtt{u}_i\mathtt{(1)}$ (resp., $\bar{\mathtt{u}}_i\mathtt{(1)}$) with the literal u_i (resp., $\bar{u}_i$); the $\mathtt{u}_i\mathtt{(0)}$ and $\bar{\mathtt{u}}_i\mathtt{(0)}$ values are used to prevent the "$\mathtt{b}_i\mathtt{(X)}$:- $\mathtt{u}_i\mathtt{(X)}$, $\bar{\mathtt{u}}_i\mathtt{(X)}$" rules from being deleted, as deleting such rules would prevent the remaining theory from answering the final set of queries correctly. Hence, we can only consider deleting the atomic $\mathtt{u}_i\mathtt{(1)}$ and $\bar{\mathtt{u}}_i\mathtt{(1)}$ clauses, which leads to the same basic proof shown above.

There are two situations to consider when dealing with $\textsc{ThRev}_{PredCal,Atom,Perf}[\Upsilon^{-R,+A}]$ and $\textsc{ThRev}_{PredCal,Atom,Perf}[\Upsilon^{+A}]$, depending on whether with underlying languages includes equality. If it does not, then the above proof also holds for $\textsc{ThRev}_{PredCal,Atom,Perf}[\Upsilon^{-R,+A}]$, as there is no advantage to adding an antecedent. Here, we can deal with $\textsc{ThRev}_{PredCal,Atom,Perf}[\Upsilon^{+A}]$ by replaying this proof, but replacing the operation of deleting a $\mathtt{u}_i\mathtt{(1)}$ atomic clause with the operation of adding an unsatisfied antecedent, to form "$\mathtt{u}_i\mathtt{(1)}$:- ℓ_{false}.".

The situation is slightly trickier if we allow equality. Here, there is a perfect theory in $\Upsilon^{+A}[\mathrm{T}_\varphi{}']$, formed by simply adding a "X = 0" to each "$\mathtt{b}_i\mathtt{(X)}$:- $\mathtt{u}_i\mathtt{(X)}$, $\bar{\mathtt{u}}_i\mathtt{(X)}$" rule, forming "$\mathtt{b}_i\mathtt{(X)}$:- $\mathtt{u}_i\mathtt{(X)}$, $\bar{\mathtt{u}}_i\mathtt{(X)}$, X=0". To get around this problem, we can use $\mathrm{T}_\varphi{}''$, which differs from $\mathrm{T}_\varphi{}'$ by including a new set of $2n$ literals, $\mathtt{u}_i\mathtt{(2)}$ and $\bar{\mathtt{u}}_i\mathtt{(2)}$ for each $i = 1..n$; and S''_φ which includes all of S'_φ as well as the n additional $\langle \mathtt{b}_i\mathtt{(2)};\ \mathtt{Yes} \rangle$ query/answer pairs. Here, the simple trick of adding the "X=0" antecedents is not sufficient; this forces the revision system to use the changes shown above.

(2d*): To show that $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{-R=K}]$ is NP-hard (where $K \geq n$), use

$$\mathrm{T}_\varphi \quad = \quad \left\{ \begin{array}{ll} \mathtt{u}_i. \quad \bar{\mathtt{u}}_i. & \text{for } i = 1..n \\ \mathtt{b}^k \ \mathtt{:-}\ \mathtt{u}_i,\ \bar{\mathtt{u}}_i. & \text{for } i = 1..n,\ k = 1..K \\ \mathtt{c}_j \ \mathtt{:-}\ \mathtt{u}_i. & \text{whenever } u_i \in c_j \\ \mathtt{c}_j \ \mathtt{:-}\ \bar{\mathtt{u}}_i. & \text{whenever } \bar{u}_i \in c_j \end{array} \right\}$$

$$S_\varphi \quad = \quad \left\{ \begin{array}{ll} \langle \mathtt{b}^k;\ \mathtt{No} \rangle & \text{for } k = 1..K \\ \langle \mathtt{c}_j;\ \mathtt{Yes} \rangle & \text{for } i = 1..m \end{array} \right\}$$

As we can only "spend" K on delete-rule transformations, we cannot delete all $3K$ symbols of the "$\mathtt{b}^k$:- $\mathtt{u}_i$, $\bar{\mathtt{u}}_i$" clauses for any i, meaning we cannot afford to leave both $\{\mathtt{u}_i,\ \bar{\mathtt{u}}_i\}$ in the final theory; the proof then reduces to the solution shown above.

Similar proofs deal with $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{+A=K}]$ and $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{-R=K,\ +A=K}]$.

(3a*): To show that $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^{+R}]$ is NP-hard, we reduce to it the NP-complete

MaxIndSet decision problem (Definition 3). Given any graph $G = \langle N, E\rangle$ with nodes N and edges E, and specified size of the independent set k, use

$$\mathrm{T}_G \quad = \quad \left\{ \texttt{b :- n, m.} \qquad \text{for } \langle \texttt{n},\texttt{m}\rangle \in E \right\}$$

and

$$S_G \quad = \quad \left\{ \begin{array}{lll} \langle \texttt{n}_j;\ \texttt{Yes}\rangle & \text{for } n_j \in N & \text{(Ask each of these } |N| \text{ queries 1 time)} \\ \langle \texttt{b};\ \texttt{No}\rangle & & \text{(Ask this query } |N| \text{ times)} \end{array} \right\}$$

Now observe that G has an independent set of size k iff there is a theory $\mathrm{T}_{opt} \in \Upsilon^{+R}[\mathrm{T}_G]$ formed by adding new rules to T_G, whose error is $p = \frac{|N|-k}{2|N|}$:

$\Longrightarrow$: Suppose G has an independent set of size k; call this independent set $U = \{n_i\}_{i=1}^{k} \subset N$. Let T_U be the theory obtained by adding to T_G the corresponding $\texttt{n}_i$ atomic clauses, $i = 1..k$. As U is independent, it contains at most one of any $\langle n,\ m\rangle \in E$ pair, which means T_U can contain at most one of any $\{\texttt{n},\texttt{m}\}$ pair, which means T_U will not entail the $\texttt{b}$ literal. As T_U also entails only k of the $|N|$ $\texttt{n}_j$ literals, its error is $|N| - k$.

$\Longleftarrow$: Suppose we can add a set of clauses to T_G, to form a theory T' whose error is p. Notice first that the obvious clauses to add are of the form $\texttt{n}_i$; adding in any other clause can only hurt. Let $\texttt{U} = \{\texttt{n}_i\}$ be the set added. If this $\texttt{U}$ includes both the literals $\texttt{n}$ and $\texttt{m}$ corresponding to any "`b :- n, m.`" rule, then T' would entail $\texttt{b}$, which alone would prevent T's error from equaling p. We can therefore assume that $\texttt{U}$ includes at most one of any pair $\{\texttt{n},\texttt{m}\}$, which means that $\texttt{U}$ corresponds to an independent set. As $\mathrm{Err}(\mathrm{T}') = p$, this set must contain k elements, as desired.

The above proof for $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^{+R}]$ deals only with transformations that add clauses; an isomorphic proof holds for $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^{-A,+R}]$ based on the observation that deleting antecedents can only be detrimental. We can also use an virtually isomorphic proof to deal with $\textsc{ThRev}_{Prop,Atom,Opt}[\Upsilon^{-A}]$: Here, use the theory

$$\mathrm{T}_{G'} \quad = \quad \left\{ \begin{array}{ll} \texttt{b :- n, m.} & \text{for } \langle \texttt{n},\texttt{m}\rangle \in E \\ \texttt{n}_j \texttt{ :- } \ell_{false}\texttt{.} & \text{for } n_j \in N \end{array} \right\}$$

(notice ℓ_{false} is *not* in $\mathrm{T}_{G'}$), and the same S_G shown above. We can then simply repeat the above proof, just substituting the operation of "deleting a ℓ_{false} literal from a '$\texttt{n}_j$:- ℓ_{false}.' clause, for the "adding in a $\texttt{n}_i$ literal" used above. Notice immediately that both operations have the same effect: of causing $\texttt{n}_i$ to be entailed. (Notice also that deleting any other antecedent, in particular, from any "`b :- n, m`" rule, can only be detrimental.)

(3b): The proof for $\textsc{ThRev}_{Prop,Disj,Perf}[\Upsilon^{+R}]$ is the same as the proof of Theorem 3(d)). Similar proofs apply to $\textsc{ThRev}_{Prop,Disj,Perf}[\Upsilon^{-A}]$ and $\textsc{ThRev}_{Prop,Disj,Perf}[\Upsilon^{+R,-A}]$.

(3c): These proofs are identical to the proof of Theorem 3(c).

(3d*): To show that $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{+R=K}]$ is NP-hard (when $K \geq n$), use

$$\mathrm{T}_\varphi \quad = \quad \left\{ \begin{array}{ll} \texttt{b :- u}_i\texttt{, }\bar{\texttt{u}}_i\texttt{.} & \text{for } i = 1..n \\ \texttt{c}_j^k \texttt{ :- u}_i\texttt{.} & \text{whenever } u_i \in c_j,\ k = 0..K \\ \texttt{c}_j^k \texttt{ :- }\bar{\texttt{u}}_i\texttt{.} & \text{whenever } \bar{u}_i \in c_j,\ k = 0..K \end{array} \right\}$$

$$S_\varphi \quad = \quad \left\{ \begin{array}{ll} \langle \texttt{b};\ \texttt{No} \rangle & \\ \langle \texttt{c}_j^k;\ \texttt{Yes} \rangle & \text{for } i = 1..m,\ k = 0..K \end{array} \right\}$$

As we can only "spend" K on add-rule transformations, we cannot add all $K+1$ symbols of the "$\texttt{c}_j^k$" atomic clauses for any j query, meaning we must add at least one $\texttt{u}_i$ or $\bar{\texttt{u}}_i$ atomic clause for each associated c_j clause. The proof then reduces to the solution shown above.

Similar proofs deal with $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{-A=K}]$ and $\textsc{ThRev}_{Prop,Atom,Perf}[\Upsilon^{+R=K,\ -A=K}]$.

$\Box$ Theorem 6

Theorem 7 *Unless $P = NP$, none of the following is* $\textsc{PolyApprox}$*:*

1. $\textsc{MinThRev}_{PredCal,Atom}[\mathcal{S}]$ *and* $\textsc{MinThRev}_{Prop,Horn}[\mathcal{S}]$
   for $\mathcal{S} \in \{\Upsilon^{-R,+A},\ \Upsilon^{-R},\ \Upsilon^{+A}\ \}$
2. $\textsc{MinThRev}_{PredCal,Atom}[\mathcal{G}]$ *and* $\textsc{MinThRev}_{Prop,Disj}[\mathcal{G}]$
   for $\mathcal{G} \in \{\Upsilon^{+R,-A},\ \Upsilon^{+R},\ \Upsilon^{-A}\ \}$
3. $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^\dagger]$
   for $\Upsilon^\dagger \in \left\{\ \Upsilon^{+A=K,\,-R=K)},\ \Upsilon^{-R=K},\ \Upsilon^{+A=K},\ \Upsilon^{-A=K,+R=K},\ \Upsilon^{+R=K},\ \Upsilon^{-A=K}\ \right\}$.

Proof: Each of these proofs is a modification of Theorem 5, based on a reduction from $\textsc{MinColor}$ (Definition 5).

(1b): Given any graph $G = \langle N, E\rangle$, let T_{DR} be

$$\mathrm{T}_{DR} \quad = \quad \left\{ \begin{array}{ll} \texttt{c}_{\texttt{n},j}\,. & \text{for } \texttt{n} \in N \text{ and } j = 1..|N| \\ \texttt{colored}_{\texttt{n}}\ \texttt{:-}\ \texttt{c}_{\texttt{n},j}\,. & \text{for } \texttt{n} \in N \text{ and } j = 1..|N| \\ \texttt{viol}\ \texttt{:-}\ \texttt{c}_{\texttt{n},j}, c_{\texttt{m},j}\,. & \text{for } \langle \texttt{n},\texttt{m}\rangle \in E, \text{ and } j = 1..|N| \\ \texttt{use\_color}_j\ \texttt{:-}\ \texttt{c}_{\texttt{n},j}\,. & \text{for } \texttt{n} \in N,\ j = 1..|N| \end{array} \right\}$$

and let S_{DR} be the following $M = |N| + |N|^2 + |N| + |E| \times |N|^2 + |N|^3$ query/answer pairs:

$$\left\{ \begin{array}{lll} \langle \texttt{use\_color}_j;\ \texttt{No} \rangle & \text{for } j = 1..|N| & \\ \langle \texttt{colored}_n;\ \texttt{Yes} \rangle & \text{for } n \in N & \text{(Ask each of these queries } |N| \text{ times)} \\ \langle \texttt{viol};\ \texttt{No} \rangle & & \text{(Ask this query } |N| \text{ times)} \\ \langle \texttt{viol}\ \texttt{:-}\ \texttt{c}_{\texttt{n},j}, c_{\texttt{m},j};\ \texttt{Yes} \rangle & \text{for } \langle \texttt{n},\texttt{m}\rangle \in E,\ j = 1..|N| & \text{(Ask each of these queries } |N| \text{ times)} \\ \langle \texttt{use\_color}_j\ \texttt{:-}\ \texttt{c}_{\texttt{n},j};\ \texttt{Yes} \rangle & \text{for } \texttt{n} \in N,\ j = 1..|N| & \text{(Ask each of these queries } |N| \text{ times)} \end{array} \right\}$$

As in Theorem 5 above, we show that there is a theory $\mathrm{T}_C \in \Upsilon^{-R}[\mathrm{T}_{DR}]$ whose error is $\textsc{Err}(\,\mathrm{T}_C\,) = C/M$, iff there is a solution to the $\textsc{MinColor}$ problem G using C colors. This proof involves first observing $\mathrm{T}_C(\texttt{viol})$ must be $\texttt{No}$, as otherwise $\textsc{Err}(\,\mathrm{T}_C\,)$ will be at least $|N|/M$, which exceeds the allowed C/M. Similarly T_C must include each "$\texttt{viol}\ \texttt{:-}\ \texttt{c}_{\texttt{n},j}, c_{\texttt{m},j}$" rule, as otherwise its error will be at least $|N|/M$, due to the fourth set of queries. A similar argument prevents T_C from excluding any of the "$\texttt{use\_color}_j\ \texttt{:-}\ \texttt{c}_{\texttt{n},j}$" rules. As removing any "$\texttt{colored}_{\texttt{n}}\ \texttt{:-}\ \texttt{c}_{\texttt{n},j}$" rule is detrimental, we can assume that T_C is formed by deleting only atomic $\texttt{c}_{\texttt{n},j}$ clauses, until only one such literal remains for each $\texttt{n}$. The rest of the proof is isomorphic to (the end of) Theorem $5(a)$.

The same arguments show that adding antecedents to any non-atomic clause is problematic, leading to a proof that involves simply adding unsatisfied ℓ_{false} antecedents to various

$\mathtt{c}_{\mathtt{n},j}$ clauses — all but one, for each $\mathtt{n}$ — which shows that $\textsc{MinThRev}_{Prop,Horn}[\Upsilon^{-R,+A}]$ and $\textsc{MinThRev}_{Prop,Horn}[\Upsilon^{+A}]$ are not-approximatable.

(1a): To deal with $\textsc{MinThRev}_{PredCal,Atom}[\Upsilon^{-R}]$, use the theory

$$\mathrm{T}_{DR}{}' \quad = \quad \left\{ \begin{array}{ll} \mathtt{c}_{\mathtt{n},j}\mathtt{(0)}. \quad \mathtt{c}_{\mathtt{n},j}\mathtt{(1)}. & \text{for } \mathtt{n} \in N \text{ and } j = 1..|N| \\ \mathtt{colored}_{\mathtt{n}}\mathtt{(X)} \ \mathtt{:-}\ \mathtt{c}_{\mathtt{n},j}\mathtt{(X)}. & \text{for } \mathtt{n} \in N \text{ and } j = 1..|N| \\ \mathtt{viol(X)}\ \mathtt{:-}\ \mathtt{c}_{\mathtt{n},j}\mathtt{(X)},\ \mathtt{c}_{\mathtt{m},j}\mathtt{(X)}. & \text{for } \langle \mathtt{n},\mathtt{m}\rangle \in E, \text{ and } j = 1..|N| \\ \mathtt{use\_color}_j\mathtt{(X)}\ \mathtt{:-}\ \mathtt{c}_{\mathtt{n},j}\mathtt{(X)}. & \text{for } \mathtt{n} \in N,\ j = 1..|N| \end{array} \right\}$$

and labeled queries

$$S'_{DR} \quad = \quad \left\{ \begin{array}{lll} \langle \mathtt{use\_color}_j\mathtt{(0)};\ \mathtt{No}\rangle & \text{for } j = 1..|N| & \\ \langle \mathtt{colored}_n\mathtt{(0)};\ \mathtt{Yes}\rangle & \text{for } n \in N & \text{(Ask each of these queries } |N| \text{ times)} \\ \langle \mathtt{viol(0)};\ \mathtt{No}\rangle & & \text{(Ask this query } |N| \text{ times)} \\ \langle \mathtt{viol(1)};\ \mathtt{Yes}\rangle & & \text{(Ask this query } |N| \text{ times)} \\ \langle \mathtt{use\_color}_j\mathtt{(1)};\ \mathtt{Yes}\rangle & \text{for } j = 1..|N| & \text{(Ask each of these queries } |N| \text{ times)} \end{array} \right\}$$

Here, the role of the $\langle \mathtt{viol(1)};\ \mathtt{Yes}\rangle$ and $\langle \mathtt{use\_color}_j\mathtt{(1)};\ \mathtt{Yes}\rangle$ queries are to prevent us from deleting either the "$\mathtt{viol(X)}\ \mathtt{:-}\ \mathtt{c}_{\mathtt{n},j}\mathtt{(X)},\ \mathtt{c}_{\mathtt{m},j}\mathtt{(X)}.$" or the "$\mathtt{use\_color}_j\mathtt{(X)}\ \mathtt{:-}\ \mathtt{c}_{\mathtt{n},j}\mathtt{(X)}.$" rules.

We can now re-use the same proofs presented above to show that we cannot approximate any of $\textsc{MinThRev}_{PredCal,Atom}[\Upsilon^{-R}]$, $\textsc{MinThRev}_{PredCal,Atom}[\Upsilon^{-R,+A}]$ or $\textsc{MinThRev}_{PredCal,Atom}[\Upsilon^{+A}]$.

(2a): To deal with $\textsc{MinThRev}_{PredCal,Atom}[\Upsilon^{+R}]$, use

$$\mathrm{T}_{AR} \quad = \quad \{\mathtt{viol(X,\ Y)}\ \mathtt{:-}\ \mathtt{c(X,\ Z)},\ \mathtt{c(Y,\ Z)}.\}$$

and

$$S_{AR} \quad = \quad \left\{ \begin{array}{lll} \langle \mathtt{c(}j\mathtt{,\ X)};\ \mathtt{Yes}[X/?]\rangle & \text{for } j = 1..|N| & \text{(Ask each of these queries } |N| \text{ times)} \\ \langle \mathtt{viol(i,j)};\ \mathtt{No}\rangle & \text{for } \langle \mathtt{n}_i, \mathtt{n}_j\rangle \in E & \text{(Ask each of these queries } |N| \text{ times)} \\ \langle \mathtt{c(Y,\ }j\mathtt{)};\ \mathtt{No}\rangle & \text{for } j = 1..|N| & \end{array} \right\}$$

Here, if there is a coloring $f: N \mapsto \{1, \ldots, C\}$ that uses C colors, we can form a theory $\mathrm{T}' \in \Upsilon^{+R}[\mathrm{T}_{AR}]$ by adding the n atomic clauses, $\mathtt{c(}j\mathtt{,}\ \ f(n_j)\mathtt{)}$; and vice versa.

We can re-use this to address $\textsc{MinThRev}_{PredCal,Atom}[\Upsilon^{+R,-A}]$ and $\textsc{MinThRev}_{PredCal,Atom}[\Upsilon^{-A}]$.

(2b): We use a propositional variant of the above proof to handle $\textsc{MinThRev}_{Prop,Disj}[\Upsilon^{+R}]$: Here, $\mathrm{T}'_{AR} = \{\}$ and S'_{AR} is

$$\left\{ \begin{array}{lll} \langle \mathtt{c}_{1,k} \vee \mathtt{c}_{2,k} \vee \cdots \vee \mathtt{c}_{|N|,k};\ \mathtt{No}\rangle & \text{for } k = 1..|N| & \\ \langle \mathtt{c}_{j,1} \vee \mathtt{c}_{j,2} \vee \cdots \vee \mathtt{c}_{j,|N|};\ \mathtt{Yes}\rangle & \text{for } j = 1..|N| & \text{(Ask each of these queries } |N| \text{ times)} \\ \langle \mathtt{viol}_{i,j};\ \mathtt{No}\rangle & \text{for } \langle \mathtt{n}_i, \mathtt{n}_j\rangle \in E & \text{(Ask each of these queries } |N| \text{ times)} \\ \langle \mathtt{viol}_{i,j}\ \mathtt{:-}\ \mathtt{c}_{i,k},\ \mathtt{c}_{j,k};\ \mathtt{Yes}\rangle & \text{for } \langle \mathtt{n}_i, \mathtt{n}_j\rangle \in E,\ k = 1..|N| & \text{(Ask each of these queries } |N| \text{ times)} \end{array} \right\}$$

Again, if there is an coloring $f: N \mapsto \{1, \ldots, C\}$ that uses C colors, we can form a theory $\mathrm{T}' \in \Upsilon^{+R}[\mathrm{T}_{AR}{}']$ by adding the n atomic clauses $\mathtt{c}_{j,f(n_j)}$ (together with the $|E| \times |N|$ $\mathtt{viol}_{i,j}$ `:-` $\mathtt{c}_{i,k}$`,` $\mathtt{c}_{j,k}$ clauses) and vice versa; and again, we can re-use the proof to deal with $\textsc{MinThRev}_{Prop,Disj}[\Upsilon^{+R,-A}]$ and $\textsc{MinThRev}_{Prop,Disj}[\Upsilon^{-A}]$.

(3): The proof for $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^{+R=K}]$ is identical to the proof of Theorem 5(c). The proofs for $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^{-A=K}]$ and $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^{+R=K,-A=K}]$ are similar.

To handle $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^{-R=K}]$: Use as initial theory $\mathrm{T}_C{}' = \mathrm{T}_C \cup \{\mathtt{cn}_{,j}\,\mathtt{.}\}_{n\in N,\ j=1..|N|}$, which includes all $|N|^2$ $\mathtt{cn}_{,j}$ literals, and let $K = |N|^2 - |N|$. Here, each plausible solution involves deleting all but $|N|$ of $\mathtt{cn}_{,j}$ literals, leaving one for each `n`. The proofs for $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^{+A=K}]$ and $\textsc{MinThRev}_{Prop,Atom}[\Upsilon^{-R=K,+A=K}]$ are similar.

□ Theorem 7

Theorem 8 *For each* $\Upsilon^\dagger \in \{\Upsilon^{-R,+A},\ \Upsilon^{-R},\ \Upsilon^{+A},\ \Upsilon^{+R,-A},\ \Upsilon^{+R},\ \Upsilon^{-A}\}$,

$$\exists B_\dagger \in \mathrm{Poly}(\,\textsc{MaxThRev}[\Upsilon^\dagger]\,), \quad \mathit{MaxPerf}[\textsc{MaxThRev}[\Upsilon^\dagger]](\,B_\dagger, x\,) \;\le\; 2$$

Proof: [17] Consider first the $\textsc{MaxThRev}[\mathcal{S}]$ situation, for any $\mathcal{S} \in \{\Upsilon^{-R},\ \Upsilon^{+A},\ \Upsilon^{-R,+A}\}$. The following 2×2 grid partitions the set of queries

$$\begin{array}{r|cc} & \multicolumn{2}{c}{\mathrm{T}(q) =} \\ & \mathtt{Yes} & \mathtt{No} \\ \hline O(\,q\,) = \mathtt{Yes} & Q_{YY} & Q_{YN} \\ O(\,q\,) = \mathtt{No} & Q_{NY} & Q_{NN} \end{array}$$

Let $p_i \;=\; Pr[\,Q_i\,]$ be the probably of encountering a query in the class Q_i. (In the predicate calculus case, we actually require that $\mathrm{T}(q) = O(\,q\,)$ for each $q \in Q_{YY}$ — *i.e.*, the binding lists must match. The Q_{YN} set contains the other queries q, for which either $\mathrm{T}(q) = \mathtt{No}$ or $\mathrm{T}(q) = \mathtt{Yes}[\cdot]$ but $\mathrm{T}(q) \ne O(\,q\,)$.) The accuracy of the initial T is $\mathrm{A}(\,\mathrm{T}\,) = p_{YY} + p_{NN}$. The optimal possible accuracy, for any $\mathrm{T}_j \in \mathcal{S}(T)$ is $\mathrm{A}(\,\mathrm{T}_{opt}\,) = opt_{\mathrm{MaxThRevS}}[\langle T, S\rangle] \le p_{NN} + p_{YY} + p_{NY}$, as any T_j, being weaker than T, can only entail fewer conclusions; *i.e.*, if $\mathrm{T}(q) = \mathtt{No}$, then $\mathrm{T}_j(q) = \mathtt{No}$ for any weaker T_j.

If we remove *all* of T's propositions (or equivalently, add a new unsatisfied literal as a new antecedent to each clause), the resulting degenerate $\mathrm{T}_\phi = \{\}$ would have an accuracy score of $p_{NY} + p_{NN}$ (as it would no longer be able to derive any of the conclusions in Q_{NY}, which is desired). Now let $B(\cdot)$ be the best possible polynomial time algorithm; *i.e.*, the algorithm that, given any $\langle \mathrm{T}, S\rangle$ can produce the revised $B(\langle \mathrm{T}, S\rangle) = \mathrm{T}^* \in \mathcal{S}(T)$ with the best score over all polynomial time algorithms. Notice trivially that $\mathrm{A}(\,B(\mathrm{T})\,) \ge \max\{p_{YY} + p_{NN},\ p_{NY} + p_{NN}\} = p_{NN} + \max\{p_{YY}, p_{NY}\}$, as $B(\cdot)$ could simply leave T as it was, or delete all of its clauses. Hence, $\frac{opt(T)}{A(B(T))} \le \frac{p_{NN}+p_{YY}+p_{NY}}{p_{NN}+\max\{p_{YY},p_{NY}\}} \le \frac{p_{YY}+p_{NY}}{\max\{p_{YY},p_{NY}\}} \le 2$, as claimed.

For the $\textsc{MaxThRev}[\mathcal{G}]$ situation, for any $\mathcal{G} \in \{\Upsilon^{+R},\ \Upsilon^{-A},\ \Upsilon^{+R,-A}\}$, just use the observation that adding a new rule (or deleting an existing antecedent) can only cause previously

[17] I am indebted to Tom Hancock for this construction.

underivable queries to be derivable, while those that could already be derived remain derivable. Hence, we need only reverse the roles of the $\mathrm{T}(q) = \texttt{Yes}$ and $\mathrm{T}(q) = \texttt{No}$ columns. □ (Theorem 8)

References

[AGM85] Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50:510–530, 1985.

[BCH90] E. Boros, Y. Crama, and P.L. Hammer. Polynomial-time inference of all valid implications for horn and related formulae. *Annals of Mathematics and Artificial Intelligence*, 1:21–32, 1990.

[BE89] Alex Borgida and David Etherington. Hierarchical knowledge bases and efficient disjunctive reasoning. In *Proceedings of KR-89*, pages 33–43, Toronto, May 1989.

[BEHW89] Anselm Blumer, Andrzei Ehrenfeucht, David Haussler, and Manfred Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the Association for Computing Machinery*, 36(4):929–965, October 1989.

[BFOS84] L. Breiman, J. Friedman, J. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[BM93] Paul T. Baffes and Raymond J. Mooney. Symbolic revision of theories with M-of-N rules. In *Proceedings of IJCAI-93*, August 1993.

[Bol85] B. Bollobás. *Random Graphs*. Academic Press, 1985.

[Bou93] C. Boutilier. Revision sequences and nested conditionals. In *Proceedings of IJCAI-93*, pages 519–525, 1993.

[Che52] Herman Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sums of observations. *Annals of Mathematical Statistics*, 23:493–507, 1952.

[Cla78] K. Clark. Negation as failure. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 293–322. Plenum Press, New York, 1978.

[Coh90] William W. Cohen. Learning from textbook knowledge: A case study. In *Proceeding of AAAI-90*, 1990.

[Coh92] William W. Cohen. Abductive explanation-based learning: A solution to the multiple inconsistent explanation problems. *Machine Learning*, 8(2):167–219, March 1992.

[Coh95a] William W. Cohen. PAC-learning recursive logic programs: Efficient algorithms. *Journal of Artificial Intelligence Research*, 2:500–539, 1995.

[Coh95b] William W. Cohen. PAC-learning recursive logic programs: Negative results. *Journal of Artificial Intelligence Research*, 2:541–573, 1995.

[Coh96] William W. Cohen. PAC-learning non-recursive prolog clauses. *Artificial Intelligence*, 79(1):1–38, 1996.

[CP91] P. Crescenzi and A. Panconesi. Completeness in approximation classes. *Information and Computation*, 93(2):241–62, 1991.

[CS90] Susan Craw and Derek Sleeman. Automating the refinement of knowledge-based systems. In L.C. Aiello, editor, *Proceedings of ECAI 90*. Pitman, 1990.

[Dal88] Mukesh Dalal. Investigations into a theory of knowledge base revision: Preliminary report. In *Proceedings of AAAI-88*, pages 475–479, 1988.

[DE92] Mukesh Dalal and David Etherington. Tractable approximate deduction using limited vocabulary. In *Proceedings of the Ninth Canadian Conference on Artificial Intelligence*, Vancouver, May 1992.

[DMR92] S. Dzeroski, S. Muggleton, and S. Russell. PAC-learnability of determinate logic programs. In *Proceedings of the Fifth Workshop on Computational Learning Theory*, Pittsburgh, 1992.

[DP91] Jon Doyle and Ramesh Patil. Two theses of knowledge representation: Language restrictions, taxonomic classification, and the utility of representation services. *Artificial Intelligence*, 48(3), 1991.

[DP92] Rina Dechter and Judea Pearl. Structure identification in relational data. *Artificial Intelligence*, 58(1–3):237–270, 1992.

[DP94] A. Darwiche and J. Pearl. On the logic of iterated belief revision. In *TARK-94*, pages 5–23, 1994.

[EG92] T. Eiter and G. Gottlob. On the complexity of propositional knowledge base revison, updates and counterfactuals. *Artificial Intelligence*, 57:227–270, 1992.

[EH89] Andrzei Ehrenfeucht and David Haussler. A general lower bound on the number of examples needed for learning. *Inform. Comput.*, 82(3):247–251, September 1989.

[FH96] N. Friedman and J. Halpern. Belief revision: A critique. In *KR-96*, 1996.

[FL94] M. Freund and D. Lehmann. Belief revision and rational inference. Technical Report TR-94-16, Hebrew University, 1994.

[FP93] Michael Frazier and Leonard Pitt. Learning from entailment: An application to propositional horn sentences. In *Proceedings of IML-93*, pages 120–27. Morgan Kaufmann, 1993.

[Gar88] Peter Gardenfors. *Knowledge in Flux: Modeling the Dynamics of the Epistemic States*. Bradford Book, MIT Press, Cambridge, MA, 1988.

[GGK97] Russell Greiner, Adam Grove, and Alex Kogan. Knowing what doesn't matter: Exploiting the omission of irrelevant data. *Artificial Intelligence*, December 1997. http://www.cs.ualberta.ca/~greiner/PAPERS/superfluous-journal.ps.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.

[GPS94] G. Gogic, C. H. Papadimitriou, and M. Sideri. Incremental recompilation of knowledge. In *Proceedings of AAAI-94*, pages 922–927, 1994.

[Gre99] Russell Greiner. The complexity of revising logic programs. *Journal of Logic Programming*, 1999, to appear. http://www.cs.ualberta.ca/~greiner/PAPERS/impure.ps.

[GS92] Russell Greiner and Dale Schuurmans. Learning useful horn approximations. In B. Nebel, C. Rich, and W. Swartout, editors, *Proceedings of KR-92*, San Mateo, CA, October 1992. Morgan Kaufmann. http://www.cs.ualberta.ca/~greiner/PAPERS/horn.ps.

[Hau88] David Haussler. Quantifying inductive bias: AI learning algorithms and Valiant's learning framework. *Artificial Intelligence*, pages 177–221, 1988.

[Hec95] David E. Heckerman. A tutorial on learning with bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, 1995.

[Hin89] Geoff Hinton. Connectionist learning procedures. *Artificial Intelligence*, 40(1–3):185–234, September 1989.

[HRJ94] Frederick Hayes-Roth and Neil Jacobstein. Knowledge engineering systems. *Communication of the ACM*, pages 27–39, March 1994.

[Kan92] Viggo Kann. *On the Approximability of NP-Complete Optimization Problems*. PhD thesis, Royal Institute of Technology, Stockholm, 1992.

[KKS93] Henry Kautz, Michael Kearns, and Bart Selman. Reasoning with characteristic models. In *AAAI-93*, pages 34–39, 1993.

[KKS95] Henry Kautz, Michael Kearns, and Bart Selman. Horn approximations of empirical data. *Artificial Intelligence*, 74:129–145, 1995.

[KM91] Hirofumi Katsuno and Alberto Mendelzon. On the difference between updating a knowledge base and revising it. In *Proceedings of KR-91*, pages 387–94, Boston, April 1991.

[KR94a] Roni Khardon and Dan Roth. Learning to reason. In *AAAI-94*, pages 682–687, 1994.

[KR94b] Roni Khardon and Dan Roth. Reasoning with models. In *AAAI-94*, pages 1148–1153, 1994.

[KS90] Michael Kearns and Robert E. Shapire. Efficient distribution-free learning of probabilistic concepts. In *Proceedings of the 31st Symposium on Foundation of Computer Science*, October 1990.

[KSS92] M. J. Kearns, R. E. Schapire, and L. M. Sellie. Toward efficient agnostic leaning. In *Proceedings COLT-92*, pages 341–352. ACM Press, 1992.

[LDRG94] Pat Langley, George Drastal, R. Bharat Rao, and Russell Greiner. Theory revision in fault hierarchies. In *Proceedings of The Fifth International Workshop on Principles of Diagnosis (DX-94)*, New Paltz, NY, 1994. http://www.cs.ualberta.ca/~greiner/PAPERS/th-rev.ps.

[Lev84] Hector J. Levesque. Foundations of a functional approach to knowledge representation. *Artificial Intelligence*, 23:155–212, 1984.

[LMR88] Nathan Linial, Yishay Mansour, and Ronald Rivest. Results on learnability and the Vapnik-Chervonenkis dimension. In *Proceedings of COLT-88*, 1988.

[LV91] Charles X.F. Ling and Marco Valtorta. Some results on the computational complexity of refining certainty factors. *International Journal of Approximate Reasoning*, 5:121–148, 1991.

[LV95] Charles X.F. Ling and Marco Valtorta. Refinement of uncertain rule bases via reduction. *International Journal of Approximate Reasoning*, 13:95–126, 1995.

[LY93] Carsten Lund and Mihalis Yannakakis. On the hardness of approximating minimization problems. In *Proceeding of Twenty-fifth Annual ACM Symposium on Theory of Computation (STOC-93)*, pages 286–93, 1993.

[MB88] S. Muggleton and W. Buntine. Machine invention of first order predicates by inverting resolution. In *Proceedings of IML-88*, pages 339–351. Morgan Kaufmann, 1988.

[Moo94] Raymond Mooney. A preliminary PAC analysis of theory revision. In T. Petsche and S. Hanson, editors, *Third Annual Workshop on Computational Learning Theory and Natural Learning Systems (CLNL-92)*. MIT Press, 1994.

[Mug92] S.H. Muggleton. *Inductive Logic Programming*. Academic Press, 1992.

[OM94] Dirk Ourston and Raymond J. Mooney. Theory refinement combining analytical and empirical methods. *Artificial Intelligence*, 66(2):273–310, 1994.

[Plo71] G. D. Plotkin. *Automatic Methods of Inductive Inference*. PhD thesis, University of Edinburgh, 1971.

[Pol85] P.G. Politakis. *Empirical Analysis for Expert Systems*. Pitman Research Notes in Artificial Intelligence, 1985.

[Qui90] J. Ross Quinlan. Learning logical definitions from relations. *Machine Learning Journal*, 5(3):239–66, August 1990.

[Qui92] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, 1992.

[RBK88] R. Ramakrishman, C. Beeri, and R. Krishnamurthy. Optimizing existential datalog queries. In *Proc. of 7th Symposium on Principles of Database Systems*, pages 89–102, Austin, TX, March 1988.

[Sha83] Ehud Shapiro. *Algorithmic Program Debugging*. MIT Press, 1983.

[SK91] Bart Selman and Henry Kautz. Knowledge compilation using horn approximations. In *Proceedings of AAAI-91*, pages 904–09, Anaheim, August 1991.

[Tow91] Geoff Towell. *Symbolic Knowledge and Neural Networks: Insertion, Refinement and Extraction*. PhD thesis, University of Wisconsin, Madison, 1991.

[Vap82] V.N. Vapnik. *Estimation of Dependencies Based on Empirical Data*. Springer-Verlag, New York, 1982.

[WM94] David C. Wilkins and Yong Ma. The refinement of probabilistic rule sets: sociopathic interactions. *Artificial Intelligence*, 70:1–32, 1994.

[WP93] James Wogulis and Michael J. Pazzani. A methodology for evaluating theory revision systems: Results with Audrey II. In *Proceedings of IJCAI-93*, pages 1128–1134, 1993.