

Chapter 1

The Semantic Complexity of some Fragments of English

IAN PRATT-HARTMANN

ABSTRACT. By a fragment of a natural language we mean a subset of that language equipped with a semantics which translates its sentences into some formal system such as first-order logic. The familiar concepts of satisfiability and entailment can be defined for any such fragment in a natural way. The question therefore arises, for any given fragment of natural language, as to the computational complexity of determining satisfiability and entailment within that fragment. This paper presents some new technical results concerning a series of fragments of English for which the satisfiability problem lies in various complexity classes. The paper thus represents a case study in how to approach the problem of determining the semantic complexity of certain natural language constructions.

1.1 Introduction

This paper presents new results on the semantic complexity of various fragments of English. A *fragment* of English is a set of English sentences equipped with a semantics translating those sentences into a formal language such as first-order logic. To investigate the *semantic complexity* of such a fragment is to determine the computational complexity of the satisfiability and entailment problems for sentences in that fragment, as defined by the associated semantics.

We begin with a simple example. Consider the following context-free grammar. This grammar defines a set of English sentences by successive expansion of nonterminals, with the expressions on the right of the obliques indicating the semantic values of the corresponding phrases in the familiar way.

Syntax

Formal lexicon

| | |
|--|--|
| $IP/\phi(\psi) \rightarrow NP/\phi, I'/\psi$ | $Det/\lambda p\lambda q[\exists x(p(x) \wedge q(x))] \rightarrow$ some |
| $I'/\phi \rightarrow$ is a N'/ϕ | $Det/\lambda p\lambda q[\forall x(p(x) \rightarrow q(x))] \rightarrow$ every |
| $I'/\neg\phi \rightarrow$ is not a N'/ϕ | $Det/\lambda p\lambda q[\forall x(p(x) \rightarrow \neg q(x))] \rightarrow$ no |
| $NP/\phi \rightarrow$ PropN/ ϕ | |
| $NP/\phi(\psi) \rightarrow$ Det/ $\phi, N'/\psi$ | |
| $N'/\phi \rightarrow$ N/ ϕ . | |

Content lexicon

| | |
|--|--|
| $N/\lambda x[\text{man}(x)] \rightarrow$ man | $PropN/\lambda p[p(\text{socrates})] \rightarrow$ Socrates |
| $N/\lambda x[\text{mortal}(x)] \rightarrow$ mortal | $PropN/\lambda p[p(\text{diogenes})] \rightarrow$ Diogenes |
| ... | ... |

Since the primary form-determining element in this fragment of English is the copula, let us denote the fragment by Cop. The content-lexicon, comprising the open word-classes of common and proper nouns, is assumed to be open-ended. Thus, Cop can be viewed as the union of a *family* of fragments, with each member of that family corresponding to a choice of content-lexicon. In this respect, the notion of a fragment of English resembles the logician's familiar notion of a fragment of a formal language such as first-order logic.

The set of sentences recognized by Cop is, in effect, the familiar language of the syllogism; and the formulas to which Cop translates those sentences are the familiar formal translations found in introductory logic courses. For example, Cop contains the sentences featured in the following argument, and assigns them the corresponding logical translations.

| | |
|--------------------------|---|
| Every man is a mortal | $\forall x(\text{man}(x) \rightarrow \text{mortal}(x))$ |
| <u>Socrates is a man</u> | <u>$\text{man}(\text{socrates})$</u> |
| Socrates is a mortal | $\text{mortal}(\text{socrates})$ |

This translation allows familiar semantic concepts to be transferred from first-order logic to the fragment Cop in the obvious way. Thus, a set of Cop-sentences E can be said to *entail* a Cop-sentence e if the formulas to which E is translated entail the formula to which e is translated in the usual sense of first-order logic; likewise, a set of Cop-sentences E can be said to be *satisfiable* if the set of formulas to which E is translated is satisfiable in the usual sense of first-order logic. The philosophical justification of this rational reconstruction of entailment and satisfiability need not detain us here, since, for the applications we have in mind, it is not open to productive doubt.

Define the *size* of an English sentence to be the number of words it contains; likewise, define the size of a set E of sentences, denoted $|E|$, to be the sum of the sizes of its members. Using this concept of size, we can formulate complexity-theoretic questions concerning fragments of English in the usual way. In particular, the computational complexity of the satisfiability problem for an English fragment is the number of steps of computation required to determine algorithmically whether a given finite set E of sentences in that fragment is satisfiable, expressed as a function of $|E|$.

Theorem 1. *The problem of determining the satisfiability of a set of sentences in Cop is in PTIME.*

Proof. Every sentence recognized by Cop is easily seen to have a first-order translation matching one of the schemata $\forall x(p(x) \rightarrow \pm q(x))$, $\exists x(p(x) \wedge \pm q(x))$, or $\pm p(c)$, where p and q are predicates and c is a constant. (We take $\pm\phi$ to stand indeterminately for ϕ or $\neg\phi$.) Clausifying such formulas results in function-free clauses matching either of the schemata $\neg p(x) \vee \pm q(x)$ or $\pm p(c)$. Since resolution only produces more clauses of this form, saturation is reached in PTIME. \square

This simple observation suggests a programme of work: take a fragment of English delineated in terms which respect the syntax of the language; then determine the computational complexity of deciding satisfiability in that fragment, if, indeed, the fragment is decidable. From this standpoint, the syllogistic is just one such fragment, with very restricted syntax and a correspondingly efficient decision procedure. In the sequel, we shall investigate what happens as we expand our syntactic horizons. Throughout the paper, we make extensive use of the standard apparatus of resolution theorem-proving, including the notions of *clause*, *A-ordering*, *ordered resolution*, and *splitting*. For the definitions of these terms, the reader is referred to a standard text such as Leitsch (1997).

1.2 Relative clauses

Let Cop+Rel be the fragment defined by the grammar rules of Cop together with the following rules. For the sake of avoiding clutter, here and in the sequel, we suppress the semantic annotations on these rules, since these are completely routine.

| Syntax | | Formal lexicon |
|----------------------------|-------------------------|---------------------------------|
| CP \rightarrow CSpec, C' | N' \rightarrow N, CP | RelPro \rightarrow who, which |
| C' \rightarrow C, IP | NP \rightarrow RelPro | C \rightarrow |
| CSpec \rightarrow | | |

In addition, we assume that, following generation of an IP by these rules, relative pronouns are subject to wh-movement to produce the observed word-order. For our purposes, we may take the wh-movement rule to require: (i) the empty position CSpec must be filled by movement of a RelPro from within the IP which forms its right-sister (i.e. which it C-commands); (ii) every RelPro must move to a closest such CSpec position.

As for Cop, so too for Cop+Rel, the semantics map its sentences to first-order logic, thus inducing natural definitions of satisfiability and validity. For example, Cop+Rel contains the sentences featured in the following argument, and assigns them the corresponding logical translations.

| | |
|---|--|
| Every man who is not a stoic is a cynic | $\forall x(\text{man}(x) \wedge \neg \text{stoic}(x) \rightarrow \text{cynic}(x))$ |
| Every stoic is a fool | $\forall x(\text{stoic}(x) \rightarrow \text{fool}(x))$ |
| Every cynic is a fool | $\forall x(\text{cynic}(x) \rightarrow \text{fool}(x))$ |
| Every man is a fool | $\forall x(\text{man}(x) \rightarrow \text{fool}(x))$ |

The following result shows us that determining satisfiability has become more difficult.

Theorem 2. *The problem of determining the satisfiability of a set of sentences in Cop+Rel is NP-complete.*

Proof. To show membership in NP, let E be a finite set of Cop+Rel-sentences, and let Φ be the corresponding first-order logic formulas. Since Φ has no nested quantifiers, it is obvious that if Φ is satisfiable, then it has a model whose size is bounded by $|\Phi|$.

To show NP-hardness, we reduce 3SAT to the problem of determining satisfiability in Cop+Rel. Let \mathcal{C} be a set of propositional clauses, each of which has at most three literals. Without loss of generality, we may assume all the clauses in \mathcal{C} to be of the forms $p \vee q$, $\neg p \vee \neg q$ or $\neg p \vee \neg q \vee r$. We then map each clause in \mathcal{C} to a Cop+Rel-sentence as follows. (The first-order translations of these sentences are also given.)

| | | |
|-----------------------------|----------------------------------|---|
| $p \vee q$ | Every el which is not a q is a p | $\forall x(\text{el}(x) \wedge \neg q(x) \rightarrow p(x))$ |
| $\neg p \vee \neg q$ | No p is a q | $\forall x(p(x) \rightarrow \neg q(x))$ |
| $\neg p \vee \neg q \vee r$ | Every p which is a q is an r | $\forall x(p(x) \wedge q(x) \rightarrow r(x))$ |

Finally, we add the sentence Some el is an el. (Read el as ‘element’.) Let the resulting set of Cop+Rel-sentences be E , and let the first-order translations of E be Φ . It is then easy to transform any satisfying assignment for \mathcal{C} into a model for Φ and vice versa. □

1.3 Adding transitive verbs

Whereas adding relative clauses to Cop increases computational complexity, other additions are computationally harmless. Perhaps the simplest involves the addition of transitive verbs. Let Cop+TV be the fragment defined by the grammar rules of Cop together with the following rules. (Again, we assume the obvious semantics.)

| Syntax | Formal Lexicon | Content Lexicon |
|--|-----------------------------------|----------------------------------|
| $I' \rightarrow VP$ | $Neg \rightarrow \text{does not}$ | $TV \rightarrow \text{admires}$ |
| $I' \rightarrow \text{NegP}$ | | $TV \rightarrow \text{despises}$ |
| $\text{NegP} \rightarrow \text{Neg}, VP$ | | ... |
| $VP \rightarrow TV, NP$ | | |

For the sake of clarity, we have suppressed the issue of verb-inflections as well as that of polarity effects of negative contexts on determiners. In the sequel, we will silently correct any such syntactic shortcomings as required. Accommodating these details in our grammar and ruling out otherwise awkward-sounding sentences can easily be seen not to affect the complexity-theoretic results reported below. For example, Cop+TV contains the sentences featured in the following argument, and assigns them the corresponding logical translations.

| | |
|---------------------------------|--|
| Every stoic admires every cynic | $\forall x(\text{sto}(x) \rightarrow \forall y(\text{cyn}(y) \rightarrow \text{adm}(x, y)))$ |
| No cynic admires any stoic | $\forall x(\text{cyn}(x) \rightarrow \neg \exists y(\text{sto}(y) \wedge \text{adm}(x, y)))$ |
| <u>No stoic is a cynic</u> | <u>$\forall x(\text{sto}(x) \rightarrow \neg \text{cyn}(x))$</u> |

Theorem 3. *The problem of determining the satisfiability of a set of sentences in Cop+TV is in PTIME.*

Proof. If E is a finite set of sentences of Cop+TV, let Φ be the corresponding set of first-order formulas, and let \mathcal{C} be the result of putting Φ into clausal form. It suffices to show that the satisfiability of \mathcal{C} can be computed in polynomial time.

It follows easily from the semantics of Cop+TV that every $C \in \mathcal{C}$ is of one of the following forms.

$$\begin{array}{llll} \pm p(c) & \pm r(c, d) & \neg p(x) \vee \pm r(x, f(x)) & \neg p(x) \vee \neg q(y) \vee \pm r(x, y) \\ \neg p(x) \vee q(f(x)) & \neg p(x) \vee \pm q(x) & \neg p(x) \vee \pm r(c, x) & \neg p(x) \vee \pm r(x, c) \end{array}$$

Call a literal *non-unary* if its predicate is not unary. Let \mathcal{C}^∞ be the result of saturating \mathcal{C} under resolution, *but only allowing non-unary literals to be resolved*

upon, and let \mathcal{D} be the result of removing from \mathcal{C}^∞ any clauses containing a non-unary literal. Since every $C \in \mathcal{C}$ contains at most 1 non-unary literal, we have $|\mathcal{D}| \leq |\mathcal{C}^\infty| \leq |\mathcal{C}| + |\mathcal{C}|^2$. We claim that \mathcal{C} and \mathcal{D} are equisatisfiable. Certainly, if \mathcal{C} is satisfiable then \mathcal{D} is. So suppose \mathcal{C} is unsatisfiable, and let \prec be the A-order defined by $A \prec B$ if A is unary and B is non-unary. Since \prec -resolution is a complete proof strategy, there must be a derivation of the empty clause from \mathcal{C} . But the ordering \prec ensures that, in this derivation (considered as a tree), no steps of resolution on unary literals can precede any step of resolution on non-unary literals. Hence there is a derivation of the empty clause from \mathcal{D} , and so \mathcal{D} is unsatisfiable.

Thus it suffices to show that the satisfiability of \mathcal{D} can be decided in PTIME. By definition, the clauses in \mathcal{D} involve only *unary* literals; and it is routine to verify that every $C \in \mathcal{D}$ satisfies the conditions:

P1: at most 2 variables and at most 4 literals occur in C ;

P2: the depth of C is at most 1;

P3: if C is non-negative, then C must be of one of the forms

$$p(c) \quad \neg p(x) \vee q(x) \quad \neg p(x) \vee q(f(x)). \quad (1.1)$$

(The *depth* of any expression X , denoted $d(X)$, is the maximum level of nesting of function-symbols in X , with non-functional expressions assigned depth 0.) Consider the familiar A-ordering \prec^d defined by setting $A \prec^d B$ if and only if: (i) $d(A) < d(B)$, (ii) $\text{Vars}(A) \subseteq \text{Vars}(B)$, and (iii) $d(x, A) < d(x, B)$ for all $x \in \text{Vars}(A)$. (Leitsch 1997, p. 100). It is routine to show that, when \prec^d -resolution is applied to the clauses \mathcal{D} , properties **P1–P3** are preserved. It follows from **P1** and **P2** that saturation is reached in PTIME. \square

If adding relative clauses increases complexity but adding transitive verbs does not, the question naturally arises as to what happens when both are added together. That is, let Cop+Rel+TV be the fragment defined by the grammar rules of Cop+Rel together with those of Cop+TV. For example, Cop+Rel+TV contains the sentences featured in the following argument, and assigns them the corresponding logical translations.

| | |
|--|---|
| Every stoic is a philosopher | $\forall x(\text{sto}(x) \rightarrow \text{phil}(x))$ |
| Every cynic whom some stoic admires is a cynic whom some philosopher admires | $\forall x(\text{cyn}(x) \wedge \exists y(\text{sto}(y) \wedge \text{adm}(y, x)) \rightarrow \text{cyn}(x) \wedge \exists y(\text{phil}(y) \wedge \text{adm}(y, x)))$ |

The result of this addition is a further jump in the complexity of reasoning.

Theorem 4. *The problem of determining the satisfiability of a set of sentences in Cop+Rel+TV is EXPTIME-complete.*

Proof. Essentially Pratt-Hartmann (forthcoming), Theorem 4.1. □

1.4 Adding ditransitive verbs

Let Cop+Rel+TV+DTV be the fragment defined by the grammar rules of Cop+Rel+TV, together with the following rules.

| Syntax | Content Lexicon |
|----------------------------------|--------------------------------------|
| VP \rightarrow DTV, NP, to, NP | DTV \rightarrow prefers . . . |

For example, Cop+Rel+TV+DTV contains the following sentence, and assigns it the corresponding logical translation.

Every stoic whom no sceptic prefers any cynic to admires every philosopher

$$\forall x(\text{sto}(x) \wedge \forall y(\text{sce}(y) \rightarrow \neg \exists z(\text{cyn}(z) \wedge \text{pref}(y, z, x)))) \rightarrow \forall y(\text{phil}(y) \rightarrow \text{adm}(x, y))$$

Theorem 5. *The problem of determining the satisfiability of a set of sentences in Cop+Rel+TV+DTV is NEXPTIME-complete.*

Proof. To show membership in NEXPTIME, let E be a finite set of Cop+Rel+TV+DTV-sentences. Let Φ be the first-order logic translations of E . Since the sentences in E contain only one main verb, and since all noun-phrases in this fragment translate to expressions with only one (λ -bound) variable, it is easy to transform Φ polynomially into an equisatisfiable set of clauses \mathcal{C} such that every $C \in \mathcal{C}$ contains at most one non-unary literal. Now let \mathcal{D} be the set of clauses defined exactly as in the proof of Theorem 3, so that \mathcal{C} and \mathcal{D} are equisatisfiable with $|\mathcal{D}| \leq |\mathcal{C}| + |\mathcal{C}|^2$. Thus, it suffices to show that the satisfiability of \mathcal{D} can be decided in nondeterministic exponential time.

By construction, \mathcal{D} involves only unary literals. And it is easy to check that, by splitting clauses if necessary, every clause $C \in \mathcal{D}$ has the properties:

- P1:** if C contains a ground literal, then C is ground;
- P2:** if a functional ground term occurs in C , then C is ground;
- P3:** C contains at most two variables;

P4: if C contains two variables x and y , then $d(C) = 1$; moreover, all binary function-symbols in C occur in atoms of the form $p(h(x, y))$, and C contains at least one such literal.

Now define the A-order \prec as follows. Let \prec^g be the ordering on ground atoms defined by setting $A \prec^g B$ iff A is a ground atom of the form $q(g(t))$ and B a ground atom of the form $p(f(t_1, t_2))$. Define \prec^d on the set of ground atoms by setting $A \prec^d A'$ iff either (i) $d(A) < d(A')$ or (ii) $d(A) = d(A')$ and $A \prec^g A'$. Finally, define \prec on the set of all atoms by setting $A \prec A'$ iff $A\theta \prec^d A'\theta$ for all ground substitutions θ . It is clear that \prec is an A-order.

Let $C \vee L$ and $C' \vee L'$ be clauses satisfying **P1–P4** which resolve under the A-ordering \prec with resolvent C'' . It is routine to show that $d(C'') \leq \max(d(C), d(C'))$ and that furthermore, by applying the splitting rule to C'' if necessary, we obtain clauses also satisfying **P1–P4**. Since the number of such clauses is exponential in $|D|$, saturation is reached, via exponentially many non-deterministic choice points, in exponentially many steps. Hence, the satisfiability problem for Cop+Rel+TV+DTV is in NEXPTIME.

We show next that satisfiability in Cop+Rel+TV+DTV is NEXPTIME-hard. To simplify the proof, we will allow ourselves to use the NPs something and everything in the fragment, with the obvious interpretation. (In fact, this facility is inessential.) Let us say that a formula is a *standard two-variable formula* if it can be written either as $\forall x \forall y \alpha$ or as $\forall x \exists y \alpha$, where α is a quantifier-free formula involving only unary predicates (and no functions or constants). It is well known that the satisfiability problem for a set of standard two-variable formulas is NEXPTIME-hard (Börger et al. 1997, pp. 253 ff). Any such problem can easily be reduced to the satisfiability problem for a set of clauses \mathcal{C} of the following form:

$$\begin{array}{ll} \neg p_{i1}(x) \vee \neg p_{i2}(y) \vee \neg p_{i3}(x, y) & (1 \leq i \leq n_1) \\ \neg q_{i1}(x, y) \vee \neg q_{i2}(x, y) \vee \neg q_{i3}(x, y) & (1 \leq i \leq n_2) \\ \neg r_i(x, f_i(x)) & (1 \leq i \leq n_3) \\ s_{i1}(x) \vee s_{i2}(x) & (1 \leq i \leq n_4) \\ t_{i1}(x, y) \vee t_{i2}(x, y) & (1 \leq i \leq n_5), \end{array}$$

where n_1, \dots, n_5 are non-negative integers, the subscripted p, q, r, s and t are predicates of the indicated arities, and the f_i ($1 \leq i \leq n_3$) are *pairwise distinct* function-symbols. It follows that the satisfiability problem for such sets of clauses is also NEXPTIME-hard.

For every binary predicate p appearing in \mathcal{C} , let p^+ be a new unary predicate. In addition, let n be a new unary predicate, c_0 a new individual constant and \oplus a new binary function-symbol. (Think of $x \oplus y$ as denoting the ordered pair $\langle x, y \rangle$,

and think of $n(x)$ as stating that x is a ‘normal’ element—i.e. not an ordered pair.)
Now let \mathcal{D} be the corresponding set of clauses:

$$\begin{array}{ll}
\neg n(x) \vee \neg n(y) \vee \neg p_{i1}(x) \vee \neg p_{i2}(y) \vee \neg p_{i3}^+(x \oplus y) & (1 \leq i \leq n_1) \\
n(z) \vee \neg q_{i1}^+(z) \vee \neg q_{i2}^+(z) \vee \neg q_{i3}^+(z) & (1 \leq i \leq n_2) \\
\neg n(x) \vee \neg r_i^+(x \oplus f_i(x)) & (1 \leq i \leq n_3) \\
\neg n(x) \vee n(f_i(x)) & (1 \leq i \leq n_3) \\
\neg n(x) \vee s_{i1}(x) \vee s_{i2}(x) & (1 \leq i \leq n_4) \\
t_{i1}^+(z) \vee t_{i2}^+(z) & (1 \leq i \leq n_5) \\
\neg n(x) \vee \neg n(y) \vee \neg n(x \oplus y), & n(c_0).
\end{array}$$

It is routine to show that \mathcal{C} and \mathcal{D} are equisatisfiable. Now let us further transform the clause-set \mathcal{D} . For each i ($1 \leq i \leq n_2$), let q_{i12}^+ be a new unary predicate and let d be a new ternary predicate. (Think of $q_{i12}^+(x)$ as standing for $q_{i1}^+(x) \wedge q_{i2}^+(x)$.)
Now let \mathcal{E} be the corresponding set of clauses:

$$\begin{array}{ll}
\neg n(x) \vee \neg n(y) \vee \neg p_{i1}(x) \vee \neg p_{i2}(y) \vee \neg p_{i3}^+(z) \vee d(x, y, z) & (1 \leq i \leq n_1) \\
n(z) \vee \neg q_{i12}^+(z) \vee \neg q_{i3}^+(z) & (1 \leq i \leq n_2) \\
\neg q_{i1}^+(z) \vee \neg q_{i2}^+(z) \vee q_{i12}^+(z) & (1 \leq i \leq n_2) \\
\neg n(x) \vee \neg r_i^+(z) \vee d(x, f_i(x), z) & (1 \leq i \leq n_3) \\
\neg n(x) \vee n(f_i(x)) & (1 \leq i \leq n_3) \\
\neg n(x) \vee s_{i1}(x) \vee s_{i2}(x) & (1 \leq i \leq n_4) \\
t_{i1}^+(z) \vee t_{i2}^+(z) & (1 \leq i \leq n_5) \\
\neg n(x) \vee \neg n(y) \vee \neg n(z) \vee d(x, y, z), & n(c_0) \\
\neg n(x) \vee \neg n(y) \vee \neg d(x, y, x \oplus y).
\end{array}$$

Again, the sets \mathcal{D} and \mathcal{E} are easily seen to be equisatisfiable. Finally, the set E of Cop+Rel+TV+DTV-sentences

Every p_{i1} which is an n ds every p_{i2} which is an n
to every p_{i3}^+ ($1 \leq i \leq n_1$)
Every q_{i12}^+ which is a q_{i3}^+ is an n ($1 \leq i \leq n_2$)
Every q_{i1}^+ which is a q_{i2}^+ is a q_{i12}^+ ($1 \leq i \leq n_2$)
Every n ds some n to every r_i^+ ($1 \leq i \leq n_3$)
Every n which is not an s_{i1} is an s_{i2} ($1 \leq i \leq n_4$)
Everything which is not a t_{i1}^+ is a t_{i2}^+ ($1 \leq i \leq n_4$)
Something is an n
Every n ds every n to every n
No n ds any n to everything.

translates to formulas which clausify to \mathcal{E} , up to renaming of of predicates and Skolem functions. (At this point we use the assumption that the f_i are pairwise distinct.) The NEXPTIME-hardness of Cop+Rel+TV+DTV then follows. \square

1.5 Anaphora

A more radical addition to the fragments considered above concerns pronouns and reflexives. Thus, for example, we might add to Cop+Rel+TV the following rules.

| Syntax | Formal lexicon |
|----------------|----------------------------------|
| NP → Reflexive | Reflexive → itself (him/herself) |
| NP → Pronoun | Pronoun → it (he/she/him/her) |

For simplicity, we shall always take pronouns and reflexives to have antecedents in the sentences in which they occur. That is to say: all anaphora is intra-sentential. Of course, we also assume the selection of such antecedents to be subject to the usual rules of binding theory, which, again, we need not rehearse here. Providing a formal semantics for pronouns and anaphora is rather more complicated than for the fragments considered above; however, from a complexity-theoretic point of view, these details may be safely ignored.

It is easy to see that adding the above grammar rules results in anaphoric ambiguities. For example, in the sentence

Every philosopher who admires a cynic despises every stoic who castigates him, the pronoun may take as antecedent either the NP headed by philosopher or the NP headed by cynic. (The NP headed by stoic is not available as a pronoun antecedent here.) We then have two options: either we resolve such ambiguities by fiat, or we decorate nouns and pronouns in these sentences with indices to record which pronouns take which NPs as antecedents.

Considering the former option, let the fragment Cop+Rel+TV+RA (RA for “restricted anaphora”) be the fragment defined by the grammar rules for Cop+Rel+TV together with the above rules for reflexives and pronouns, subject to the restriction that *pronouns must take their closest allowed antecedents*. Here, *closest* means “closest measured along edges of the phrase-structure” and *allowed* means “allowed by the principles of binding theory”. (We ignore case and gender agreement.) It turns out that this fragment corresponds closely to the two-variable fragment of first-order logic. Because of this correspondence, we have:

Theorem 6. *The problem of determining the satisfiability of a set of sentences in Cop+Rel+TV+RA is NEXPTIME-complete.*

Proof. See Pratt-Hartmann (2003), Corollaries 1 and 2. □

Turning attention now to the latter option for dealing with anaphoric ambiguity, let Cop+Rel+TV+GA (GA for “general anaphora”) be the fragment defined by the

| Fragment | Complexity of satisfiability |
|----------------|------------------------------|
| Cop | P |
| Cop+Rel | NP-complete |
| Cop+TV | P |
| Cop+Rel+TV | EXPTIME-complete |
| Cop+Rel+TV+DTV | NEXPTIME-complete |
| Cop+Rel+TV+RA | NEXPTIME-complete |
| Cop+Rel+TV+GA | undecidable |

Table 1.1: Lattice of fragments of English and their complexity classes

grammar rules for Cop+Rel+TV together with the above rules for reflexives and pronouns, with anaphoric antecedents indicated by coindexing in the usual way, subject only to the usual rules of binding theory.

Theorem 7. *The problem of determining the satisfiability of a set of sentences in Cop+Rel+TV+GA is undecidable.*

Proof. See Pratt-Hartmann (2003), Theorem 5. □

1.6 Conclusions and relation to other work

This paper has extended results obtained in Pratt-Hartmann (2003, forthcoming), where the general programme of determining the semantic complexity of fragments of natural languages is outlined. The new results reported here concern the complexity of the fragments Cop+TV and Cop+Rel+TV+DTV. The complexity of satisfiability for all of the fragments considered above is shown in Table 1.1.

Several authors have proposed formalisms based on logical constructions inspired by the syntax of natural language, apparently in the belief that such formalisms increase inferential efficiency. These include Fitch (1973), Suppes (1979), Purdy (1991) and ‘traditional’ logicians such as Englebretsen (1981) and Sommers (1982). But none of these analyses yields any immediate complexity-theoretic consequences of the sorts reported here. On the other hand, McAllester and Givan (1992) present a natural-language-inspired formal language with NP-complete satisfiability (PTIME in certain cases). However, this formal language is not shown to be generated by any linguistically natural fragment such as those considered above. More recently, Purdy (1996, 1999, 2002) has analysed the *fluted fragment* of first-order logic, alleging some (not completely specified) affinity between this

fragment and quantification in natural language. We note in this regard that the fragment Cop+Rel+TV+ DTV does not translate into the fluted fragment.¹

Bibliography

- Börger, E., E. Grädel, and Y. Gurevich (1997). *The Classical Decision Problem. Perspectives in Mathematical Logic*. Springer-Verlag, Berlin.
- Englebretsen, G. (1981). *Three Logicians*. Van Gorcum, Assen.
- Fitch, F. B. (1973). Natural deduction rules for English. *Philosophical Studies*, **24**:89–104.
- Leitsch, A. (1997). *The Resolution Calculus*. Springer, Berlin.
- McAllester, D. A. and R. Givan (1992). Natural language syntax and first-order inference. *Artificial Intelligence*, **56**:1–20.
- Pratt-Hartmann, I. (2003). A two-variable fragment of English. *Journal of Logic, Language and Information*, **12**:13–45.
- Pratt-Hartmann, I. (forthcoming). Fragments of language. *Language and Computation*.
- Purdy, W. C. (1991). A logic for natural language. *Notre Dame Journal of Formal Logic*, **32**(3):409–425.
- Purdy, W. C. (1996). Fluted formulas and the limits of decidability. *Journal of Symbolic Logic*, **61**(2):608–620.
- Purdy, W. C. (1999). Quine’s ‘limits of decision’. *Journal of Symbolic Logic*, **64**(4):1439–1466.
- Purdy, W. C. (2002). The complexity and nicety of fluted logic. *Studia Logica*, **71**:177–198.
- Sommers, F. (1982). *The Logic of Natural Language*. Clarendon Press, Oxford.
- Suppes, P. (1979). Logical inference in English: a preliminary analysis. *Studia Logica*, **38**:375–391.

¹The author wishes to thank Allan Third and Mark Steedman for their comments on the manuscript, and gratefully acknowledges the support of the EPSRC (grant reference GR/S22509) and the hospitality of the University of Edinburgh Institute for Communicating and Collaborative Systems.