Complexity Problems for Substructural Logics

Alasdair Urquhart

University of Toronto

July 18 2007

What is the complexity of the following problems?

1. The decision problem for pure relevant implication?
2. The reachability problem for Petri nets?
3. The decision problem for affine linear logic (linear logic with weakening)?

Common features

1. The best known upper bound is Ackermann (proved in essentially the same way in each case, using Dickson's Lemma)
2. The only known lower bound is exponential space, so the problems are indeed intractable.

Can't we reduce this huge gap?

The system $\mathbf{R}_{\rightarrow\wedge\circ}$

1. t
2. $t \rightarrow (A \rightarrow A)$
3. $(A \rightarrow B) \rightarrow [(B \rightarrow C) \rightarrow (A \rightarrow C)]$
4. $[A \rightarrow (B \rightarrow C)] \rightarrow [B \rightarrow (A \rightarrow C)]$
5. $[A \rightarrow (B \rightarrow C)] \rightarrow [(A \rightarrow B) \rightarrow (A \rightarrow C)]$
6. $(A \wedge B) \rightarrow A$
7. $(A \wedge B) \rightarrow B$
8. $[(A \rightarrow B) \wedge (A \rightarrow C)] \rightarrow [A \rightarrow (B \wedge C)]$
9. $A \rightarrow (B \rightarrow (A \circ B))$
10. $[A \rightarrow (B \rightarrow C)] \rightarrow [(A \circ B) \rightarrow C]$

The rules of the system are *modus ponens* and *adjunction*:

1. From $A \rightarrow B$ and A infer B (*modus ponens*);
2. From A and B infer $A \wedge B$ (*adjunction*).

Fusion is associative, commutative and square-increasing, and t forms the semigroup identity. Thus, using $(A \leftrightarrow B)$ as an abbreviation for $((A \rightarrow B) \wedge (B \rightarrow A))$, the following are theorems:

$$\begin{aligned} ((A \circ B) \circ C) &\leftrightarrow (A \circ (B \circ C)) \\ (A \circ B) &\leftrightarrow (B \circ A) \\ A &\rightarrow (A \circ A) \\ (t \circ A) &\leftrightarrow A \end{aligned}$$

The logic $\mathbf{R}_{\rightarrow}$ is decidable (Saul Kripke 1959); the extension to $\mathbf{R}_{\rightarrow\wedge\circ}$ is due to Robert K. Meyer in 1966.

Kripke's decision procedure

1. Formulate the system as a cutfree Gentzen system.
2. Absorb the contraction rule into the inference rules, with restrictions on the number of contractions that can be performed in a given step.
3. Prove the finiteness of the proof search tree by using a form of Dickson's Lemma.

Dickson's Lemma

A *multiset* can be written as a freely permutable word on a finite set of letters, for example $aabbccccd$ or $a^2b^2c^4d$. Let's say that one multiset *contracts* to another if the second can be derived from the first by applying the contraction rule.

Example: $a^2b^2c^4d$ contracts to abc^3d.

Dickson's Lemma: Let S be a set of multisets over a fixed finite set, in which no element of S contracts to any other. Then S is finite.

This is the key to proving the finiteness of the proof search tree for $\mathbf{R}_{\rightarrow}$ and $\mathbf{R}_{\rightarrow\wedge\circ}$.

Proof theory of Dickson's Lemma

Dickson's Lemma is the key step in proving the Hilbert basis theorem. Analysis by Steve Simpson in a paper of 1988 shows that it is unprovable in primitive recursive arithmetic.

Upper bounds on the complexity of the Kripke/Meyer algorithm employ an analysis by Ken McAloon of 1984. This shows that the size of the proof search tree is bounded by a function that is a version of the Ackermann function. The decision procedures for the other two problems are also based on Dickson's Lemma, and so have essentially the same upper bound.

Commutative semi-Thue system

is a finite set of rewrite rules for multisets.

Example: $a \longrightarrow aabb, bbc \longrightarrow c$.

The reachability problem for such a system is the problem, given a fixed word (multiset) α, whether another word β can be derived from it. This problem is equivalent to the reachability problem for Petri nets.
A commutative semigroup presentation includes the converses of all the rewrite rules. Mayr and Meyer (1982) proved that the word problem for finitely presented commutative semigroups is *exponential space complete*. The complexity of the general reachability problem is unsolved (this is our **Problem Number 2**).

The Mayr and Meyer lower argument adapts to prove exponential space lower bounds for the first and third problems.

Key idea: code the halting problem for a restricted version of a counter machine (register machine) as the word problem for a finitely presented commutative semigroup. The contents of the registers are bounded by 2^{2^n}, where n is the number of states of the machine.

Essential trick: simulating the zero-test capability of such a machine.

Using semi-Thue rules, we can test for the *presence* of a symbol in a word, but not its *absence*. Mayr and Meyer deal with this problem by:

1. Doubling the number of registers, so that each register has a shadow register;
2. Keeping the sum of the register and its shadow register constant;
3. Using a succinct compression/decompression algorithm to compress exponentially many tokens into a single token;
4. Testing to see if a given register is empty by seeing if the token representing the compressed set of tokens is in the shadow register.

The argument of Mayr and Meyer adapts to prove the exponential space lower bound for both linear affine logic and pure relevant implication. Basic idea: Represent a rewrite rule such as $a \longrightarrow aabb$ by an implication $A \rightarrow (A \circ A \circ B \circ B)$. Complications arise because of the general validity of $A \rightarrow (A \circ A)$, that is to say, we can arbitrarily increase the number of occurrences of any symbol in a word. This difficulty is overcome by proving that the compression/decompression rules are "tight," in the sense that if an expansion rule were applied during a computation, the computation would not terminate, because of the presence of extraneous symbols.

In the case of $\mathbf{R}_{\rightarrow\wedge}$, the upper bound and lower bounds coincide. This is accomplished by simulating a zero test in a more direct fashion, by using the notion of AND-branching, derived from the proof by Lincoln, Mitchell, Scedrov and Shankar that propositional linear logic is undecidable.

In adapting this idea to $\mathbf{R}_{\rightarrow\wedge}$, we face the same problem as before, namely the presence of unrestricted contraction. However, it can be overcome by essentially the same trick as previously.

As before, we can prove that this modified machine halts if and only if the original machine halts, because if an expansion instruction were applied at any point during the computation, the machine could not terminate (because extraneous tokens would be present).
It follows that there is no primitive recursive decision procedure for $\mathbf{R}_{\rightarrow\wedge}$, and so the upper and lower bounds for the complexity of this logic match. $\mathbf{R}_{\rightarrow\wedge}$ is perhaps the most complex decidable "natural" propositional logic.

Problem 1

It would be good if we could nail down the exact complexity of $\mathbf{R}_{\rightarrow}$, but the argument we have just given seems to depend on the presence of conjunction. The key property of conjunction is that it is an "additive" connective, in the sense of linear logic, that is to say, the side formulas in the premisses and conclusion are exact copies of each other. This copying ability is what makes the zero-test work.

Problem 3

In the case of Problem 3, we *do* have additive connectives, but the method employed to simulate zero-test instructions using AND-branching does not seem to work, because of the unrestricted weakening rule (which means in the context of our machine model that we can arbitrarily decrement a register at any time).

I published a claim of a doubly exponential space upper bound in 2000, but sad to say, my proof had a gaping hole in it, and so the complexity of this logic is still very much an open problem.

Problem 2

No advance has been made on the results of Mayr and Meyer from the early 1980s, so the huge gap between the upper and lower bounds for the complexity of the reachability problem for Petri nets remains.

Zakariae Bouziane published a primitive recursive algorithm for this problem in 1998. However, a serious flaw was exposed in his proof by Petr Jančar, and the problem remains unsolved after more than 30 years.