

# Automatic Solving of Problems stated in Natural Language

Tore Amble

Knowledge Systems Group

Faculty of Electrical Engineering and Computer Science

The Norwegian Institute of Technology

The University of Trondheim

amble@idt.unit.no

## Abstract

The general idea of this paper is to give a report on some recent work on pragmatics and understanding in computers; “The Understanding Computer” - TUC. This is a query system with a natural language (NL) interface being developed in Prolog.

One of the aims of TUC is to use a general natural language processor for a variety of purposes including intelligent reasoning about the natural language statements that are processed.

The three wise men’s problems is a classical challenge in AI, putting the logical models to test. Although there are a number of published solutions formulated in various logics, this paper will show how TUC will solve the problem after a translation from a natural language formulation of the problem.

## The Understanding Computer

TUC is an acronym for The Understanding Computer and is an effort to implement a conversational agent [Amb94]. The language part is a subset of English, and a cooperative interface will hopefully make it a robust and useful vehicle. The focus in this paper is how to make TUC really understand the Natural Language when seen as a Naturally Readable Logic (NRL) for knowledge base query and update. One version is taken from [GN87], which we take as our starting point. [Portions bracketed out are just talk about the problem, and not the problem formulation itself.]

*Suppose there are three wise men who are told by their king that at least one of them has a white spot on his forehead; actually, all three have white spots on their forehead. We assume that each wise man can see the others’ forehead but not his own, and thus each knows whether the others have white spots. [There are various versions of this puzzle, but suppose we are told that] the first wise man says, “ I do not know whether I have a white spot, and that the second wise man then says, “I also do not know whether I have a*

*white spot.” [We can formulate this puzzle in our logic of belief to] show that the third man then knows that he has a white spot.*

A paraphrase in NRL of the three wise mens problem is shown below, just to show the level of language, and the ambitions to achieve some reasoning capabilities. Actually, a simpler version will be used for demonstration later on. The syntactical translation from natural language to the logical form, however, is outside the scope of this paper.

The problem is set as a kind of benchmark for the language understanding capabilities of the system. It should have at least this level of introspective understanding. On the other hand, much more of the kind may not be necessary for practical purposes.

There are three wise men who are told by their king that at least one of them has a white spot on his forehead. All three have white spots on their forehead. Each wise man can see the others forehead but not his own forehead and each knows whether the others have white spots . The first wise man says that he did not know whether he had a white spot. Also the second wise man says that he did not know whether he had a white spot. How does the third man know that he has a white spot ?

The underlying mechanism will be a theorem prover for first order predicate logic. The goal is to represent the meaning of the natural language sentences in logic so that the answers are produced by a theorem prover.

## **The Logical Formulation**

The 3 wise mens problem (or its generalisation N wise mens problem) is an exercise in logic, especially the logic of knowledge and belief [Hin62]. This logic tries to model idealised rational agents who are capable of drawing all valid conclusions from their base of knowledge. Contrary to cognitive science, we model what an agent should know, more than what he is likely to know.

There is a choice between using *beliefs* or *knowledge* (i.e. true beliefs) as the basis. However, if there are no beliefs that are explicitly considered to be false, the reasoning is quite independent of whether the propositions are “actually” true. We shall formulate a logic called CSL (Common Sense Logic) which is capable of supporting such reasoning.

Without going into axiomatic details, we outline a formulation and proof of the 2 wise mens problem. Let  $A_1, \dots, A_n$  be the agents (wise men). Let  $W_1, \dots, W_n$  be the propositions that agent  $A_n$  has a white spot and let  $K(A, P)$  be the proposition A knows the proposition P.

(1):  $K(A1, \text{not } W2 \Rightarrow W1)$ .           % A1 knows this rule of the game  
(2):  $\text{not } K(A1, W1)$                                % A1 did not say that he knew W1  
(3):  $\text{not } W2 \Rightarrow K(A1, \text{not } W2)$            % A1 would (see and) know if not W2  
(4):  $K(X, P \Rightarrow Q)$  and  $K(X, P) \Rightarrow K(X, Q)$  % Axiom

The first proof is presented using the implicit assumption that the the proof is based on commonly known facts and rules, and that therefore agent A2 can do the reasoning of the proof himself.

(5):  $K(A1, \text{not } W2) \Rightarrow K(A1, W1)$        % from (1), (4)  
(6):  $\text{not } K(A1, \text{not } W2)$                        % (5), (2), Modus Tollens  
(7):  $W2$    % (3), (6), Modus Tollens  
(8):  $K(A2, W2)$                                    % A2 knows (1)-(7)

QED

The last step depended on the previous proof history. To avoid this, we base the theorem proving on Resolution [Rob65] where each proof step a is single deductive step depending on a fixed number of earlier results.

A set of the axioms that can be used to model this kind of reasoning is given. They are called Axioms of Knowledge [Dav90]. The axioms are valid in all normal worlds with rational agents. The axioms are given labels for further reference.

### Axioms of Knowledge

PC:       Every axiom of Propositional Calculus  
          (PC) is an axiom of knowledge.

KN:       If P is an axiom, then  $K(A, P)$  for all agents A

KMP:      $K(A, P \Rightarrow Q)$  and  $K(A, P) \Rightarrow K(A, Q)$

KT:        $K(A, P) \Rightarrow P$

KKP:      $K(A, P) \Rightarrow K(A, K(A, P))$

KKN:      $\text{not } K(A, P) \Rightarrow K(A, \text{not } K(A, P))$

KAND:     $K(A, P)$  and  $K(A, Q) \Rightarrow K(A, P \text{ and } Q)$

## Common Knowledge

We must also include an explicit formalisation of the notions of common knowledge.

Epistemic reasoning involves nested levels of knowledge. For example, the rule of modus ponens applies inside nested levels of knowledge:

$$\frac{K(A2, K(A1, P \Rightarrow Q)) \quad K(A2, K(A1, P))}{K(A2, K(A1, Q))}$$

This follows from the axioms of knowledge. The chain of agents (A2,A1) can easily be generalised, and such chains of agents behave in most respects like any other agent. The concept of Common Knowledge transcends these chains.

Let  $CK(P)$  denote “P is common knowledge”. Then, not only does every agent know P, but it is also common knowledge that every agent knows P.

An additional axiomatisation follows.

### Axioms of Common Knowledge

TAUT: If P is an axiom, then  $CK(P)$

CMP:  $CK(P \Rightarrow Q)$  and  $CK(P) \Rightarrow CK(Q)$

CAND:  $CK(P)$  and  $CK(Q) \Rightarrow CK(P \text{ and } Q)$

CKP:  $CK(P) \Rightarrow CK(K(A, P))$

KALL:  $CK(P) \Rightarrow K(A, P)$

### Knowing Whether

An important issue is the weaker and derived concept of **knowing whether** a proposition is true. For example, A1 can see A2’s front, so A1 knows whether A2 is white (W2). This means that

$$W2 \Leftrightarrow K(A1, W2)$$

Accordingly, we define the operator KW (*know whether*) by:

KWP:  $KW(A, P) \Rightarrow (P \Rightarrow K(A, P))$

KWN:  $KW(A, P) \Rightarrow (\text{not } P \Rightarrow K(A, \text{not } P))$

## Knowledge and Time

Also time is important in epistemic reasoning as the knowledge of an agent varies over time. Time may be modelled as a series of discrete events, where the events cause new information to be true, and by deduction, agents get new knowledge at these events.

We will assume that the rational agents are not forgetful, and that for an agent, the knowledge will not decrease over time. The time points of the events change the current states of affairs, but the historic knowledge will increase.

If we introduce time in the knowledge operator, we get

$$t1 < t2 \Rightarrow K(A, t1, P) \Rightarrow K(A, t2, P)$$

This is common knowledge, and applies in nested levels. We will not go further into details at this point, but leave it to a further treatise.

## Derived composite axioms

Minimality of axiom sets is a virtue of a theory, but not necessarily a blessing in practice. It is sometimes convenient to derive axioms that are the logical consequence of two other axioms. In the implemented rulebase of CSL, this is done to a certain degree. The axioms used are

CMT:  $CK(P \Rightarrow Q) \text{ and } CK(\text{not } Q) \Rightarrow CK(\text{not } P)$ .

DUAL:  $CK(K(A, P \Rightarrow Q)) \text{ and } CK(\text{not } K(A, Q)) \Rightarrow CK(\text{not } K(A, P))$ .

CIF:  $CK(K(A, P \Rightarrow Q)) \Rightarrow CK(K(A, P) \Rightarrow K(A, Q))$ .

CIF2:  $CK(K(A, P \text{ and } Q \Rightarrow R)) \Rightarrow CK(K(A, P) \text{ and } K(A, Q) \Rightarrow K(A, R))$

NKWP:  $CK(\text{not } KW(A, P)) \Rightarrow CK(\text{not } K(A, P))$ .

NKWN:  $CK(\text{not } KW(A, P)) \Rightarrow CK(\text{not } K(A, \text{not } P))$ .

TAUT1:  $CK(\text{not } Q \Rightarrow P) \Rightarrow CK(\text{not } P \Rightarrow Q)$

TAUT2:  $CK(Q \text{ and } R \Rightarrow P) \Rightarrow CK(\text{not } P \text{ and } Q \Rightarrow \text{not } R)$ .

TAUT3:  $CK(\text{not } Q \text{ and } R \Rightarrow P) \Rightarrow CK(\text{not } P \text{ and } R \Rightarrow Q)$ .

## The implementation

Common Sense logic is realised by a theorem prover INGER (Intelligent Resolution) implemented in Prolog.

The axiom names from CSL have been retained. Because of the top down character of the proof system, the reversed notation “*A* if *B*” is used to denote implication “ $B \Rightarrow A$ ”

%%% Axioms of Common Sense Logic

% k(A,P) means: agent A knows P

% ck(P) means: P is COMMON KNOWLEDGE

%%% Basic Axioms

kall:: k(\_,P) if ck(P).

ckp:: ck(k(\_,R)) if ck(R).

cand:: ck(P and Q) if ck(P) and ck(Q).

kwp:: ck(k(A,P) if P) if  
ck(kw(A,P)).

kwn:: ck( k(A,not P) if not P) if  
ck(kw(A,P)).

cmp:: ck(P) if  
ck(P if Q) and ck(Q).

%%% Derived Axioms

cmt:: ck(not Q) if  
ck(P if Q) and ck(not P).

dual:: ck(not k(A,P)) if  
ck(k(A,Q if P)) and  
ck(not k(A,Q)).

cif:: ck(k(A,P) if k(A,Q)) if  
ck(k(A,P if Q)).

cif2:: ck(k(A,P) if k(A,Q) and k(A,R)) if  
ck(k(A,P if Q and R)).

nkwp:: ck(not k(A,P)) if  
ck(not kw(A,P)).

nkwn:: ck(not k(A,not P)) if  
ck(not kw(A,P)).

taut1:: ck( Q if not P) if ck(P if not Q).

taut2:: ck( not R if not P and Q) if ck(P if Q and R).

taut3:: ck( Q if not P and R) if ck(P if not Q and R).

## The Problem Formulation

The purpose of this section is to demonstrate how TUC translates NRL into Common Sense Logic. TUC's semantics is based on a Davidsonian semantics [Dav67], whereby each action is augmented by an event parameter. Let us take the example

*John knew on Friday that Fred kissed Mary on Thursday*

The way to understand such a composite sentence is to decompose it into labelled clauses, and use the labels for reference. The labels are then interpreted as named events, and the clauses further decomposed into into a conjunctive normal form.

```
event(w0,s1).          % s1 is an event in the world
ontime(friday,s1).    % s1 happened on Friday
knowthat(john,s2,s1). % In s1, John knows s2

event(s2,s3).          % s3 is an event in s2
ontime(thursday,s3).  % s3 happened on Thursday
kiss(fred,mary,s3).   % In s3, Fred kissed Mary
```

From this representation, TUC is able to extract information into the form of CSL statements. In order to show the proof mechanism, we take a very simplified version of the two wise mens problem formulated in NRL. The phrase "it is known that" is given the precise meaning of common knowledge. That an agent "has a white spot" is simplified as "being white".

A1 is a man. A2 is a man.

```
It is known that A1 knows whether A2 is white.
It is known that A2 knows whether A1 is white.
It is known that if A1 is not white then A2 is white.
It is known that A1 does not know whether A1 is white.
```

How does A2 know that A2 is white ?

This is translated by TUC into the following.

```
ck(kw(a1,w2)). % w2 = white/a2
ck(kw(a2,w1)). % w1 = white/a1

ck( w2 if not w1).
ck( not k(a1,w1)).

problem:: k(a2,w2).
```

## The theorem prover

The proof is done by a theorem prover that is a modification of a Prolog metainterpreter [Amb87]. The axioms of CSL are non Horn clauses at the object level. Negation is treated as logical negation, and no negation by failure is evoked unless explicitly programmed. The proof method is top down, depth first, depth bounded. Together with the proving, a complete proof tree is grown, and printed out when a proof has been found. However, the proof search process and all the failures are not shown.

The proof process is bounded by a depth bound "Max Depth". The depth bound makes the theorem prover inherently incomplete, but by iterative deepening, the theorem prover will be "complete in the limit". Note that in the axiomatisation of CSL, only axioms of PC were supposed to be common knowledge. A theorem in PC is only known to the extent that its proof is not too complicated. In a qualitative way, this solves the paradox of omniscience of rational agents.

## The proof

```
*** INGER THEOREM PROVER ***
```

```
Max Depth = 7
```

```
* PROBLEM: k(a2,w2)
```

```
* PROOF *
```

```
k(a2,w2) BECAUSE [ ka11 ]  
    ck(w2)
```

```
ck(w2) BECAUSE [ cmp ]  
    ck(w2 if not k(a1,not w2)) AND  
    ck(not k(a1,not w2))
```

```
ck(w2 if not k(a1,not w2)) BECAUSE [ taut1 ]  
    ck(k(a1,not w2)if not w2)
```

```
ck(k(a1,not w2)if not w2) BECAUSE [ kwn ]  
    ck(kw(a1,w2))
```

```
ck(kw(a1,w2)) is TRUE
```

```
ck(not k(a1,not w2)) BECAUSE [ cmt ]  
    ck(k(a1,w1)if k(a1,not w2)) AND  
    ck(not k(a1,w1))
```

```
ck(k(a1,w1)if k(a1,not w2)) BECAUSE [ cif ]  
    ck(k(a1,w1 if not w2))
```

```
ck(k(a1,w1 if not w2)) BECAUSE [ ckp ]
```



```
ck(w1 if not w2)

ck(w1 if not w2) BECAUSE [ taut1 ]
ck(w2 if not w1)

ck(w2 if not w1) is TRUE

ck(not k(a1,w1)) is TRUE

* QED *
```

Time 204 ms

## Three wise mens problem

The Three wise mens problem is formulated and solved in a similar way.

A1 isa man. A2 isa man. A3 isa man.

It is known that  
A1 knows whether A2 is white. %% ...etc

It is known that  
if A1 is not white and A2 is not white then A3 is white.

It is known that  
A1 does not know whether A1 is white.

It is known that  
A2 does not know whether A2 is white.

It is proved that  
if A3 is not white then A2 knows that A2 is white .

How does A3 know that A3 is white ?

However, when the problem was scaled up, problems of combinatorial complexity emerged, and had to be dealt with by heuristic methods.

## Sequencing

An obvious way of optimisation is to put frequently successful axioms at the front, while axioms occurring infrequently in successful proofs are but behind. Also, ordering the sequencing of conditions is a similar optimisation method.

## Composite Axioms

Another remedy is to make composite axioms. By resolving two axioms that are frequently applied after each other, we obtain a more specialised axiom that could solve two basic steps by one resolution. Thereby, the proofs become shorter, at the expense of searching through more axioms.

It is an exciting thought that the derivation of composite axioms actually can be done automatic as an adaptive self improving process.

## Lemmatisation

The use of lemmas is a general method of dividing a problem into subproblems, solving the subproblems individually and use them in the proof. In NRL, the lemma is declared as

```
it is proved that Lemma
```

```
where Lemma is in the form
```

```
if condition then conclusion
```

which TUC interprets as a command to prove the Lemma and then add it as a new premise. The lemmas of the form of implications are proved in the manner of Natural Deduction, by proving the *conclusion* under the assumption that *condition* is true.

With the help of the lemma, the three wise mens problem was proved in a matter of seconds. Without it, the theorem prover went astray. We can argue whether TUC should be smart enough to find the lemmas by itself. There is no conflict however, the ability to accept advice is a mark of wisdom.

## The missing Background Knowledge

The NRL version is a simplification of the original formulation. In order to handle the full NRL version, a number of problems of semantical and pragmatcal nature has to be solved:

- the men are identified as three individuals
- the wise men have one forehead each, but one king together
- the white spot is a situation independent property
- what the king says is common knowledge
- what the men says is common knowledge

This knowledge has to be stated in a declarative way, and not be programmed into the intended interpretation by any black magic.

As such, the problem is still a fruitful challenge for natural language understanding.

## References

- [Amb87] T. Amble. *Logic Programming and Knowledge Engineering*. Addison-Wesley, 1987.
- [Amb94] T. Amble. Domain Modelling and Automated Natural Language Interfaces. In S.L Hansen, editor, *Topics in Knowledge Based NLP Systems*. Workshop on KBS,Copenhagen Business School, Samfundslitteratur, 1994.
- [Dav67] D Davidson. The Logical Form of Action Sentences. In N Rescher, editor, *The Logic of Decision and Action*. Pittsburgh, 1967.
- [Dav90] E. Davies. *Representations of Commonsense Knowledge*. Morgan Kaufman, 1990.
- [GN87] M Genesereth and N.J Nilsson. *Logical Foundations of Artificial Intelligence*. Morgan Kaufman, 1987.
- [Hin62] J. Hintikka. *Knowledge and Belief*. Itacha, N.Y.:Cornell University Press, 1962.
- [Rob65] J.A. Robinson. A machine oreinted logic based on the resolution principle. *Journal of The ACM*, 12(1), 1965.