CS 475 — Artificial Intelligence, Fall 1997

Assignment 2

Due Thursday, 16 October

1 Part I

1. (Unification)

   (a) Trace through the unification algorithm on your own for the following pairs of variables. **There is nothing to hand in for this question. Just use it to confirm that you understand the algorithm.**

      i. $f(X)$ and $f(Y)$
      ii. $f(spot)$ and $f(fluffy)$
      iii. $w(fatherOf(A))$ and $w(B)$
      iv. $z(spot, c(X, Y))$ and $z(X, c(fluffy, spot))$

   (b) Explain why the following sets of literals do not unify (uppercase letters are used for variables, lower case for constants):

      i. $\{r(f(Y, f(Y, a)), a), r(f(X, X), a)\}$
      ii. $\{r(f(a), X), r(X, a)\}$
      iii. $\{\neg r(a), r(X)\}$

2. (Resolution)

   (a) Convert the following predicate-calculus sentence into clause form:

   $$\forall X(p(X) \rightarrow \{\forall Y[p(Y) \rightarrow p(f(X, Y))] \wedge \neg\forall Y[q(X, Y) \rightarrow p(Y)]\})$$

   (b) Consider the following sentences:

   - Maria knows about all kinds of safe food.
   - If someone ate something and did not die, then that thing is a safe food.
   - Apples are a safe food.
   - Oranges are a safe food.
   - Daniel eats bananas and is still alive.
   - Maria eats everything Daniel eats.

      i. Translate these English sentences into predicate-calculus sentences.
      ii. Convert the results of the first part into clausal form.
      iii. Prove that Maria knows about bananas using resolution.

iv. Use resolution to answer the question, "What safe food does Maria eat?"

Notes: (1) Just translate the following into single symbols: "safe food", "still alive", "did not die".

(2) You may need to add an assertion to your knowledge base. If you do, please point it out.

2 Part II

2.1 Background

Within the domain of electronic circuits, we would like to build an AI system to analyze circuits such as the one shown on page 223 of the text.

The circuit purports to be a one-bit full adder. The first two inputs (I1 and I2) are the two bits to be added, and the third input (I3) is a carry bit. The first output (Out1) is the sum, and the second output (Out2) is a carry bit for the next adder.

A1 and A2 are AND-gates, X1 and X2 are XOR gates, and O1 is an OR-gate. Each gate, obviously, has both inputs and outputs. The signal at an input or output port is either 0 or 1.

The goals of our AI system are to determine whether or not a given circuit is an adder, and to answer questions about the value of current flow at various points in the circuit. **This is not a fault diagnosis system; all circuits and gates are assumed to be functioning properly.**

The system reasons by performing resolution over first-order predicate calculus sentences that have been converted to clausal form.

We now turn to the representation scheme used. The connectivity between gates is represented using the predicate *conn*: $conn(X, Y)$ means that ports X and Y are connected. For example, in the diagram, the output port of gate X1 is connected to the top input port of gate X2, and the top input port (I1) of the circuit is connected to the top input port of (X1).

The value of the signal at a port is represented using the function symbol *value*: $value(X)$ designates the value of the signal at port X (either 0 or 1).

Type information is represented with obvious predicates. For example, $port(X)$ means that X is a port and $andGate(Y)$ means that Y is an AND-gate.

Suppose that the knowledge base of the system contains, among others, the following axioms and facts:

i $\forall A1 \forall A2\ conn(A1, A2) \rightarrow value(A1) = value(A2)$.
(Recall that we are assuming that the circuits tested by the system are not faulty.)

ii $\forall X \forall Y \forall Z\ X = Y \wedge Y = Z \rightarrow X = Z$.

iii $conn(p1, p2)$.

iv $value(p1) = 1$.

v $port(p1)$.

vi $port(p2)$.

2.2 The Questions

1. Show that:

$$\{i, ii, iii, iv\} \vdash value(p2) = 1.$$

2. Items (a)-(d) below are four attempts to rewrite (i) using *value* as a predicate symbol rather than a function symbol. (Assume that (iv) has been rewritten as (iv)': $value(p1, 1)$.) Imagine that they are answers students gave on a final exam. For each answer (a)-(d):

Case 1 If the answer is correct, and if no additional axioms are needed to infer $value(p2, 1)$, then show the latter. That is, show that

$$\{\text{the answer}, ii, iii, iv', v, vi\} \vdash value(p2, 1).$$

Case 2 If the answer is essentially correct, except that some supplementary axioms are needed to preserve some inferences supported by the original (i)-(vi), then do the following. State additional axioms that would be needed for the system to be able to infer $value(p2, 1)$. That is, state some axioms such that

$$\{\text{the answer, the new axioms}, ii, iii, iv', v, vi\} \vdash value(p2, 1).$$

The axioms should be general, useful ones.

Case 3 : If the answer does not mean the same thing as (i) at all, explain to the student what is wrong with his or her answer. For example, the answer might not support the inferences that are supported by the original version of (i), and there are no natural supplementary axioms to add to the knowledge base to save the answer. Or, the answer might allow the system to infer things we clearly don't want it to infer, in this problem domain. In other words, case 3 is for answers that we clearly would not want to add to the knowledge base, as is.

(a) $\forall T1\ \forall T2\ \exists Z\ conn(T1, T2) \rightarrow value(T1, Z) \wedge value(T2, Z)$.

(b) $\forall P1\ \forall P2\ \forall V1\ \ conn(P1, P2) \wedge value(P1, V1) \rightarrow value(P2, V1)$.

(c) $\forall T1\ \forall T2\ \forall X\ conn(T1, T2) \rightarrow value(T1, X) \wedge value(T2, X)$.

(d) $\forall X\ \forall Y\ \ conn(X, Y) \rightarrow value(X) \wedge value(Y)$.

3. Considering the problem being solved and the domain, discuss the relative advantages and disadvantages of using *value* as a predicate or a function symbol.