

CSC 244/444 Assignment 3: Theorem Proving and Planning

Due: October 26th *New date for midterm Oct 28th*

This assignment will be discussed as part of the review before the midterm exam. So it must be handed in *before* class on the 26th, and no late hand-ins will be allowed. Working this assignment is one of the best ways to study for the midterm.

1. State whether the following formulas unify, and if so, give a most general unifier. In the following x, y, z , and w are variables and other terms are constants.

- a. $P(x, f(x), g(f(x)))$ and $P(z, z, y)$
- b. $P(f(x), y, g(y))$ and $P(z, f(a), w)$
- c. $P(f(x, y), f(y, x))$ and $P(z, f(z, w))$

2. Determine whether the following formula is valid using resolution. Show all steps carefully and clearly.

$$\exists x \forall y \forall z ((P(y) \supset Q(z)) \supset (\neg P(x) \vee Q(x)))$$

3. The resolution rule as presented in the text is not complete as it stands. As an example, the KB consisting of the clauses $\{[P(x), P(y)]\}$ and $\{[\neg P(u), \neg P(w)]\}$ is logically unsatisfiable but the resolution rule in the text cannot derive the empty clause.

- a) Give an argument that shows the resolution rule in the text is incomplete, using this KB as your example.
- b) State a more general resolution rule that does handle this case (note - you can produce this yourself, or consider the handouts on resolution and discuss what the generalization is.
- c) Show that using your generalized rule that you can prove that the KB is unsatisfiable.

4. Consider the following attempt to avoid the expressive limitations of Horn clause system. We will introduce new predicates to stand for the negations of existing predicates. So, for an y predicate P , we introduce a predicate $\text{Not}P$, which we intend to be equivalent to $\neg P$. Now a non-Horn clause such as $[P, Q]$ can be expressed as the Horn clause $[\neg \text{not}P, Q]$.

a) Show that the following non-Horn clause KB is unsatisfiable (say by presenting a resolution proof)

$$\{[P, Q], [\neg P, Q], [\neg Q, R], [\neg R]\}$$

b) Show that the following equivalent Horn-clause KB is is satisfiable:

$$\{[\neg \text{Not}P, Q], [\neg P, Q], [\neg Q, R], [\neg R]\}$$

c) Could we extend the Horn-clause database in part (b) with additional Horn clauses that capture the relationship between P and $\text{Not}P$ so that we get the desired behavior. Give an argument why this cannot be done.

d) Can we get partially there - i.e., sanction some inferences about disjunctive positive predicates, but not be complete. Give examples of clauses you might be able to add to extend the KB and give an example of something that can now be proven that otherwise could not without the technique.

5. Consider the following information about a one-sexed species, the *unis*.

Every uni has exactly one mother.

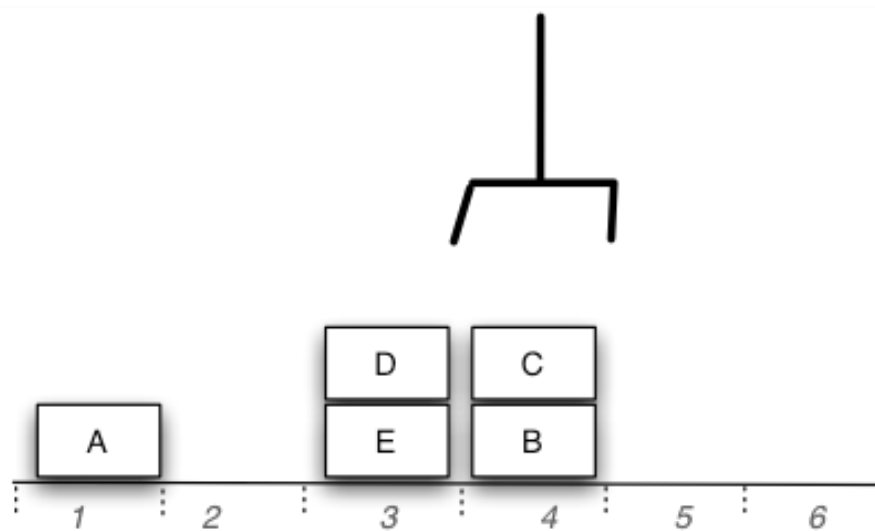
Two unis are sisters if they have the same mother.

Sarah's mother is Sissy.

Sandra's mother is Sinda.

- Express the knowledge above, taking each assertion literally and without adding any assumptions, as a set of axioms using first-order logic with equality.
- Can you prove, given only the knowledge above, that Sarah and Sandra are not sisters? If so, give the resolution proof (using paramodulation if necessary). If not, show that they could be sisters.
- If you showed they could be sisters in part (b), give the minimal number of new facts that you need to assume in order to prove that they are not sisters. Your facts should agree with your intuitions about this problem if you had just been told it as a story. Now show a proof using your extended KB that Sarah and Sandra are not sisters.

6. Assume a 2-dimensional discrete blocks world as shown here.



The objects in the world consist of identical blocks named A, B and C, etc, a robot arm and a table with n discrete positions. There are three actions possible in this world. (1) The robot can move to any position, (2) it can close its hand (If the robot closes its hand when it is at the same position as a block, it grasps the topmost block at that position); (3) it can open its hand (If it opens its hand, it drops any block it was holding, which becomes the topmost block at that position).

We will formalize this domain using the situation calculus.

- Define intuitively the predicates that will be the set of **fluents** you use to represent this world. Describe what you intend each to mean and what its arguments are.
- Give a set of formulas that capture all the essential relationships between the predicates in (a). For each, give a gloss as to why this knowledge would be useful for planning in this domain. This is often called the **domain knowledge**.

- c) Give a set of formulas that capture all the necessary **preconditions** and **effects** of the three actions. Give a gloss for each stating why this knowledge is useful for planning. Make sure you express complete knowledge about the actions. For instance, what happens if the robot closes its hand when it is already holding a block?
- d) Give a set of formulas that capture all the necessary **frame axioms** for this domain (including a gloss for each).
- e) Give a set of assertions for that describe the situation S_0 , which is shown in the figure above.
- f) Show using illustrative examples, explanations, and sketches of proofs how the knowledge defined supports reasoning about the following situations:
- i) How do you know what the topmost block is at a location? How do the open and close actions change (or not change) this information. Demonstrate that your knowledge agrees with our intuitions about the blocks world. To answer this, you'll need to show how your action definitions, domain axioms and frame axioms work together to change what should be changed, and leave the unaffected parts of the world alone.
 - ii) State how you would pose the goal that *block E is on block A*. Identify a situation (constructed using the *do* function and your actions starting from situation S_0 defined in part (e)) in which this goal is achieved - and give a sketch of the proof that the goal holds in that situation.
- g) (444 students) The situation calculus is a richer representation than used in most of the planning systems we'll examine in the rest of the course. By this, we mean that we can express goals that often can't be posed in other system, and we can prove general facts about the domain that are not considered "planning problems". Here we explore some examples. *Before answering each question make sure you precisely state the proposition you are trying to prove.* If your current knowledge is not sufficient to prove these facts, then discuss what additional knowledge you need, add it, and then do the proof.
- i) Consider the goal that *no block is on any other block*. Identify a situation (expressed in terms of the *do* function starting from situation S_0) in which this goal is true, and give a sketch of the proof that the goal holds in this situation.
 - ii) Give a sketch of a proof that, that starting from S_0 , that we cannot reach a state in which block A is on block D, by taking only one action.
 - iii) Give a sketch of a proof: *the robot can never hold two blocks simultaneously*. If you have to add facts, describe them. Note that adding the fact that says explicitly that the robot cannot hold two objects simultaneously is not a good solution!