Assignment 2 – COMP-424
Submission on MyCourses Feb 14, 11:59PM

TA: Pierre-Luc Bacon

Submission Instructions

Submit all your answers in a **single PDF** file. You need to answer all questions to receive full points. However, only one or two questions will be graded due to the size of the class.

Question 1: Propositional Logic

Assume you want to play a game of Clue using propositional logic. If you don't know the rules, see `http://en.wikipedia.org/wiki/Cluedo`. Assume you are playing a game with only three players (Alice, Bob and Charlie). Also assume that there are only three possible suspects (Miss Scarlet, Prof Plum, and Col Mustard), three weapons (knife, rope, candle), and three rooms (hall, library, ballroom). Each player holds only two cards.

Now suppose you are playing the game from Charlie's perspective, and you see the following:

- Charlie queries Prof **Plum** with the **candle** in the **library**.
- Bob replies: "I have none."
- Alice shows Charlie the **candle**.
- Bob queries Col. **Mustard** with the **knife** in the **ballroom**.
- Alice replies: "I have none."
- Charlie shows Bob Col. **Mustard**.
- Alice queries Miss **Scarlet** with the **rope** in the **hall.**
- Charlie shows Alice the **Hall.**
- Charlie queries Miss **Scarlet** with the **rope** in the **library.**
- Bob shows Charlie Miss **Scarlet.**
- Bob queries Prof **Plum** with the **knife** in the **library.**
- Alice shows Bob a card (Charlie does not see which card.)
- Alice queries Prof **Plum** with the **rope** in the **ballroom.**
- Charlie replies: "I have none."
- Bob shows Alice a card (Charlie does not see which card.)

1. Design a propositional logic representation of this version of Clue. Include any necessary sentence to express useful known facts about the game in general, as well as each of the plays listed above. Clearly explain your choice of propositional symbols and what facts each sentence represents.
2. Include your knowledge base in the **single PDF** file submitted to myCourses.
3. Using the propositional logic prover at `www.lwb.unibe.ch/index.html` try to prove, for each of Scarlet, Plum and Mustard, who is the murderer. Repeat to determine the murder weapon, and the room. Include the prover output for each case in your submission (in the same PDF file).

Instructions for using the online solver

Go to solver website: `www.lwb.unibe.ch/index.html` On the first line, load the solver:

```
load(cpc);
```

Next, input your sentences (one per line), using the syntax explained below. Make sure to define each sentence by name, and terminate it with a semicolon. You can use brackets as appropriate.

```
A1 := a v b;
A2 := ~b v c;
A3 := c & d;
A4 := d -> ( a v b);
```

Define your knowledge-base to include all sentences.

```
T := [A1, A2, A3, A4];
```

Define the conclusion sentence you want to prove.

```
B := d;
```

Call the proof procedure to verify whether your conclusion is true or false:

```
provable(B,T);
```

To represent	Use syntax
Negation	`~`
Conjunction	`&`
Disjunction	`v`
Implication	`->`
Equivalence	`<->`

Table 1: Documentation available at `http://www.lwb.unibe.ch/documentation`

Question 2: Translation

Represent the following sentences in first-order logic, using a consistent vocabulary (which you must define).

1. Not all students like both AI and Logic.
2. Every student who dislikes AI also dislikes Logic.
3. All assignments are disliked by some of the students.
4. Some assignments are disliked by all of the students.
5. There is a student who likes AI but dislikes all assignments.

Question 3: FOL Resolution

From the sentence "Zebras are animals", it follows that "The head of a zebra is the head of an animal." Demonstrate that this inference is valid by carrying out the following steps:

1. Translate the premise and the conclusion into first-order logic. Use three predicates HeadOf(h,x), Zebra(x), and Animal(x).
2. Negate the conclusion, and convert the premise and the negated conclusion into conjunctive normal form.
3. Use resolution to show that the conclusion follows from the premise.

Question 4: STRIPS Planning

1. Imagine that you want to improve the waiting time for the elevator in the McConnell building using some of the planning techniques that you learned in this course. Your first step is to specify a model amenable to planning. For this question, you will need to translate the following domain description in the STRIPS language.

Passenger states

A passenger $p \in \mathcal{P}$ can be in the following dynamic states:

(a) The passenger has boarded: `Boarded(p)`
(b) The passenger has been served (ie she arrived at the destination floor): `Served(p)`

Once she has boarded, a passenger can then be:

(a) `GoingUp(p)`
(b) `GoingDown(p)`

Floors

Assume that McConnell has a secret floor only accessible to some special people: a secret lab or an amusement park, you decide. We can establish whether a passenger p has access to a floor f by `Access(p, f)`. The current location of the lift is specified through `LiftAt(f)`. For a pair of floors f_1, f_2, one can determine which one is above using the `Above(f1, f2)` predicate (evaluating to $\top$ if f_1 is above f_2). For the system considered here, there are only seven floors $f_1, f_2, \ldots, f_7$ for which $f_i, f_j : i > j \implies \mathit{Above}(f_i, f_j)$. The seventh floor is the special/restricted/magic floor in which passengers p_2 and p_4 are denied access. Their access status is immutable.

Actions

The elevator system has three actions:

(a) `GoUp`(f_1, f_2) The lift can move up to a floor f_2 only if f_1 is lower.
(b) `Stop(f)`
When the elevator stops on floor f, every passenger who has reached reached their destination floor leaves and all of those waiting outside enter.
(c) `GoDown`(f_1, f_2) The lift can move down from f_1 only if f_1 is above f_2.

Requirements

A passenger cannot change her destination during the trip. The system needs to serve every passenger. The set of passengers is restricted to $p_1, p_2, \ldots, p_6$ and the system is instantiated with:

(a) p_1 is on the first floor and wants to go to the second floor.

(b) p_2 is on the second floor and wants to go to the fourth floor.

(c) p_3 is on the third floor and wants to go to the first floor.

(d) p_4 is on the fifth floor and wants to go to the first floor.

(e) p_5 is on the fourth floor and wants to go to the third floor.

(f) p_6 is on the seventh floor and wants to go to the sixth floor.

The origin and destination of a passenger can be established using the predicates `Origin(p, f)` and `Destination(p, f)`, where $p \in \{p_1, p_2, \ldots, p_6\}$ and $f \in \{f_1, f_2, \ldots, f_7\}$.

Constraints

Whenever the elevator stops on a restricted floor, the system must ensure that none of its unauthorized passenger will see anything around. In some cases, the planner will have to delay the boarding of a restricted passenger in order to safely serve all of the *special* passengers. You don't need to think about the details of what the plan would look like in this problem. However, you do have to take this constraint into account when specifying the `Stop` action.

We finally add the constraint that if a passenger tells the system that she wants to go up or down, then the planner must guarantee that this person will never travel in the opposite direction. For example, if someone wants to leave from the first floor to reach the fourth floor, we want to guarantee that no time is wasted by taking a detour down to the basement.

Your answer

Using STRIPS, you need to specify the initial and goal states as well as the three operators described above. You must use the same predicate names as above and should not introduce new ones. Give the pre and post-conditions for each operator.

2. In the previous modelling exercise, we have neglected the capacity constraint which would arise in real life. For example, elevator #40 in the McConnell building has a capacity of 1814Kg and can accommodate 25 people. How would you go about modelling a capacity constraint in STRIPS ? Don't forget that state transitions result from the application of an operator and change a set of conditions. Explain why this is a problem and what could be used as an effective replacement strategy (without redefining the language).