CS 188
Spring 2024

Introduction to
Artificial Intelligence

HW 3B Sol.

Solutions for HW 3 (Written)

Q1. [39 pts] Propositional Logic

(a) Consider a propositional language with 4 symbols: A, B, C, D. For each of the following sentences, mark how many models satisfy the sentence out of the 16 possible models.

(i) [3 pts] $\alpha_1 = A \vee B$: $\boxed{12}$

There are 12 possible worlds where $A \vee B$ is true: 3 out of 4 possible assignments to A and B satisfy $A \vee B$, and for each of those, all four possible assignments to C and D.

(ii) [3 pts] $\alpha_2 = (A \wedge B) \Rightarrow C$: $\boxed{14}$

There are 14 possible worlds where $(A \wedge B) \Rightarrow C$ is true. The sentence is *False* only when A and B are true and C is false, i.e., exactly one assignment to A, B, C, and for that case, there are two possible assignments to D. So there are two worlds where it is false and hence 14 where it is true.

(iii) [4 pts] $\alpha_3 = (A \vee B) \wedge (\neg C \vee D)$: $\boxed{9}$

There are 4 possible worlds where $A \vee B$ is false and 4 possible worlds where $\neg C \vee D$ is false, and 1 when they are both false. So the total is 16 - (4+4-1) = 9.

(b) Suppose there are three chairs in a row, labeled L(eft), M(iddle), R(ight) and three persons A, B, and C. Everyone has to sit down but, unfortunately,

- A doesn't want to sit next to B
- A doesn't want to sit in the left chair
- C doesn't want to sit to the (adjacent) right of B

We will formulate these constraints in propositional logic using only variable $X_{p,c}$ to mean that person p sits in chair c. **Please express the constraints in CNF.**

Note: As long as your solutions is in CNF form and is logically equivalent to our expressions below, you can give yourself full credit.

(i) [2 pts] A doesn't want to sit next to B.

$(\neg x_{A,L} \vee \neg x_{B,M}) \wedge (\neg x_{A,R} \vee \neg x_{B,M}) \wedge (\neg x_{A,M} \vee \neg x_{B,L}) \wedge (\neg x_{A,M} \vee \neg x_{B,R})$

(ii) [1 pt] A doesn't want to sit in the left chair.

$\neg x_{A,L}$

(iii) [1 pt] C doesn't want to sit to the (adjacent) right of B.

$(\neg x_{B,L} \vee \neg x_{C,M}) \wedge (\neg x_{B,M} \vee \neg x_{C,R})$

(iv) [4 pts] Besides the constraints from the first 3 parts, are there any other constraints needed to solve the problem? If yes, express them in propositional logic, including only the minimum number of constraints needed to solve the problem.

There are two other constraints that need to be satisfied: each person is placed and there's only one person per chair.

Each person is placed: $(x_{A,L} \vee x_{A,M} \vee x_{A,R}) \wedge (x_{B,L} \vee x_{B,M} \vee x_{B,R}) \wedge (x_{C,L} \vee x_{C,M} \vee x_{C,R})$

At most one person per chair:

$(\neg x_{A,L} \vee \neg x_{B,L}) \wedge (\neg x_{A,L} \vee \neg x_{C,L}) \wedge (\neg x_{B,L} \vee \neg x_{C,L}) \wedge$

$(\neg x_{A,M} \vee \neg x_{B,M}) \wedge (\neg x_{A,M} \vee \neg x_{C,M}) \wedge (\neg x_{B,M} \vee \neg x_{C,M}) \wedge$

$(\neg x_{A,R} \vee \neg x_{B,R}) \wedge (\neg x_{A,R} \vee \neg x_{C,R}) \wedge (\neg x_{B,R} \vee \neg x_{C,R}) \wedge$

(v) [1 pt] Lastly, can we satisfy these constraints? (A yes/no answer with some justification is sufficient for the problem. You do not need a formal proof.)

No these constraints can't be satisfied. An acceptable brief argument can be as follows:

- The first constraint means that the seating is either A C B or B C A.
- The second constraint means that the seating can only be B C A.
- This forces the third constraint to be violated, since C is to the right of B in our only possible remaining assignment.

(c) Logical Inference

Given

$$KB = (A, A \Rightarrow B, A \Rightarrow C, B \wedge C \Rightarrow D)$$

(i) [4 pts] Show the steps in a forward chaining algorithm for proving $KB \models D$. Make sure to list the agenda, inferred, and count at the beginning and at each iteration of the algorithm.

1. We start out with:
   - `agenda` $= \{A\}$
   - `inferred` $= \{A : F, B : F, C : F, D : F\}$ (where T, F mean True and False).
   - `count` $= (0, 1, 1, 2)$
2. After iteration 0:
   - `agenda` $= \{B, C\}$
   - `inferred` $= \{A : T, B : F, C : F, D : F\}$
   - `count` $= (0, 0, 0, 2)$
3. After iteration 1:
   - `agenda` $= \{C\}$
   - `inferred` $= \{A : T, B : T, C : F, D : F\}$
   - `count` $= (0, 0, 0, 1)$
4. After iteration 2:
   - `agenda` $= \{D\}$
   - `inferred` $= \{A : T, B : T, C : T, D : F\}$
   - `count` $= (0, 0, 0, 0)$, so we add the conclusion of the clause $C \Rightarrow D$ to agenda.
5. On iteration 3, we meet the stopping condition since the symbol D is popped off the agenda, which is the conclusion (consequent) we wanted to find in the sentence $KB \models D$.

(ii) [2 pts] How do we prove $KB \models D$ using a SAT solver? (Hint: The solution is simple and takes just one line.)

Show that $KB \wedge \neg D$ is unsatisfiable

(iii) [4 pts] Write out the necessary clauses in CNF representation of the sentence required for the previous part.

First we convert each clause to CNF.
$A : A$
$A \Rightarrow B : \neg A \vee B$
$A \Rightarrow C : \neg A \vee C$
$B \wedge C \Rightarrow D : (\neg B \vee \neg C \vee D)$
Therefore, the CNF representation will be:

$$(A) \wedge (\neg A \vee B) \wedge (\neg A \vee C) \wedge (\neg B \vee \neg C \vee D) \wedge (\neg D)$$

(iv) [10 pts] Show the steps in the operation of DPLL, assuming a fixed variable ordering (A, B, C, D) and a fixed value ordering (true before false). Remember to apply early termination, pure literals (repeatedly), and unit clauses (repeatedly), keeping track of which clauses have already been satisfied in the process. Make sure to list the intermediate steps for the model, symbols, and clauses at the beginning and at each iteration of the algorithm.

We are trying to show that $A \wedge (\neg A \vee B) \wedge (\neg A \vee C) \wedge (\neg B \vee \neg C \vee D) \wedge \neg D$ is unsatisfiable.

1. We start out with
   - Model: $\{\}$
   - Symbols: A, B, C, D
   - Clauses: $A \wedge (\neg A \vee B) \wedge (\neg A \vee C) \wedge (\neg B \vee \neg C \vee D) \wedge \neg D$ (each clause separated by the $\wedge$)
2. On iteration 0, there are no pure literals, but there are unit clauses $(A, \neg D)$. We stick to our fixed variable ordering and assign $A = True$ first. So we make a recursive call to DPLL with the following params:
   - Model: $\{A = T\}$
   - Symbols: B, C, D
   - Clauses: $A \wedge (\neg A \vee B) \wedge (\neg A \vee C) \wedge (\neg B \vee \neg C \vee D) \wedge \neg D$ (nothing has changed from initialization yet)
3. On iteration 1, there are no pure literals, but there are unit clauses $(B, C, \neg D)$. We assign B first, tiebreaking per our fixed variable ordering. So we make a recursive call to DPLL with the following params:
   - Model: $\{A = T, B = T\}$
   - Symbols: C, D
   - Clauses: $B \wedge C \wedge (\neg B \vee \neg C \vee D) \wedge \neg D$
4. On iteration 2, there are no pure literals, but there are unit clauses (C). So we make a recursive call to DPLL with the following params:
   - Model: $\{A = T, B = T, C = T\}$
   - Symbols: D
   - Clauses: $C \wedge (\neg C \vee D) \wedge \neg D$
5. On iteration 3, there are no pure literals, but there are unit clauses $(D, \neg D$. So we make a recursive call to DPLL with the following params:
   - Model: $\{A = T, B = T, C = T, D = T\}$
   - Symbols: $\emptyset$
   - Clauses: $D \wedge \neg D$
6. On iteration 4, we meet the stopping condition since the clauses, after plugging in our model, evaluate $D \wedge \neg D$, which is $True \wedge False$. Since some clause is false via our model, we return false (no solution).

Since $KB \wedge \neg D$ is unsatisfiable, we can conclude that $KB \models D$