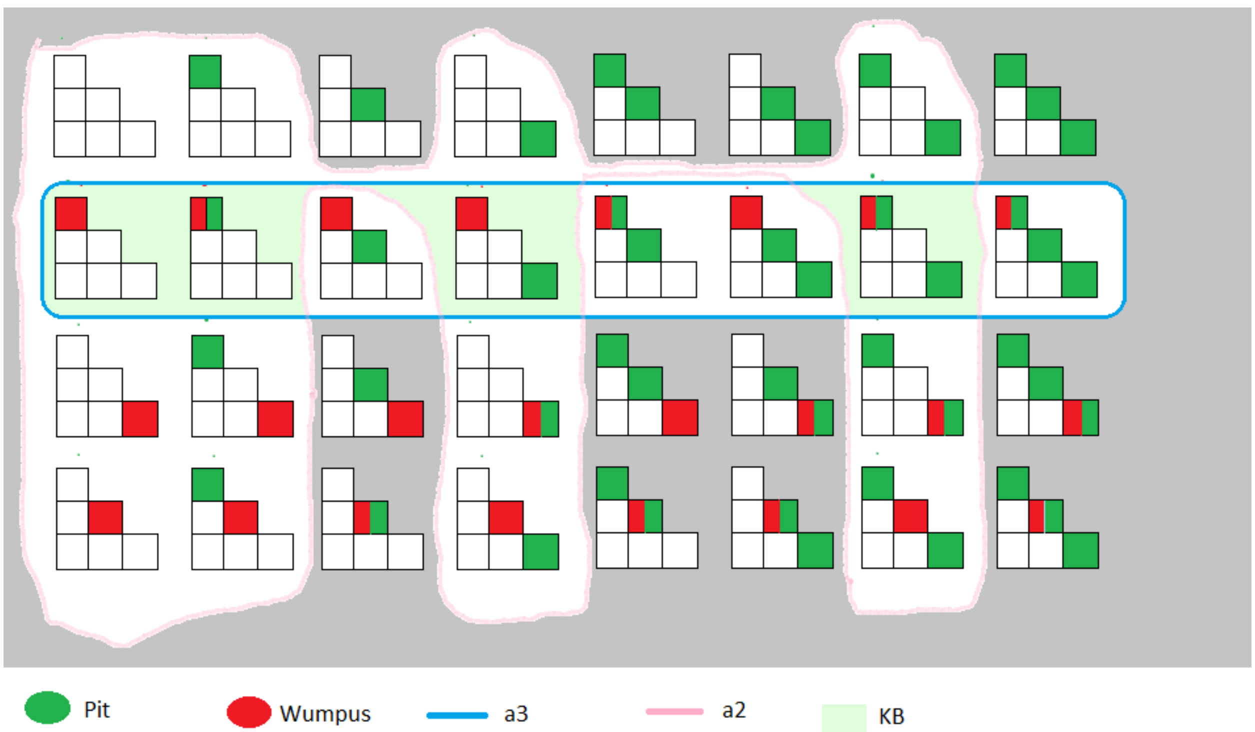
Paul Philip Mitchell

TDT4136 Assignment 2 – Propositional and Predicate Logic

1 Models and Entailment in Propositional Logic

1.1 Exercise 7.1

Build the complete model table and show both entailments using model checking.



The world matching the knowledge base (KB) is highlighted in light green.

1.2 Exercise 7.4

a. **False |= True**
    a. True. *False* does not have any models and therefore entails all sentences. In addition, *True* is true for all models and therefore it is entailed by all sentences.
b. **True |= False**
    a. False. See above.
c. **$(A \land B) \models (A \Leftrightarrow B)$**
    a. True. Only solutions for LHS are *A = B = True*, and for RHS only *A = B* applies.
d. **$A \Leftrightarrow B \models A \lor B$**
    a. False. One of the models of LHS have *A* and *B* false values. This does not satisfy RHS.
e. **$A \Leftrightarrow B \models \neg A \lor B$**
    a. True. RHS is equal to *A => B*. *A => B* is one of several conjuncts of $A \Leftrightarrow B$.
f. **$(A \land B) \Rightarrow C \models (A \Rightarrow C) \lor (B \Rightarrow C)$**
    a. True. *P => Q* is only false if *P* is true and *Q* is false. Therefore, on LHS, $A \land B = P$ which is true and *C* is false.
g. **$(C \lor (\neg A \land \neg B)) \equiv ((A \Rightarrow C) \land (B \Rightarrow C))$**
    a. True. Done by truth table.
h. **$(A \lor B) \land (\neg C \lor \neg D \lor E) \models (A \lor B)$**
    a. True. The set will be smaller by a conjunct, and therefore LHS must be a subset of RHS.
i. **$(A \lor B) \land (\neg C \lor \neg D \lor E) \models (A \lor B) \land (\neg D \lor E)$**
    a. False. If you delete a disjunct, you will allow for fewer models.
j. **$(A \lor B) \land \neg (A \Rightarrow B)$ is satisfiable**
    a. True. If A is true and B is false, it is satisiable.
k. **$(A \Leftrightarrow B) \land (\neg A \lor B)$ is satisfiable**
    a. True. This is the same as task e.
l. **$(A \Leftrightarrow B) \Leftrightarrow C$**
    a. True. $(A \Leftrightarrow B) \Leftrightarrow C$ is satisfiable if *(A, B, C) = (f, f, t) v (f, t, f) v (t, f, f) v (t, t, t).* Similarly, $A \Leftrightarrow B$ is satisfiable if *(A, B, C) = (f, f, f) v (f, f, t) v (t, t, f) v (t, t, t).*

1.3 EXERCISE 7.7

Consider a vocabulary with only four propositions, *A, B, C,* and *D.* How many models are there for the following sentences?

- **a.** A can be either true or false and D can be either true or false, which leads to four possible values. Both B and C have three valid values. This leaves us with four times three values, which is twelve.
- **b.** This expression can be written as $\neg (A \wedge B \wedge C \wedge D)$. All the possible values are 16. The expression is only insatisfiable if all values are false, which leaves us with 15 models.
- **c.** In this case, A has to be true and B has to be false, but that leaves *A => B* to be false. Therefore, there are no models for this expression.

1.4 EXERCISE 7.10

- **a.** Valid
- **b.** Neither
- **c.** Neither
- **d.** Valid
- **e.** Valid
- **f.** Valid
- **g.** Valid

1.5

Consider a logical knowledge base with 100 variables, A1, A2, . . . , A100. This will have Q = 2^{100} possible models. For each logical sentence below, give the number of models that satisfy it. Feel free to express you answer as a fraction of Q (without writing out the whole number 1267650600228229401496703205376 = 2^{100}) or to use other symbols to represent large numbers.

- **a.** $\mathbf{A_1 \vee A_{73}}$. There are four possible values. It is only false if A_1 and A_{73} *is false.* That leaves $\frac{3}{4} * 2^{100}$ models.
- **b.** $A7 \vee (A19 \wedge A33)$. There are $2^3 = 8$ possible values. Of these, there are three expressions which will make the sentence false. This leaves us 5/8 * 2^{100} remaining models.
- **c.** $A11 \rightarrow A22$. There are four possible values. Of these, the sentence will be false only if *A11* is false. This leaves us ¾ * 2^{100} remaining models.

2 RESOLUTION IN PROPOSITIONAL LOGIC

2.1

Convert each of the following sentences to Conjunctive Normal Form (CNF).

a. **A ⋀ B ⋀ C.** This sentence is already written in Conjunctive Normal Form.
b. **A ⋁ B ⋁ C.** This sentence is already written in Conjunctive Normal Form.
c. **A → (B ⋁ C).** In Conjunctive Normal Form, this would be written as *¬A ⋁ (B ⋁ C)*.

2.2

Consider the following Knowledge Base (KB)
Use resolution to show that KB |= ¬E.

(A ⋁ ¬B) → ¬C. $\neg(A \vee \neg B) \vee \neg C = (\neg A \wedge B) \vee \neg C = (\neg A \vee \neg C) \wedge (B \vee \neg C)$

D ⋀ E → C $\neg(D \wedge E) \vee C = (\neg D \vee \neg E) \vee C = \neg D \vee \neg E \vee C$

A ⋀ D $A \wedge D$

$$(\neg D \vee \neg E \vee C) \wedge A \wedge D = (\neg E \vee C) \wedge A \wedge D$$

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

$$\big((\neg E \vee C) \wedge A \wedge D\big) \wedge (\neg C \wedge A \wedge D) = A \wedge D \wedge \neg E$$

$$A \wedge D \wedge \neg E \ (proper\ subset\ of) \ \neg E \ \Rightarrow KB \neq \neg E$$

3 REPRESENTATIONS IN FIRST-ORDER LOGIC

3.1 EXERCISE 8.9

This exercise uses the function *Map Color* and predicates *In(T, y), Borders (x ,y),* and *Country(x),* whose arguments are geographical regions, along with constant symbols for various regions. In each of the following we give an English sentence and a number of candidate logical expressions. For each of the logical expressions, state whether it (1) correctly expresses the English sentence; (2) is syntactically invalid and therefore meaningless; or (3) is syntactically valid but does not express the meaning of the English sentence.

a. Paris and Marseilles are both in France.

(i) In(Paris* A *Marseilles, France). Is syntactically invalid and therefore meaningless.
(ii) In(Paris, France)* A *In(Marseilles, France). Correctly expresses the English sentence.
(iii) In(Paris, France) v In(2)4Urseilles, France). Is syntactically valid but does not express the meaning of the English sentence.

b. There is a country that borders both Iraq and Pakistan.

(i) Correctly expresses the English sentence.
(if) Is syntactically valid but does not express the meaning of the English sentence. Reveals true for any object that is not a country.
(iii) Syntactically invalid and therefore meaningless. In the right hand side of the sentence, c is not defined.
(iv) Syntactically invalid and therefore meaningless. You cannot use a conjunction inside a term.

c. All countries that border Ecuador are in South America.

(i) Correctly expresses the English sentence.
(ii) Correctly expresses the English sentence.
(iii) Syntactically invalid and therefore meaningless.
(iv) Syntactically invalid and therefore meaningless.

3.2 EXERCISE 8.21

In Chapter 6, we used equality to indicate the relation between a variable and its value. For instance, we wrote *WA = red* to mean that Western Australia is colored red. Representing this in first-order logic, we must write more verbosely *ColorOf (WA) = red.* What incorrect inference could be drawn if we wrote sentences such as WA = *red* directly as logical assertions?

WA refers to a certain state, Western Australia, in Australia. This state has a set of properties, including color. The only thing we want to do, is to compare a single property of this state. If we do not compare only a single property, we could deduce that anything that is red is equal to Western Australia. That is, if WA is red and Q is red, we could deduce that WA is equal to Q, which we of course do not want.

3.3 EXERCISE 8.23

For each of the following sentences in English, decide if the accompanying first-order logic sentence is a good translation. If not, explain why not and correct it. (Some sentences may have more than one error!)

a. No two people have the same social security number.
   a. This sentence does not handle the situation where both *x* and *y* are equal. Correct version is:

$$\neg\exists x, y, n\ Person(x) \wedge Person(y) \wedge \neg(x = y) \wedge [HasSS\#(x,n) \wedge HasSS\#(y,n)]$$

b. John's social security number is the same as Mary's.
   a. This is a correct sentence.

c. Everyone's social security number has nine digits.
   a. This sentence only specifies that everyone has <u>all</u> social security numbers. Correct version is:

$$\forall x, n\ Person(x) \wedge HasSS\#(x,n) \Rightarrow Digits(n, 9)$$

3.4

Translate into first-order logic the sentence "Everyone's DNA is unique and is derived from their parents' DNA." You must specify the precise intended meaning of your vocabulary terms. (Hint: Do not use the predicate Unique(x), since uniqueness is not really a property of an object in itself!)

We have the function $Equals(x, y)$ which tells us if *x* and *y* are equal.
We have the function $Derives(x, z)$ which tells us if *x* is derived from *y*.

$$\forall xy\ \neg Equals(x, y) \land \exists xz\ Derives(x, z)$$

4 RESOLUTION IN FIRST-ORDER LOGIC

4.1

Find the unifier (θ) - if possible - for each pair of atomic sentences. Here, Owner, Horse and Rides are predicates, while FastestHorse is a function that maps a person to the name of their fastest horse:

c. $\boldsymbol{Owner}(\boldsymbol{Leo}, \boldsymbol{x}) \ldots \boldsymbol{Owner}(\boldsymbol{y}, \boldsymbol{Rocky})$

$$\theta = \{\frac{y}{Leo}, \frac{x}{Rocky}\}$$

d. $\boldsymbol{Owner}(\boldsymbol{Leo}, \boldsymbol{x}) \ldots \boldsymbol{Rides}(\boldsymbol{Leo}, \boldsymbol{Rocky})$

$$\theta = \{\frac{x}{Rocky}\}$$

e. $\boldsymbol{Owner}\big(\boldsymbol{x}, \boldsymbol{FastestHorse}(\boldsymbol{x})\big) \ldots \boldsymbol{Owner}(\boldsymbol{Leo}, \boldsymbol{Rocky})$

$$\theta = \{\frac{x}{Leo}\}$$

4.2

Use resolution to prove Green(Linn) given the information below. You must first convert each sentence into CNF. Feel free to show only the applications of the resolution rule that lead to the desired conclusion. For each application of the resolution rule, show the unification binding, θ.

$$\boldsymbol{Hybrid}(\boldsymbol{Prius})$$

$$\boldsymbol{Drives}(\boldsymbol{Linn}, \boldsymbol{Prius})$$

$$\forall \boldsymbol{x}: \boldsymbol{Green}(\boldsymbol{x}) \leftrightarrow \boldsymbol{Bikes}(\boldsymbol{x}) \vee [\exists \boldsymbol{y}: \boldsymbol{Drives}(\boldsymbol{x}, \boldsymbol{y}) \wedge \boldsymbol{Hybrid}(\boldsymbol{y})]$$

$$\forall x\ Green(x) \Rightarrow Bikes(x) \vee [\exists y\ Drives(x,y) \wedge Hybrid(y)]] \wedge [\forall x\ Bikes(x) \vee [\exists y\ Drives(x,y) \wedge Hybrid(y)] \Rightarrow Green(x)]$$

$$\forall x\ [\neg Green(x) \vee Bikes(x) \wedge [\exists y\ Drives(x,y) \wedge Hybrid(y)]] \wedge [\neg[Bikes(x) \vee [\exists y\ Drives(x,y) \wedge Hybrid(y)]] \vee Green(x)]$$

$$\forall(x)[\neg Green(x) \vee Bikes(x) \vee [\exists y\ Drives(x,y) \wedge Hybrid(y)]] \wedge [[\neg Bikes(x) \wedge \neg[\exists y\ Drives(x,y) \wedge Hybrid(y)]] \vee Green(x)]$$

$$\forall x\ [\neg Green(x) \vee Bikes(x) \vee [\exists y\ Drives(x,y) \wedge Hybrid(y)]] \wedge [[\neg Bikes(x) \wedge [\forall y\ \neg Drives(x,y) \vee \neg Hybrid(y)]] \vee Green(x)]$$

Next, in order to remove the existential quantifier, we have to replace *y* with *F(x)*

$$\forall x\ [\neg Green(x) \vee Bikes(x) \vee [Drives(x, F(x)) \wedge Hybrid(F(x))]] \wedge [[\neg Bikes(x) \wedge [\forall y\ \neg Drives(x,y) \vee \neg Hybrid(y)]] \vee Green(x)] [\neg Green(x) \vee Bikes(x) \vee [Drives(x, F(x)) \wedge Hybrid(F(x))]] \wedge [[\neg Bikes(x) \wedge [\neg Drives(x,y) \vee \neg Hybrid(y)]] \vee Green(x)] [\neg Green(x) \vee Bikes(x) \vee Drives(x, F(x))] \wedge [\neg Green(x) \vee Bikes(x) \vee Hybrid(F(x))] \wedge [Bikes(x) \vee Green(x)] \wedge [\neg Drives(x,y) \vee \neg Hybrid(y) \vee Green(x)]$$

Next, we convert our sentences to Conjunctive Normal Form.

$$[\neg Green(x) \vee Bikes(x) \vee Drives(x, F(x))]$$

$$[\neg Green(x) \vee Bikes(x) \vee Hybrid(F(x))]$$

$$[\neg Bikes(x) \vee Green(x)]$$

$$[\neg Drives(x, y) \vee \neg Hybrid(y) \vee Green(x)]$$

$$Hybrid(Prius)$$

$$Drives(Linn, Prius)$$

Finally,

$$\theta = \{\frac{y}{Prius}\}$$

$$[\neg Drives(x, Prius) \vee \neg Hybrid(Prius) \vee Green(x)], Hybrid(Prius) \neg [\neg Drives(x, Prius) \neg Green(x)]$$

$$\theta = \{\frac{x}{Linn}\}$$

$$[\neg Drives(Linn, Prius) \vee Green(Linn)], Drives(Linn, Prius) \neg Green(Linn)$$