

# THE CALCULUS OF COMPUTATION

HIDENORI SHINOHARA

## 1. CHAPTER 1

### Exercise (1.1).

(a) Assume that there is a falsifying interpretation  $I$ .

1.  $I \models P \wedge Q \rightarrow P \rightarrow Q$  (assumption)
2.  $I \models P \wedge Q$  (by 1 and semantics of  $\rightarrow$ )
3.  $I \not\models P \rightarrow Q$  (by 1 and semantics of  $\rightarrow$ )
4.  $I \models Q$  (by 2 and semantics of  $\wedge$ )
5.  $I \not\models Q$  (by 3 and semantics of  $\rightarrow$ )
6.  $I \models \perp$  (4 and 5 are contradictory)

There is only one branch and it is closed. Thus  $F$  is valid.

### Exercise (1.2).

(a) To prove that  $\top \Leftrightarrow \neg\perp$ , we prove that  $\top \leftrightarrow \neg\perp$  is valid. Assume that there is a falsifying interpretation  $I$  such that  $I \not\models \top \leftrightarrow \neg\perp$ . We apply the semantics of  $\leftrightarrow$ .

The first branch is:

- 1a.  $I \models \top \wedge \neg(\neg\perp)$
- 2a.  $I \models \neg(\neg\perp)$  (by 1a and semantics of  $\wedge$ )
- 3a.  $I \not\models \neg\perp$  (by 2a and semantics of  $\neg$ )
- 4a.  $I \models \perp$  (by 3a and semantics of  $\neg$ )

The second branch is:

- 1b.  $I \models \neg\top \wedge \neg\perp$
- 2b.  $I \models \neg\top$  (by 1b and semantics of  $\wedge$ )
- 3b.  $I \not\models \top$  (by 2b and semantics of  $\neg$ )
- 4b.  $I \models \top$  (Under any interpretation,  $\top$  has value true)
- 5b.  $I \models \perp$  (3b and 4b are contradictory)

Thus both branches are closed, and thus  $\top \leftrightarrow \neg\perp$  is valid.

(b) We will apply a strategy similar to that of Example 1.13. To prove  $\perp \Leftrightarrow \neg\top$ , we prove that  $F : \perp \leftrightarrow \neg\top$  is valid. Suppose  $F$  is not valid; there exists an interpretation  $I$  such that  $I \not\models F$ . There are exactly two branches.

The first branch is:

- 1a.  $I \models \perp \wedge \neg(\neg\top)$  (by semantics of  $\leftrightarrow$ )
- 2a.  $I \models \perp$  (by 1a and semantics of  $\wedge$ )

The second branch is:

- 1b.  $I \models (\neg\perp) \wedge \neg(\neg\top)$  (by semantics of  $\leftrightarrow$ )
- 2b.  $I \models \neg\top$  (by 1b and semantics of  $\wedge$ )
- 3b.  $I \not\models \top$  (by 2b and semantics of  $\neg$ )
- 4b.  $I \models \top$  (by definition, P.7)

5b.  $I \models \perp$  (by 3b and 4b)  
 Both of these two branches are closed;  $F$  is valid.

**Exercise (1.3).**

- $\perp$  is equivalent to  $\neg\top$ . In other words,  $\perp \Leftrightarrow \neg\top$  and that is proved in Exercise 1.2(b).

**Exercise (1.4).** We claim that  $\neg F \Leftrightarrow F \overline{\wedge} F$ . To prove that, we prove  $\neg F \leftrightarrow F \overline{\wedge} F$  is valid.

$F$	$\neg F$	$F \overline{\wedge} F$	$\neg F \leftrightarrow F \overline{\wedge} F$
0	1	1	1
1	0	0	1

**Exercise (1.5).**

- (a) By using the list of template equivalences on P.19, we can obtain the negation normal form of the original formula as following:
- $F : \neg(P \rightarrow Q)$ .
  - $F' : \neg(\neg P \vee Q)$ .
  - $F'' : \neg(\neg P) \wedge \neg Q$ .
  - $F''' : P \wedge \neg Q$ .
- The only connectives in  $F'''$  are  $\neg$ ,  $\wedge$ , and  $\vee$  and the negations appear only in literals. Thus  $F'''$  is in NNF. Furthermore,  $F'''$  is actually in CNF and DNF since it is the disjunction of one conjunction, and it is the conjunction of two clauses.
- (b)  $F' : (\neg P \vee \neg Q) \wedge R$  is the NNF of  $F$  that can be obtained using the same strategy as above.  $F'$  is already in CNF.  $F'' : (\neg P \wedge R) \vee (\neg Q \wedge R)$  is an equivalent formula in DNF.

**Exercise (1.6).**

- (a)  $\bigwedge_{v \in V} \bigvee_{c \in C} P_v^c$ .
- (b)  $\bigwedge_{(v,w) \in E} \bigvee_{c_1 \neq c_2 \in C} (\neg P_v^{c_1} \vee \neg P_w^{c_2})$ .
- (c)  $\bigwedge_{(v,w) \in E} \bigvee_{c \in C} (\neg P_v^c \vee \neg P_w^c)$ .
- (d) No clue. The problem statement seems too ambiguous.
- (e) They are already in CNF. We have  $N \cdot M$  variables and  $N \cdot M + K \cdot M \cdot (M - 1) + K \cdot M$  clauses in the encoding above.

**Exercise (1.7).**

- (a)  $P_{(F)}$  contains 1 term. Each  $\text{En}(G)$  contains  $1, 1, 2 \cdot 2 \cdot 3, 1, 1, 2 \cdot 2 \cdot 3, 3 \cdot 2 \cdot 2$  clauses. Thus the expansion would contain 1728 clauses, which is the product of the numbers of clauses.
- (b)
- $2^n$  clauses.
  - $\text{Rep}(F_n) = P_{F_n}$ , so it is just one clause. We have the subformula set  $S_{F_n} = \{Q_1, \dots, Q_n\} \cup \{R_1, \dots, R_n\} \cup \{Q_i \wedge R_i \mid 1 \leq i \leq n\}$ . We will consider how many clauses  $\text{En}(G)$  has for each  $G \in S_{F_n}$ .
    - For each  $Q_i$  and  $R_i$ , we have  $\text{En}(Q_i) = \text{En}(R_i) = \top$ . This adds  $2n$  clauses to  $F'$ .

- $\text{En}(Q_i \wedge R_i)$  contains 3 clauses as defined on P.25. This adds  $3n$  clauses to  $F'$ .

Therefore, in total,  $F'$  contains  $1 + 2n + 3n = 5n + 1$  clauses. Note that  $5n + 1 < 30n + 2$  where  $30n + 2$  is the upper bound described on P.26.

I am not sure if a subformula can be bigger. In other words, for instance, should  $(Q_1 \wedge R_1) \vee (Q_2 \wedge R_2)$  be a subformula of  $F_3$ ? It seems unnecessary to include such cases for the purpose of CNF conversion, but it seems to be more consistent with the definition of a subformula.

- (iii) When  $n \leq 4$ ,  $2^n$  is smaller than  $5n + 1$ . For  $n \geq 5$ ,  $2^n$  is bigger than  $5n + 1$ .

**Exercise (1.8).**

- (a) We will follow the format described in Example 1.30. Branching on  $Q$  or  $R$  will result in unit clauses; choose  $Q$ . Then  $F\{Q \mapsto \top\} : (P \vee \neg R) \wedge (R)$ .  $P$  appear only positively, so we consider  $F\{Q \mapsto \top, P \mapsto \top\} : R$ . Then  $R$  appears only positively, so the formula is satisfiable. In particular,  $F$  is satisfied by interpretation

$$I : \{P \mapsto \text{true}, Q \mapsto \text{true}, R \mapsto \text{true}\}.$$