The Second Assignment

3190102721 Xu Shengze

Note: Some topics in this homework were completed after discussing with Zhou Yuxin.

Problem 1.1 Construct an algorithm that determines whether a given set of Boolean functions $\mathcal{A}$ constitues a complete basis. (Functions are represented by tables.)

Answer 1.1

According to the meaning of the question, all functions here are based on two variables. We consider formulating these functions on the basis of $\mathcal{A}$, then $\mathcal{A}$ is complete. Because functions are represented by tables, and there are four combinations of two variables, each of which has two values, the number of functions for two variables is $2^4 = 16$.

Next we need to construct an algorithm to check whether $\mathcal{A}$ contains these 16 functions. First, we need to check whether there are functions in $\mathcal{A}$ that can handle two variables $p_1(x, y) = x$ and $p_2(x, y) = y$. If the conclusion is negative, it means that $\mathcal{A}$ is incomplete. If it exists, the next step is for the set $\mathcal{F}$ of already constructed functions. We add to the set $\mathcal{F}$ all functions of the form $f(g_1(x_1, x_2), g_2(x_3, x_4), \cdots, g_k(x_{2k-1}, x_{2k}))$, where $x_j \in \{x, y\}, g_j \in \mathcal{F}, f \in \mathcal{F}$. If the set $\mathcal{F}$ increases, we repeat the procedure. Otherwise, there are two possibilities. One is that we have got all the functions in two variables when the algorithm stops, which shows that $\mathcal{A}$ is complete. Therefore, there is another possibility that $\mathcal{A}$ is incomplete.

Problem 2.2 Let c_n be the maximum complexity $c(f)$ for Boolean functions f in n variables. Prove that $1.99^n < c_n < 2.01^n$

Answer

As for the upper bound, it's obvious that an upper bound is $O(n2^n)$. A circuit composed of n variables ends in at most n steps, and each step has at most 2^n assignments. Hence the upper bound of the size is $O(n2^n)$ which is less than 2.01^n when n is large enough.

As for the lower bound, we compare the number of Boolean functions in n variables (obviously, we have 2^{2^n} functions) and the number of all circuits of a given size. Assume that the standard complete basis is used. For the k-th step of assignment of the circuit there are at most $O((n+k)^2)$ possibilities (two arguments can be chosen among $n+k-1$ variables, where there are n input and $k-1$ auxiliary variables, then the lower bound is $O(C_{n+k-1}^2) = O((n+k)^2)$.) Therefore, the number N_s of different circuits of size s does not exceed

$$O(((n+s)^2)^s) = 2^{2s(\log(n+s)+O(1))}$$

But the number of Boolean functions in n variables equals 2^{2^n}. If

$$2^n > 2s(\log(n+s)+O(1))$$

there are more functions than circuits, so that $c_n > s$. If $s = 1.99^n$, then the inequality above is satisfied for suffiiently large n.