1901210488 时绍森

设计思路

本次作业的核心点在于理解 **DDT** 表和 **ACT** 表是如何计算出来的。

1.祖冲之 S 盒的介绍

根据国密祖冲之算法的介绍可以知道，祖冲之算法的 **S** 盒共分四个小 **S** 盒，分别为 **S0**、**S1**、**S2**、**S3**，其中 **S2=S0**、**S3=S1.**本次作业让计算的就是其中两个小盒 **S0** 和 **S1** 的 **DDT** 表和 **ACT** 表。

S 盒的输入输出是 **4** 字节，即 **32** 位，每个小盒输入输出为 **8** 位。

设 **S0** 的 **8** 比特输入为 **X**，将 **X** 视作两个 **16** 进制数的连接，即 **X=H||L**，则 **S0** 盒中第 **H** 行和第 **L** 列交叉的元素即为 **S0** 的输出 **S0(X)**

设 **S** 盒 **S** 的 **32** 比特输入 **X** 和 **32** 比特输出 **Y** 分别为：

X= x0 || x1 || x2 || x3

Y= y0 || y1 || y2 || y3

其中 **xi** 和 **yi** 均为 **8** 比特**,i=0,1,2,3**。

则有则有 **yi=Si(xi),i=0,1,2,3**。

由于我们仅要求 **S0** 盒和 **S1** 盒的 **DDT** 表和 **ACT** 表，因此输入时仅有 **8** 位。

2.DDT 的计算

DDT 表的行为 **detX**，列为 **detY**，其中 **detX** 为 **X1** 与 **X2** 异或得到，**detY** 为 **X1** 与 **X2** 根据 **S** 盒得到的输出 **Y1** 与 **Y2** 异或得到。

由于输入 **X** 为 **8** 位，输出 **Y** 也为 **8** 位，因此表示的整数为 **2** 的 **8** 次方即 **256** 个数，因此 **DDT** 表大小为 **256*256**。

填充时采取按行填充的策略，因此以 **detX** 为变量，每给定一个 **x1**，将 **detX** 与 **x1** 异或得出 **x2**，将 **x1** 与 **x2** 分为前后两部分各 **4** 位，作为行与列去 **S** 盒中交叉寻找对应输出 **y1** 与 **y2**，做异或后得到 **detY**，然后根据 **detX** 与 **detY** 将 **DDT** 中对应表项自增。

由于有 **8** 位共 **256** 种 **detX**，每种 **detX** 有 **256** 对 **X1** 与 **X2** 来计算 **detY**，因此程序复杂度为 **256*256**。

代码以及算出的 **DDT** 表见文件夹。

3.LAT 的计算

与 **DDT** 不同，**LAT** 表的行为 **8** 位 **X** 输入，列为由 **8** 位 **X** 输入根据 **S** 盒得出的 **8** 位 **Y** 输出。**8** 位二进制数可以表示 **256** 个整数，因此 **LAT** 表的大小也为 **256*256**。

我们要寻找 **X** 与 **Y** 的不同位异或之后的线性关系来填充 **LAT** 表，由于一共有 **8** 位，因此 **X** 的 **8** 位每位均可参与异或或不参与异或，所以 **X** 有 **2** 的 **8** 次方即 **256** 种异或情况，同理 **Y** 也有 **256** 种异或情况，因此线性关系共有 **256*256** 种，填充 **LAT** 表时共有 **256** 种 **X** 输入，因此程序复杂度为 **256*256*256.**

我在程序中进行不同位异或的方法为设置一个 **int** 变量，从 **0** 自增到 **255**，将其转化为 **8** 位 **2** 进制数，为 **1** 的位数对应的 **X** 的相应位参与异或，为 **0** 的则不参与异或，因此随着 **int** 变量从 **0** 自增 **255**，**X** 所有位的所有异或组合均被涵盖。**Y** 的不同情况异或关系也同理。

代码以及算出的 **LAT** 表见文件夹。