ISSN 1000-9825 Journal of Software 软件学报 2000,11(10):1299~1302

# 满足若干密码学性质的S-盒的构造*

刘晓晨[1] 冯登国[2,3]

[1](中国科学技术大学研究生院信息安全国家重点实验室 北京 100039)

[2](中国科学院软件研究所信息安全国家重点实验室 北京 100080)

[3](中国科学院信息安全技术工程研究中心 北京 100080)

E-mail: fengdg@hotmail.com

**摘要** S-盒是许多密码算法的唯一非线性部件,它的密码强度决定了整个密码算法的安全强度.但是对于大的S-盒的构造比较困难,而且软硬件实现也比较难,目前比较流行的是8×8的S-盒.基于m-序列,提出一种构造8×8与8×6的S-盒的方法,通过测试法从中选出了一批非线性性质与差分均匀性都比较好的S-盒.同时,基于正形置换构造了一批4×4的S-盒.这些S盒对进一步设计密码算法提供了非线性资源.

**关键词** 分组密码,S-盒,正形置换.

**中图法分类号** TP393

替换或置换是构成对称密钥分组密码算法的基本变换函数,扮演着非线性变换的核心角色.S-盒首次出现于Lucifer算法中,随后因DES(data encryption standard)的使用而广为流行.S-盒也因此成为密码学界的重要研究对象,研究内容集中于S-盒的设计准则及构造方法[1].纵观近几年的研究成果,S-盒主要有以下一些设计准则:非线性度($N_S$)、差分均匀度($\delta$)、代数次数及项数分布、完全性和正交性.

基于以上设计准则,人们提出了许多构造方法,如随机选取并测试、使用数学函数等方法.随机选取方法要求设计者有足够的时间和计算能力.使用数学函数可以构造一些好的S-盒,目前常用的此类S-盒有:指数函数和对数函数(SAFER系列密码采用)、有限域$GF(2^n)$上的逆映射(SHARK,SQUARE)以及有限域上的幂函数.

一般不直接用幂函数作S-盒,而是以它为基础构造新的S-盒.例如,E2的S-盒是用有限域$GF(2^n)$中的幂函数和模加复合而成.但以上应用数学函数构造的S-盒,其代数结构都比较简单,并且所构造的性质好的S-盒的数量也比较少.本文提出了一种基于m-序列的构造8×8的S-盒的方法.该方法构造简单,而且构造出的S-盒满足较好的密码学性质.稍加改进便可构造8×6的S-盒.所谓$n\times m$的S-盒是指具有$n$端输入、$m$端输出的变换.不失一般性,这里只讨论二元域的情况,此时,$n\times m$的S-盒实际上是一个从$F_2^n$到$F_2^m$的变换.

## 1 S-盒的构造方法

### 1.1 构造8×8的S-盒的方法

第1步:先生成$F_2$上的8级m-序列.

第2步:定义函数$\varphi$.

$\varphi: F_2^6 \rightarrow F_2^8$,

$X\rightarrow$m-序列中从第$X+1$为开始截取8位所得的二进制数,

$255\rightarrow 0$.

第3步:所得的函数$\varphi$即为8×8的S盒的置换:

---

* 本文研究得到国家自然科学青年基金(No.69703012)和国家重点基础研究发展规划项目(No.G1999035800)资助.作者刘晓晨,1975年生,助教,主要研究领域为密码学.冯登国,1965年生,博士,研究员,博士生导师,主要研究领域为信息安全.

本文通讯联系人:冯登国,北京 100080,中国科学院软件研究所信息安全国家重点实验室

本文2000-05-31收到原稿,2000-06-30收到修改稿

$$S(X)=(f_1(X),f_2(X),\ldots,f_8(X)):F_2^6\to F_2^8.$$

在$F_2[x]$中的8次本原多项式有

0x11d,0x12d,0x14d,0x15f,0x163,0x165,0x169,0x171,

0x187,0x18d,0x1a9,0x1c3,0x1cf,0x1e7,0xf5,0x1f9.

根据不同的初始值,即$a(x)$和$h(x)$,可构造出许多S-盒,选择适当的$a(x)$,$h(x)$可构造S-盒,$N_S=2^7-2^{4.7}$,$\delta=2^{-5}$.

利用上述方法构造的S-盒,其结构有一定的规律,为了改变这种结构,可通过函数复合来达到目的.

**引理1.** 令$S(X)=(f_1(X),f_2(X),\ldots,f_m(X)):F_2^n\to F_2^m$是一个多输出函数,$A:F_2^m\to F_2^m$和$B:F_2^n\to F_2^n$都是仿射双射,则有

(1) $N_S=N_{A\cdot S\cdot B}$,

(2) $\delta_S=\delta_{A\cdot S\cdot B}$.

### 1.2 构造8×6的S-盒的方法

在DES中利用了6×4的S-盒,对于$n\times m$的S-盒,一般要具有正交性.

**定义1.** 若对任意的$\beta\in F_2^m$,恰有$2^{n-m}$个$X\in F_2^n$,使得$S(X)=\beta$,则称$S(X)=(f_1(X),f_2(X),\ldots,f_m(X)):F_2^n\to F_2^m$是正交的.

**引理2.** 令$S(X)=(f_1(X),f_2(X),\ldots,f_m(X)):F_2^n\to F_2^m$是一个多输出函数,S-盒是正交的⇔$f_1,f_2,\ldots,f_m$任意的非零线性组合均是平衡函数.

利用m-序列构造$n\times m$的S-盒方法如下:

第1步:用同一个$h(x)$构造$m$个m-序列$C_1,C_2,\ldots,C_m$.

第2步:取$C_1,C_2,\ldots,C_m$序列的第$X+1$位$C_1^{(X)},C_2^{(X)},\ldots,C_m^{(X)}$,$0\leqslant X\leqslant 254$,按位排列$C_1C_2\ldots C_m$,组成一个$m$位的二进制数$Y$.

第3步:定义映射$S:F_2^n\to F_2^m$,

$$S(X)=Y,\quad 0\leqslant X\leqslant 254,$$

$$S(255)=0.$$

**定理1.** 用如上方法构造的S-盒,若$m$个m-序列的初始向量$a_1(x),a_2(x),\ldots,a_m(x)$是线性无关的,则S-盒是正交的.

证明:由引理2可知,S-盒正交⇔对于S-盒的输出$f_1(X),f_2(X),\ldots,f_m(X)$的任一非零线性组合$f_{i_1}(X)\oplus\ldots\oplus f_{i_k}(X)$为平衡函数.而$f_{i_1}(X)\oplus\ldots\oplus f_{i_k}(X)$为平衡函数⇔$\frac{a_{i_1}(x)+a_{i_2}(x)+\ldots+a_{i_k}(x)}{h(x)}$生成的序列为m-序列⇔$a_{i_1}(x)+a_{i_2}(x)+\ldots+a_{i_k}(x)\neq 0$(这里利用了m-序列的性质).所以,当$a_1(x),a_2(x),\ldots,a_m(x)$线性无关时⇒S-盒是正交的.

## 2 基于正形置换的4×4的S-盒的构造

**定义2.** 设$\theta:x\to\theta(x)$是$GF(2^n)$上的一个双射,定义$\varphi:x\to\varphi(x)=x\oplus\theta(x)$,若$\varphi$也是双射,则称$\theta$为正形变换.

正形置换的构造可等价于下面的问题[2].构造正交拉丁方表,见表1.

*Table 1*

表1

| 0 | 1 | 2 | … | $2^n-1$ |
|---|---|---|---|---|
| $1\oplus 0$ | $1\oplus 1$ | $1\oplus 2$ | … | $1\oplus 2^n-1$ |
| $2\oplus 0$ | $2\oplus 1$ | $2\oplus 2$ | … | $2\oplus 2^n-1$ |
| ⋮ | ⋮ | ⋮ | … | ⋮ |
| $2^n-1\oplus 0$ | $2^n-1\oplus 1$ | $2^n-1\oplus 2$ | … | $2^n-1\oplus 2^n-1$ |

正形变换等价于从表1中找一序列,使得每行、每列中均唯一地取一个数,并且使得这个序列是 $F_2^n \to F_2^n$ 的置换.

根据正交拉丁方表的取法(见表2),可得以下几个规则:

(1) 每个格子里只能取一个数.

(2) 每一行、每一列均取两个格子,相同的格子要取两次.

(3) 第一部分和第四部分共取8个格子,第二部分和第三部分共取8个格子.

(4) 表2中所示的4个部分,每部分均取4个格子.

**Table 2**

**表2**

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 1 | 0 | 3 | 2 | 5 | 4 | 7 | 6 | 9 | 8 | 11 | 10 | 13 | 12 | 15 | 14 |
| 2 | 3 | 0 | 1 | 6 | 7 | 4 | 5 | 10 | 11 | 8 | 9 | 14 | 15 | 12 | 13 |
| 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 11 | 10 | 9 | 8 | 15 | 14 | 13 | 12 |
| 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 12 | 13 | 14 | 15 | 8 | 9 | 10 | 11 |
| 5 | 4 | 7 | 6 | 1 | 0 | 3 | 2 | 13 | 12 | 15 | 14 | 9 | 8 | 11 | 10 |
| 6 | 7 | 4 | 5 | 2 | 3 | 0 | 1 | 14 | 15 | 12 | 13 | 10 | 11 | 8 | 9 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 9 | 8 | 11 | 10 | 13 | 12 | 15 | 14 | 1 | 0 | 3 | 2 | 5 | 4 | 7 | 6 |
| 10 | 11 | 8 | 9 | 14 | 15 | 12 | 13 | 2 | 3 | 0 | 1 | 6 | 7 | 4 | 5 |
| 11 | 10 | 9 | 8 | 15 | 14 | 13 | 12 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 |
| 12 | 13 | 14 | 15 | 8 | 9 | 10 | 11 | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 |
| 13 | 12 | 15 | 14 | 9 | 8 | 11 | 10 | 5 | 4 | 7 | 6 | 1 | 0 | 3 | 2 |
| 14 | 15 | 12 | 13 | 10 | 11 | 8 | 9 | 6 | 7 | 4 | 5 | 2 | 3 | 0 | 1 |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

一 二 三 四

下面说明一下上述4点的正确性.规则(1)和规则(2)由正形变换的等价命题显然可得.规则(3)可由规则(2)得出.规则(4)用反证法:设第一部分取 $m \neq 4$ 个格子,由规则(2)可知前4行应取8个格子,所以第二部分应取 $8-m$ 个格子.考虑后4列,同理,第四部分应取 $m$ 个格子.但由规则(3)知,第四部分应取 $8-m$ 个格子.$m \neq 4 \Rightarrow 8-m \neq m$,矛盾.

搜索所有的 $S_4$ 中的正形变换,用穷搜法要试 $16! = 20922789888000$ 种,这是不可能实现的.基于上述规则,将表2化简为表3.

**Table 3**

**表3**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| A | B | C | D | E | F | G | H |
| B | A | D | C | F | E | H | G |
| D | A | B | A | G | H | E | F |
| D | C | B | A | H | G | F | E |
| E | F | G | H | A | B | C | D |
| F | E | H | G | B | A | D | C |
| G | H | E | F | C | D | A | B |
| H | G | F | E | D | C | B | A |

由表3可先计算出16个格子的取法,每种取法用4个int型整数存储,将所有取法存入一个文件box4p.dat.

(1) 因为4个部分的结构相同,根据规则(2)和规则(4),将一个部分中的所有可能取法(共1 232种)存储在数组 $M$ 中.

(2) 根据规则(2),构造3个 $1232 \times 1232$ 的数组 $A,B,C$(其中 $A[i][j]$ 表示第一部分取法为 $M[i]$,第四部分取法为 $M[j]$,若满足规则(2)则为真值,否则为假;$B[i][j]$ 表示第一部分取法为 $M[i]$,第二部分取法为 $M[j]$,

若满足规则(2)则为真值,否则为假;$C[i][j]$表示第一部分取法为$M[i]$,第三部分取法为$M[j]$,若满足规则(2)则为真值,否则为假;而第四部分与第二、三部分的关系同第一部分与第二、三部分的关系).

(3) 遍历4个部分的所有取法的组合,将合法的组合存储起来(共465 977种).

从第3步存储的每种取法中找出合法的置换(即满足每行、每列只取一个数,并且每个数只取一次),可知共有244 744 192种正形置换.

寻找满足一定性质的4×4的S-盒:通过测试,从上面所得到的正形置换中发现,基于正形变换的所有S-盒的非线性度$N_S \leqslant 4$,差分均匀度$\delta \geqslant 2^{-2}$.我们通过软件测试的办法构造出了一大批达到这两个界并满足其他若干密码学性质的4×4的S-盒.

## 3 结束语

用数学方法构造出的S-盒一般从理论上可证明其具有一条或两条性质,但往往很难同时达到若干密码学性质,实用性也比较差.但对于比较小的S-盒,目前国际上比较流行的构造方法是从理论上先构造出一批具有主要密码学性质的候选对象,然后再通过软件测试方法找出满足要求的S-盒.本文就是这种方法的一种尝试.

## 参考文献

1 Feng Deng-guo, Pei Ding-yi. Guide to Cryptography. Beijing: Science Press, 1999
(冯登国,裴定一.密码学导引.北京:科学出版社,1999)

2 Liu Zhen-hua, Shu Chang. A method for constructing orthomorphic permutations of degree $2^m$. In: Pei Ding-yi, Zhao Ren-jie, Zhou Jin-jun eds. Advances in Chinacrypt'96. Beijing: Science Press, 1996. 56~59
(Liu Zhen-hua, Shu Chang. A method for constructing orthomorphic permutations of degree $2^m$. 见:裴定一,赵仁杰,周锦君编.密码学进展 Chinacrypt'96. 北京:科学出版社,1996. 56~59)

## Construction of S-Boxes with Some Cryptographic Properties

LIU Xiao-chen[1] FENG Deng-guo[2,3]

[1](*State Key Laboratory of Information Security Graduate School of University of Science and Technology of China Beijing* 100039)

[2](*State Key Laboratory of Information Security Institute of Software The Chinese Academy of Sciences Beijing* 100080)

[3](*Engineering Research Center for Information Security Technology The Chinese Academy of Sciences Beijing* 100080)

**Abstract** S-boxes are the only nonlinear component in many algorithms for encryption, intension of which decides the security strength of the whole algorithm. By theoretical analysis and statistics, there is evidence showing that large S-boxes have better cryptographic properties than small S-boxes. But they are harder to design and to implement. Now the 8×8 S-box is popular. In this paper, a method for constructing the 8×8 S-box and 8×6 S-box based on m-sequence is described. Through testing, some S-boxes with better nonlinearity and better difference uniformity are obtained. Finally some 4×4 S-boxes based on orthomorphic permutation are designed. These S-boxes offer the nonlinear resouce for further design of cryptographic algorithms.

**Key words** Blockcipher, S-Box, orthomorphic permutation.