文章编号:1000-582X(2008)01-0052-05

一种新的混沌分组密码算法

刘加伶[1],杨华千[2],廖晓峰[3]
(1. 重庆工学院 计算机科学与工程学院 重庆 400050;2. 重庆教育学院 计算机与现代教育系 重庆 400067
3. 重庆大学 计算机学院 重庆 400030)

摘　要:提出了一种新的分组密码算法,该算法把 128 比特的明文加密为 128 比特的密文。算法的密钥由 128 位的比特流 K 和 Logistic 映射的初值 x_0 两部分组成。整个加密过程包含了一个初始变换、8 个轮变换和最后的一个输出变换。每一轮使用一个 128 比特的轮密钥 $K^{(r)}$ 来加密上一轮的输入 $C^{(r-1)}$,并把输出反馈到下一轮的输入。所有的轮密钥都是由 128 位的比特流 K 和由 Logistic 映射产生的 128 比特随机二进制序列导出。理论与实验分析表明该算法克服了一些纯混沌密码系统的固有缺陷,具有较高的性能。

关键词:分组密码;混沌映射;代数模乘运算;置换
中图分类号:TP309.7　　文献标志码:A

New Block Cryptosystem Based on Chaotic System

LIU Jia-ling[1], *YANG Hua-qian*[2], *LIAO Xiao-feng*[3]
(1. College of Computer Science and Engineering, Chongqing Institute of Technology, Chongqing 400050, P. R. China;
2. Department of Computer and Modern Education Technology, Chongqing Education College, Chongqing 400067, P. R. China; 3. College of Computer Science, Chongqing University, Chongqing 400030, P. R. China)

Abstract: A new block cipher is proposed based on the study of some existing chaotic encryption algorithms. The proposed cipher encrypts 128-bit plaintext to 128-bit ciphertext blocks, using a 128-bit key K and the initial value x_n and the control parameter μ of logistic map. The block cipher process consists of an initial permutation and eight computationally identical rounds followed by an output transformation. Round r uses a 128-bit roundkey $K^{(r)}$ to transform a 128-bit input $C^{(r-1)}$, which is fed to the next round. The output after round 8 enters the output transformation to produce the final ciphertext. All roundkeys were derived from K and a 128-bit random binary sequence generated from a chaotic map. Analysis shows that the proposed block cipher does not suffer from the flaws of pure chaotic cryptosystems and possesses high security.

Key words: block cipher; chaotic map; modulo multiplication; permutation

大多数的基于混沌的软件加密技术都是使用混沌映射来产生伪随机序列[1-15]。然而,Wheeler 等人在[1-2]中指出当混沌系统用有限精度的计算机来实现的时候,数字化的混沌系统表现出了许多明显的不同的行为。它们的数字动力学行为也远不如于连续混沌系统的动力学行为。例如,非常短的周期;依赖于特定的数字精度等。假设采用定点运算,并且有限精度为 L 位(设为 2 进制),则混沌系统的性能将由于以下两个原因而降低:1)整个系统中,只有 2^L 有限个离散值来表示混沌轨道。因此,混沌序列的周期将小余等于 2^L。2)计算机的量化误差使得混沌轨道的性能也远不如理论值[3]。

Xun Yi 等人提出了另一种混沌密码系统[4]。在这个密码系统中,由混沌 Tent 映射产生的实值序

收稿日期:2007-10-15
基金项目:国家自然科学基金项目资助(60573047)
作者简介:刘加伶(1963-),女,副教授,硕士,主要研究方向:信息安全、数据库技术及应用。(E-mail)jiall@hotmail.com。

列通过一个域值函数来确定 $4n$ 比特的噪声向量。同时,也确定了一个 4 比特位和 1 ~ 4 的排列之间的一个查询表。然后,噪声向量和排列置换操作交替应用到 $4n$ 比特明文上以产生 $4n$ 比特的密文($n \geqslant 16$)。显然,该密码系统存在如下两个缺陷:1)查询表太小,只有 16 项(因为,4 比特位至多有 16 种取值)。2)v_{ji} 和排列 w_{ji} 之间的关系是固定不变的,与密钥无关。在选择明文攻击下,这两个缺陷有可能成为密码系统的安全漏洞。

结合混沌和代数群上的⊙运算, 文中构造了一种新的且更具安全性的混沌分组密码。在这个密码系统中,128 比特的明文用 128 比特的密钥 K、Logistic 映射的控制参数 u 和初值 x_0,经过 8 轮计算上相同的轮加密和一个输出变换加密成 128 比特的密文。在第 r 轮使用了 128 比特的轮密钥 $K^{(r)}$ 把一个 128 比特的输入 $C^{(r-1)}$ 变换成 128 比特的输出块作为下一轮的输入。第 8 轮的输出经过一个输出变换形成最后的密文。所有的轮密钥是由 128 比特密钥 K 和混沌映射产生的 128 比特的随机二进制序列导出的。

1　新的加密算法

具有良好性质的伪随机数序列在保密通信和密码学中有着广泛的应用,文献[5]提出了三种从混沌映射中产生独立同分布的二进制随机序列的方法,并且还讨论了这些随机序列的充分性及其统计特性。根据 Kohda 和 Tsuneda 在文献[5]中所述,Logistic 映射

$$\tau^{n+1}(x) = \mu\tau^{n}(x)(1-\tau^{n}(x)), x \in I = [0,1] \quad (1)$$

具有很多的与密码学相关的优良特性。这些属性在产生独立同分布的随机数序列方面具有重要意义。在本章,我们将采用如下的方法来获得随机变量序列[5]。一个实数 x 表示成如下的二进制形式:

$$x = 0.b_1(x)b_2(x)\cdots b_i(x)\cdots,$$
$$x \in [0,1], \quad b_i(x) \in \{0,1\}。 \quad (2)$$

在这个表示形式中,第 i 比特可以表示成:

$$b_i(x) = \sum_{r=1}^{2^{i-1}} (-1)^{r-1} \Theta_{(r/2^i)}(x), \quad (3)$$

此处,$\Theta_t(x)$ 是一个域值函数,其定义如下:

$$\Theta_t(x) = \begin{cases} 0, & x < t, \\ 1, & x \geqslant t \end{cases}。 \quad (4)$$

这样,我们就得到了一个独立同分布的二进制随机序列,$B_i^n = \{b_i(\tau^n(x))\}_{n=0}^{\infty}$。

在该加密算法中,一个 128 比特的明文块用一个 128 比特的密钥 K 和 Logistic 映射产生的双射函数 g,经过 8 次相似的轮加密,最后加密成一个 128 比特的密文块。在每一轮中采用了一个 128 比特的轮密钥 $K^{(r)}, r = 1,2,\cdots,8$。

1.1　密钥编排

文中加密算法,所有的轮密钥 $K^{(r)}, r = 1,2,\cdots,8$ 都是由密钥 K 和上面描述的方法产生的伪随机二进制序 PRN 列。下面的密钥扩展算法描述了轮密钥的生成过程。

Algorithm: keyschedule (K, PRN)

输入:128 比特密钥 $K = k_1\cdots k_{128}$;按 2.1 节方法产生的 128 比特的伪随机二进制序列 *PRN*
输出:8 个 128 比特的轮密钥 $K^{(r)}$。
步骤 1: for($r = 1; r \leqslant 8; r++$)

$$K^{(r)} = PRN \oplus (K >>> 16 \cdot (r-1)) \oplus 2^{r-1}$$

注:
1. ⊕ 表示按位异或
2. $>>>$ 表示 K 循环左移 $16 \cdot (r-1)$ 位;
3. 在⊕运算过程中,2^{r-1} 被扩展乘 128 比特参与运算

1.2　初始变换

文中的加密算法中,输入和输出可看成编号为 0 到 15 的 16 个字节,所以其长度是 128 比特。InitialPermu 算法重排 128 个比特的输入 P,得到 128 个比特的输出 P'。

Algorithm: initialPermu(P)

输入: 128 比特明文,$P = b_{0,0}, b_{0,1}, \cdots, b_{i,j}, \cdots. 0 \leqslant i \leqslant 15, 0 \leqslant j \leqslant 7.$
输出:128 比特经过初始排列的比特流 P'.
步骤 1: for $i \leftarrow 0$ *to* 7 {
$P_i \leftarrow$ the ith bit of all 16 bytes.
}
步骤 2: $P' \leftarrow P_0P_1P_2P_3P_4P_5P_6P_7$

该初始变换的好处在于:使得每轮加密都同时作用在 16 个字节的不同比特位上,从而增强该算法安全性。

显然,InitialPermu 变换是可逆的,并把这种可逆变换记为 Inv_ initialPermu。

1.3　替换变换

MessageSub 变换是一个非线性的字(16 比特)变换,独立地作用在每一个 16 比特的子块上。变换过程如下框图所示。

Algorithm: MessageSub($K^{(r)}, C^{(r-1)}$)

INPUT: 第 r 轮的轮密钥 $K^{(r)}$; 第 $(r-1)$ 轮的输出 $C^{(r-1)}$
OUTPUT: 128 比特的中间结果 $I^{(r)}$.
步骤 1: 把 $K^{(r)}$ 分成 8 个 16 比特的子块 $K_i^{(r)}, 1 \leqslant i \leqslant 8$; 把 $C^{(r-1)}$ 也分成 8 个 16 比特的子块 $C_i^{(r-1)}, 1 \leqslant i \leqslant 8$
步骤 2: *for* $i \leftarrow 1$ *to* 8 {
$a \leftarrow$ Bin2Int($K_i^{(r)}$); $b \leftarrow$ Bin2Int($C_i^{(r-1)}$);
if $a = 0$ then $a \leftarrow 2^{16}$;
if $b = 0$ then $b \leftarrow 2^{16}$;
$c_i \leftarrow a \odot b$
if $c_i = 256$ then $c_i \leftarrow 0$
$I_i^{(r)} \leftarrow$ Int2Bin(c_i)
}
步骤 3: $I^{(r)} \leftarrow (I_1^{(r)}I_2^{(r)}I_3^{(r)}I_4^{(r)}I_5^{(r)}I_6^{(r)}I_7^{(r)}I_8^{(r)})$
注意:
1. ⊙ 表示群 $Z_{2^{16}+1}^*$ 上的模 $2^{16}+1$ 乘运算.
2. Bin2Int(.) 把 16 比特的二进制变换成群 $Z_{2^{16}+1}^*$ 上的元素;Int2Bin(.)是其逆变换

显然,MessageSub 变换也是可逆的,并且其逆变

换也是非线性的字变换。其可逆性由群$(Z^*_{2^{16}+1},\odot)$的可逆性决定。即 $C_i^{(r-1)}\leftarrow(K_i^{(r)})^{-1}\odot I_i^{(r)}$,$(K_i^{(r)})^{-1}$是 $K_i^{(r)}$ 在群 $Z^*_{2^{16}+1}$上的逆元。我们记 MessageSub 的逆变换为 Inv_ MessageSub。

1.4 移位变换

在 MessageShift 的移位变换过程中,不同的子块,根据轮密钥的不同,循环左移不同的位数。其具体过程如下所示:

Algorithm: MessageShift $(K^{(r)},I^{(r)})$

输入:第 r 轮的轮密钥 $K^{(r)}$;第 r 轮 MessageSub 变换的输出 $I^{(r)}$.
输出:128 比特的中间结果 $T^{(r)}$.
步骤 1:把 $K^{(r)}$ 分成 8 个 16 比特的子块 $K_i^{(r)}$,$1\leqslant i\leqslant 8$;把 $I^{(r)}$ 分成 8 个 16 比特的子块 $I_i^{(r)}$,$1\leqslant i\leqslant 8$;
步骤 2: *for* $i\leftarrow 1$ *to* 8
do $T_i^{(r)}\leftarrow I_i^{(r)}<<<(\mathrm{Bin2Int}(K_i^{(r)}) \bmod 2^4)$
步骤 3: $T^{(r)}\leftarrow(T_1^{(r)}T_2^{(r)}T_3^{(r)}T_4^{(r)}T_5^{(r)}T_6^{(r)}T_7^{(r)}T_8^{(r)})$
注意:<<< 表示 $I_i^{(r)}$ 循环左移 $(\mathrm{Bin2Int}(K_i^{(r)}) \bmod 2^4)$位.

MessageShift 移位变换的可逆变换是循环右移,即 $I_i^{(r)}\leftarrow T_i^{(r)}>>>(\mathrm{Bin2Int}(K_i^{(r)}) \bmod 2^4)$。我们把这种可逆变换记为,Inv_ MessageShift 。

1.5 排列变换

在 MessagePermu 变换过程中,实质是对 128 比特的 8 个 16 比特子块进行重排列。其具体过程如下:

Algorithm: MessagePermu $(K^{(r)},T^{(r)})$

输入:第 r 轮的轮密钥 $K^{(r)}$;第 r 轮 MessageShift 变换的输出 $T^{(r)}$.
输出:128 比特的 第 r 轮 的输出密文 $C^{(r)}$.
步骤 1:把 $K^{(r)}$ 分成 8 个 16 比特的子块 $K_i^{(r)}$,$1\leqslant i\leqslant 16$;把 $T^{(r)}$ 分成 8 个 16 比特的子块 $T_j^{(r)}$,$1\leqslant j\leqslant 8$
步骤 2:$w_r=g(\bigoplus_{i=1}^{16}K_i^{(r)})$
步骤 3:shuffle $(T_1^{(r)}T_2^{(r)}T_3^{(r)}T_4^{(r)}T_5^{(r)}T_6^{(r)}T_7^{(r)}T_8^{(r)})$ by the sequence of permutation w_r in map g to get a new permutation $T_{i_1}^{(r)}T_{i_2}^{(r)}T_{i_3}^{(r)}T_{i_4}^{(r)}T_{i_5}^{(r)}T_{i_6}^{(r)}T_{i_7}^{(r)}T_{i_8}^{(r)}$.
步骤 4:$C^{(r)}\leftarrow(T_{i_1}^{(r)}T_{i_2}^{(r)}T_{i_3}^{(r)}T_{i_4}^{(r)}T_{i_5}^{(r)}T_{i_6}^{(r)}T_{i_7}^{(r)}T_{i_8}^{(r)})$.

例如,设 $\bigoplus_{i=1}^{16}K_i^{(r)}=01011001$,$w_r=g(01011001)=43582167$,下面的列表形式 $\begin{pmatrix}1&2&3&4&5&6&7&8\\4&3&5&8&2&1&6&7\end{pmatrix}$ 解释了 MessagePermu 变换的过程。所以,$T_1^{(r)}T_2^{(r)}T_3^{(r)}T_4^{(r)}T_5^{(r)}T_6^{(r)}T_7^{(r)}T_8^{(r)}$ 通过 MessagePermu 变换,变成了 $T_4^{(r)}T_3^{(r)}T_5^{(r)}T_8^{(r)}T_2^{(r)}T_1^{(r)}T_6^{(r)}T_7^{(r)}$。

显然,MessagePermu 变换是可逆的,其逆变换记做 Inv_ MessagePermu。

1.6 加密与解密过程

加密步骤如下(Encryption(K,P,x_0,μ)),其加密过程示意如图 1。

Algorithm: Encryption(K,P,x_0,μ)

输入:P;密钥 K、x_0 和控制参数 μ.
输出:128 比特的密文 C.
步骤 1:按照第二节的方法产生 128 比特的二进制随机序列 PRN
步骤 2:按照第二节的方法构造双射映射 g.
步骤 3:按照算法 keyschedule (K, PRN),得到 8 个轮密钥 $K^{(r)}$($1\leqslant r\leqslant 8$).
步骤 4:按照算法 initialPermu,重排明文 P。即 $P'\leftarrow$initialPermu(P)
步骤 5:把 P'赋给 $C^{(0)}$ *i. e.* $C^{(0)}\leftarrow P'$.
步骤 6: For round r from 1 to 8 do:
$I^{(r)}\leftarrow$MessageSub$(K^{(r)},C^{(r-1)})$.
$T^{(r)}\leftarrow$MessageShift$(K^{(r)},I^{(r)})$.
$C^{(r)}\leftarrow$MessagePermu$(K^{(r)},T^{(r)})$.
步骤 7:输出密文 C。*i. e.* $C\leftarrow$Inv_ initialPermu$(C^{(8)})$.

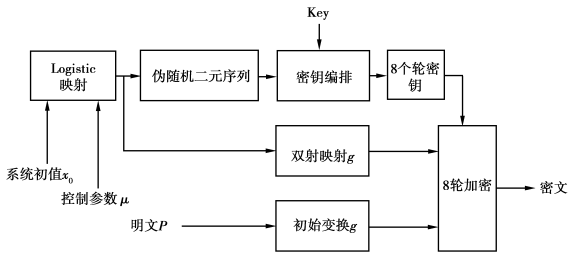


图 1 所提加密算法流程

在这个密码系统中,每一轮实际上只有一个非线性变换,即 MessageSub 变换。在解密过程中,每一轮的变换顺序是与加密过程相反的。其具体步骤如下:

Algorithm: Decryption(K,C,x_0,μ)

输入:密文 C;密钥 K、x_0 和控制参数 μ.
输出:128 比特的明文 P.
步骤 1:按照第二节的方法产生 128 比特的二进制随机序列 PRN
步骤 2:按照第二节的方法构造双射映射 g.
步骤 3:按照算法 keyschedule (K, PRN),得到 8 个轮密钥 $K^{(r)}$($1\leqslant r\leqslant 8$).
步骤 4:$C'\leftarrow$initialPermu(C)
步骤 5:把 C'赋给 $C^{(8)}$。*i. e.* $C^{(8)}\leftarrow C'$.
步骤 6: For round r from 8 to 1 do:
$T^{(r)}\leftarrow$Inv_MessagePermu$(K^{(r)},C^{(r)})$.
$I^{(r)}\leftarrow$Inv_MessageShift$(K^{(r)},T^{(r)})$
$C^{(r-1)}\leftarrow$Inv_MessageSub$(K^{(r)},I^{(r)})$.
步骤 7:输出明文 P。*i. e.* $P\leftarrow$Inv_ initialPermu$(C^{(0)})$.

2 仿真实验结果

在一个密码系统中,安全性是首要的问题。下面我们将从理论和仿真实验方面来阐述文中提出的加密算法的安全性。

在实验中,为了评估算法的性能,我们采用了一个 3 200 字节的文本文件和一个 256×256 像素的灰度图像文件。

设 $x_0=0.436\ 567\ 349\ 535\ 648$,$\mu=3.999\ 999\ 96$,$K=$"abcdefghijklmnop"。为了避免瞬态效应,忽略 Logistic 映射开始迭代的 250 次。

图 2 表明,该算法能够正确地加/解密文件。注

意,当用文中的算法来加密文本文件时,在密文中可能存在一些不可打印的字符。

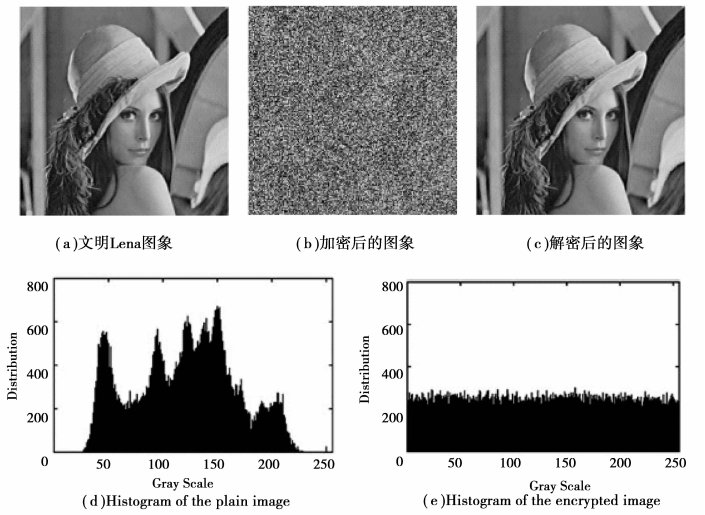


图 2　用明文 Lena 图象加密结果.

3　安全性与性能分析

3.1　密钥空间

文中的密码算法,Logistic 映射在迭代过程中采用了 IEEE754 浮点数标准[6]。设 $x_0 = 0.x_1x_2\cdots x_{15}$,又因为任意的 128 比特都可以作为密钥 K。所以,算法的密钥空间约为$(10^{15}) \times 2^{128} \approx 2^{177.83}$。

如果密码分析人员采用蛮力攻击,他们不需要知道 Logistic 映射的细节,譬如初始值 x_0 和控制参数 μ。但他们必须知道密钥 K 和双射映射 $g: r \rightarrow w_r$。根据前面第二节双射映射的构造知道,此时的密钥空间大约是 $8! \cdot (8!-1)\cdots(8!-255)\cdot 2^{128}$,这对于当今的计算能力来说,是一个非常大的数了。

3.2　排列分析

正如在[1]中所述,PRN 是独立同分布的。除非知道混沌系统的初始值 x_0 和控制参数 μ,否则很难从 PRN 序列的前面位来预测下一位。同时,只有混沌实值轨道 $\tau^n(x)$ 的部分位参与了构造 *PRN* 序列。因此,在进行密码分析时,对 x_0 的猜测和加密系统的重构变成了不可能。

排列几乎是所有的传统密码系统的基本操作。在许多的密码系统中,排列只是根据设计者预先定义的方式重新排列输入元素,是与密钥无关的。在实际的密码分析过程中,由于这种排列很容易被差分分析攻破,所以它对算法安全性几乎没有什么意义。然而,在文中提出的加密算法中,排列是与密钥相关的,不同的消息块有不同的排列方式。从而增加了密码分析的难度。

3.3　统计测试

根据 Shannon 理论,一个密码系统在抗统计攻击方面应该具有很好的性质。下面的实验表明本章的密码系统保留了这个好的特性。实验结果如图 3(第四节的文本加密前后的统计直方图)。我们发现密文的直方图分布已经相当均匀了,并且完全不同于明文的直方图分布。

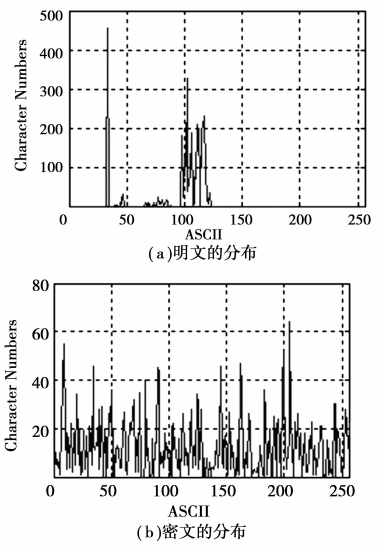


图 3　明文和密文分布

文本文件通常是由可打印字符组成的,其 ASCII 码一般在 033 ~ 126 之间。然而,经过文中的算法加密后,其密文的 ASCII 码分布在整个 0 ~ 255 之间(如图 3)。

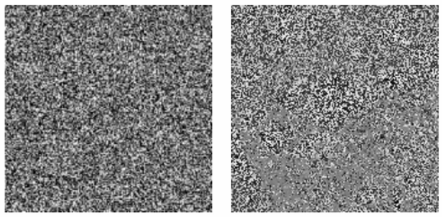
3.4　密钥敏感性测试

由于本章的加密算法的密钥是由 x_0,K 两部分组成的,所以我们将从两个方面来进行密钥敏感性测试。

Ⅰ、保持 x_0 不变,改变密钥 K 的最后一位。修改后的密钥 K' = "abcdefghijklmnoq"。然后用密钥 K' 和 x_0 解密图 2(b),实验结果如图 4(a)。

Ⅱ、保持 K 不变,改变 x_0 的最后一位。修改后的密钥 $x'_0 = 0.436\ 567\ 349\ 535\ 649$。然后用密钥 K 和 x'_0 解密图 2(b),实验结果如图 4(b)。

实验结果表明,尽管密钥只有微小的差异也导致了解密的失败。因此,这个新的加密算法仍然保



(a)用μ，x_0和K'解密图象 (b)用μ，x'_0和K'解密图象

图4 密钥敏感性测试

持了密钥敏感性。同时,我们在实验中也发现,两个只有2^{-15}微小差异的初值x_0和x'_0,按照构造的双射映射也几乎完全不同。

4 结 论

文中提出了一种新的基于混沌映射和代数群上运算的密码系统。在这个新的密码系统中,每个128比特的明文块产生一个同样长度的密文块,同时密文也依赖于明文、密钥、混沌映射和群上的运算。该算法弥补了纯混沌密码系统的一些缺陷。另外,大的密钥空间、比特位的替换与移位和基于密钥的子块排列变换都大大增强了算法的各种抗攻击能力。当然,该算法在某些性能方面还不能和AES相比,但它提供了一种结合混沌映射和代数群运算来构造密码系统的新思路。对于该算法,我们接下来的工作将是进一步对它进行各种密码分析找出其可能存在的某些缺陷,同时进一步改善其加/解密速度。

参考文献:

[1] WHEELER D D. Problems with chaotic cryptosystems [J]. Cryptologia 1989, 8(3): 243-250.

[2] WHEELER D D, MATHEWS R A J. Supercomputer investigations of a chaotic encryption algorithm [J]. Cryptologia 1991, 15(2): 140-152.

[3] WEI J, LIAO X F, WONG K W, et al. A new chaotic cryptosystem [J]. Chaos, Solitons and Fractals, 2006, 30: 1143-1152.

[4] XUN Y, CHIK H T, and CHEE K S. A new block cipher based on chaotic tent maps [J]. IEEE Trans. Circuits and Systems, 2002; 49(12): 1826-1829.

[5] KOHDA T, TSUNEDA A. Statistics of chaotic binary sequences [J]. IEEE Transactions on Information Theory, 1997, 43: 104.

[6] GOLDBERG D, PRIEST D. What every computer scientist should know about floating-point arithmetic [J]. ACM Comp. Surv, 1991, 23(1): 5-48.

[7] YANG H. A new block cipher based on chaotic map and group theory [J]. Chaos, Solitons & Fractals 2007, (10):1016.

[8] TANG G, LIAO X F. A method for designing dynamical S-boxes based on discretized chaotic map [J]. Chaos, Solitons & Fractals 2005,23, 1901-1909.

[9] JAKIMOSKI G, KOCAREV L. Chaos and cryptography: block encryption ciphers based on chaotic maps [J]. IEEE Trans. Circuits Syst I 2001, 48(2): 163-169.

[10] STOJANOVSKI T, KOCAREV L. Chaos-based random number generators—part I: analysis [J]. IEEE Trans. Circuits Syst I 2001, 48(3): 281-288.

[11] STOJANOVSKI T, KOCAREV L. Chaos-based random number generators—part II: practical realization [J]. IEEE Trans. Circuits Syst I 2001, 48(3): 382-385.

[12] LI S J, LI Q, LI W, et al. Statistical properties of digital piecewise linear chaotic maps and their roles in cryptography and pseudo-random coding [J]. Lecture Notes in Computer Science, 2001.

[13] LI S, MOU X Q and CAI Y L. Pseudo-random bit generator based on couple chaotic systems and its applications in stream-cipher cryptography [J]. Lecture Notes in Computer Science, 2001, 2247: 316-329.

[14] FRIDRICH J, Symmetric ciphers based on two-dimensional chaotic maps [J]. Int. J. Bifurcat Chaos 1998, 8(6):1259-84.

[15] KNUTH D E. The Art of Computer Programming [M]. 3rd. MA: Addison Wesley, 1998.

(编辑 吕建斌)

(上接第51页)

参考文献:

[1] 薛睿峰,钟顺时. 微带天线圆极化技术概述与进展[J]. 电波科学学报,2002,17(4):331-336.
XUE RUI-FENG,ZHONG SHUN-SHI. Survey and progress in circular polarization technology of microstri Pantennas [J]. Chinese Journal of Radio Science,2002, 17(4):331-336.

[2] 韩庆文,易念学,李忠诚,等. 圆极化微带天线的设计与实现[J]. 重庆大学学报:自然科学版,2004:27(4):57-60.
HAN QING-WEN, YI NIAN-XUE, LI ZHONG-CHENG, et al. Design and realization of circular polarization microstrip antenna [J]. Journal of Chongqing University: Natural Science Edition,2004,27(4):57-60.

[3] 鲍尔 I J,布哈蒂亚 P. 微带天线[M]. 梁联倬,寇廷耀,译. 北京:电子工业出版社,1984.

[4] 张 钧,刘克诚. 微带天线理论与工程[M]. 北京:国防工业出版社, 1988.

[5] 林昌禄,聂在平. 天线工程手册[M]. 北京:电子工业出版社,2002.

[6] DAVID M POZAR. Microwave engineering [M]. 3rd ed. Beijing:Electronic Industry Press,2005.

(编辑 张 苹)