Advances in Applied Mathematics 应用数学进展, 2020, 9(2), 257-262
Published Online February 2020 in Hans. http://www.hanspub.org/journal/aam
https://doi.org/10.12677/aam.2020.92030

Hans汉斯

Design of Digital Image Encryption Algorithm Based on Latin Square

Yanhua Tang, Chuanjun Tian

College of Electronics and Information Engineering, Shenzhen University, Shenzhen Guangdong
Email: tiancj@sina.com.cn

Received: Feb. 5th, 2020; accepted: Feb. 20th, 2020; published: Feb. 27th, 2020

Abstract

Firstly, a basic cryptosystem is designed by using a 16 order Latin square, and formulas of the nonlinear reversible transformations of encryption and decryption are presented. Secondly, a new stream cipher algorithm is proposed by using this basic cryptosystem and the existing Logistic chaotic system. Finally, this stream cipher algorithm is applied to encrypt digit image, and the image encryption and decryption are simulated. The simulation shows that this algorithm has good encryption and decryption effect.

Keywords

Stream Cipher Algorithm, Digital Image Encryption Algorithm, 16 Order Latin Square, Formula of Nonlinear Reversible Transformation

基于拉丁方的数字图像加密算法设计

汤艳华，田传俊

深圳大学电子与信息工程学院，广东 深圳
Email: tiancj@sina.com.cn

收稿日期：2020年2月5日；录用日期：2020年2月20日；发布日期：2020年2月27日

摘 要

首先利用一个16阶拉丁方设计了一种基本密码系统，给出了它所决定非线性可逆变换和加解密变换的代数计算公式。然后，综合利用这种基本密码系统和现有的Logistic混沌系统，提出了一种新的流密码算法。最后，将这种流密码算法应用在图像加密中，并对加解密效果进行了仿真，仿真效果说明该算法具有良好的加解密效果。

文章引用: 汤艳华, 田传俊. 基于拉丁方的数字图像加密算法设计[J]. 应用数学进展, 2020, 9(2): 257-262.
DOI: 10.12677/aam.2020.92030

关键词

流密码算法，数字图像加密算法，16阶拉丁方，可逆变换代数式

1. 引言

流密码算法是当前密码学理论的关键研究问题之一，在数字信息的加密中有重要作用。普遍认为，完善保密系统是流密码算法的理论基础[1]。自从 Shannon 提出了完善保密系统模型以来[2]，已有不少文献都对相关的理论模型及其实际应用进行了研究，但还是有不少相关问题值得进一步研究。例如，当前的常见流密码算法都是基于模 2 加法运算来设计的，但模 2 加法运算过于简单，因而会影响到流密码算法设计的应用效果。最近的文献[3]建立了一种新的更广泛完善保密系统模型。该理论模型将流密码系统设计细分为两个阶段：基本密码系统与应用密码系统。当以新旧两种完善保密系统模型作为理论基础时，尽管两种模型在应用密码系统中关键的密钥流序列的设计上的要求形式上差别不算太大，但是，它们在基本密码系统的设计上却有非常明显的区别：旧模型所能设计的基本密码系统的类型是很少的，而新模型所能设计的基本密码系统的类型与技巧都非常丰富。因此，本文将提出一种基本密码系统的新设计方法，进而再构造一种新的流密码算法。

由文献[3]可知，现有常见流密码算法的基本密码系统是利用模 2 加法设计的，这等价于利用单一的 2 阶拉丁方所设计的基本密码系统。文献[3]的主要结果将 2 阶拉丁方推广为更一般的任意阶拉丁方组来设计基本密码系统了。这样，一些常见流密码算法如 A5 和 RC4 等所用的基于 2 阶拉丁方或模 2 加法运算的设计方法都是在新模型下所能提出的设计方法的特殊情形。除了 2 阶拉丁方和文献[4]所用的 4 阶拉丁方来设计基本密码系统之外，当前很少有文献讨论利用高于 4 阶拉丁方来设计基本密码系统。因此，本文将研究基于 16 阶拉丁方设计基本密码系统的新方法，并结合基于常见的 Logistic 离散混沌系统来设计一种新的流密码算法。

2. 一些基本概念

参照现有密码学文献，流密码算法需要对基本明文单元依次进行加解密变换，相关步骤如下：1) 设明文单元序列为 $m=m_0m_1m_2\cdots$；2) 设任一密钥为 k，利用一个以 k 为参数的密钥流产生器来产生一个密钥流序列 $z=k_0k_1k_2\cdots$；3) 设利用加密变换 E 依次加密得到的密文为 $c=c_0c_1c_2\cdots$，其中，$c_j=E\left(k_j,m_j\right)$，对任意 $j=0,1,2,\cdots$；4) 最后利用解密变换 D 将 c 依次解密可恢复出明文单元序列 $m=D\left(k_0,c_0\right)D\left(k_1,c_1\right)D\left(k_2,c_2\right)\cdots$。

当前，常见的明文单元 $m_j\in Z_2=\{0,1\}$ 是 1 比特，加密变换 E 为模 2 加法：$E\left(k,m\right)=m\oplus k$ 等。参照文献[3]，在 Z_2 上的可逆变换只有恒等变换与取反变换，因而所能设计的基本密码系统会很简单。为了设计更复杂的基本密码系统，先介绍如下一些概念[5]。

设 n 阶方阵 $A=\left(a_{ij}\right)_{n\times n}$ 和 $B=\left(b_{ij}\right)_{n\times n}$ 满足 $a_{ij},b_{ij}\in\{0,1,\cdots,n-1\}$，并记

$$\left(A,B\right)=\left(\left(a_{ij}\right),\left(b_{ij}\right)\right)=\begin{bmatrix}\left(a_{11},b_{11}\right) & \left(a_{12},b_{12}\right) & \cdots & \left(a_{1n},b_{1n}\right)\\ \left(a_{21},b_{21}\right) & \left(a_{22},b_{22}\right) & \cdots & \left(a_{2n},b_{2n}\right)\\ \vdots & \vdots & \ddots & \vdots\\ \left(a_{n1},b_{n1}\right) & \left(a_{n2},b_{n2}\right) & \cdots & \left(a_{nn},b_{nn}\right)\end{bmatrix}. \tag{2-1}$$

定义 2.1. 设 $n\in\{2,3,4,\cdots\}$。如果 $Z_n=\{0,1,\cdots,n-1\}$ 上所有不同的数字在 n 阶方阵 A 的每行和每列中都出现，则称 A 为 n 阶拉丁方。

定义 2.2. 如果 A 和 B 都是由 $0,1,2,\cdots,n-1$ 构成的 n 阶方阵，且 (A,B) 的 n^2 个元素组成的集合等于 $\{(i,j)\,|\,i,j=0,1,\cdots,n-1\}=Z_n^2$，则称 A 和 B 是正交的。特别地，当 $k(\geq 2)$ 个拉丁方 $A_1,A_2,\cdots,A_k$ 两两正交时，称 $A_1,A_2,\cdots,A_k$ 为正交拉丁方组。

例如，如下的 4 阶矩阵都是拉丁方，且 A,B,C 是两两正交的。

$$A=\begin{bmatrix}1&0&2&3\\0&1&3&2\\3&2&0&1\\2&3&1&0\end{bmatrix},\ B=\begin{bmatrix}0&3&1&2\\1&2&0&3\\3&0&2&1\\2&1&3&0\end{bmatrix},\ C=\begin{bmatrix}3&1&0&2\\2&0&1&3\\0&2&3&1\\1&3&2&0\end{bmatrix}. \tag{2-2}$$

文献[3] [4]研究了 4 阶拉丁方组来设计基本密码系统。本文将考虑利用如下更高的 16 阶拉丁方来设计基本密码系统。

$$L=\begin{bmatrix}1&0&2&3&5&4&6&7&13&12&14&15&9&8&10&11\\0&1&3&2&4&5&7&6&12&13&15&14&8&9&11&10\\3&2&0&1&7&6&4&5&15&14&12&13&11&10&8&9\\2&3&1&0&6&7&5&4&14&15&13&12&10&11&9&8\\5&4&6&7&1&0&2&3&9&8&10&11&13&12&14&15\\4&5&7&6&0&1&3&2&8&9&11&10&12&13&15&14\\7&6&4&5&3&2&0&2&11&10&8&9&15&14&12&13\\6&7&5&4&2&3&1&0&10&11&9&8&14&15&13&12\\9&8&10&11&13&12&14&15&1&0&2&3&5&4&6&7\\8&9&11&10&12&13&15&14&0&1&3&2&4&5&7&6\\11&10&8&9&15&14&12&13&3&2&0&1&7&6&4&5\\10&11&9&8&14&15&13&12&2&3&1&0&6&7&5&4\\13&12&14&15&9&8&10&11&5&4&6&7&1&0&3&2\\12&13&15&14&8&9&11&10&4&5&7&6&0&1&3&2\\15&14&12&13&11&10&8&9&7&6&4&5&3&2&0&1\\14&15&13&12&10&11&9&8&6&7&5&4&2&3&1&0\end{bmatrix}=\begin{bmatrix}T_0\\T_1\\T_2\\T_3\\T_4\\T_5\\T_6\\T_7\\T_8\\T_9\\T_{10}\\T_{11}\\T_{12}\\T_{13}\\T_{14}\\T_{15}\end{bmatrix}. \tag{2-3}$$

其中，$T_0:Z_{16}\leftrightarrow Z_{16}$ 表示如下可逆变换：$T_0(0)=1$，$T_0(1)=0$，$T_0(2)=2$，…，$T_0(15)=11$ 等。显然，Z_{16} 与 Z_2^4 是一一对应的，因而可将 Z_{16} 与 Z_2^4 中相互对应的数不加区别[3] [4]。因此，基于 L 设计的基本密码系统 (M,C,T) 满足 $M=C=Z_2^4=Z_{16}$ 和 $T=\{T_0,T_1,\cdots,T_{15}\}$。这样，下面将所设计的流密码系统的具体明文单元设为 $m_j\in M$，对任意 $j=0,1,2,\cdots$，等等。

3. 一种新的流密码算法设计

由文献[6] [7] [8]可知，流密码算法的设计是当前密码算法研究的一个重要问题。下面将设计一种新的流密码算法。参照文献[3]，流密码系统可分为基本密码系统和应用密码系统，其中，应用密码系统设计的关键是密钥序列空间的设计。上面已设计出一个理论基本密码系统 (M,C,T)。为了方便实际应用，还要将理论加密变换 T 和解密变换 T^{-1} 转化为实际密钥空间 K 与加密变换 E 和解密变换 D 来实现。下面先讨论与 (M,C,T) 相应的实际基本密码系统 (M,C,K,E,D) 的设计问题。

1) 实际基本密码系统 (M,C,K,E,D) 的设计

将 $T=\{T_0,T_1,\cdots,T_{15}\}$ 中每个可逆变换利用简单运算表示如下：对任意 $m=m_1m_2m_3m_4\in Z_2^4$，

$$T_0(m)=\left(m_3\times m_3m_4-m_4-m_3\times\overline{m_4}+1\right)\bmod 4+4\times\overline{m_1}\times m_2+12\times m_1\times\overline{m_2}+8\times m_1\times m_2$$

$$T_1(m)=\left(m_4\times m_3m_4-m_3\right)\bmod 4+4\times\overline{m_1}\times m_2+12\times m_1\times\overline{m_2}+8\times m_1\times m_2$$

$$T_2(m)=\left(m_3\times m_3m_4-m_4-m_3\times\overline{m_4}+3\right)\bmod 4+4\times\overline{m_1}\times m_2+12\times m_1\times\overline{m_2}+8\times m_1\times m_2$$

其中，m_i 是二进制数，m_im_j 表示十进制数，$\overline{m_i}=1-m_i$，$i,j=0,1,2,3$。类似地，可将 $T_3,T_4,\cdots,T_{15}$ 的代数计算公式一一写出来，在此省略。这样，可将基本实际密钥空间设计为 $K=Z_{16}$：$k\leftrightarrow T_k$，对任意 $k\in K=Z_2^4$。因此，基本加密函数 E 的计算公式为 $c=E(k,m)=T_k(m)$。

更进一步，基本加密函数和解密函数的具体设计方法如下：

a) 基本加密函数 E：对任一 2 比特明文 $m=m_1m_2\in Z_4$ 和 4 比特密钥 $k=k_1k_2k_3k_4\in Z_{2^4}$，其中，$m_1,m_2,k_1,k_2,k_3,k_4\in Z_2$，可利用如下统一代数式将加密变换 $c=E(k,m)$ 设计为

$$\begin{aligned}c=&\tilde{k}_0\times T_0(m)+\tilde{k}_1\times T_1(m)+\tilde{k}_2\times T_2(m)+\tilde{k}_3\times T_3(m)+\tilde{k}_4\times T_4(m)+\tilde{k}_5\times T_5(m)\\&+\tilde{k}_6\times T_6(m)+\tilde{k}_7\times T_7(m)+\tilde{k}_8\times T_8(m)+\tilde{k}_9\times T_9(m)+\tilde{k}_{10}\times T_{10}(m)\\&+\tilde{k}_{11}\times T_{11}(m)+\tilde{k}_{12}\times T_{12}(m)+\tilde{k}_{13}\times T_{13}(m)+\tilde{k}_{14}\times T_{14}(m)+\tilde{k}_{15}\times T_{15}(m)\end{aligned}$$

其中，$\tilde{k}_0=\overline{k}_1\wedge\overline{k}_2\wedge\overline{k}_3\wedge\overline{k}_4$，且

$$\tilde{k}_1=\overline{k}_1\wedge\overline{k}_2\wedge\overline{k}_3\wedge k_4,\quad \tilde{k}_2=\overline{k}_1\wedge\overline{k}_2\wedge k_3\wedge\overline{k}_4,\quad \tilde{k}_3=\overline{k}_1\wedge\overline{k}_2\wedge k_3\wedge k_4,$$

$$\tilde{k}_4=\overline{k}_1\wedge k_2\wedge\overline{k}_3\wedge\overline{k}_4,\quad \tilde{k}_5=\overline{k}_1\wedge k_2\wedge\overline{k}_3\wedge k_4,\quad \tilde{k}_6=\overline{k}_1\wedge k_2\wedge k_3\wedge\overline{k}_4,$$

$$\tilde{k}_7=\overline{k}_1\wedge k_2\wedge k_3\wedge k_4,\quad \tilde{k}_8=k_1\wedge\overline{k}_2\wedge\overline{k}_3\wedge\overline{k}_4,\quad \tilde{k}_9=k_1\wedge\overline{k}_2\wedge\overline{k}_3\wedge k_4,$$

$$\tilde{k}_{10}=k_1\wedge\overline{k}_2\wedge k_3\wedge\overline{k}_4,\quad \tilde{k}_{11}=k_1\wedge\overline{k}_2\wedge k_3\wedge k_4,\quad \tilde{k}_{12}=k_1\wedge k_2\wedge\overline{k}_3\wedge\overline{k}_4,$$

$$\tilde{k}_{13}=k_1\wedge k_2\wedge\overline{k}_3\wedge k_4,\quad \tilde{k}_{14}=k_1\wedge k_2\wedge k_3\wedge\overline{k}_4,\quad \tilde{k}_{15}=k_1\wedge k_2\wedge k_3\wedge k_4,$$

b) 基本解密函数 D：对任一 2 比特密文 $c=c_1c_2\in Z_4$ 和 4 比特密钥 $k=k_1k_2k_3k_4\in Z_{2^4}$，其中，$c_1,c_2,k_1,k_2,k_3,k_4\in Z_2$，可将解密变换 $m=D(k,c)$ 设计为

$$\begin{aligned}m=&\tilde{k}_0\times T_0(c)+\tilde{k}_1\times T_1(c)+\tilde{k}_3\times T_2(c)+\tilde{k}_2\times T_3(c)+\tilde{k}_4\times T_4(c)+\tilde{k}_5\times T_5(c)\\&+\tilde{k}_7\times T_6(c)+\tilde{k}_6\times T_7(c)+\tilde{k}_8\times T_8(c)+\tilde{k}_9\times T_9(c)+\tilde{k}_{11}\times T_{10}(c)\\&+\tilde{k}_{10}\times T_{11}(c)+\tilde{k}_{12}\times T_{12}(c)+\tilde{k}_{13}\times T_{13}(c)+\tilde{k}_{15}\times T_{14}(c)+\tilde{k}_{14}\times T_{15}(c)\end{aligned}$$

至此就完成了实际基本密码系统 (M,C,K,E,D) 的设计。

2) 应用密码系统中密钥流序列的设计

上面已设计出实际基本密码系统 (M,C,K,E,D)，还需要设计应用密钥空间才构成一个完整的流密码算法。下面再来讨论应用系统中密钥序列空间的设计问题，将利用现有的 Logistic 混沌系统所决定的一个密钥流发生器来对密钥流序列空间进行设计。该混沌系统表达式如下：

$$x_{m+1}=\mu x_m(1-x_m),$$

其中 $x_m\in[0,1]$，对任意 $m=0,1,2,\cdots$，且 $\mu\in[0,4]$。当 $\mu\in[3.571448,4]$ 时，系统会处于混沌的状态。利用该混沌系统是容易产生 2 元密钥流序列的。更进一步，为了能与基本密码系统配合使用，还需要将所产生 2 元密钥序列变成 16 元密钥序列，以便下面使用。

当基本密码系统和密钥流序列组成密钥空间设计好后就能综合构造出一种流密码算法了，可将它用于数字信息的加解密变换之中，并可利用 Matlab 仿真来实现数字图像信息的加解密变换。

下面就给出一种新的流密码算法设计步骤：

a) 选择一幅数字灰度图像作为明文，在 Matlab 软件中，该明文可表示成一个矩阵 $I=\left(m_{ij}\right)_{256\times 256}$，其中，$m_{ij}\in Z_{256}$，对任意 $i,j=0,1,\cdots,255$；

b) 将图像矩阵 I 中每个 8 比特像素转换为比特序列而得到 2 元明文序列 $m=m_1m_2m_3\cdots$。之后，将明文序列依次转化为 16 元明文单元序列 $m=\tilde{m}_1\tilde{m}_2\cdots$，其中，$\tilde{m}_1=m_1m_2m_3m_4$，$\tilde{m}_2=m_5m_6m_7m_8\in Z_{16}$，等等；

c) 利用 Logistic 混沌系统迭代产生 16 元密钥流序列 $z=\tilde{k}_1\tilde{k}_2\cdots$，其中 $\tilde{k}_1=k_1k_2k_3k_4\in Z_{16}$，等等；

d) 加密变换：依次对明文单元加密 $\tilde{c}_j=E\left(\tilde{k}_j,\tilde{m}_j\right)$，对任一 $j=1,2,\cdots$，可得到 16 元密文序列 $c=\tilde{c}_1\tilde{c}_2\cdots$。若有必要，并可将 c 变换为 2 元密文序列，以便得到加密后的灰度图像；

e) 解密变换：依次对密文序列解密 $\tilde{m}_j=D\left(\tilde{k}_j,\tilde{c}_j\right)$，对任一 $j=1,2,\cdots$，可得到 16 元明文序列 $m=\tilde{m}_1\tilde{m}_2\cdots$。然后将 m 可还原为原数字图像矩阵 $I=\left(m_{ij}\right)_{256\times 256}$。

按照上述步骤，在 Matlab 中仿真的加解密效果图可见图 1，其中，对照的流密码算法为利用模 2 加法和 Logistic 系统所设计的密码算法。

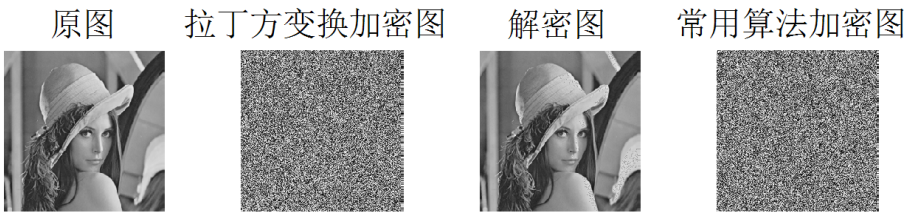

Figure 1. Simulation effect of algorithm
图 1. 算法仿真效果图

更进一步，对两种算法加密图像之后的图像信息的相关性进行数值计算的结果可见表 1。

Table 1. Simulation data of correlation
表 1. 相关性仿真数据

方向	原图	本算法	**常见密码算法**
水平	0.9357	0.0026	0.0076
竖直	0.9682	0.00016	0.0012
对角	0.9084	0.0053	0.0039

由该表可知，明文在各个方向的相关系数都接近于 1，经过加密后的图像的相关系数在所有的方向都接近 0，说明像素间的相关性已被加密变换打乱了。与常见流密码算法相比，新算法在水平，竖直方向更好，在对角方向的相关系数和常用密码算法有微小差异。

下面再进行信息熵分析，可分析出加密后的图像像素值之间分散的均匀程度。根据图像信息熵的定义式(3-1)和最大熵原理知，由于本文所选取的Lena图像的灰度取值范围是[0, 255]，因而图中各像素值的概率出现的最大信息熵为8。数值越接近8就说明图像像素之间分散得越均匀。

$$H\left(X\right)=-\sum_{i=0}^{255}P\left(x_i\right)\log P\left(x_i\right). \tag{3-1}$$

通过仿真计算，可得到原始图像信息熵为7.4442，利用新流密码算法加密图像后的信息熵为7.9974，利用常用密码系统加密图像后的信息熵为7.9972，两种密文图的熵都比明文图的熵更接近最大理想值8。这说明利用高阶拉丁方设计的流密码系统具有一定的参考价值。

4.小结

本文研究了基于16阶拉丁方的基本密码系统的设计问题，并结合常见的密钥流产生器提出了一种流密码算法。经过仿真分析可知，该算法的加密效果良好，为后续研究更高阶的拉丁方变换矩阵打下了一定的基础。

参考文献

[1] 张斌, 徐超, 冯登国. 流密码的设计与分析: 回顾, 现状与展望. 密码学报, 2016, 3(6): 527-545.

[2] Shannon, C.E. (1949) Communication Theory of Secrecy System. *Bell System Technical Journal*, **28**, 656-715.

[3] 田传俊. 密钥非均匀分布的完善保密通信系统[J]. 通信学报, 2018, 39(11): 1-9.

[4] 田传俊. 基于4阶正交拉丁方组实际基本密码系统设计[J]. 深圳大学学报, 2020, 待发表.

[5] 李超. 用线性取余变换造正交拉丁方和幻方[J]. 应用数学学报, 1996, 19(2): 231-238.

[6] 丁存生, 肖国镇. 流密码学及其应用[M]. 北京: 国防工业出版社, 1994.

[7] 杨刘洋, 吕翔. 一种基于完备拉丁方的图像加密算法[J]. 计算机应用研究, 2015, 32(11): 3435-3442.

[8] Yin, Q. and Wang, C. (2018) A New Chaotic Image Encryption Scheme Using Breadth-First Search and Dynamic Diffusion. *International Journal of Bifurcation and Chaos*, **28**, Article ID: 1850047.