第 10卷 第 2期 集美大学学报 (自然科学版) Vol 10 No. 2
2005年 6月 Journal of Jimei University(Natural Science) Jun 2005

[文章编号] 1007 - 7405(2005)02 - 0125 - 09

一种新的基于神经网络混沌吸引子的公钥密码算法

刘年生[1], 郭东辉[2]

(1. 集美大学计算机工程学院, 福建 厦门 361021; 2. 厦门大学物理与机电工程学院, 福建 厦门 361005)

[摘要] 论述了一种新的基于神经网络混沌吸引子的公钥密码算法, 在过饱和贮存的 Hopfield神经网络模型中混沌吸引子与初始状态之间存在一种单向函数关系, 改变神经网络的联结权矩阵, 混沌吸引子及其相应的吸引域会随之发生改变, 如果以联结权矩阵为陷门, 并利用可交换的随机变换矩阵来改变神经网络的联结权矩阵, 则可以创建一种新的 Diffie-Hellman公钥加密算法. 将随机变换矩阵作为私钥, 而将变换后的神经网络联结突触矩阵作为公钥, 介绍了这种新的公钥加密方案, 并分析和讨论其安全性和加密效率.

[关键词] 神经网络; 公钥密码体制; 混沌吸引子; 矩阵分解

[中图分类号] TP 309. 02 [文献标识码] A

0 引言

网络安全问题是下一代互联网研究的关键问题之一[1], 而加密算法又是网络安全问题的核心. 为了满足下一代互联网多媒体实时性安全通信的要求, 既需要选用复杂性高的加密算法, 以增强信息的安全性, 又希望所选用的加密算法能以并行方式实现快速运算, 以缩短加密和解密的时间来保证实时通信. 其中, 既能实现快速并行运算又有混沌动力学复杂行为的人工神经网络一直被认为是用来设计下一代互联网通信所需的加密算法的最佳选择之一[2].

自 1976年 W. Diffie和 M. Hellman首次提出公钥密码思想以来[3], 公钥密码体制因其不需要安全信道来进行密钥的分配与传送, 并且在多用户保密通信时可有效地减少密钥数量, 方便和简化了密钥管理工作. 因此, 它倍受计算机网络安全通信的重视, 提出了许多公钥密码算法[4], 但是基于神经网络的公钥密码算法还未见相应的报道.

神经网络尽管是由简单的元器件所构成的, 但是它具有复杂的动力学特性, 如神经网络的混沌特性. 在 20世纪 90年代初 L. M. Pecora和 T. L. Carroll发现了混沌同步现象后[5], K. R. Crounse和 V. Milanovic分别提出了基于神经网络混沌同步的对称密码算法[6,7], 这种密码算法具有算法简单, 易于硬件实现等特点, 但是对敏感消息所提供的保密性不是很强, 且在硬件兑现中还存在信道噪声影响和参数匹配等问题.

为了克服上述困难, 郭东辉等提出了一种新的基于神经网络吸引子的对称几率加密算法[8], 他们发现在过饱和的 Hopfield神经网络中存在不按照 Hamming最小规则进行联想的混沌吸引子, 混沌吸引子的个数随神经元个数的增加而增多, 并且从理论上证明了每个吸引子及其相应的吸引域因联接突触矩阵的改变而发生相应的变化, 而且该算法所提供的信息安全性随神经元个数增多时呈指数方式提高. 因而, 该算法能提供很高的实际的信息保密性, 如当神经元个数超过 28时, 所提供的信息安

[收稿日期] 2004 - 10 - 21
[基金项目] 国家自然科学基金项目 (69886002; 60076015); 福建省自然科学基金项目 (A0010019)
[作者简介] 刘年生 (1967 -), 男, 副教授, 从事人工智能与网络通讯方向研究.

全性就高于一般公认可接受的安全性水平 10^{12} MIPS Years

本文引入了 Diffie-Hellman公钥思想，利用可交换矩阵族的特性，提出一种新的基于神经网络吸引子的公钥密码算法，并对算法的安全性和加密效率能进行较深入的分析.

1 算法加密原理

在本节中首先介绍所采用的神经网络模型，然后根据 Diffie - Hellman公钥体制来构造一种新的加密算法.

1.1 神经网络模型

Hopfield神经网络 (Hopfield Neural Networks, HNN)是 J. J. Hopfield在 20世纪 80年代初提出的一类神经网络模型[9]，可以进行硬件兑现. 对于离散 Hopfield神经网络而言，如果神经网络的某一初始状态根据 MHD (Minimum Hamming Distance, 最小汉明距离) 规则收敛到一个系统吸引子，那么它就是稳定状态，这些稳定状态通常被作为 HNN的联想贮存样本. 但是联想神经网络的记忆容量是有限的，对于由 N 个神经元组成的 HNN而言，对随机样本的记忆，其存贮容量仅约为 0.14N. 当所要存贮的样本数超过该模型的存贮容量，那么该神经网络系统的稳定吸引子将发生畸变，使得系统不能按汉明距离最小规则进行联想，出现了过饱和存贮的混沌吸引性质，这时的 HNN就变成为 OHNN (Overstoraged HNN).

假设离散 Hopfield神经网络有 N 个互联神经元，每个神经元状态只为 0或 1，它的下一个状态 $S_i(t+1)$ 取决于当前各神经元的状态，即:

$$S_i(t+1) = \left\{ f\left[\sum_{j=0}^{N-1} T_{ij} S(t) + \theta_j \right] \right\}, \quad i = 0, 1, 2, \cdots, N-1. \tag{1}$$

其中，T_{ij}为神经元 i与 j之间的联接权值，θ_i 为神经元 i的域值，$f(x)$ 为任一非线性函数，不妨设 $f(x) = \sigma(x)$ 为一符号函数，则:

$$\sigma(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{2}$$

在 HNN模型中，神经元的域值 θ_i可定义为 $0 (i = 0, 1, 2, \cdots, N-1)$，$T_{ij}$为一对称矩阵 T. 根据公式 (1) 和 (2)，则有:

$$S(t+1) = F_T(S(t)) = \sigma(S(t)T) \tag{3}$$

其中，$S(t+1) = \{S_0(t), S_1(t), S_2(t), \cdots\cdots, S_{N-1}(t)\}$，系统在时间 t时刻时状态为:

$$S(t) = F_T(S(t-1)) = F_T^t(S(0)) \tag{4}$$

其在 t时刻时系统的能量函数为:

$$E(t) = -\frac{1}{2} \sum_{ij} T_{ij} S_i(t) S_j(t) \tag{5}$$

J. J. Hopfield已证明能量函数随系统状态的演进而单调下降[10]，由于神经网络的能量是有限的，它最终会达到一种稳定状态，即吸引子. 而郭东辉等人的进一步研究结果表明它是混沌吸引子，吸引子与初始状态之间不按 MHD规则进行联想，每个吸引子的吸引域中的状态消息之间存在不可预测的关系[8]; 如果改变联接权值矩阵 T, 则吸引子及其相应的吸引域都会随之发生改变[8]. 在引入随机变换矩阵 H后，原初始状态 S和吸引子 S^{μ} 分别变为新的初始状态 $\hat{S}$和吸引子 $\hat{S}^{\mu}$:

$$\hat{S}^{u} = S^{\mu} * H \tag{6}$$

$$\hat{S} = S * H \tag{7}$$

1.2　基于混沌吸引子的 Diffe-Hellman公钥体制

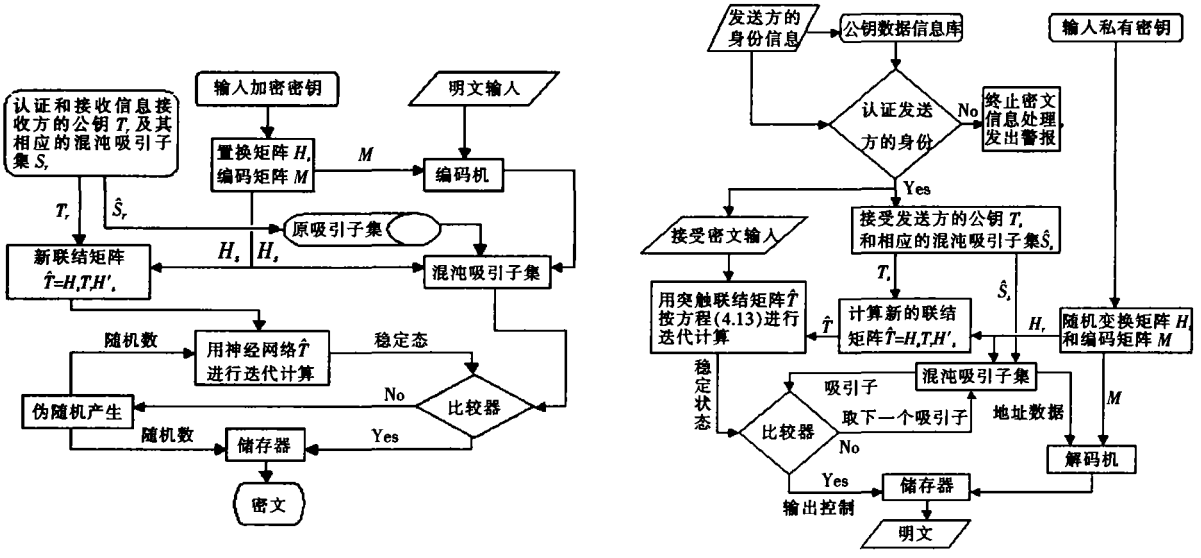
根据矩阵理论[11]，当联结突触矩阵 T为 n阶奇异方阵时，假设任取一 n阶可对角化随机变换矩阵 H，并保密，则计算 $\hat{T}=HTH$是容易的，并且它是矩阵 T的相合矩阵，也是 n阶奇异方阵. 同时，在随机变换矩阵中存在一类特殊的矩阵族，即可交换矩阵族，假设 H_1和 H_2为可交换矩阵族中任意两个同阶方阵，则它们满足 $H_1 * H_2 = H_2 * H_1$.

根据 Diffie-Hellman公钥密码体制的思想，在一组通信用户中共同选取一个联结突触矩阵 T_0，它为 n阶奇异方阵. 每个用户在 n阶方阵交换族中随机选取一个变换方阵，如用户 A任意选取一个非奇异变换方阵 H_a，首先计算 $T_a = H_a T_0 H_a'$，H_a'为 H_a的转置矩阵，然后将 H_a保密，而把 T_a公开. 当同一组内的用户 A与 B需要保密通信时，他们就可以把 $\hat{T}=H_a T_b H_a' = H_b T_a H_b'$作为他们之间保密通信的共同密钥，用户 A (或用户 B) 均可以根据自己的私钥和对方的公钥很容易地计算出公共密钥. 但是第三者将很难从公钥 T_a和 T_b中直接计算出 $\hat{T}$或 H_a和 H_b，特别当 n较大时.

为了进一步增强信息传输安全，防止中间欺骗者攻击，采用带认证的 Diffie-Hellman密钥交换协议，对保密通信的双方用数字签名和公钥证书来相互认证对方的身份是否合法[12].

2　加密方案

由上述神经网络过饱和存贮的混沌吸引性质可以知道：只要改变少量的存贮样本 (S^{μ})或少量神经元之间的连接矩阵元 (T_{ij})，该神经网络系统就可以获得具有大范围混沌的、随机的吸引域的分类吸引子. 为此，可以根据 Diffie-Hellman公钥密码体制设计出安全性较高的计算机公钥加密通信系统，如图 1所示.



(a)基于神经网络混沌吸引子的加密算法　　(b)基于神经网络混沌吸引子的解密算法

图 1　基于神经网络混沌吸引子的加密通信系统方案

Fig.1　The encryption scheme of communication system based on chaotic attractors of neural networks

2.1　密钥产生与分配

一组需要保密通信的用户，共同使用一个经过仔细选择的突触联结矩阵 T_0，应注意两点：1) T_0为奇异方阵；2) 能使神经网络产生较多的不可预测的吸引子，即方阵每行或每列具有相同的 -1，0和 +1的数目[13]. 因此，在矩阵阶数 n较高时，最好是通过编程由计算机随机产生 T_0，计算出 T_0所

对应的混沌吸引子集.

组内用户不应对任何组外用户公开 T_0，以增加中间攻击者破解密钥的计算难度，并且组内用户还应共同确定一种较理想的明文编码和解码方法，即编码矩阵 M. 每个用户在 n阶方阵交换族中随机选择一个互不相同的变换矩阵，比如用户 i所选择的 H_i，计算 $T_i = H_i T_0 H_i'$及其相应的混沌吸引子集 (根据公式 1)，将 H_i保密，而将 T_i公开，然后每个使用者将自己的公开密钥及其对应的混沌吸引子集，连同用户名、地址等其他信息用数字签名方式保存在某认证中心公钥信息库中，并获得该认证中心颁发的公钥证书，并每次把经过数字签名过的信息和公钥证书等放在需保密的数据信息的头部，便于带认证的 Diffie-Hellman密钥交换协议中通信双方相互认证对方身份，防止中间者攻击.

2.2 加密算法

1) 密钥的生成 信息的发送方首先输入自己的私有密钥 H_s，并且接收方经认证过的合法的公钥 T_r及其相应的吸引子集，从而计算出神经网络系统新的联接突触矩阵 $\hat{T} = H_s T_r H_s' = H_s H_r T_0 H_r' H_s'$，并且将 T_r所对应的吸引子集转换为 $\hat{T}$所对应的吸引子集.

2) 明文的编码处理 利用编码矩阵 M和属于 $\hat{T}$的吸引子 $\hat{S}^{\mu}$，将明文 Y映射到编码明文 $Y_x = \{\hat{S}^{\mu}\}$中.

3) 密文的生成 利用伪随机数生成器生成一由 0和 1所组成的数组，作为 OHNN基于突触矩阵 $\hat{T}$的初始状态 $S(0)$，按公式 (1) 和 (2) 进行迭代运算，得到一个稳定状态 $S(\infty)$，并与 $\hat{S}^{\mu}$相比较；如果 $S(\infty)$等于 $\hat{S}^{\mu}$，则说明 $S(0)$为 $\hat{S}^{\mu}$吸引域中一个状态，这个随机数 $S(0)$就作为明文 Y所对应的密文 X输出，在公共信道中传输. 如果 $S(\infty)$不等于 $\hat{S}^{\mu}$，则伪随机数生成器重新产生新的数组，依上述方法重新进行计算和比较，直至找到一个随机数的稳定状态 $S(\infty)$等于 $\hat{S}^{\mu}$为止，之后再进行新的明文处理，产生新的密文.

2.3 解密算法

1) 接收方首先对发送方身份进行认证 根据发送方所提供的身份信息，通过数字签名和公钥证书检查与核对对方的身份，如果身份认证信息是真实的，发送方是合法的用户，才对密文进行解密处理，否则，对接收到的信息隔离删除，并发出警报.

2) 解密的过程 首先输入自己的私有密钥 H_r以及发送方的合法公钥 T_s，计算新的联接突触矩阵 $\hat{T} = H_r T_s H_r' = H_r H_s T_0 H_s' H_r'$及其相应新的吸引子集 $\hat{S}^{\mu}$，对神经元所有状态按公式 (6)、(7) 进行吸引子及其相应吸引域的重新计算，生成新的混沌吸引子集；其次，生成编码明文，输入密文 X，利用公式 (1) 进行迭代运算，得到相应的一个稳定状态 $S(\infty)$，而 $S(\infty)$等于 $\hat{S}^{\mu}$，即编码明文 $Y_x = \{\hat{S}^{\mu}\}$；最后恢复原明文，利用编码矩阵 M，将编码明文 $Y_x = \{\hat{S}^{\mu}\}$解码为原明文 Y，从而完成密文的解密，恢复原明文.

现以由 8个神经元所组成的 OHNN为例，假设某用户组内所选的公共联接突触矩阵 T_0为：

$$T_0 = \begin{pmatrix} 1 & -1 & 0 & 1 & -1 & -1 & 0 & 1 \\ -1 & 1 & 1 & 0 & -1 & -1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 0 & 1 & 1 & -1 & 0 & -1 & -1 \\ -1 & -1 & 0 & -1 & 0 & 1 & 1 & 1 \\ -1 & -1 & -1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & -1 & -1 & 1 & 0 & 1 & -1 \\ 1 & 0 & -1 & -1 & 1 & 1 & -1 & 0 \end{pmatrix}$$

而不妨设发送方和接收方的私有密钥分别为如下随机变换矩阵 H_s 和 H_r:

$$H_s = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad H_r = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

发送方 T_s 和接收方 T_r 的公钥分别为:

$$T_s = H_s T_0 H_s' = \begin{pmatrix} 1 & 1 & 0 & -1 & -1 & 1 & 0 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 & 0 \\ 0 & 1 & 1 & -1 & 0 & -1 & -1 & 1 \\ -1 & 0 & -1 & 0 & 1 & 1 & 1 & -1 \\ -1 & -1 & 0 & 1 & 1 & 0 & 1 & -1 \\ 1 & -1 & -1 & 1 & 0 & 1 & -1 & 0 \\ 0 & -1 & -1 & 1 & 1 & -1 & 0 & 1 \\ -1 & 0 & 1 & -1 & -1 & 0 & 1 & 1 \end{pmatrix},$$

$$T_r = H_r T_0 H_r' = \begin{pmatrix} 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 1 & 1 & -1 & 0 & -1 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 & 1 & 1 & -1 & -1 \\ -1 & 0 & 1 & 1 & 0 & 1 & -1 & -1 \\ -1 & -1 & 1 & 0 & 1 & -1 & 0 & 1 \\ -1 & -1 & 1 & 1 & -1 & 0 & 1 & 0 \\ 0 & 1 & -1 & -1 & 0 & 1 & 1 & -1 \\ 1 & 0 & -1 & -1 & 1 & 0 & -1 & 1 \end{pmatrix};$$

而通信双方的共同密钥 (新的联接突触矩阵 $\hat{T}$) 为:

$$\hat{T} = H_s H_r T_0 H_r' H_s' = H_r H_s T_0 H_s' H_r' = \begin{pmatrix} 1 & -1 & 0 & -1 & -1 & 1 & 0 & 1 \\ -1 & 0 & 1 & 1 & 1 & -1 & -1 & 0 \\ 0 & 1 & 1 & 0 & 1 & -1 & -1 & -1 \\ -1 & 1 & 0 & 1 & -1 & 0 & 1 & -1 \\ -1 & 1 & 1 & -1 & 0 & 1 & 0 & -1 \\ 1 & -1 & -1 & 0 & 1 & 1 & -1 & 0 \\ 0 & -1 & -1 & 1 & 0 & -1 & 1 & 1 \\ 1 & 0 & -1 & -1 & -1 & 0 & 1 & 1 \end{pmatrix}.$$

对应不同的联接突触矩阵如 T_0、T_s、T_r 和 $\hat{T}$ 等，均有相应不同的吸引子和吸引域，其中对于 $\hat{T}$ 而言，则有 $\hat{S}^0 = (11001011)$、$\hat{S}^1 = (11001101)$、$\hat{S}^2 = (01011101)$、$\hat{S}^3 = (01111100)$、$\hat{S}^4 = (01110110)$、$\hat{S}^5 = (10110110)$、$\hat{S}^6 = (10110011)$、$\hat{S}^7 = (10101011)$. 若采用 3位的编码矩阵 M,

即: 000→0, 001→1, 010→2, 011→3, 100→4, 101→5, 110→6, 111→7, 则它们分别对应前面的8个吸引子 $\hat{S}^{\mu}$. 当加密端输入的明文为 110时, 它所对应的吸引子为 $\hat{S}^{6}$ = (10110011), 再从 $\hat{S}^{6}$ 的吸引域中随机选择一个非稳定状态如 (10010010) 作为它对应的密文输出. 而在解密端, 首先对发送方的身份进行认证, 若合法, 则输入自己的私有密钥和对方的合法公钥, 计算出共同的密钥联接突触矩阵 $\hat{T}$, 然后, 当收到密文 (10010010) 时, 根据公式 (1) 和联接突触矩阵 $\hat{T}$, 计算出密文所对应的吸引子 (10110011), 即 $\hat{S}^{6}$ 为密文所相应的编码明文; 再利用 3位的编码矩阵 M, 6→110, 就恢复出原明文.

3 算法性能分析

评价一种加密算法优劣很重要的两个因素就是要其所提供的安全性和加密速度, 对本加密方案而言, 数据加密速度主要受加密效率的影响.

3.1 加密方案的安全性

在公钥密码系统中, 它的安全性是基于奇异矩阵的分解困难性和 OHNN混沌分类特性. 对密码系统的攻击, 其主要目的是寻找密钥, 就本密码系统而言, 它既可以根据 OHNN混沌分类特性来攻击密钥, 也可以通过对密码系统所采用的奇异矩阵进行分解来寻找密钥.

3.1.1 对矩阵分解的攻击

在加密方案中已说明了联结突触矩阵 T_0 为奇异方阵, 因而 T_0、T_r、T_s 和 $\hat{T}$ 均为奇异方阵, 由 T_0、H_r 和 H_s 计算 T_r、T_s 和 $\hat{T}$ 是容易的, 而由 T_r、T_s 和 $\hat{T}$ 求解 H_r 和 H_s 是困难的, 原因如下: 对于奇异方阵的分解而言, 其分解不是唯一的. 例如: 假设奇异矩阵 $\hat{T}=B\times C$ 是 $\hat{T}$ 的一种分解, 则对任何相同阶的左可逆矩阵 R (即 $R^{-1}\times R$ 等于单位矩阵), $\hat{T}(B\times R^{-1})\times(R\times C)$ 也是一种分解, 其中 R^{-1} 为矩阵 R 的逆. 而且难以找到甚至不存在可行的求出全部分解的算法, 因而从公钥 T_r、T_s 分别推出私有密钥 H_r 和 H_s 是困难的.

常见的矩阵分解方法主要有 QR (正交三角分解) 分解、奇异值分解和 LU (三角分解) 分解. 首先, T_0、T_r、T_s 和 $\hat{T}$ 均为奇异方阵, 因此它是不能通过 QR分解方法来分解的. 其次, 当 T_0、T_r、T_s 和 $\hat{T}$ 的阶数 $n(n>64)$ 较大时, 利用矩阵的奇异值分解是行不通的, 其困难性在于两个方面: 其一, 至今尚无切实可行的方法能求出一般高阶矩阵的全部准确特征值[14]; 其二, 奇异方阵的奇异值分解一般也不是唯一的, 因为分解式中存在多个正交矩阵[15], 至少有 $(2+2^{m-n+1}+2\sum_{k=2}^{r}(2^{k}!)^{2^{m-nk-1}})$, 其中 m 为矩阵的阶, 可令 $m>n$, $r=\left[\frac{m-1}{n}\right]$; 而且, 还应注意到正交矩阵具有如下两个重要性质: 1) 正交矩阵的逆阵是正交矩阵; 2) 任意两个正交矩阵的乘积仍是正交矩阵. 因此, 尽管不知道 n 阶正交矩阵的确切个数, 但仍可以初步推定其的空间范围是比较大的, 遍历其空间将是很困难的. 再次, 对于 LU 分解来说, 大多数情况下, 它们的分解并不是唯一的, 目前尚无法遍历所有的分解; 同时, LU 分解所得的分解并不是密钥产生那种的形如 $\hat{T}=HT_0H$ 的方式.

退一步讲, 即使第三者知道原突触联结矩阵 T_0, 采用试凑的方法, 要从公钥 T_i 中推出私钥 H_i 在计算上仍存在难以克服的困难性; 例如采用穷举攻击法来寻找私钥 H_i, 一种方式直接虚构一个 H_i, 测试 $H_iT_0H_i'$ 是否等于 T_i, 在这样情况下, 即使在 n阶变换矩阵 H_i 中的所有元素只为 0或 1, 那么, 它可能的数目为 2^{n^2}, 即它的计算时间复杂性为 $O(2^{n^2})$, 即随矩阵阶数 n的平方而呈指数性增长, 当

n较大时，由于计算量太大，实际上是不可能计算的. 另一种方式就是采用矩阵变换，这在变换矩阵 H_i为正交矩阵时才能使用，即先将 T_0转化为 Hessenberg矩阵，然后将 T_i也转化为 Hessenberg矩阵，而 T_i与 T_0可以具有一个相同的 Hessenberg矩阵，如果将 T_i与 T_0约化成同一 Hessenberg矩阵，则可以求出私钥 H_i来，但同样存在计算困难的问题，其一是将任一方阵约化为 Hessenberg矩阵，其计算量为 $O(n^3)$，n为矩阵的阶数；其二是在一般情况下，Hessenberg分解是不唯一的[14]，至少有 2^n个. 因此，当 n比较大 (如大于 128) 时，要遍历其所有的 Hessenberg分解形式在计算上是不可能的.

另外，对于任何第三者知道通信双方的公钥 T_s和 T_r，这样他是否能从中推出通信双方的公共密钥 $\hat{T}$. 对于这个问题，第三者要知道公共密钥只有两种途径，一种就是从公钥 T_s (或 T_r) 推出私钥 H_s (或 H_r) 来求公共密钥 $\hat{T}$，如前面所述将是很困难的，当 n比较大时，计算上是不可行的；另一种用已知的公钥 T_s和 T_r进行矩阵变换来公共密钥 $\hat{T}$ 如试求一矩阵 X，让它满足：

$$\hat{T} = T_s X T_r \tag{8}$$

将其 T_s和 T_r代入公式 (8)，则

$$\hat{T} = H_s T_0 H_s' X H_r T_0 H_r' \tag{9}$$

而 $$\hat{T} = H_s H_r T_0 H_r' H_s' = H_r H_s T_0 H_s' H_r' \tag{10}$$

由于 T_0是奇异方阵，$\hat{T}$、T_s和 T_r也都为奇异方阵，不存在相应的逆矩阵，所以不可能从理论上求解出一个矩阵 X使得满足方程 (9) 与 (10) 相等，这种想法也是行不通的.

3.1.2　抗常用密码攻击的能力

目前无论是选择性明文攻击还是已知明文攻击都不可能找到其随机变换矩阵 H，即密钥，而且整个密码系统是不规则的，在加密过程中它是随机选取密文的，同一个明文块可对应多个密文块，而在解密过程中又采用自吸引的方法. 差分密码分析法不可能有效地破译这种不规则的密码算法. 基于明文特性统计几率的穷举攻击法可能是唯一能破解本密码系统的有效方法，但是代价是巨大的.

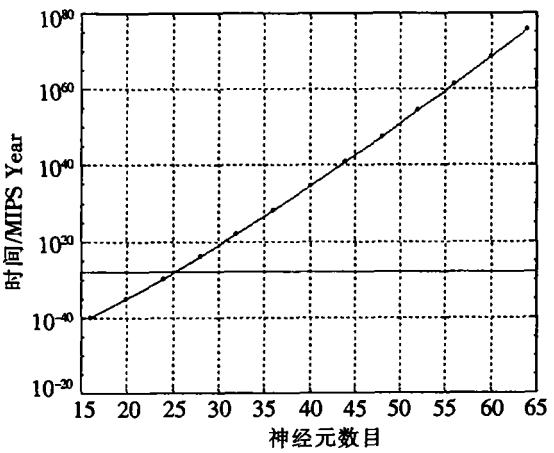


图 2　穷举搜寻密钥 H 所需时间与 OHNN 神经元个数之间的关系

Fig.2　The relation between the time of exhaustive searching encryption key and the number of neuron OHNN

由前面的加密方案可知，对由 N个神经元所组成的 OHNN，所选取的吸引子数目为 p，则对于每种编码矩阵，随机变换矩阵 H有 $N!$种可能，即密钥空间为 $N!$. 即使是已知明文攻击，采用穷举法搜寻随机变换矩阵 H，将要运行 $N!$次，如果用计算能力为每秒 10^6个变换矩阵 H的专业计算机来穷举搜寻确定变换矩阵 H，则遍历变换矩阵 H空间所需的时间取决于网络的神经元个数 N，如图 2所示. 当 $N=32$时，成功地搜寻到一次变换矩阵 H所需的就为 10^{20}M IPS Years数量级，高于目前可接受的安全水平 10^{12}M IPS Years

另一方面，在本密码方案的加密过程中，要求在每个吸引域内随机选取一个非稳定状态 A^{μ}来替代相应的吸引子，以消除明文字符的统计相似性，避免基于统计分析的密码攻击. 因而，在加密过程中，每个吸引域内非稳定状态的数目 Λ对本密码系统的安全性而言是另一个密钥参数. 较大的 Λ可降低相同明文出现相同密文的几率. 而参数 Λ主要取决于 OHNN的神经元个数 N，N越大，Λ就越

大，例如，$N=8$时，$\Lambda=20$，而 $N=32$时，$\Lambda=2^{16}$. 加密时替代吸引子的非稳定状态 A^{μ} 是用伪随机生成器产生的，如果密码系统中将伪随机生成器设计成具有时变性，则相同的明文在不同的时刻被加密成为不同的密文，从而进一步增强了密码系统的安全性.

3.2 加密效率

笔者提出的这一加密方案与基于混沌同步的加密方案相比，避免了同步混沌通讯系统中必须要求收发两端严格同步的诸多麻烦和不便，只要算法和密钥相同，就可以准确地进行信息的加密与解密. 同时，它采用 Diffie-Hellman公钥密码体制，与对称密码体制相比，可更好地满足现代计算机多媒体保密网络通信的需要，有效地减少用户之间的密钥量，方便了密钥管理. 不过它的密文长度比原明文的要长许多，存在着密文数据膨胀的问题. 从密码学的观点来说，一般不希望过度的密文数据膨胀，它会影响加密和传输的效率.

在本加密方案中，假设是由 N 个神经元所组成的 OHNN，每次加密 n比特长的二进制明文，则有 $p=2^n$ 个 OHNN的吸引子被作为替代 n比特长的二进制明文 (应注意的是：只有 OHNN的吸引子样本数多于或等于 2^n 个，加密算法才能有效)，每次所产生的密文的二进制长度为 N 比特，则密文数据的膨胀率为：

$$e = N/n = N/\log_2 p \qquad (11)$$

从公式 (11) 显示出密文数据膨胀率与 OHNN神经元个数和明文编码长度之间的关系，OHNN神经元个数 N越大，相应的吸引子数目就越多，如果采用适当的明文编码长度，就可有效降低密文的膨胀率；又由于 OHNN是采用并行运算模式，神经元个数的增多并不降低它的加密或解密的速度.

4 结论

根据神经网络的混沌吸引子性质提出了一种新的公钥加密算法，该算法具有较高的安全性，有效地抵抗常规的密码分析方法的攻击，并利用神经网络在专用芯片中对敏感信息进行并行计算处理，数据加密速度比较高，因此，该加密算法为下一代互联网的安全通信提供一种新的候选加密算法，满足其对信息传输的安全性和实时性双重要求.

[参考文献]

[1] Marcus Goncalves, Kitty Niles. IPv6网络 [M]. 黄锡伟，杨震，译. 北京：人民邮电出版社，2000.

[2] Simon Haykin. Neural Networks -A Comprehensive Foundation (Second Edition) [M]. 北京：清华大学出版社，2001.

[3] Diffie W, Hellman M. New Directions In Cryptography [J]. IEEE Transactions on Information Theory, 1976, 22 (6): 644-654.

[4] William Stallings. Cryptography and Network Security: Principles and Practice (2nd) [M]. New Jersey: Prentice-Hall 2003.

[5] Pecora L M, Carroll T L. Synchronization in Chaotic Systems [J]. Physical Review Letters, 1990, 64 (8): 821-824.

[6] Crounse K R, Yang T, Chua L O. Pseudo-random sequence generation using the CNN universal machine with applications to cryptography [A]. IEEE. Proceedings of the IEEE International Workshop on Cellular Neural Networks and their Applications [C]. Singapore: World Scientific Press, 1996. 433-438.

[7] Veljko Milanovic, Mona E. Zaghloul. Synchronization of chaotic neural networks for secure communications [J]. IEEE International Symposium on Circuits and Systems, 1996, 3: 28-31.

[8] Donghui Guo, Zheng L M, Zheng L L. A New Symmetric Probabilistic Encryption Scheme Based on Chaotic Attractors of

Neural Networks [J]. Applied Intelligence, 1999, 10 (1): 71-84.
[9] Hopfield J J. Neural Networks and Physical Systems with Emergent Collective Computational Abilities [J]. Proceedings of the National Academy of Science, 1982, 79 (4): 2554-2558.
[10] Hopfield J J. Neurons, Dynamics and Computation [J]. Physics Today, 1994, 47: 40-46.
[11] 陈景良，陈向晖. 特殊矩阵 [M]. 北京：清华大学出版社，2001.
[12] Bresson Emmanuel, Chevassut Olivier, Pointcheval David, et al. Probably Authenticated Group Diffie-Hellman Key Exchange [A]. Pierangela Samarati Proceedings of the 8th ACM Conference on Computer and Communications Security [C]. New York: ACM Press, 2001. 255-264.
[13] Gardner E. Maximum Storage Capacity in Neural Networks [J]. Europhys Lett, 1987, 4 (4): 481-485.
[14] 程云鹏. 矩阵论 [M]. 西安：西北工业大学出版社，2001.
[15] 温巧燕，肖国镇. m阶相关免疫函数的构造与计数 [J]. 西安电子科技大学学报，1997, 24 (1): 36-39.

A New Public-key Cryptography Based on Chaotic Attractors of Neural Networks

LIU Nian-sheng[1], GUO Dong-hui[2]
(1. School of Computer Engineering, Jimei University, Xiamen 361021, China;
2. School of Physics and Mechanical & Electrical Engineering, Xiamen University, Xiamen 361021, China)

Abstract: A new public-key cryptography based on chaotic attractors of neural networks is described. There is a one-way function between chaotic attractors and initial states in an Overstoraged Hopfield Neural Networks (OHNN), and each attractor and its corresponding domain of attraction are changed with permutation operations on the neural synaptic matrix. If the neural synaptic matrix is used as a trap door and changed by commutative random permutation matrix, a new cryptography technique according to Diffie-Hellman public-key cryptosystem is proposed. By keeping the random permutation operation of the neural synaptic matrix as the secret key, and the neural synaptic matrix after permutation as public-key, a new encryption scheme for a public-key cryptosystem is introduced. Security of the new scheme is discussed.

Key words: neural networks; public-key cryptosystem; chaotic attractor; matrix decomposition

(责任编辑　马建华)