

基于二维超混沌映射的单向 Hash 函数构造^{*}

彭 飞[†] 丘水生 龙 敏

(华南理工大学电子与信息学院, 广州 510640)

(2004 年 9 月 8 日收到; 2005 年 2 月 21 日收到修改稿)

快速有效的单向 Hash 函数的设计是现代密码学研究的一个热点. 提出了一种基于二维超混沌映射的单向 Hash 函数的构造方法. 该方法将原始输入报文按字节进行线性变换以及混沌映射, 在所产生的混沌序列中选值作为二维超混沌映射的初始值, 对迭代后所得的超混沌信号经过调制构造单向 Hash 函数. 实验与分析结果表明, 这种构造方法实现简单、对初值有高度敏感性、具有很好的单向性、弱碰撞性和保密性能.

关键词: 单向 Hash 函数, 二维超混沌, 混沌映射

PACC: 0545

1. 引 言

随着全世界电子商务技术的迅猛发展, 以公钥密码术、数字签名等为代表的加密安全技术已成为研究的热点^[1-3]. 单向 Hash 函数是实现数字签名和数字认证的一个关键环节, 并广泛应用于消息的完整性检测, 内存的散布分配, 操作系统以及数据库中帐号口令的安全存储等^[4]. 传统的单向 Hash 方法有 MD2, MD4, MD5 以及 SHA 等标准^[2, 5], 但大多是基于计算复杂度假设的. 他们通过大量的类似于异或的逻辑运算或是通过分组加密多次迭代得到 Hash 结果^[1]. 后种方法运算量很大, 且难以找到快速同时可靠的加密方法; 而前种方法中由于异或运算中固有的缺陷, 虽然每步运算简单, 但计算轮数即使在被处理的文本很短情况下也很大^[4, 6].

针对以上问题, 有人提出了基于混沌映射模型的单向 Hash 算法^[4, 7, 8]. 但这些方案均存在以下不足: (1) 这些系统能够实现摘要提取, 是基于混沌同步的鲁棒性, 而这种鲁棒性也可被攻击者所利用, 从而降低了系统的保密性^[9]; (2) 这些系统均是基于特定的简单混沌系统来实现的, 可以利用各种混沌预测技术成功地破译、提取信息信号^[10-15], 保密性能有限; (3) 由于实际实现中的有效字长精度效应, 混沌映射本身所产生的混沌序列也会退化为周期序列^[9].

文献[6]提出了一种基于广义混沌映射切换的单向 Hash 函数算法, 虽然在性能上有一定程度的提高, 但由于其采用的混沌模型简单, 仍然存在 (2)、(3) 问题. 因此, 如何增加混沌信号的复杂度和减小有效字长精度效应的影响是提高混沌 Hash 单向性、置乱性和弱碰撞性的主要问题. 本文通过采用二维超混沌映射模型产生超混沌信号, 大大增强了信号的复杂度. 同时抽取多个超混沌信号进行调制来减小有效字长精度效应的影响. 实验结果表明, 这种方式能有效地提高混沌 Hash 函数的性能, 而同时由于采用简单形式的二维超混沌映射模型, 因而也具有较快的运算速度.

2. 二维超混沌映射构造单向 Hash 函数的可行性

单向函数的定义: 如映射 $H: X \rightarrow Y$ 对所有的 $x \in X$, $H(x)$ 都容易计算. 但反过来, 给定一个 $H(x)$ 要求 x 在计算上是困难的, 满足以上条件的函数称之为单向函数. 单向 Hash 函数是一种特殊的单向函数, 它满足以下 4 个条件^[1]:

- (1) 输入为任意长度的序列, 但输出长度固定;
- (2) 不可逆性: 即已知 $c = \text{Hash}(m)$, 求 m 计算困难, 除穷举外没有好办法;
- (3) 防伪造性: 已知 $c = \text{Hash}(m)$, 求 n 使 $\text{Hash}(n) = c$ 计算困难;

^{*} 国家自然科学基金项目 (批准号 70372004) 和广东省自然科学基金项目 (批准号 31445, 20820) 资助的课题.

[†] 通讯联系人. E-mail: eepengf@yahoo.com.cn

(4)初值敏感性 : $c = \text{Hash}(m)$ 中 c 的每一 bit 都与 m 的每一 bit 相关,并有高度的敏感性,即每改变 m 的 1bit,都将对 c 产生明显的影响.

从数学角度来看,报文空间可以是无限的,而 Hash 结果总是一段定长字节的数字,可能会有无数的报文具有同样的 Hash 函数值.但在 Hash 结果达到一定长度,比如结果为固定的 128 bit 长时,结果空间将达到 $2^{128} \approx 3.4028 \times 10^{28}$ 个,在现有的计算环境下同时在这样大的空间中进行穷举计算是困难的.

混沌^[16, 17]是指确定性非线性动力系统的长期行为对系统初始状态或系统参数异常敏感,却又不发散,但无法精确重复的现象,它是非线性系统普遍存在的一种复杂动力学行为.混沌序列具有对初始条件和混沌参数的极端敏感性、白噪声的统计特性和混沌序列的遍历特性^[6].因此无法对混沌序列进行正确长期的预测.

超混沌是一类特殊的混沌现象.它们的区别在于超混沌比混沌具有更多方向的不稳定性,而混沌只在一个方向上发生轨迹扩张(发散),前者至少有两个表征这种发散特性(轨迹对初始条件的敏感性)的李雅普诺夫指数 $\lambda > 0$,而后者只有一个 $\lambda > 0$ ^[18].从安全的角度考虑,由于超混沌系统比低维混沌系统具有更复杂的相空间^[10, 19],用它设计单向 Hash 函数不易被一类基于混沌同步的分析方法攻击.

二维超混沌系统一种二维非线性动力系统,它具有两个正的李雅普诺夫指数.相对于高维超混沌系统,它产生超混沌序列的速度快、时间短,能满足现代通讯实时性的要求^[20].

以上表明二维超混沌序列拥有单向 Hash 函数所要求的较好的散布性和不可逆性、防伪造性、初值敏感性.本文通过二维超混沌映射模型产生超混沌信号,大大增强了信号的复杂度,使得信号更加难以

预测,同时通过抽取多个超混沌信号进行调制来减小有效字长精度效应的影响.由于采用简单形式的二维超混沌系统,在保证较好的单向 Hash 性能的同时拥有较快的运算速度.

3. 基于二维超混沌映射的单向 Hash 函数构造

3.1. 简单形式的二维超混沌映射

二维超混沌映射的一般形式可以表示为^[20] :

$$\begin{cases} x_{n+1} = f_1(x_n, y_n) \\ y_{n+1} = f_2(x_n, y_n) \end{cases} \quad (1)$$

其中,

$$\begin{cases} f_1(x_n, y_n) = a_1 + a_2 x_n + a_3 x_n^2 + a_4 y_n + a_5 y_n^2 + a_6 x_n y_n; \\ f_2(x_n, y_n) = a_7 + a_8 x_n + a_9 x_n^2 + a_{10} y_n + a_{11} y_n^2 + a_{12} x_n y_n. \end{cases} \quad (2)$$

式中 $a_i (i = 1, 2, \dots, 12)$ 均为待定常数.

利用 Wolf 提出的数值算法^[21],计算出系统(1)的两个李氏指数,这样就可以得到一系列参数不同的二维超混沌方程.系统(1)所描述的是最高项为二次的二维离散方程的一般形式.为了使方程形式更加简单,尽可能地减少迭代函数中的非线性项,可对(1)式中非线性项的系数加以限制,使其大部分为零.调整非零部分的参数,进一步通过计算方程的李氏指数来判断,最后得到一些有代表性的形式简单的二维超混沌映射.表 1 列出了计算得到的部分结果.由表 1 可见,列出的系统都具有形式简单、总的项数少、非线性项数少等特点.每一个系统在确定参数下均有两个大于零的李雅普诺夫指数.

表 1 简单形式的二维超混沌映射

序号	二维超混沌映射	参数值	李雅普诺夫指数
(A)	$x_{n+1} = a_4 y_n + a_5 y_n^2$	$a_4 = 1.55, a_5 = -1.3$	0.238
	$y_{n+1} = a_8 x_n + a_{10} y_n$	$a_8 = -1.1, a_{10} = 0.1$	0.166
(B)	$x_{n+1} = a_2 x_n + a_5 y_n^2$	$a_2 = 0.1, a_5 = 1.8$	0.139
	$y_{n+1} = a_7 + a_8 x_n + a_{10} y_n$	$a_7 = 0.7, a_8 = -2.2, a_{10} = 0.2$	0.063
(C)	$x_{n+1} = a_5 y_n^2$	$a_5 = 1.3,$	0.221
	$y_{n+1} = a_7 + a_8 x_n + a_{10} y_n$	$a_7 = -1.05, a_8 = 1.15, a_{10} = -0.2$	0.046
(D)	$x_{n+1} = a_2 x_n + a_4 y_n$	$a_2 = -0.95, a_4 = -1.3$	0.302
	$y_{n+1} = a_7 + a_9 x_n^2 + a_{10} y_n$	$a_7 = -0.45, a_9 = -2.2, a_{10} = 1.05$	0.240
(E)	$x_{n+1} = a_2 x_n + a_4 y_n + a_5 y_n^2$	$a_2 = 0.55, a_4 = -2.3, a_5 = 0.9$	0.208
	$y_{n+1} = a_7 + a_8 x_n$	$a_7 = 0.65, a_{10} = -1.0$	0.063
(F)	$x_{n+1} = a_1 + a_2 y_n + a_4 y_n$	$a_2 = 0.2, a_2 = 0.3, a_4 = 0.5$	0.161
	$y_{n+1} = a_7 + a_9 x_n^2$	$a_7 = -1.6, a_9 = 3.7$	0.095

3.2. 单向 Hash 函数构造

以表 1 中序号(A)的二维超混沌映射为例.

分析表明^[22], 当 $a_4 = 1.66$ 时, 系统进入超混沌状态, 此时的序列区间为: $-1.577702 < x_n < 0.529923$; $-0.634822 < y_n < 1.862664$, 此时的二维混沌吸引子如图 1 所示.

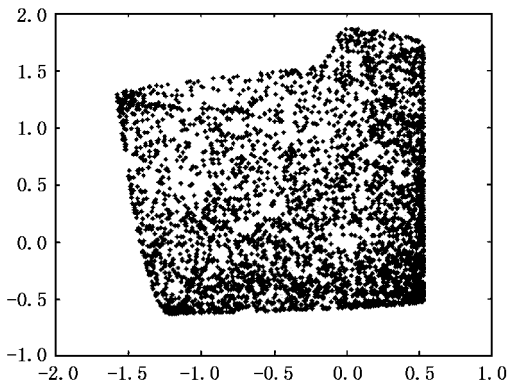


图 1 二维混沌吸引子

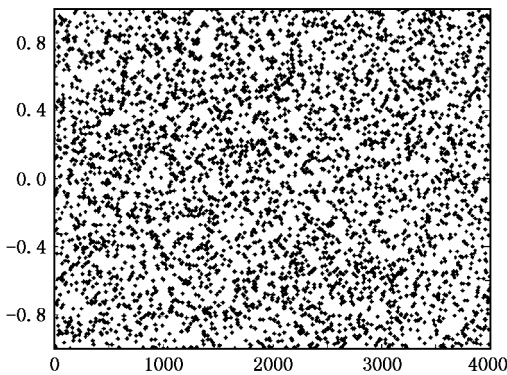


图 2 混沌映射(3)分布

单向 Hash 函数构造原理 将原始输入报文以字节为单位经过线性变换以及通过混沌映射进行一系列的迭代, 保证报文的每个字节都经过迭代, 得到一个离散的混沌时间序列. 从中任取两个作为二维超混沌映射的初始值进行迭代, 最后将 $x_R, x_{2R}, x_{3R}, y_R, y_{2R}, y_{3R}$ (R 为某一设定值) 六个迭代结果分别映射成 40bit, 40bit, 48bit, 40bit, 40bit, 48bit 的二进制数, 然后从中选择 20bit, 20bit, 24bit, 20bit, 20bit, 24bit, 合起来组成 128bit 的 Hash 值.

算法描述如下:

(1) 将待处理报文按对应字节的 ASCII 码转换为数字, 线性变换为 $[-1, 1]$ 范围内的数, 这样整个

报文得到一个大数组, 记为 T , 记数组长度即报文字节数为 N ;

(2) 令 $x_1 = 2 \times T_1/255 - 0.8, x_2 = 2 \times T_2/255 - 0.8$;

(3) 进行迭代运算. 令迭代轮数为 $r = R \times ([N/R] + K)$, 这里, $[]$ 表示取整运算; K, R 均为大于 0 的整数;

a) 在混沌映射(3) (其混沌映射分布如图 2 所示)的基础上,

$$a_j = ((1 + 0.3 \times (a_{j-2} - a_{j-3}) + 2917 \times a_{j-1}^2) \bmod 2) - 1 \quad (3)$$

令 $c = a_{j-3}$, j 从 2 到 r 按如下混沌映射计算:

$$a_j = ((1 + 0.3 \times (a_{j-2} - c) + 2917 \times a_{j-1}^2) \bmod 2) - 1 \quad (4)$$

当 $j \leq N$ 时, 令 $c = 2 \times T_j/255 - 0.8$;

当 $j > N$ 时, 令 $c = a_{j-2}$;

b) 将 a_{r-1}, a_r 分别进行线性变换: $a'_k = 1.045 \times a_k - 0.55$; $a'_l = 1.245 \times a_l + 0.615$ 得到 a'_{r-1} 和 a'_r ;

c) 令 $x_1 = a'_{r-1}, x_2 = a'_r, m$ 从 2 到 $3R$ 按

$$\begin{cases} x_{m+1} = 1.66 \times y_m - 1.3 \times y_m^2 \\ y_{m+1} = -1.1 \times x_m + 0.3 \times y_m^n \end{cases} \quad (5)$$

进行计算;

(4) 将 $x_R, x_{2R}, x_{3R}, y_R, y_{2R}, y_{3R}$ 六个迭代结果分别通过线性变换映射成 40bit, 40bit, 48bit, 40bit, 40bit, 48bit 的二进制数, 然后按照从 40bit 中选出序号第 15 位到第 34 位, 从 48bit 中选出序号第 13 位到第 36 位的规则, 分别从中选择出 20bit, 20bit, 24bit, 20bit, 20bit, 24bit, 合起来组成 128bit 的 Hash 值.

对于相同族类的其他二维超混沌映射, 算法的设计思想基本相同, 只是所选择的二维超混沌系统及其参数有所不同而已.

4. 仿真结果

4.1. 文本 Hash 结果

对上述单向 Hash 函数构造算法进行计算机仿真, 取初始文本 1 为 "The delegates on Monday began a three-day process to formally nominate Bush and Vice

President Dick Cheney to a second term with a roll call full of home-state bragging and lots of praise for their beloved incumbents. Officials tried to keep secret which state would give Bush the 1,255 delegate votes to seal the nomination. That was likely to come Tuesday night, with the roll ending Wednesday.”. 文本 2 将文本 1 中首字母 T 改为 S; 文本 3 将文本 1 中的 President 改为 Presidents; 文本 4 将文本 1 最后的句号改为逗号; 文本 5 在文本 1 的第一句话后加一个空格; 文本 6 将文本 1 中的 Officials 改为 officials. 基于二维超混沌映射方法的 Hash 结果用十六进制数表示如下. 用 0,1 序列的图形化表示如图 3 所示.

文本 1 9DE0CEDCE8A1FAACC7134F6DC149A921

文本 2 31F01FA8C538B0D7E92CD4F2DFF34E57

文本 3 753C37F2B4706406BFEBABA2A023C30A

文本 4 3070C18E9E420531480C8FFCA98D4ACC

文本 5 1D3E3EDF6B532BBB9839E667DDBAF8853

文本 6 363683B5DB98AFED41232B7DA2FFFF63

从以上数据可看出,初值的每 1bit 变动,结果都以很大的概率发生很大的变化,可见该算法具有很好的单向 Hash 性能高度的初值敏感性.

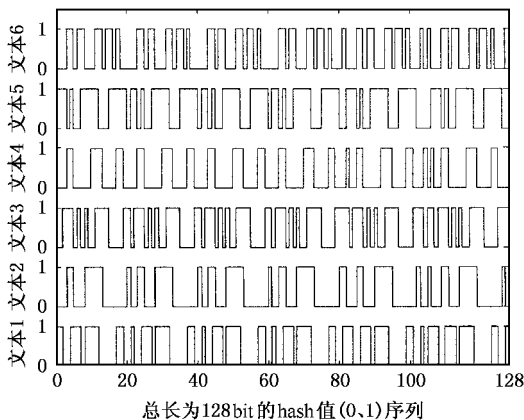


图 3 六个文本的 Hash 值比较

4.2. 混乱与散布统计性质分析

Shannon 提出了混乱与散布的概念来定义隐藏明文消息的冗余度^[23]. Hash 函数和加密体制一样,均要求能充分地利用密文空间,使得 Hash 值尽可能在密文空间均匀的分布. 因此,就要求在 Hash 的过程中,Hash 值与所对应的明文在内容上不相关,却又要求对明文有高度的敏感性. 这对于用二进制表示的结果来说,每 bit 只有 0 或 1 两种可能,因此理

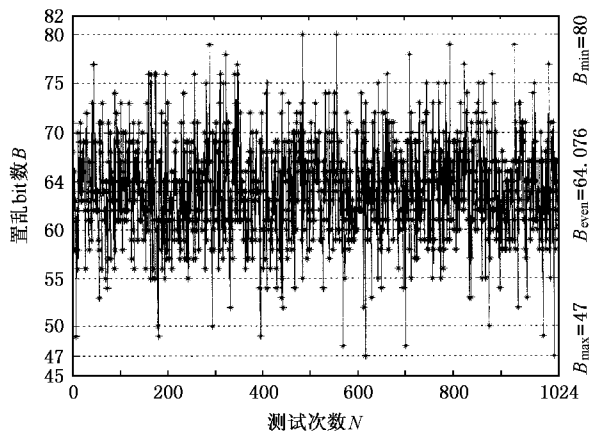


图 4 置乱 bit 数分布图

想 Hash 的散布效果应该是初值的细微变化将导致 Hash 值每 bit 都以 50% 的概率发生变化.

考察算法在明文发生 1bit 变化的情况下,引起 Hash 值的变化 bit 数. 定义:

平均变化 bit 数:

$$\bar{B} = \frac{1}{N} \sum_{i=1}^N B_i, \quad (6)$$

平均变化概率:

$$P = (\bar{B}/128) \times 100\%, \quad (7)$$

B 的均方差:

$$\Delta B = \frac{1}{N-1} \sqrt{\sum_{i=1}^N (B_i - \bar{B})^2}, \quad (8)$$

P 的均方差:

$$\Delta P = \frac{1}{N-1} \sqrt{\sum_{i=1}^N (B_i/128 - P)^2}, \quad (9)$$

其中 N 为统计总次数, B_i 为第 i 次测试时结果的变化 bit 数. 测试方法为: 在明文空间中随机选取一段明文进行 Hash 测试, 然后改变明文 1bit 的值得到另一 Hash 结果, 比较两个结果得到变化 bit 数 B_i . 在 N = 1024 次测试下得到的置乱数分布情况如图 4 所示.

从图 4 可以看出, 测试 1024 次的 128bit Hash 值的置乱 bit 数的分布情况. 其中最小变化 bit 数 $B_{\min} = 47$; 最大变化 bit 数 $B_{\max} = 80$; 平均变化 bit 数为 $B_{\text{even}} = 64.076$, 这和理想状态的变化 bit 数 $B_{\text{ideal}} = 64$ 非常接近. 同时, 经过统计, 有 812 次变换 bit 数在 58 与 70 之间, 占测试总数的 79.297%; 有 120 次变换 bit 数在 70 和 80 之间, 占测试总数的 11.719%; 有 92 次变化 bit 数在 47 和 58 之间, 仅占测试总数的 8.984%. 变化 bit 数绝大部分集中在以 64 为中心的

一个较小的邻域中,这表明了该 Hash 算法具有较强且稳定的置乱能力. 另外,分别对该 Hash 算法进行 256,512,1024 以及 2048 次测试,得到在每变化 1bit 的情况下的平均 Hash 值变化 bit 数 $\bar{B}$ 、变化 bit 数的均方差 ΔB 、平均变化概率 P 以及平均变化概率的均方差 ΔP 的值. 具体如表 2 所示.

表 2 平均 Hash 值变化 bit 数及其相关指标列表

	$N = 256$	$N = 512$	$N = 1024$	$N = 2048$	总平均数
$\bar{B}/\text{bit}$	63.938	64.143	64.076	64.079	64.058
$\Delta B/\text{bit}$	5.251	5.253	5.255	5.253	5.253
$P/\%$	49.951	50.060	50.111	50.062	50.046
$\Delta P/\%$	4.102	4.104	4.105	4.014	4.104

由表 2 可见,该 Hash 算法的 Hash 值总平均变化 bit 数和总平均变化率分别为 64.058 和 50.046%,这和理想状态的 64bit 和 50% 非常接近. 这表明该 Hash 算法充分均匀地在密文空间中取值. 因此,从统计效果来看,即使攻击者已知一些明文与 Hash 值对,也无法帮助其伪造或反推其他明文及其 Hash 值对. 因为明文只要有 1bit 的变化,Hash 值从统计意义上在密文空间中的取值都接近均匀分布,故无法获得 Hash 值分布的有效信息. $\Delta B, \Delta P$ 标志着 Hash 混乱与散布性质的稳定性. 这两个值越接近 0,就表明 Hash 算法越稳定. 文中的 Hash 算法中,总平均 $\Delta B = 5.253$ 、总平均 $\Delta P = 4.104$,这两个数都已很小,从而也可看出基于二维超混沌映射的 Hash 构造算法对明文具有强而稳定的混乱与散布能力.

5. 算法的速度和碰撞分析

从上面 Hash 算法的描述来看,算法的速度与原始报文长度基本成正比的关系. 算法的总迭代步数为 $N_{\text{total}} = r + 3 \times R = R \times ([N/R] + K + 3)$,则 $N + (2 + K) \times R \leq N_{\text{total}} \leq N + (4 + K) \times R$. 其中 K, R 值的选择与所用的混沌映射有密切的关系. 若混沌映射对初始值具有高度的敏感性,则 K, R 的值可选择较小,否则,就应选择较大的数值. 本算法中的所选择的混沌映射与二维超混沌映射均对初值有很高的敏感性,因此 K, R 的数值可以选择较小,从而使得 N_{total} 较小,从而保证算法具有较快的速度. 因此原始文本很短时只需很小的迭代步数. 本算法同时具有一定并行的特点,直接迭代出 6 个大数,每个对应最终结果的 40bit 左右,这就约相当于 5 个字节并行计

算. 更重要的是,算法可以容易地改造为真正的并行,只要将原始文本分割为适当长度的子段,分别迭代,再将结果组成新的序列进行迭代,这在原始文本十分长的情况下,可以成倍地缩短时间.

所谓碰撞,是指不同的初值 Hash 映射结果相同,即发生了多对一映射. 取初始文本为一字节,即包含 8bit,ASCII 码的对应值为 0—255,同样 Hash 值也取为 8bit,也对应为一个 0—255 的数. 这样,初值空间与终值空间相同. 设终值空间(也就是像空间)中任一值对应初值空间中原像的个数记为 k ,记终值空间中具有 k 个原像的点的个数记为 $n(k)$. 如果 $n(1)$ 越大, $n(0)$ 和其他各项越小,表明碰撞越少,Hash 函数的散乱能力越强.

用值域空间与定义域空间的测度之比来定量衡量碰撞发生程度,令:

$$C_e = \frac{256 - n(0)}{256} \tag{10}$$

C_e 的值越接近 1,表示碰撞程度越低;当 C_e 等于 1 时,表示完全没有碰撞发生.

基于二维超混沌映射算法的碰撞分布图如图 5 所示. 图 5 中 $n(0)$ 至 $n(8)$ 依次为 98,91,46,16,4,1,0,0,0, $C_e = 0.617$. 可见,该 Hash 算法的碰撞程度还是较低的. 参数 C_e 的计算与 Hash 值长度密切相关,对于不同的 Hash 值长度,算法的 Hash 性能变化很大, C_e 值变化也较大,故难以进行精确的衡量与预测. 倘若选择在 128bit 等应用尺度上的进行碰撞分析,计算量则过于庞大而在现实无法操作.

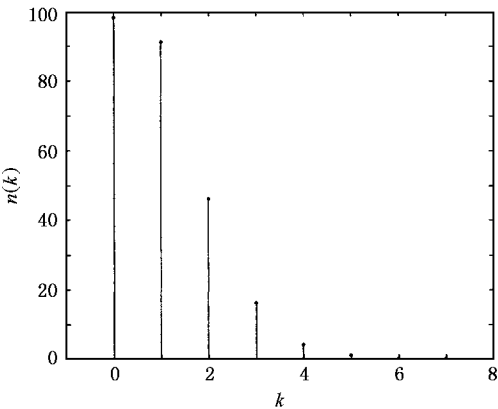


图 5 $n(k) - k$ 分布图

6. 结 论

本文研究了一种基于二维超混沌映射的单向

Hash 函数构造方案. 该方法将原始输入报文按字节进行线性变换以及混沌映射, 在所产生的混沌序列中选值作为二维超混沌映射的初始值, 对迭代后所得的超混沌信号经过调制构造单向 Hash 函数. 研究结果表明: (1) 采用简单的二维超混沌系统使该算法实现简单, 对初值有高度的敏感性, 具有很好的单向

Hash 函数性能; (2) 由于超混沌信号相对于一般的混沌信号或其他简单的组合混沌信号更加复杂, 故基于超混沌信号的单向 Hash 函数在混乱与散布性质上更加均匀, 抵抗攻击的能力更强. 同时, 该方案的迭代步数与待处理报文长度成正比, 并且易于并行实现, 可望成为一种快速有效的单向 Hash 算法.

[1] Weidong Kou 1997 *Network Security and Standards* (Boston : Kluwer Academic Publishers)

[2] Pieprzyh J and Sadeghiyan B 1993 *Design of Hashing Algorithm* (Berlin : Springer-Verlag)

[3] Peter Wayner 1997 *Digital Cash, Commerce on the Net* (Boston : Academic Press)

[4] Liu J N, Xie J C and Wang Pu 2000 *J. Tsinghua Univ.* (Sci & Tech) **40** 55 (in Chinese) [刘军宁、谢杰成、王 普 2000 清华大学学报(自然科学版) **40** 55]

[5] Knudsen L and Preneel B 2002 *IEEE Transactions on Information Theory* **48** 2524

[6] Wang X M, Zhang J S and Zhang W F 2003 *Acta Phys. Sin.* **52** 2737 (in Chinese) [王小敏、张家树、张文芳 2003 物理学报 **52** 2737]

[7] Scott H, Celso G and Edward O 1993 *Phys. Rev. Lett.* **70** 3031

[8] Heileman G L, Abdallah C, Hush D R and Baglio S 1993 *Proceedings of International Symposium on Nonlinear Theory and its Applications* 1183

[9] Zhang J S and Xiao X C 2001 *Acta Phys. Sin.* **50** 2121 (in Chinese) [张家树、肖先赐 2001 物理学报 **50** 2121]

[10] Short K M 1994 *Int. J. Bifur. Chaos* **4** 959

[11] Short K M 1997 *Int. J. Bifur. Chaos* **7** 1579

[12] Zhang J S and Xiao X C 2000 *Acta Phys. Sin.* **49** 403 (in Chinese) [张家树、肖先赐 2000 物理学报 **49** 403]

[13] Zhang J S and Xiao X C 2000 *Acta Phys. Sin.* **49** 1221 (in Chinese) [张家树、肖先赐 2000 物理学报 **49** 1221]

[14] Zhang J S and Xiao X C 2001 *Acta Phys. Sin.* **50** 124 (in Chinese) [张家树、肖先赐 2001 物理学报 **50** 124]

[15] Zhang J S and Xiao X C 2000 *Chin Phys. Lett.* **17** 88

[16] Pecora L M and Carroll T L 1990 *Phys. Rev. Lett.* **64** 82

[17] Yang T and Shao H H 2002 *Acta Phys. Sin.* **51** 742 (in Chinese) [杨 涛、邵惠鹤 2002 物理学报 **51** 742]

[18] Fang J Q 1996 *Science and Technology Review* **4** 6 (in Chinese) [方锦青 1996 科技导报 **4** 6]

[19] Kevin M S 1996 *Int. J. Bifur. Chaos* **2** 367

[20] Cheng L, Tao L, Huang Q N and Peng J H 2002 *J. Northeast Normal Univ.* (Sci & Tech) **34** 47 (in Chinese) [程 丽、陶 路、黄秋楠、彭建华 2002 东北师大学报(自然科学版) **34** 47]

[21] Wolf A, Swift J B, Swinney H L and Vastano J A 1985 *Physica* **16** 285

[22] Li X J, Peng J H, Xu M *et al* 2003 *J. Image and Graphics* **18** 1172 (in Chinese) [李雄军、彭建华、徐 明等 2003 中国图像图形学报 **18** 1172]

[23] Shannon C E 1949 *Bell. Syst. Tech. J* **28** 656

One-way Hash function construction based on two-dimensional hyper-chaotic mappings^{*}

Peng Fei[†] Qiu Shui-Sheng Long Min

(College of Electronic & Information, South University of Technology, Guangzhou, 510640, China)

(Received 8 September 2004 ; revised manuscript received 21 February 2005)

Abstract

The design of an efficient one-way Hash function with good performance is a hot spot in modern cryptography researches. In this paper, a Hash function construction method based on two-dimensional hyper-chaotic mappings is proposed. The initial values of the two-dimensional hyper-chaotic system come from the chaotic sequence, produced by a series of operations including chaotic mappings and linear transformation based on the plain text, and then modulates the signals resulting from the iteration of the hyper-chaotic system to construct a one-way Hash function. Simulation and analysis demonstrate that the new method has the merit of convenience, high sensitivity to initial values, good Hash performance, weak collision property, and better security.

Keywords : one way Hash function, two-dimension hyper-chaos, chaotic mappings

PACC : 0545

^{*} Project supported by the National Natural Science Foundation of China (Grant No. 60372004), the Natural Science Foundation of Guangdong Province, China (Grant Nos. 51445 and 20820)

[†] E-mail : eepengf@yahoo.com.cn