基于 RBF 神经网络和混沌映射的 Hash 函数构造*)

陈 军[1,2] 韦鹏程[1,3] 张 伟[1,3] 杨华千[1,3]

(重庆教育学院计算机与现代教育技术系 重庆 400067)[1]

(西南大学计算机与信息科学学院 重庆 400715)[2] (重庆大学计算机科学与工程学院 重庆 400044)[3]

摘 要 单向 Hash 函数在数字签名、身份认证和完整性检验等方面得到广泛的应用,也是现代密码领域中的研究热点。本文中,首先利用神经网络来训练一维非线性映射产生的混沌序列,然后利用改序列构造带秘密密钥的 Hash 函数,该算法的优点之一是神经网络隐藏混沌映射关系使得直接获得映射变得困难。模拟实验表明该算法具有很好的单向性、弱的碰撞性,较基于传统的 Hash 函数具有更强的保密性且实现简单。

关键词 RBF 神经网络,混沌映射,Hash 函数

Construct Hash Function Based on RBF Neural Network and Chaotic Map

CHEN Jun[1,2] WEI Peng-Cheng[1,3] ZHANG Wei[1,3] YANG Hua-Qian[1,3]

(Department of Computer and Modern Education Technology, Chongqing Education College, Chongqing 400067)[1]

(School of Computer & Information, Southwest China University, Chongqing 400715)[2]

(Department of Computer Science and Engineering, Chongqing University, Chongqing 400044)[3]

Abstract How to design an efficient and security keyed hash function is always the point in modern cryptography researches. In this paper, A better chaotic sequence is generated by the RBF neural network through training the known chaotic sequence generated by a piecewise non-linear, then the sequence is used to construct keyed Hash function. One advantage of the algorithm is that the hidden-mapping model of neural network makes it difficult to get the direct mapping function of the ordinary chaos encryption algorithm. Simulation results show that the keyed Hash function based on the neural network has good one-way, weak collision, better security property and it can be realized easily.

Keywords RBF neural network, Chaotic map, Hash function

1 引言

Hash 函数是从全体消息集合到一个具有固定长度的消息摘要集合的变换,可分为两类[1,2]:带秘密密钥的 Hash 函数和不带密钥的 Hash 函数。不带密钥的 Hash 函数是数字签名中的一个关键环节,可以大大缩短签名时间,在消息完整性检测和操作系统中帐号口令的安全存储中也有重要应用;带密钥的 Hash 函数可用于认证、密钥共享和软件保护等方面[2,3]。Hash 函数具有三个特性:

① 给定的消息 m 和 Hash 函数 H,很容易计算 Hash 值 $h=H(m)$;

② 给定 Hash 值 h,根据 $H(m)=h$ 计算 M 很难(也叫单向性);

③ 给定 m,要找到另一消息 m' 并满足 $H(m')=H(m)$ 很难(也叫抗碰撞性)。

传统的单向 Hash 方法有 MD2、MD5、SHA 等标准,多是采用基于异或等逻辑运算的复杂方法得到 Hash 结果,安全性低,运算量大,难以找到快速同时可靠的加密方法。在文[3~7]中,提出了一些基于混沌的 Hash 函数构造运算,但是这些算法都是基于对某一种非线性影射的设计实现,对混沌影射的参数和状态模拟精度的限制,使混沌序列表现出短周期、强相关以及局部线性的缺点,因此在较小精度实现下的混沌系统不适于构造 Hash 函数,本文基于神经网络的混沌 Hash 算法就是针对此而提出的。

神经网络的非线性、联想记忆、大规模并行分布和强容错性等特点适合于密码通信,只要用集成电路来直接兑现它的并行运算方式。本文利用径向基函数神经网络(RBFNN)来训练已知混沌序列,由此产生的非线性序列来构造带秘密密钥的 Hash 函数,充分利用神经网络的灵活性,在统一的系统结构下,通过变更网络的连接权值就可实现不同混沌系统产生的各种混沌序列,同时将混沌影射关系变为隐式形式,使其更具隐蔽性。这种算法具有对初值有高度敏感性、很好的单向性、弱碰撞性,较基于混沌影射的 Hash 函数具有更强的保密性能,且实现简单。

2 RBF 神经网络对混沌序列的学习

2.1 RBF 神经网络(RBFNN)

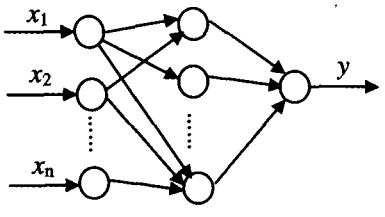

图 1 RBF 神经网络

RBF 径向基神经网络的结构如图 1 所示,是一类结构简单、性能优越、具有局部逼近能力的神经网络。它包含两层:隐层和输出层。设网络的输入节点数为 n,隐层节点为 m,输

*)国家自然科学基金资助项目(60573047)。重庆市科委自然科学基金资助项目(CSTC,2005B2286),重庆市教委资助项目(No. kj051501)。

陈 军 硕士研究生,主要研究方向为信息安全。韦鹏程 博士研究生,主要研究方向为信息安全,混沌理论。张 伟 教授,博士后,主要研究方向为信息安全、计算智能与数据挖掘。杨华千 博士研究生,主要研究方向:信息安全,混沌数字水印。

出节点数为1。

RBFNN网络的输入为：$X=[x_1,x_2,\cdots,x_n]^T$

则网络的输出为：$y_n=f(X_n)=b_0+\sum_{i=1}^{h}w_i\Phi_i(X_i)$

取高斯函数：$\Phi_i(X_n)=\exp(-\|x_n-C_i\|/2\sigma^2)$

2.2 逐段非线性混沌映射特性

如图1所示，具有良好动力学特性的一维非线性映射 f：$I\to I, I_i=[0,1]$，$I_i(i=0,1,\cdots,N)$定义如下[12,13]：

$$F(x_{k+1})=\begin{cases}\left(\dfrac{1}{I_{i+1}-I_i}+a_i\right)(x_k-a_i)-\dfrac{a_i}{I_{i+1}-I_i}(x_k-a_i)^2 & \text{if } x_k\in[I_i,I_{i+1})\\ 0 & \text{if } x_k=0.5\\ F(x_k-0.5) & \text{if } x_k\in(0.5,1)\end{cases} \tag{1}$$

其中，$x_k\in[0,1]$，$0=I_0<I_1\cdots<I_i<\cdots<I_{N+1}=0.5$，$N\geqslant 2$，$a_j\in(-1,0)\cup(0,1)$，$j=0,1,\cdots,N$ 和

$$\sum_{i=0}^{N-1}(I_{i+1}-I_i)a_i=0 \tag{2}$$

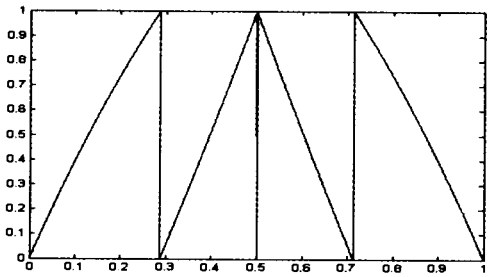

图2 逐段非线性映射

我们发现映射 $F(\cdot)$ 具有如下性质：

i)迭代系统 $x_{k+1}=F(x_k)(k\geqslant 0)$是混沌；ii) 序列$\{x_k\}_{k=1}^{\infty}$在[0,1]区间内是均匀分布的，且分布函数 $f(x)=1$；iii)序列$\{x_k\}_{k=1}^{\infty}$具有 δ 函数自相关函数：

$$R_F(r)=\lim_{j\to\infty}\frac{1}{J}\frac{\sum_{k=1}^{J}x_kx_{k+r}}{\sum_{k=1}^{J}x_k^2},r\geqslant 0 \tag{3}$$

证明：i)因为 $x\in[I_i,I_{i+1})$和 $0\leqslant i\leqslant N$，

$$|F'(x)|=\left(\frac{1}{I_{i+1}-I_i}+a_i\right)-\frac{2a_i}{I_{i+1}-I_i}(x-I_i)>\frac{1}{I_{i+1}-I_i}-a_i>1$$

同时 $F(x)$ 是一个偶对称映射，于是 $|F'(x)|>1$ 得出 $x\in[0,1]$，也就是说，Lyapunov 指数为：

$$\lambda=\lim_{J\to\infty}\frac{1}{J}\log_2\left(\prod_{k=1}^{J}|F'(x_k)|\right)>0 \tag{4}$$

根据混沌系统的定义，Lyapunov指数大于零意味者迭代系统是混沌的，故性质(i)成立。

ii)如(i)一样，当 $x\in[I_i,I_{i+1})$和 $0\leqslant i\leqslant N$ 时

$$|F''(x)|/|F'(x)|^2=2a_i(I_{i+1}-I_i)/[(1+a_i(I_{i+1}-I_i))-2a_i(x-I_i)]^2<2a_i(I_{i+1}-I_i)/(1-(I_{i+1}-I_i)a_i)^2<2/(1-(I_{i+1}-I_i)a_i)^2 \tag{5}$$

注意到 $F(x)$ I 偶对称映射，可得到

$|F''(x)|/|F'(x)|^2<+\infty$对任意 $x\in[0,1]$

根据以上的理论，

$P_r\circ f(x)=\sum|f(y_i)|(I_{i+1}-I_i)(1+a_ix),x\in[0,1]$

其中 P_r 是映射 $f(x)$上的 Frobenius-Perron 算子，在[kohda]中有详细的定义。

$$P_r\circ f(x)=\frac{d}{dy}\int_{F^{-1}([0,x])}f(x)dx \tag{6}$$

其中 $y_i=I_i+(I_{i+1}-I_i)(1+a_iI_{i+1}+a_iI_i-2a_ix)$。

根据等式(2)，$f(x)=1$ 是方程 $f(x)=P_r\circ f(x)$的唯一解，故性质(ii)成立。

iii)正如在[kohda]中定义一样，我们重写自相关函数：

$$\rho_F(r)=\frac{\int_0^1xF^r(x)f(x)dx}{\int_0^1x^2f(x)dx} \tag{7}$$

当 $r=0$，$F^r(x)=x$，$\rho_F(r)=1$；根据对称轴 $x=0.5$，当 $r>0$，x 是奇数和 $F^r(x)=x$ 是偶数于是性质(iii)成立。

2.3 产生混沌序列的RBFNN(CS-RBFNN)

RBF网络具有良好的逼近任意非线性影射能力，因此该网络通过对混沌序列的学习和建模可具有混沌性态。基于RBF神经网络的混沌序列产生模型系统结构如图3所示。网络连接权值和初值数据库用来存储由学习样本训练好的网络权值和相应的初值，由网络的输出至输入端的反馈形成闭环结构，使输出的混沌序列反馈至输入端，作为下次输出序列的初始值，从而可以源源不断地输出混沌序列。最后将网络产生的模拟混沌序列转化成二进制混沌序列。

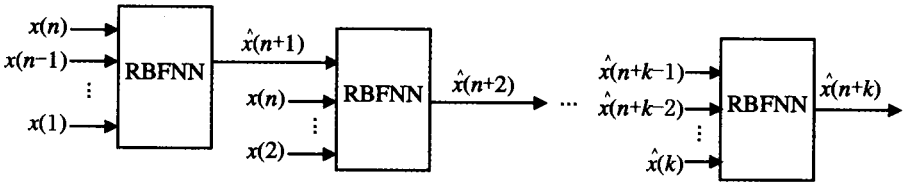

图3 RFB神经网络训练混沌序列模型

考虑时间序列 $x(1),x(2),\cdots,x(n)$需要预测 $x(n+1)$，则可以将 $x(n-k+1),\cdots,x(n)$个值作为神经网络的输入，输出即可 $\hat{x}(n+1)$(预测值)。模型的第1步骤是将 $x(1),x(2),\cdots x(n)$，n 个符合一维非线性影射的序列作为初值馈入神经网络以产生新的序列：$\hat{x}(n+1),\hat{x}(n+2),\cdots,\hat{x}(n+m)$，这就归纳为如何用 $x(1),x(2),\cdots,x(n)$来预测第 k 步：$\hat{x}(n+k)$的问题。

我们采用递归的方法：每次只预测一步，即 $\hat{x}(n+k)$，再不断加上预测出的值作为 RBFNN 的是输入，得到下一步的值，如此循环，最终得到需要的所有值。

3 Hash函数的构造

用RBFNN构造的带秘密密钥的Hash函数构造如图4所示，描述如下：

(a)假设消息 M 是二进制序列，Hash值的长度为 $N(N=128*i,i=1,2,\cdots)$，若M的长度$|M|$不是 N 的整数倍，可以连接适当的随机序列使之满足要求。将 M 按长度 N 分组，记 $M=M_1,M_2,\cdots M_t$ 并且 $M_i=M_i^1M_i^2\cdots M_i^N$。

(b)选取随机序列N作为初始向量 H_0，计算 $H_0\oplus M_1$。

(c)以 $x(0)$及神经网络的参数作为秘密密钥，通过CS-

RBFNN 训练，得到一个长度为 N 位的伪随机序列，然后对 $H_0 \oplus M_1$ 进行加密，然后转化为二进制序列作为 M_1 的 Hash 值 H_1，然后计算 $H_1 \oplus M_2$，再对 $H_1 \oplus M_2$ 进行加密得到 H_2。重复上述过程至消息结束，得到 M 的 Hash 值 H_k。

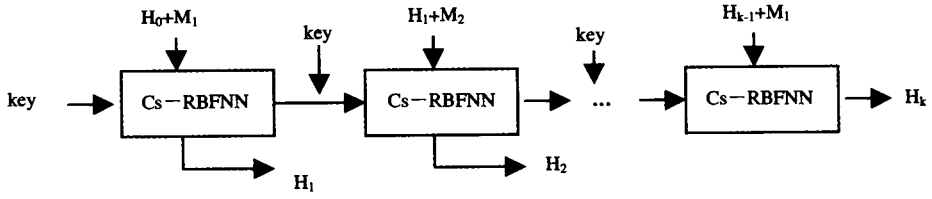

图 4　Hash 算法结构

4　仿真实验

4.1　单向性分析

单向性意味着仅仅能从消息得到 Hash 值，而要从 Hash 值得到原始信息是很困难的[8,10]。清晰的描述这一特性，我们对如下明文 1 分别进行 Hash 实验：

Cryptologyist the science of overt secret writing(cryptography), of its authorized decryption(cryptanalysis), and of the rules which are in turn intended to make that unauthorized decryption more difficult(encryption security).

为了形象地展示明文与 Hash 值之间的区别，我们使用二维图形来表达。从图 5 的明文的 ACII 值分布图来看，ASCII 码值比较集中，其码值主要分布在一个较小的范围内；从图 6 的十六进制 Hash 值可以看出，经过 Hash 函数进行运算后，Hash 值的十六进制值却非常分散而且分布很均匀。也就是说，通过扩散、扰乱等作用后，Hash 值中不包含消息的任何信息(包括消息的统计概率信息)，这正是我们想要达到的 Hash 效果。

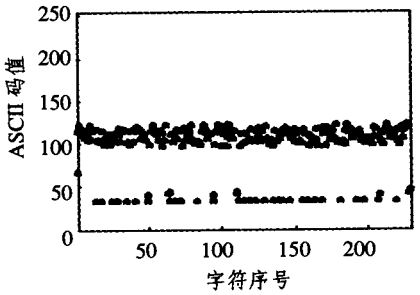

图 5　明文的 ASCII 分布图

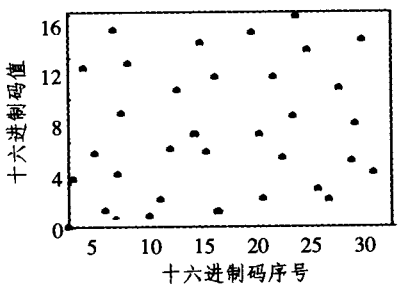

图 6　Hash 值十六进制分布图

4.2　碰撞分析

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[3,5]。取初始文本为一字节，即 8bit，ASCII 码对应值为 0～255，Hash 结果取为 8bit，即也为的 0～255 数，这样初值空间与终值空间相同。记终值空间即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 $n(k)$，$n(1)$ 越大，$n(0)$ 和其他各项越小，说明碰撞越少，混沌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

$$L=\frac{256-n(0)}{56} \tag{8}$$

L 的值越接近 1，碰撞程度越低，等于 1 时，完全没有碰撞发生。

基于 RBF 神经网络和混沌影射的 Hash 函数的碰撞分布图如图 7 所示。在图 7 中，$n(0)$ 至 $n(8)$ 依次为：78，96，58，6，1，0，0，0，0，$k>8$ 均为 0，$P=0.6953125$。可见，该算法的碰撞程度较低。参数 L 的计算目前较难与其他算法比较，因为其他算法的设计多与结果长度相关，改变结果长度 Hash 性能变化难以预测，相当于重新设计算法，而在 128bit 等应用尺度上的碰撞分析，计算量过于庞大而不现实。而本算法可以将结果取为任意长度，从而可以方便地在小尺度下进行算法碰撞程度的定量分析，这正是一个独特的优点。

4.3　Hash 值对密钥的敏感依赖

基于 RBF 神经网络和混沌影射的 Hash 函数结构对密钥是非常敏感的[12,13]。使用微小差异的密钥进行 Hash 运算，就会产生截然不同的 Hash 值，这正符合定义中要求。图 8 是把密钥做 10^{-3} 改动，通过 Hash 运算后得到的十六进制值分布图，从图 5 和图 8 可以看出，密钥的微小改变，结果都是以很大概率发生变化，得到的 Hash 值截然不同，可见该算法单向性好，并且具有十分高的初值敏感性。

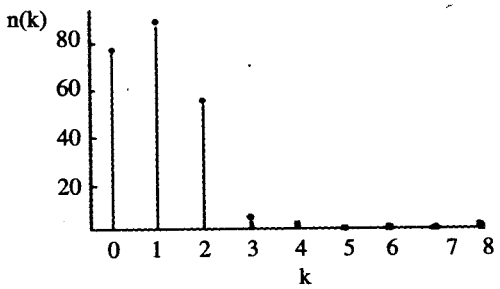

图 7　$k-n(k)$ 分布图

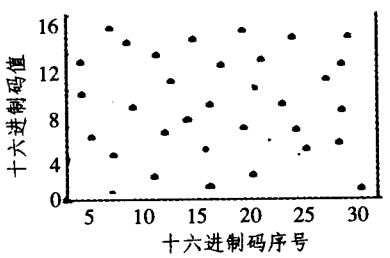

图 8　密钥微小失配时的 Hash 值十六进制分布图

结论　基于 RBF 神经网络和混沌影射的 Hash 函数的算法简单、速度快，而且算法利用了离散混沌系统对初始条件的敏感性和迭代过程的单向性，使 Hash 值的每个 bit 都与消

息 M 有关，并且这种关系对消息及密钥的微小改变很敏感。由于迭代过程将使初始值之间的差异不断放大，经过第一轮的迭代将会使差异大到足以影响 Hash 值，因此，对同一个消息用不同的密钥，将得到完全不同的 Hash 值，Hash 值与消息之间的这种复杂而敏感性的非线性关系可以有效地抵抗线性分析；由于具有很大的密钥空间，可以有效抵抗穷举攻击；神经网络和混沌序列有机结合使得混沌理论的攻击方法和传统密码的分析方法变得困难，从而使 Hash 算法具有很高的安全性。

参 考 文 献

1 李红达，冯登国. 复合离散混沌动力系统与 Hash 函数[J]. 计算机学报，2003，(26)4：21～26
2 冯登国，裴定一. 密码学导论[M]. 北京：科学出版社，1999
3 刘军宁，谢杰成，王普. 基于混沌映射的单向 Hash 函数构造[J]. 清华大学学报(自然科学版)，2000，(40)7：55～58
4 Pieprzyh J，Sadeghiyan B. Design of Hashing Algorithm. Berlin [J]：Springer-verlag，1993
5 王小敏，张家树，张文芳. 基于广义混沌映射切换的单向 Hash 函数构造[J]. 物理学报，2003，11(52)：37～43
6 Wong K-W. A Combined Chaotic Cryptographic and Hashing Scheme [J]. Physics Letters A，2003，(307)307：292～298
7 Bakhtiari S，Safavi-Naini R，Pieprzyk J. Keyed Hash Function [J]. Lecture Note in Computer Science，1996，1029：201～204
8 Zhang W，Peng J，Yang Huaqian，Pengcheng Wei. A Digital Image Encryption Scheme Based on the Hybrid of Cellular Neural Network and Logistic[A][C] Map. Advances in Neural Networks-ISNN2005，Chongqing，China，May/June 2005 Proceedings，Part Ⅱ，2005. 860～867
9 Xiao D，Liao X F，Deng S J. One-way Hash Function Construction Based on the Chaotic Map with Changeable-parameter [J]. Chaos Solitons & Fractals，2005，24(1)：65～71
10 Sang Tao，Wang Ruli，Yixun Yan. Generating Binary Bernoulli Sequences Based on a Class of Even-Symmetric Chaotic Maps [J]. IEEE Trans on Communications，2001，4(49)：620～623
11 Hu Guojie，Feng Zhengjin. Theoretical Design for a Class of New Chaotic Stream Cipher [J]. Communications Technology，2003，4：73～74
12 Li Shujun，Mou Xuanqin，Cai Yuanlong. Improving Security of a Chaotic Encryption Approach. Physics Letters A [J]，2001，290(4)：127～133
13 Li Shujun，Mou Xuanqin，Cai Yuanlong. Pseudo-random Bit Generator Based on Couple Chaotic Systems and its Application in Stream-ciphers[A][C]Cryptography. In：Progress in Cryptology-INDOCRYPT 2001，Lecture Notes in Computer Science，2001，(2247)：316～329

(上接第 197 页)

表 2 GB_MGA 算法与 Ge_GA 算法时间的比较

TSP 实例	Ge_GA		GB_MGA	
	最好时间(s)	最差时间(s)	最好时间(s)	最差时间(s)
Krob200	1	2	0.280	0.982
a280	0.5	2	0.538	1.002
fl417	9	51	3.157	30.593
d493	3	13	3.651	7.238
att532	6	19	8.590	21.154
gr666	6	20	9.137	18.300
vm1084	8	593	26.356	112.795
fl1577	734	6305	184.101	431.483

实验结果证明，该文算法的求解质量要优于 GA、PGA、MMGA 算法，而求解速度方面则优于 Ge_GA 算法，特别是对于大规模城市的 TSP 问题求解效果尤其明显，具有快速收敛的特性。Ge_GA 算法对于中等城市规模的 TSP 实例求解，其运算时间就大幅度增加，如果把该算法用于求解大规模和超大规模 TSP 问题，那么时间上的代价就让人无法忍受。而该文的 GB_MGA 算法在单亲遗传演化中就使用了基因库中的优质基因，使得单个个体的进化速度大大提高，从而为进一步的演化提供了条件，群体演化过程的选择机制和收敛准则的恰当选取使得算法在注重了求解质量的同时兼顾了算法的效率。

结束语 该文在对 TSP 问题进行分析的基础上，通过对全局最优解和局部最优解中的边关系的分析发现，通过最小生成树的求解保存最有希望成为全局最优解的边，可以提高算法的效率，同时并不降低搜索的性能。在实验中发现，通过生成基因库快速提高种群质量，虽然可以快速收敛，但是 TSP 问题的求解质量并没有达到一个可以接受的程度，因此在群体演化阶段中又加入了 2-Opt 局部搜索算子和多重搜索策略，对不同类型的染色体以不同几率进行选择交叉操作。用该算法测试 TSP 库中的典型实例，结果显示，对初始种群运用单亲遗传算法，并引入基因库方法是可行的，有效地提高了算法的效率和收敛速度，在群体演化过程中引入多重搜索策略的方法，提高了算法的并行性和全局寻优性能，即使在较少的寻优步数也能得到适应度不错的局部优化解。GB_MGA 算法在提高算法求解质量的同时兼顾了算法的效率，但是其中还有许多有待解决的问题，比如如何构建高质量的基因库，如何对现有基因库进行优化和扩充，以及算法中一些参数的选取问题等，这些方面，还需要进一步的研究。

参 考 文 献

1 Bck T B，Hammel U，Schwefel H P. Evolutionary computation：Comments on the history and current state [J]. IEEE Transactions On Evolutionary Computation，1997，2(6)：3～17
2 王磊，潘进，焦李成. 免疫算法[J]. 电子学报，2000，28(7)：74～78
3 Dorigo M，Gambardella L M. Ant Colony System：A Cooperative Learning Approach to the Traveling Salesman Problem [J]. IEEE Transactions on Evolutionary Computation，1997，2(8)：53～66
4 Holland J H. Adaptation in Natural and Artificial Systems [M]. Ann Arbor：The University of Michigan，1975. 10～15
5 杨辉，康立山，陈毓屏. 一种基于构建基因库求解 TSP 问题的遗传算法[J]. 计算机学报，2003，26(12)：1753～1758
6 Tsai Cheng-Fa，Tsai Chun-Wei，Yang Tzer. A Modified Multiple-Searching Method to Genetic Algorithms for Solving Traveling Salesman Problem [J]. IEEE Transactions on Systems，Man and Cybernetics，2002，3(10)：6～12
7 Baraglia R，Hidalgo J I，Perego R. A Hybrid Heuristic for the Traveling Salesman Problem [J]. IEEE Transactions on Evolutionary Computation，2001，5(12)：613～622
8 Tsai Cheng-Fa，Tsai Chun-Wei. A New Approach for Solving Large Traveling Salesman Problem Using Evolutionary Ant Rules [C]. In：Proc. of the 2002 International Joint Conf. on Neural Networks，Hawaiian：honolulu，2002
9 Jung S，Moon B R. Toward Minimal Restriction of Genetic Encoding and Crossovers for the Two-Dimensional Euclidean TSP [J]. IEEE Transactions on Evolutionary Computation，2002，6(12)：557～565
10 李茂军，童调生，罗隆诵. 单亲遗传算法及其应用研究[J]. 湖南大学学报，1998，25(6)：56～59
11 陈国良，王煦法，庄镇泉，等. 遗传算法及其应用[M]. 徐修存，王晓丹. 北京：人民邮电出版社，1996. 75～81
12 Watson J，Ross C，Eisele V，et al. The Traveling Salesrep Problem，Edge Assembly Crossover，and 2-opt [C]. Parallel Problem Solving from Nature V，Netherlands：Amsterdam，1998
13 Burkowski F J. Proximity and priority：applying a gene expression algorithm to the Traveling Salesperson Problem [J]. Parallel Computing，2004，30(5-6)：803～816