遗传算法中排列问题的编码研究

韩建枫　李敏强　寇纪淞
（天津大学系统工程研究所，天津 300072）
E-mail：hjf@shannon.com.cn

摘　要　针对排列问题的编码方法一直是遗传算法应用中的重要研究领域。采用各种传统编码方法的编码表示空间通常远远大于实际的问题空间，这不但提高了各算子设计的复杂性，同时很大程度上降低了收敛速度。文章提出了一种针对排列问题基于次序的一维二进制编码方案和两种改良方案，使排列与编码形成了一一映射，最大限度地缩小了编码表示空间与问题空间的差距。采用 TSP 问题的实验结果表明，文章提出的编码方式具有很好的性能。

关键词　排列问题　次序　编码　遗传算法　TSP

文章编号 1002-8331-（2002）12-0029-04　**文献标识码** A　**中图分类号** TP301.6

Encoding Schemes of Array Problem in Genetic Algorithms

Han Jianfeng　Li Minqiang　Kou Jisong
（Institute of Systems Engineering，Tianjin University，Tianjin 300072）

Abstract：Encoding of genetic algorithm（GA）about array problem is an important problem in GA's application.The size of representation space by using traditionary encoding schemes is far great than the size of problem space，which not only increase complexity of genetic operators but also expend generations much more.A kind of new one dimension binary encoding scheme and its two improved encoding schemes based on order of array are presented for solving the array problem，using in such as Traveling Salesman Problem（TSP）or Scheduling Problem et.The difference size between of representation space and problem space is reduced.The new encoding schemes accelerate search process in problem space by reducing encoding space.The experimental results show that the new encoding scheme has great advantage of speed over other encoding schemes.

Keywords：Array problem，Order，Encoding，Genetic Algorithm（GA），Traveling Salesman Problem（TSP）

1　引言

遗传算法[1]（GA）是一类用机器模拟生物界自然选择和自然遗传机制来求解复杂问题的随机搜索和优化方法。应用 GA 求解时，首先要把实际问题的有关参数和可能解集（问题空间，记为 Ω）转换成 GA 空间的代码串，该代码串能够表示的全体编码组成编码表示空间（简称表示空间，记为 ψ），这一转换操作称为编码（coding）；当用 GA 方法求得满意解后，再将其对应的代码串转换为实际问题的解，这一转换操作称为解码（decoding）。从数学角度看，编码为 Ω 到 ψ 的映射，解码为其逆映射。显然[2]，编码与解码应符合完备性（completeness）、健全性（soundness）、非冗余性（nonredundancy）三级标准。同时，考虑 GA 算子的特性，该映射应尽量保持两空间中的距离一致性，即 Ω 中相近解（需根据不同类型问题加以定义）映射到 ψ 后应具有同样相近的距离（通常用海明距离加以度量）。作为 GA 应用的第一步，编码方案决定了各种遗传算子的设计方式和复杂性，因此始终成为 GA 研究的热点问题，在迄今为止的历届 ICGA 中，研究编码表示的论文均在 10 篇以上。

NP 问题是 GA 主要应用的领域之一[1]，排列问题因为具有比指数复杂性更甚的 $n!$ 复杂性，所以为典型的 NP 问题。排列问题具有广泛的应用代表性，如在 TSP 问题中 Ω 是 N 个城市的全排列空间；调度问题中某设备上任务排列子问题的 Ω 是 N 个该设备上任务集合的全排列空间等。因此，对于排列的编码研究是利用 GA 解决上述问题的关键，目前常见的有一维显式编码[3]、序表示编码[4]、基于次序对或边关系的编码[5]、间接"节点"编码[3]、基于矩阵类的编码表示[11,12,9]等。其中，一维显式编码采用十进制整数编码形式，ψ 空间大小为 n^n 而 Ω 空间为 $n!$，ψ 远远大于 Ω 且存在大量非法编码，因此需要设计复杂的交叉算子、选择算子，利用预处理函数、惩罚函数、过滤函数、转换函数等技术消除 ψ 中的非法解，而这些操作将导致随机性的下降、算法性能降低，一定程度上影响甚至抵消了 GA 自身的优势；序表示编码和基于次序对关系编码虽然通过巧妙的编码定义消除了 ψ 中的非法部分，但同基于矩阵类的编码表示一样，面临着 Ω 到 ψ 的映射为一对多映射，ψ 仍远大于 Ω 且算子设计复杂等问题；间接"节点"编码由于其特征遗传性较差，因此很少采用[10]。由于编码方式的限制，此类 GA 算法的改良仍以算子设计为主[3-8]，其执行性能依旧受到限制。

该文针对上述编码方式的不足，提出了一种基于有序排列编号的从 Ω 到 ψ 的二进制编码方法，并针对其海明距离给出了两种进一步优化编码策略。

基金项目：国家自然科学基金项目资助（编号：69974026）
作者简介：韩建枫，男，1970 年生，博士研究生，主要研究方向为遗传算法和并行调度算法。李敏强，1965 年生，教授，博士生导师，主要研究方向为遗传算法，多目标决策。寇纪淞，1947 年生，教授，博士生导师，主要研究方向为计算机应用，遗传算法。

2 排列问题的一维线性编码

2.1 线性权重编码(LR 码)

为便于说明,现以 4 个事件的全排列为例,说明编码过程。不失一般性,设该 4 个事件分别用字符“1”、“2”、“3”、“4”表示。则其全排列的个数为 24 个,如表 1 所示。首先,任意设定参与全排列事件的一维逻辑次序,该次序可以没有任何意义;也可以根据具体问题赋予特定意义并以权值形式量化表示,如重要性、执行时间、规模等,再配合如重要优先、短任务优先等基本次序原则进行排序。同样不失一般性,这里设定该例中该次序为“1”-“2”-“3”-“4”。

定义 1:参与全排列的每个事件具有的“事件权重”,为该事件在一维逻辑次序中从前到后,所处的位数。

定义 2:在某个排列中,排列中的每一位具有的“排列位权”,为该位在排列中从后到前所处的位数 * 以全排列中事件总个数+1)为进制的该位的权。

定义 3:在某个排列中,排列中的每一位具有的“排列实例位权”,为该位的排列位权 * 该位事件的事件权重。

定义 4:在全排列中,每个排列的“排列权重”,为各位的排列实例位权和。

根据定义,上例中事件“1”、“2”、“3”、“4”的事件权重分别为 1、2、3、4;(,*,*,*)的排列位权分别为(25,25,5,1);全排列中,排列“3124”的各位排列实例位权为(*125,1*25,2*5,4*1);排列“3124”的排列权重为 3*125+1*25+2*5+4*1。表 1 显示了各排列的排列权重。

定义 5:某个 n 事件排列的线性权重编码(LR 码)为该排列在 n 个事件全排列中按排列权重不减排序时次序号所对应的编码,分别可用十进制、二进制、格雷码表示。

根据以上定义,4 事件十进制 LR 码如表 1 所示。算法 1、2 分别给出了任意事件排列的十进制 LR 码的编、解码算法。由于参与的排列事件数量可以任意,所以在算法中排列事件元素可以根据情况定义为 1 个字符、字符串或结构等。相应格雷 LR 码可在该算法的基础上附加格雷码变换算法,该文从略。

定理 1:具有 n 个事件(n>0)的全排列空间 Ω 与其十进制 LR 码空间 ψ 构成双射。

证明:设排列权重空间为 A,根据定义 1–4 建立的 Ω 到 A 间的对应关系为 R1,根据定义 5 建立的 A 到 ψ 间的对应关系为 R2。

(1)因为 n>0 及定义 4,所以 Ω、A 为非空集合;对于 Ω 中不同元素,根据定义 2,可知排列位权即为进制转换中的位权定义规则,又根据定义 3、4 及进制公理,可知在 A 中有唯一元素与之对应。由上可知,R1 为映射。

(2)根据定义 4,A 中每个元素必有 Ω 中元素与之对应,又根据进制公理,其必唯一。所以,R1 为双射。

(3)R2 为双射(证明从略)。

(4)根据双射传递律,R1(R2)为双射,命题得证。

定理 2:具有 n 个事件(n>0)的全排列空间 Ω 与其二进制 LR 码、格雷 LR 码空间 ψ 构成双射。

证明:(从略)。

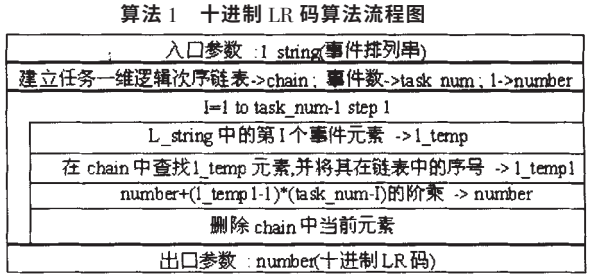
2.2 线性折中编码

根据格雷码定义可知,相临数值的格雷码海明距离恒为 1。因此,根据定义 5 可知,在 GA 中衡量线性编码海明距离的指标可由衡量相应十进制相临数值编码所对应的排列海明距离来说明。

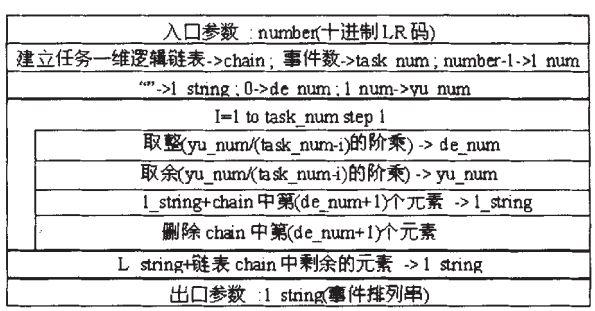
表 1 四事件线性权重、折中、海明十进制编码表

事件排列	排列权重	线性权重编码	线性折中编码	线性海明编码
1234	194	1	1	1
1243	198	2	2	2
1324	214	3	4	4
1342	222	4	3	3
1423	233	5	5	5
1432	241	6	6	6
2134	294	7	7	12
2143	298	8	8	11
2314	334	9	10	9
2341	346	10	9	10
2413	358	11	11	8
2431	366	12	12	7
3124	414	13	13	13
3142	422	14	14	14
3214	434	15	16	16
3241	446	16	15	15
3412	482	17	17	17
3421	486	18	18	18
4123	538	19	19	24
4132	542	20	20	23
4213	558	21	22	21
4231	566	22	21	22
4312	582	23	23	20
4321	586	24	24	19

算法 1 十进制 LR 码算法流程图

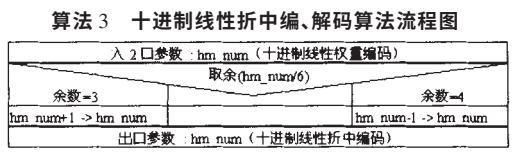


算法 2 十进制 LR 码算法流程图



定义 6 在与 n 事件全排列空间 Ω 构成双射的线性编码空间 ψ 中,平均海明距离为 ψ 全部(n!–1)对相临编码所对应排列的海明距离总和除以(n!–1)。

算法 3 十进制线性折中编、解码算法流程图



由于排列中元素的非重复性质决定了当元素个数大于 1 时,不同排列间海明距离的最小值为 2。因此,当 n>1 时,ψ 的平均海明距离应>=2。LR 码的平均海明距离如表 2 所示。通过对 Ω、ψ 的分析,该文针对十进制 LR 码提出了一种时间复杂度

仅为 O（）的简单变换，其十进制编码通过在原算法 1、2 后面加上算法 3 实现，再变为二进制码和格雷码，可明显降低平均海明距离。LR 码严格体现了排列权重的增序性质，经算法 3 转换后，实现了权重有序和平均海明距离之间的折中；同时，该算法采用模 6 后折中数字交换方法实现，因此这里称该编码为线性折中编码，简称 LTO 码。

定理 3：具有 n 个事件（k>0）的全排列空间 Ω 与其十进制、二进制、格雷 LTO 码空间 ψ 构成双射。证明：从略）。

表 2　三种线性编码方法平均海明距离对比表

n=	2	3	4	5	6	7	8	9	10
线性权重	2	2.4	2.5217	2.605	2.6231	2.6283	2.6291	2.6292	2.6292
线性折中	2	2	2.1739	2.2689	2.2893	2.2949	2.2957	2.2959	2.296
线性海明	2								

2.3　线性海明编码

对于很多全排列情况，维持其权重有序没有意义，在 GA 应用中更多应在编码过程中保持其平均海明距离最小化。在对按照十进制 LTO 码排序的排列序列分析过程中，发现当 n>3 时保持平均海明码距离恒为 2 的如下规律：

（）将全排列空间按首位分段，第一段正序、接第二段反序、接第三段正序……；

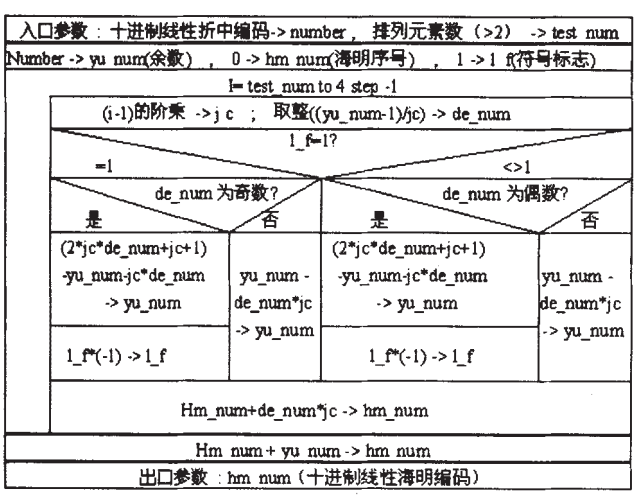
（）递归对第 i 段再按次位分段，重复上述步骤；

（）重复上面步骤，直到每一段长度为 6 为止。注意：不同段间衔接与其嵌套的反序层数相关。

这里将符合上述规律的针对排列的线性编码规律称为线性海明编码，简称 LHM 码。其十进制编码过程为算法 1、3 后面加上算法 4 实现，解码过程为算法 2、3 后面加上算法 4 的逆（略）实现。同上，LHM 码可分为十进制、二进制、格雷码等三种编码形式，该编码在已经测试的范围内满足平均海明距离为 2 的规律。

定理 4：具有 n 个事件（k>0）的全排列空间 Ω 与其十进制、二进制、格雷 LHM 码空间 ψ 构成双射。证明：从略）。

算法 4　十进制线性海明编码算法流程图



3　编码分析

为进一步分析在 GA 应用过程中不同编码方案的优劣，这里从编码长度、编解码时间复杂性等方面对不同编码方案进行了比较。

3.1　编码长度与表示空间

由于上面三种二进制编码方案实现了排列全空间与编码的双射且编码步长为 1，因此，其二进制编码长度必为最小值。表 3 显示了几种针对排列问题的二进制编码方案对应的编码长度情况。

表 3　几种常见的排列问题二进制编码方案编码长度对照表

编码方案	N（）元素全排列的编码长度	n=150 时，等价长度二进制编码
该文提供的三种编码方案	int（$\log_2$（!））+1	872
基于次序对或边关系的编码方案[5]	n*（−1）/2	11175
一维显式编码（近邻表示）[3]	n*（int（$\log_2$（）+1）	1200

3.2　编/解码时间复杂性

根据算法 1–4 所示，对于具有 n 个元素的全排列，每次线性权重、折中、海明编/解码的时间复杂性均为 O（）。基于次序对或边关系的编码方案[5]时间复杂性为 O（^{2}），基于矩阵类的编码[11,12,9]时间复杂性为 O（^{2}）。

3.3　海明距离

该文所示三种编码方式的平均海明距离如表 2 所示，根据 n<15 的有限次测试，线性权重平均海明距离接近 2.6293；线性折中平均海明距离接近 2.2961。其他编码方案由于没有实现双射或编码步长无定义，因此平均海明距离无定义，不可比。

3.4　相关算子分析

虽然该文提出的三种编码方式实现了从问题空间到编码空间的双射，但由于二进制编码的限制，因此编码空间与编码表示空间还存在着一定差异。其中：编码空间体积为 n!；编码表示空间体积为 $2^{\text{int}(\log_2(!))+1}$；表示空间中具有 $2^{\text{int}(\log_2(!))+1}-n!$ 个盲点。对于交叉或变异后表示空间的个体编码值 x 可采用如下取阀（!）模分段函数进行预处理：

$$x=\begin{cases} x & (x<=n!) \\ \text{int}(!)*((x-n!)/(2(\text{int}(\log_2(!)))+1-n!))) & (x>n!) \end{cases} \quad (\ )$$

其中，比较与减法操作可映射到十进制域或二进制域中进行。

对于该文提出的三种编码方式，交叉算子可采用普通单点或多点交叉，实现简单且可保证宏观搜索性能；由于 LR 码有稳定的权重次序、LHM 码有稳定的海明距离、LTO 码在微观上兼顾了权重与海明距离，因此变异算子可采用随机加/减[1,k]方式（k 为最大变异度，一般取值小于 n）即可保证得到类似爬山或模拟退火等算法所具有的较高的微观搜索性能。

4　实验结果

由于该文参照的诸多编码方案很多源自求解 TSP 问题，因此，笔者针对 TSP 问题选用了两种典型传统编码和文中介绍的 LR 码进行了实验比较。采用 TC 编译器，城市数为 50–150，硬件配置为 P133，24M RAM，240M 虚拟内存，软件执行平台为 WIN98 的 DOS-SHELL，5 次实验取均值。三种实验中的算法取相同的入口参数：种群规模 80；交叉概率 0.9；变异概率 0.005；精英策略。为了更有力地说明不同编码方式对进化速度的影响，采用的终止条件为“当种群内最佳值达到参考最佳值的 1.3 倍之内的 20 代后终止进化”。

参考最佳结果采用 Stein [13] 的理想解估算的经验公式 L=0.765*SQR（N*S）得到，其中：0.765 为经验值、N 为城市数量、S 为城市随机分布区域的面积，以 max（,y）为边的正方形）。“*”表示由于设备所限缺少实验结果。针对该文提出的编码规则，为克服某些编程语言采用的科学记数法的保留限制，笔者开发了针对二进制字符串的加，减，乘，除，取模，阶乘，比较等专用

函数。

表 4　三种编码方案解 TSP 问题的实验结果比较

问题规模	参考最佳结果	一维显示编码		基于次序对或边关系的编码方案[5]		线性权重编码	
		最佳值	时间 s	最佳值	时间 s	最佳值	时间 s
50	530.118	601.2816	25	585.942	72	588.0048	44
60	574.789	698.7492	30	667.3548	168	685.596	73
70	627.244	799.0188	38	732.5172	410	773.5824	115
80	670.552	877.0536	90	850.3548	925	827.37	179
90	696.713	945.5652	165	904.0752	2235	936.5904	281
100	757.35	1003.1484	288	986.4792	5029	994.248	424
110	786.292	1095.9732	558	1064.6304	12405	1036.421	664
120	829.635	1124.5236	998	*	*	1053.925	1018
130	846.067	1233.0648	1948	*	*	1071.91	1621
140	878.005	1253.388	3677	*	*	1149.356	2429
150	936.93	1220.0388	8131	*	*	1186.698	3644

实验结果说明：在要求完成一定优化目标时，三种方法的时间相差较大，在问题规模稍大情况下，基于次序编码方案的耗时几乎无法承受，而较一维显示编码而言，LR 码有较明显优势；在问题规模较小时，LR 码在时间上较一维显示编码有一定不足。

5　结束语

利用 GA 算法在全排列空间范围内进行优化、选择是 GA 应用的重要领域，文章针对此类问题，通过最大限度地压缩编码表示空间而提出了一种编码方式及其两种改良策略。理论分析及实验结果表明，在求解搜索空间较大问题时的 GA 应用中，该文所提编码方式较同类编码具有一定时间与空间优势，下一步工作将侧重针对此类编码的高效算子研究。

（收稿日期：2002 年 4 月）

参考文献

1.Hnoll J H.Adaptation in Natural and Artificial Systems[M].Univ of Michigan Press，1975

2.Goldberg D E.Genetic Algorithm in Search，Optimization and Machine Learning[M].Addision-Wesley，1989

3.Grefenstette J J et al.Genetic Algorithms for the Traveling Salesman Problem[C].In：Proceedings of an International Conference on Genetic Algorithms and Their Applications，1985：160~168

4.Back T.Selective Pressure in Evolutionary Algorithms：A Characterization of Selection Mechanisms.1994：57~62

5.于达.基于 petri 网模型的任务调度问题研究[D].博士学位论文.清华大学，1996：80~81

6.Goldberg D E，Lingle R Aleles.The Traveling Salesman Problem[C].In：Proceedings of an International Conference on Genetic Algorithms and Their Applications，1985：154~159

7.Davis L.Job Shop Scheduling with Genetic Algorithms[C].In：Proceedings of an International Conference on Genetic Algorithms and their Applications，1985：136~140

8.Ulder N L J et al.Genetic Local Search Algorithm for the Traveling Salesman Problem.1991：109~116

9.Seniw D.A Genetic Algorithm for the Traveling Salesman Problem.MSc Thesis，University of Norith Carolina at Charlotte，1991

10.陈国良等.遗传算法及其应用[M].人民邮电出版社，1996

11.Fox b r，McMahon m b.Genetic Operators for Sequencing Problems.1991：284~300

12.Homaifar A，Guan S.A New Approach on the Traveling Salesman Problem by Genetic Algorithm.1993：460~466

13.Stein D.Scheduling Dial a Ride Transportation Systems：An Asymptotic Approach[D].Ph D Dissertation.Harvard University，1977

（上接 11 页）

```
vulnerability application.licq.licq_logon_Vulnerability (application a)
{
property string CVE_ID = "CVE-2001-0440";
preconditions { a instanceof licq_Application & a.version<v1.0.3 }
way string way_ID = "buffer overflow";
threat integer threat_NO = 2;
threat string threat_ID1 = "denial of service";
threat string threat_ID2 = "execute arbitrary commands";
}
```

上例中，将脆弱点命名为“application.licq.licq_logon_Vulnerability”，从中可以明显地看出脆弱点位于应用程序 licq 的 logon 模块中，属性中设置 CVE 安全字典中的标识符“CVE-2001-0440”，以便与其它信息源交叉参考，前提条件是 licq 的版本号小于 1.0.3，出现的漏洞方式为‘缓冲区溢出’，导致的威胁数目为 2 个，分别是‘拒绝服务’和‘执行二进制命令’。

4　结论

比较现有的枚举法、CVE 法与笔者提出的 VDL 法，使用 VDL 描述脆弱点的优越性主要体现在以下几点：

（1）借助 CVE 的映射机制，解决脆弱点标识符混乱问题；

（2）采用类 DNS 的命名机制，方便脆弱点的定位、归类与对比；

（3）采用类程序设计语言结构，可以清楚、明确地描述脆弱点内容；

（4）语言结构便于扩展，可以有效管理脆弱点的扩展部分及其更新过程。

在实践过程中，笔者发现 VDL 还存在一些问题，如威胁集合的定义不够灵活等等，这些还有待于进一步的研究。

（收稿日期：2002 年 4 月）

参考文献

1.The ICAT team.ICAT Comprehensive Vulnerability Ranking System.http：//icat.nist.org

2.Mann D E，Christey S M.Towards a Common Enumeration of Vulnerabilities[C].In：Presented at 2nd Workshop on Research with Security Vulnerability Databases，Purdue University，West Lafayette，IN，1999

3.Baker D W，Christey S M，Hill W H et al.The Development of a Common Enumeration of Vulnerabilities and Exposures[C].In：the Second International Workshop on Recent Advances in Intrusion Detection，1999

4.Elz R，Bush R.Clarifications to the DNS Specification[S].RFC 2181，1997-07

5.Mockapetris P.Domain names－concepts and facilities[S].RFC 1034，1987-11

6.Mockapetris P.Domain names－implementation and specification[S].RFC 1035，1987-11