维普资讯 http://www.cqvip.com

一种基于区间分割的遗传算法及其在连续交通网络设计中的应用

周和平　胡列格

（长沙理工大学交通运输学院　长沙 410076）

摘　要　将区间分割与遗传算法相结合，构造了一种新型遗传算法，通过对变量与区间同时进行遗传操作，增强了遗传算法的局部搜索能力。笔者将该算法用于连续交通网络设计，并用一个简单算例进行了仿真试验，通过与其它算法的结果进行对比，可以发现笔者提出的方法明显优于其他方法。

关键词　区间分割法，遗传算法，连续网络设计，仿真

A Genetic Algorithm Based on Interval Partition and its Application to Continuous Road Network Design Problem

ZHOU He-Ping　HU Lie-Ge

(School of Transportation, Changsha Uni. of Sci. and Tech., Changsha 410076)

Abstract　Combined interval partition method with genetic algorithm, the author brings forward a new approach, which improves local search ability of GA by operating variables and their intervals simultaneously. The new algorithm is then applied to continuous network design problem. A simulation is made to test the algorithm and compare the output with other two method. The result indicates the author's approach is better than both of them.

Keywords　Interval partition method, Genetic algorithm, Continuous network design, Simulation

遗传算法以其良好的全局搜索性能在实际工作中得到了大量应用，但由于其局部搜索功能较差，算法效率受到影响，如何增强其局部搜索功能一直是研究者探寻的目标之一。笔者针对有界连续变量规划问题，将区间分割法与遗传算法相结合，设计了一种新型遗传算法，并将其应用于连续交通网络设计问题，通过一个简单路网的仿真试验，结果表明该方法明显优于目前已有的两种算法。

1　连续交通网络设计模型

交通网络设计问题（network design problem, NDP）目前已成为交通规划的热点之一，一般分为离散型网络设计问题（discrete network design problem, DNDP）与连续型网络设计问题（continuous network design problem, CNDP）。前者是对关于添加新路段的投资决策问题，后者则是关于已有交通网络路段通行能力的改进提高的投资决策问题，即选择哪些路段进行改进及改进程度多大的问题。自 NDP 问题被提出以来，CNDP 一直非常受研究者的关注，关于它的研究成果也最多[1]。

1.1　CNDP 的双层规划模型

无预算约束的连续交通网络设计广泛采用的是双层规划模型，上层为网络设计模型（NDP），目标为预算约束条件下的系统总时间，下层为用户平衡交通分配模型（TAP），如式（1）至（6）所示。

$$[\mathrm{NDP}]:\min Z(y)=\sum_{a\in A}x_a^*(y)\cdot t_a(x_a^*(y),y_a)+\lambda\sum_{a=1}^{m}g_a(y_a) \tag{1}$$

$$\text{s.t. } y_a\geqslant 0,(a=1,2,\cdots,m) \tag{2}$$

$$[\mathrm{TAP}]:\min_x Z(x)=\sum_{a\in A}\int_0^{x_a}t_a(w,y_a)\,\mathrm{d}w \tag{3}$$

$$\text{s.t. } \sum_k f_k^{rs}=q_{rs},\ \forall r,s \tag{4}$$

$$x_a=\sum_{r,s}\sum_k f_k^{rs}\cdot\delta_{a,k}^{rs},\ \forall a \tag{5}$$

$$f_k^{rs}\geqslant 0,\ \forall r,s\ \forall k \tag{6}$$

其中，$x^*=(\cdots,x_a^*,(y),\cdots)$是下层数学规划的极值；

y_a：表示路段 a 的通行能力的增加值，共 m 个，它们组成的向量为 $y=(y_1,y_2,\cdots,y_m)$，称之为“扩容向量”；

$g_a=g_a(y_a)$：表示路段上通行能力增加 y_a 所需的投资值；λ 为投资约束惩罚系数；

$t_a(x)$为路段 a 的走行时间是流量的函数；f_k^{rs} 为点对(r,s)间的第 k 条路径的交通流量；q_{rs}：点对(r,s)间的 PA 交通量；$\delta_{a,k}^{rs}$为路段——路径相关变量（0 或 1）。

改进后路段上的 BPR 阻抗公式采用：

$$t_a=t_a(x_a,y_a)=A_a+B_a\left(\frac{x_a}{K_a+y_a}\right)^4 \tag{7}$$

其中，A_a：路段中的自由走行时间；B_a：非负参数；K_a：表示路段 a 的现状通行能力。

1.2　算法

由于双层规划是 NP 完全问题，对于复杂路网很难得到最优解。对于以上模型，目前常用的算法有迭代优化算法、Hooke-Jeeves 算法、平衡分解优化算法等多种，参见文[2]。

迭代优化算法是将一层规划的极值问题的解作为固定值，代入另一层规划问题求解极值，如此反复迭代，直至得到满意解。该算法思路简单，编程方便，但可能不收敛。Hooke 和 Jeeves 于 1961 年提出一种无约束的非凸规划问题的启发式算法，该算法可用于 CNDP 问题。算法由两层循环构成，外层循环使目标函数按固定步长向下降的方向移动，内层循环在每个解点搜索使目标函数下降的方向，先在其所有的坐标轴的正方向上搜索，如不能改善其目标，则在反方向上搜索。该算法相对迭代优化算法需要更多的交通分配次数。1987 年 Suwansirikul 和 Friesz 提出平衡分解算法（equilibri-

周和平　讲师，博士。

维普资讯 http://www.cqvip.com

um decomposed optimization, EDO)，该方法将 CNDP 的上层问题按路段分解成不独立的子问题，然后分别采用区间分割法求解。从应用情况来看该方法优于前面两种方法，但其要求上层规划目标函数是单峰的，不能处理多峰情况。

以上三种方法都只能得到局部最优解，为寻求该问题的全局最优解，许多学者进行了研究，目前模拟退火法、遗传算法等全局搜索算法已广泛应用于网络设计问题。尽管这些方法也不能保证得到全局最优解，但可以得到比以上三种方法更好更合理的结果。笔者采用基于区间分割的遗传算法对 CNDP 问题进行了计算，结果表明该方法明显优于前面三种方法。

2 区间分割算法

区间分割法是一类缩短包含极值点区间的方法，主要有斐波那契法、黄金分割法和对分法，通过计算及比较函数在区间内某些点的值得到较小的包含极值点的区间[2]。其中，斐波那契法是最优的分割法，但由于该法的分划比是随区间的长度与所求精度之比的变化而变化的，每次缩小区间的分划比都不相同，而且需要计算斐波那契数。黄金分割法是斐波那契法的逼近及极限情况，每次缩小区间的分划比是固定的，只要多计算一次函数值就可以达到斐波那契法的精度。以上两种方法都只需计算函数值，计算方便。由于黄金分到法不需要计算斐波那契数，因此它的应用远比斐波那契法广泛。对分法是一种利用函数导数值的区间分割法，每次迭代都将区间分成相等长度的两半，其计算也比较简单，在实际中应用很广，但其搜索效率不如前面两种方法。以上三种方法都只针对单峰函数，如果函数是多峰的则不再适用。

3 基于区间分割的遗传算法

遗传算法是一种全局搜索算法，但其局部搜索功能较差。而区间分割算法具有很强的局部搜索能力，但其不能处理目标函数是多峰的情况。笔者将区间分割法与遗传算法相结合，提出了一种新的遗传算法。其基本思路是：利用遗传算法进行全局搜索，利用区间分割法进行局部搜索，通过区间分割逐步缩小搜索空间，通过遗传算法保证最优解不丢失。与传统的遗传算法不同的是，该方法对变量及其可能区间同时进行遗传操作。该算法在保持遗传算法的全局搜索能力的前提下，增加了遗传算法的局部搜索能力与搜索效率。

(1) 变量编码

笔者采用的变量编码以三维向量表示，如(200,500,382.0528)，前两位数为变量区间，第三位为变量值。

(2) 适应度函数

适应度函数采用如下表达式：

$$F(i)=V-O(i)\cdot M/\sum_{i=1}^{M}O(i) \tag{8}$$

式中：$O(i)$为第 i 个个体的目标函数值；V 为上式右边第二部分的最大值；M 为种群个数。

(3) 初始种群

初始种群产生方法是：首先对变量初始区间进行黄金分割，得到两个区间，然后在变量初始区间上产生随机数，选择随机数所在黄金分割后的区间，与随机数一起构成新的变量编码，得到一个个体。如一个个体有两个变量，其初始取值范围都是[200,600]，在该区间内产生均匀随机数 380.2 和 450.8，对区间进行黄金分割，得到两个区间[200,447.2]和[447.2 600]，而 380.2 和 450.8 分别处在第一和第二个区间，则产生一个个体(200,447.2,380.2)(447.2,600,450.8)。按此方法随机产生另外个体，即可得到初始种群。

(4) 遗传算子

笔者采用四个遗传算子，即选择、粘贴、换位和压缩。

选择算子采用轮盘赌的方法，适应度高的个体优先选择，并保证保留当前最优的个体。

粘贴算子是为了保证群体尽可能覆盖变量的完整区间，防止最优解的丢失。在进行区间分割的过程中，由于要舍弃部分区间，有可能会丢失有最优解。为避免出现这种情况，必须补充区间。笔者采用的方法是：每次迭代，均从初始区间中随机产生部分新个体(新个体产生的方法与初始种群产生的方法相同)，将其粘贴到种群中，替代适应度最差的几个个体。

换位算子对应于传统的交叉算子，主要功能是对变量的搜索区间进行调整。依概率选择交叉个体，并对个体进行随机配对，在随机产生的交叉位上进行换位，与一般的交叉算子不同的是，该操作是针对向量进行的。如图 1 所示，p1 和 p2 在第一个变量处交叉，互换对应的编码。

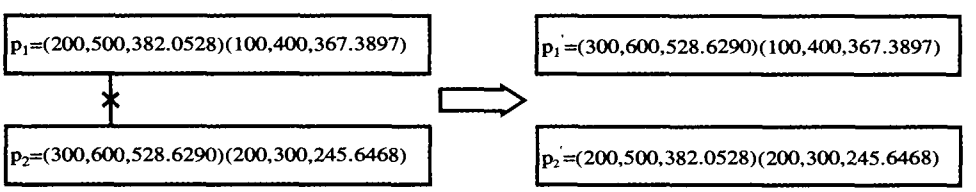

图 1 换位操作

压缩算子对应于传统的变异算子，主要功能是通过对变量的区间进行压缩以增加局部搜索能力。依概率随机产生压缩个体，随机产生压缩位置，在对应变量的区间上产生随机数，并对区间进行黄金分割，得到两个区间，选择随机数所在的区间，与随机数一起构成新的编码，得到新的个体。如图 2 所示，对个体 p1 的第一个变量进行区间压缩，其对应的区间是[300,600]，在该区间内产生随机数 594.2848，将[300,600]进行黄金分割，得到两个区间[300,485.4]和[485.4,600]，由于随机数 594.2848 在第二个区间内，故选择该区间作为新的区间，故将变量的区间由[300,600]压缩到了[485.4,600]。

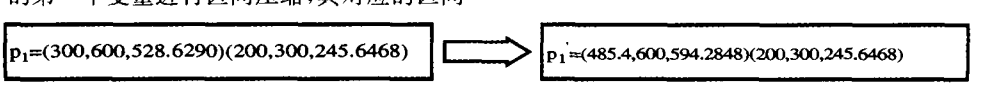

图 2 压缩操作

维普资讯 http://www.cqvip.com

(5)算法步骤

算法步骤如下：

Step 1 给定扩容变量初始区间，如不能较准确把握，可给定一个较大范围的区间；

Step 2 在初始区间上生成初始种群；

Step 3 将扩容变量值与初始值叠加，得到新的路段容量；

Step 4 转入下层规划进行用户平衡配流，得到路段流量及时间；

Step 5 转入上层规划，计算目标函数值；

Step 6 如果满足目标要求或目前最优解变量区间小于指定的范围，停止，当前最优解即为所求，否则转 step 7；

Step 7 计算适应度函数值并排序，并记录对应的变量区间；

Step 8 按轮盘赌进行选择操作；

Step 9 进行粘贴操作；

Step 10 进行换位操作；

Step 11 进行压缩操作，转 step4。

4 仿真试验

以下用文[3]中的一个简单路网进行仿真试验，该路网由4个节点5个路段构成，如图3所示，路段相关参数如表1所示。该路网只有一个OD对，即1-2，分别取100、150、200和300进行仿真试验。上层规划目标函数如式(9)所示。仿真过程如图4至图7所示。

$$Z(y)=\sum_a t_a(x_a,y_a)x_a+1.5\sum_a g_a(y_a)^2 \tag{9}$$

表1 路段相关参数

路段	A_a	B_a	K_a	g_a
1	4.0	0.60	40.0	2.0
2	6.0	0.90	40.0	2.0
3	2.0	0.30	60.0	1.0
4	5.0	0.75	40.0	2.0
5	3.0	0.45	40.0	2.0

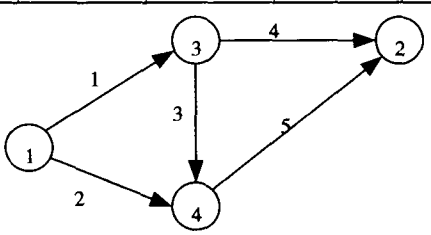

图3 测试路网

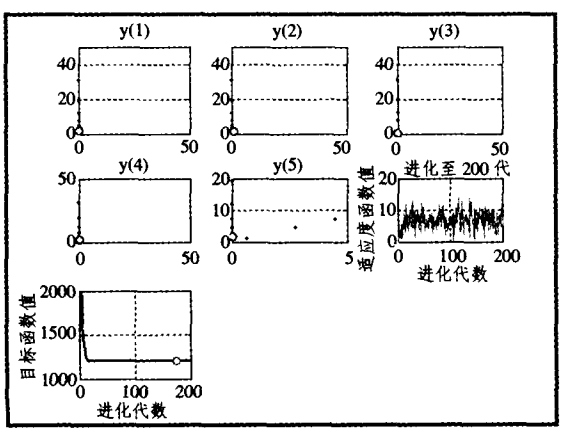

图4 q=100的仿真过程

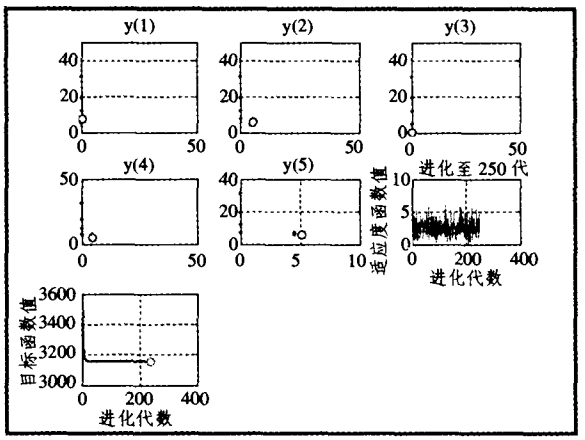

图5 q=150的仿真过程

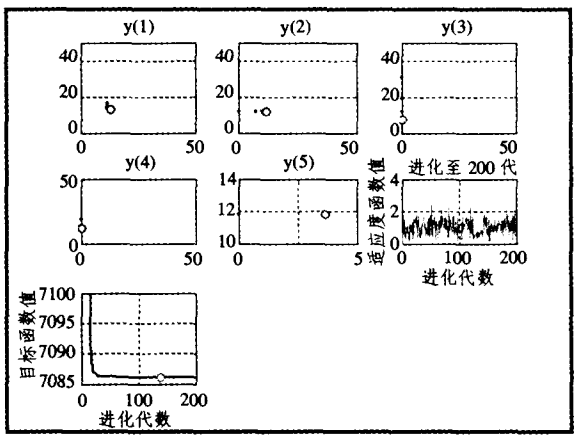

图6 q=200的仿真过程

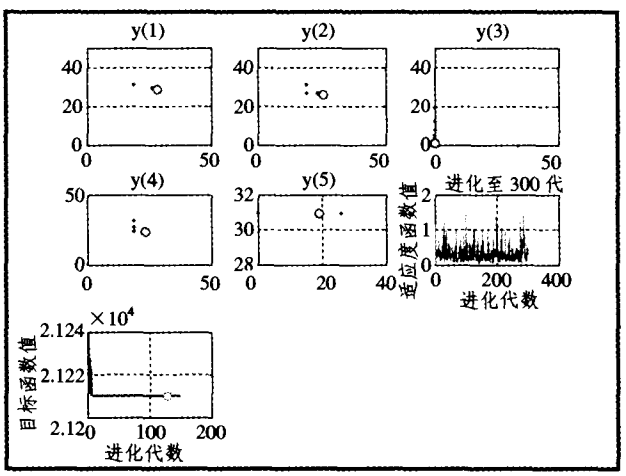

图7 q=300的仿真过程

图4～图7中，每张图中的前5个小图分别为5个扩容变量的区间变化过程，每个点对应一个区间，横坐标表示变量的下限，纵坐标表示上限，圆圈表示最终的区间。第6与第7个小图分别为适应度函数值与目标函数值的变化过程。从以上四张图可以看出，算法运行良好，进化代数均不超过300。为比较算法的有效性，将计算结果与Hooke-Jeeves(简称H-J)和EDO算法的结果进行比较，如表2所示。可以发现，在不同需求情况下，笔者提出的方法都优于其他两种方法，目标值均有不同程度的降低，表明该方法是有效的。但算法需要更多的交通分配次数，且分配次数随需求量的增加而增加。

维普资讯 http://www.cqvip.com

表 2 三种不同方法计算结果比较

需求量	路段	H-J	EDO	GA
100	y1	1.25	1.31	1.33
	y2	1.20	1.19	1.22
	y3	0.00	0.06	0.00
	y4	0.95	0.94	0.97
	y5	1.10	1.06	1.09
	目标值	1200.61	1200.64	1200.58
	分配次数	11	8	5400
150	y1	5.95	5.98	6.04
	y2	5.65	5.52	5.47
	y3	0.00	0.02	0.00
	y4	4.60	4.61	4.65
	y5	5.20	5.27	5.26
	目标值	3156.38	3156.24	3156.21
	分配次数	14	11	7500
200	y1	13.00	12.85	13.03
	y2	11.75	12.02	11.68
	y3	0.00	0.02	0.01
	y4	10.25	10.33	10.30
	y5	11.75	11.77	11.75
	目标值	7086.21	7086.45	7086.15
	分配次数	20	12	4200
300	y1	28.44	28.11	28.44
	y2	25.75	26.03	25.74
	y3	0.00	0.01	0.00
	y4	23.44	23.39	23.44
	y5	26.56	26.58	26.54
	目标值	21209.91	21210.54	21209.91
	分配次数	28	14	7710

结束语 笔者提出的算法具有良好的搜索性能，对于处理有界连续变量的规划问题具有普遍适用性。对于求解连续交通网络设计问题，算法也具有一定实用价值。但从算例可以看出，算法需要较多的交通分配次数，对于大型路网来说，势必消耗大量的 CPU 时间。因此，有必要对算法进行更进一步的改进。可以从两个方面着手解决交通分配的问题，一是寻求更快速的交通分配算法；二是在遗传算法中尽量减少交通分配的次数，可以通过构造一个标准（例如采用类似隐枚举法的标准），仅对满足一定条件的方案进行交通分配，不满足条件的方案的目标函数依照“满足一定条件的方案比不满足条件的目标值好”的原则，参照给出目标值。由于篇幅限制，笔者将另文给出解决办法。

参 考 文 献

1 刘灿齐. 现代交通规划学[M]. 北京：人民交通出版社，2001

2 袁亚湘. 非线性规划数值方法[M]. 上海：上海科技出版社，1993

3 Suwansirikul C, Friesz, Tobin R L. Equilibrium Decomposed Optimization: A Heuristic for the Continuous Equilibrium Network Design Problem. Transportation Science, 1987, 21:254～263

（上接第 57 页）

以运行时间作为性能评价的标准。由于本文所给方法和模拟运行的方法的运行时间都与在超周期中所有任务的请求个数相关，因此以任务集合中的任务实例个数进行分类。实验中共生成 20 个集合类。集合类 1 包括所有任务实例个数小于等于 10 的集合；集合类 2 包括所有任务实例个数大于 10 且小于等于 100 的集合；集合类 3 包括所有任务实例个数大于 100 且小于等于 200 的集合；…；集合类 20 包括所有任务实例个数大于 1900 且小于等于 2000 的集合。每个集合类中有 20 个任务集。

图 2 与图 3 是模拟运行方法与采用本文提出新方法的运行时间比值分布图，图中比值表示模拟运行方法的运行时间除以新方法的运行时间。从图 2、图 3 中可以看出在任务实例个数的变化过程中，新方法的运行时间始终小于模拟运行方法的运行时间。

小结 本文针对抢占阈值调度策略，给出了具有释放抖动和特定释放偏移的周期任务最大响应时间的计算方法。本文提出的新方法是对强实时任务的可调度性分析方法的补充和完善，对判断强实时周期任务的可调度性具有重要意义。

参 考 文 献

1 宾雪莲，杨玉海，金士尧. 基于 EDF 抢占式调度的周期任务最小响应时间分析. 计算机科学，2004，31(9)：114～116

2 Redell O, Törngren M. Caculating exact worst-case response times for static priority scheduled tasks with offsets and jitter. In: Proceedings of the 8th real-time and embedded technology and applications symposium. RTAS'02, San Jose. CA. USA. IEEE computer society press, 2002. 164～172

3 Bate I, Burns A. Schedulability analysis of fixed priority real-time systems with offsets. PEUROMICRO-RTS'97. IEEE computer society press, 97

4 Redell O. Response time analysis for implementation of distributed control systems:[Phd thesis]. Royal institue of technology, KTH, Sweden, 2003

5 Goosesens J, Devillers R. The non-optimality of the monotonic priority assignments for hard real-time offset free systems. Real-time System, 1997, 13(2):107～126

6 杨玉海，宾雪莲，金士尧. 基于抢占阈值调度的周期任务最小响应时间分析. 计算机应用研究，2004，21(11)：41～44

7 宾雪莲，金士尧，杨玉海. 周期多帧任务模型的响应时间分析. 计算机工程与科学，2003，25(6)：104～108

8 宾雪莲，实时任务调度分析：[博士论文]. 长沙：国防科技大学研究生院，2004

9 宾雪莲，杨玉海，金士尧. 一种基于分组与适当选取策略的实时多处理器的动态调度算法. 计算机学报，2006，29(1)：81～91

（上接第 125 页）

4 He H. Automatic Integration of Web Search Interfaces with WISE-Integrator. WWW2004, New York, NY, 2004

5 Wu W S, Yu C. An Interactive Clustering-based Approach to Integrating Source Query Interfaces on the Deep Web. SIGMOD2004, Paris, France, 2004

6 Nambiar U, Kambhampati S. Answering Imprecise Queries over Autonomous Web Databases. ICDE, 2006

7 Zhang W, Li J Z. Processing Probabilistic Range Query over Imprecise Data Based on Quality of Result. LNCS3842, 2006. 441～449

8 Bilke A. Schema Matching using Duplicates. ICDE2005, Tokyo, Japan, 2005

9 Wang J Y. Instance-based Schema Matching for Web Databases by Domain-specific Query Probing. VLDB2004, Toronto, Canada, 2004

10 He B. Statistical Schema Matching across Web Query Interfaces. SIGMOD2003, San Diego, California, 2003

11 Anjomshoaa A. The Design and Implementation of Grid Database Services in OGSA-DAI. In: Proceedings of UK e-Science All Hands Meeting, Nottingham, EPSRC, 2003

12 Paques H, Liu L, Pu C. Distributed Query Adaptation and Its Trade-offs. SAC 2003, Melbourne, Florida, USA, 2003

13 Urhan T. Dynamic Pipeline Scheduling for Improving Interactive Query Performance. VLDB2001, Rome, Italy, 2001

14 Naughton J, DeWitt D, Maier D. The Niagara Internet Query System. VLDB2000, Cairo, Egypt, 2000

15 Ives Z. Adaptive Query Processing for Internet Applications. IEEE Data Engineering Bulletin, 2000, 23(2):19～26