利用遗传算法实现试题库自动组卷问题①

孟祥娟[1]　王俊峰[2]　曹锦梅[1](1 新疆医科大学 高职学院 新疆 乌鲁木齐 8300541;
2 新疆信息产业厅 新疆 乌鲁木齐 8300112)

摘　要： 提出并实现了利用遗传算法求解试题库组卷的数学模型，定义了组卷问题的适应度函数，讨论了运用遗传算法求解在一定约束条件下的多目标参数优化问题，通过初始化种群、选择算子、交叉算子和变异算子,等过程不断进化，最后得到最优解，实验结果表明,遗传算法相对于其它算法更能有效的解决试题库自动组卷问题,提出了实现不相邻试卷分配的补遗随机算法，为求解类似的多目标约束问题及不相邻组合问题提供一种新的方法。

关键词： 遗传算法 随机算法 自动组卷 试题库 多目标约束

Testpaper Auto-Assembling from Question Database on Genetic Algorithm

MENG Xiang-Juan[1], WANG Jun-Feng[2], CAO Jin-Mei[1] (1.Vocational College, Xingjiang Medical University, Urumqi 830054, China; 2.Xinjiang Information Industries Hall, Urumqi 830011, China)

Abstract: The paper introduces a mathematical model of testpaper assembling on genetic algorithm, defines an adaptive function on testpaper assembling, and provides some ideas on multi-object parameter optimization on restricted terms by genetic algorithm. In the evolutionary processes of seeds initialization, operators selecting, operator crossing, operation differentiation, the best solution is finally worked out. Results of experiments indicate, genetic algorithm is more efficient than other algorithms on testpaper auto-assembling. Random algorithm which could achieves testpaper non-adjacency distribution, is a new method for similar multi-object restriction and non-adjacency combination problems.

Key words: genetic algorithm; random algorithm; testpaper auto-assembling; question database; multi-object restriction

计算机辅助考试系统自动组卷的效率与质量完全取决于抽题算法的设计[1]。常用组卷方法大致可分为两类：(1)随机抽取法，即根据状态空间的控制指标，由计算机随机抽取一道符合控制指标的试题放入组卷题库，此过程不断重复，直到组卷完毕或已无法从题库中抽取满足控制指标的试题为止。该方法结构简单，具有很大的随意性和不确定性，无法从整体上把握题库不断变化的要求，不具有智能性。(2)回溯试探法[2]，即将随机选取法产生的每一状态类型纪录下来，当搜索失败时释放上次纪录的状态类型，然后再依据一定的规律变换一种新的状态类型进行试探，通过不断的回溯试探直到试卷生成完毕或退回出发点为止，是一种有条件的深度优先算法，适合于试题类型和题库数量都比较小的考试系统，同时系统应用程序结构相对比较复杂，而且选取试题缺乏随机性，组卷时间长，对于考生随机即时调题的考试过程来说，它已不符合要求。

如何设计一个算法从题库中既快又好的抽出一组最佳解或是抽出一组非常接近最佳解的实体，涉及到一个全局寻优和收敛速度快慢的问题，本文描述了利用遗传算法求解组卷问题的处理方法，讨论了运用遗传算法求解在一定约束条件下的多目标参数优化问题，最后收敛到一个最适应环境条件的个体上，得到问题的

① 基金项目:国家“十一五”科技支撑计划(2006BAD10A15)
收稿时间:2009-04-29

最优解，试验结果表明，遗传算法具有简单、通用、寻优性好、收敛速度快、适合于并行处理等特点。

1　组卷问题

计算机辅助考试系统中，一个非常重要的课题就是在已经建立的试题库中自动生成满足约束条件的试卷，一套试卷的构成需要涉及很多因素，在试卷中每一道题又包含多个属性，其中与组卷有关的属性有如下七项：①题型；②知识点；③难度系数；④能力层次；⑤区分度；⑥时间；⑦分数。组卷中决定一道题，就是决定它的上述 7 个属性，组成一份 n 道题的就是从试题库中抽取 n 道题，组成一个 n×7 的矩阵，矩阵中每一列代表一个属性，每一行代表一道试题。即

$$s=\begin{bmatrix} a_{11}a_{12}a_{13}a_{14}a_{15}a_{16}a_{17} \\ a_{21}a_{22}a_{23}a_{24}a_{25}a_{26}a_{27} \\ a_{31}a_{32}a_{33}a_{34}a_{35}a_{36}a_{37} \\ \cdots\cdots\cdots\cdots \\ a_{n1}a_{n2}a_{n3}a_{n4}a_{n5}a_{n6}a_{n7} \end{bmatrix}$$

该矩阵应满足以下相应的约束条件[4]:

① 题型：试题类型。按照现代教育理论对于考试的要求，试卷既要考核考生对基础知识基本理论的掌握和理解，又要考察考生应用所学知识，分析问题解决问题的能力，试卷同时还要能测试考生思维水平与能力、分析运算能力、科学表达能力、实验操作能力，考虑到上述要求，试卷要包含客观试题(选择题、判断题、填空题)和主观试题(改错题、问答题、证明题、计算题、操作题)共八种题型。

② 知识点：试题内容要有足够的覆盖面，每个章节都要有相应的试题，且各章节内容所占的分数比重能反映教学时数和知识要点。

③ 难度系数：在试题库组卷时，针对不同的考试对象、不同阶段的考试、命题难度也不同。试题的难度系数定义为:d =1-(该项平均分/该项满分),根据难度系数不同，将试卷分为容易、中等、较难、难四个难度等级:

容易：0.05～ 0.20　中等：0.25～ 0.40

较难：0.45～ 0.70　　难：0.75～ 0.95

组卷中通过调整难度系数，获得不同难度命题。

④ 能力层次：分为四个类型：A 类考核考生识记能力，B 类考核考生的理解能力，C 类考核考生简单应用能力，D 类考核考生综合应用能力。试卷的难度越大，能力层次中 A 类，B 类题目比例越小，能力层次中 C 类，D 类题目比例越大。

⑤ 区分度：试题对考生水平的鉴别和区分程度的指标。为了估计区分度，将考生分成[0，24]、[25，49]、[50，74]和[75，100]得分区间，分别统计不同得分区间的得分率。

⑥ 分数：每份试卷的总分为 100 分(或由用户指定)，每道试题小分。

⑦ 时间：每份试卷的考核时间 120 分钟(或由用户指定)，每道试题预期时间。

除了上面的 7 种试题属性约束以外还可以增加其它相关性约束，但约束指标过多，组卷问题难度加大，组卷效率随之降低。

2　试题库修正

如图 1 所示，图中横坐标为得分，纵坐标为得分的分布概率。试卷中能力层次高的题目占的比例大，得分低的考生人数比例也大，得分的分布区线重心偏左（见曲线 1）；反之，得分高的考生比例大，得分的分布区线重心偏右（见曲线 2）；试卷中能力层次中等的题目占的比例大，得分中等的考生人数比例也大，得分的分布区线应大致呈正态分布（见曲线 3），同时该曲线也是试题期望的考生成绩曲线。将得分区间[0，100]分为[0，24]、[25，49]、[50，74]和[75，100]四个部分，四个区间上分 布的得分概率分别为 A、B、C、D 各能力层次占总分的百分比。符合考题的能力层次与得分的分布之间的对应关系。

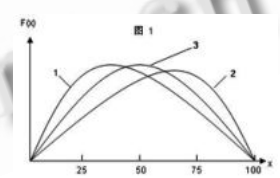

图 1　考试成绩分布曲线

$$A=\int_0^{24} f(x)dx(\%) \qquad B=\int_{25}^{49} f(x)dx(\%)$$

$$C=\int_{50}^{74} f(x)dx(\%) \qquad D=\int_{75}^{100} f(x)dx(\%)$$

根据统计学理论，在理想状态下，考生的成绩分布应大致呈正态分布，分布函数为：

$$f(x)=\frac{100}{\sqrt{2\pi}\sigma}e^{-\frac{(x-p)^2}{800}}(\%)$$

式中 σ 为正太分布的方差，p 为均值。

分析试题的区分度，能鉴别考生水平，利用考生实际成绩曲线与试题期望曲线的偏差计算，修正能力层次试题的分值，得到下次考试的期望曲线，通过试题的实际得分率修正试题难度系数及优化试题库。

3　多目标参数组卷的优化算法

由于生成的试卷要满足：题型、知识点、难度系数、能力层次、区分度、时间、分数等多种属性约束条件，组卷时间相对较长，在计算机网络化考试中，如果每次考生登录考试窗口后动态自动组卷，响应时间过长，在不牺牲组卷质量前提下，缩短响应时间，是组卷算法的难题之一。同时解决相邻考生试卷不同，也是组卷算法的另一难题。

采用在考试前使用遗传算法完成 L 套试卷{1,2,3,…,L}的自动组卷，使用“补遗随机算法”实现套题号分配，达到科学组卷、动态分配，相邻考生、难度相同、内容不同的考试效果。

3.1 遗传算法的组卷问题求解

遗传算法是一种并行的、能够有效优化的算法，以 Morgan 的基因理论及 Eldridge 与 Gould 间断平衡理论为依据，同时融合了 Mayr 的边缘物种形成理论和 Bertalanffv 一般系统理论的一些思想，模拟达尔文的自然界遗传学：继承(基因遗传)、进化(基因突变)优胜劣汰，其实质就是一种把自然界有机体的优胜劣汰的自然选择、适者生存的进化机制与同一群体中个体与个体间的随机信息交换机制相结合的搜索算法。然后根据环境进行基本的操作：selection(选择)，crossover(交叉)，mutation(变异)…这样进行不断的所谓“生存选择”，遗传算法可描述为[3]：

① 随机产生初始种群;

② 利用评价函数 适应度函数 对个体计算函数值;

③ 按一定的概率对个体进行选择、交叉、变异等操作产生新种群;

④ 重复 2 、3 两步,直到收敛(找到最佳解或迭代次数足够多);

上述框架中的参数往往与待解决的具体问题密切相关。针对自动组卷问题,我们给出相应的算法步骤如下:

步骤 1 :染色体的编码

假设试题库中有 n 道不同类型试题，每题型分别有{$m_1,m_2\cdots,m_n$}道小题,可以将每个类型试题用一个用 m_i(i=1,2, …,n)位的二进制子串来表示,形式为:$x_1x_2\ x_3\cdots x_{mi}$ 其中若 x_i 为 1，则表示该题被选中,若 x_i 为 0，则表示该题未被选中,各题型的二进制子串组合在一起，构成整个试卷二进制串。即:

$$x_i=\begin{cases}1,\text{当第}i\text{道题被选中}\\0,\text{当第}i\text{道题未被选中}\end{cases}$$

若试卷中有 n 道试题,则:$x_1x_2\ x_3\cdots x_m$ 串中应有 n 个 1.

步骤 2:初始化群体

通过随机的方法生成初始化的串群体。在串群体中,串的长度是相同的,群体的大小根据需要按经验或实验给出。

步骤 3:计算当前种群每个个体的适应度

本问题的适应度函数可定义为:

$$f=\sum_{i=1}^{7}\delta_i w_i$$

δ_i 表示第 i 个属性指标与用户要求的误差的绝对值,W_i 表示第 i 个指标对组卷重要程度的权值,f 是所有指标与用户要求的误差绝对值之和.

步骤 4:选择

按照一定的选择概率对种群进行复制,选择较好的串生成下一代(个体的适应度函数值越小,该串的性能越好,选择概率越大),去掉较差的串。

步骤 5：交叉

交叉是两个串按照一定的概率(交叉概率 p_c)，从某一位开始逐位互换。首先，对每个二进制串产生一个在 0～1 随机数，若该数小于 p_c，则选择该串进行交叉，否则不选择。随机地对被选择的二进制串进行配对,并根据二进制串的长度 n ,随机产生交叉位置 i,i 为[1,n-1]上的一个整数,然后按下面的方式交叉:

交叉前	交叉后
$a_1a_2a_3\cdots a_ia_{i+1}\cdots a_n$	$a_1a_2a_3\cdots a_ib_{i+1}\cdots a_n$
$b_1b_2b_3\cdots b_ib_{i+1}\cdots b_n$	$b_1b_2b_3\cdots b_ia_{i+1}\cdots b_n$

步骤 6:变异

变异是二进制串的某一位按照一定的概率(突变概率 P_m)发生反转，1 变为 0,0 变为 1，由于在每个单元串中，“1”的数目即是该题型试题的数目，因此在变异过程中应保证整个种群所有单元串中“1”的数目不变。可执行如下过程，首先，由变异概率决定某位取反；然后，检查、修正字符串中“1”的数目，保证不发生变化，这里 P_m 较小，P_m 可小于 0.001，但在实际中发现，在有些遗传算子中，Pm 为 0.1 时更好[5]。

步骤 7：终止

记录进化的代数，并判断是否满足终止条件。若满足，则输出结果，否则转向步骤 3 继续执行。终止条件如下：

(a) 出现种群满足 f=0

(b) 某个个体适应度值达到指定要求；

(c) 达到指定的进化代数；

(d) 当前种群中最大适应度值与以前各代中最大适应值相差不大，进化效果不显著。

步骤 8：全局最优解替换

为保证好的字符串不至于流失，每次遗传操作前记录本次迭代的最优解，若该解优于全局最优解则替换全局最优解，否则全局最优解则替换全局最优解，否则全局最优解保持不变。此次遗传操作后，用全局最优解替换本代的最差解。

3.2 补遗随机算法的试卷分配问题求解

对一个方形教室的考场，可以看成是一个 m×n 矩阵，考场中每位考生对应于一个矩阵元素，如果每位考生都从 L 套试卷，随机获得一套试卷，遗憾的是在试卷总套数不多的情况下，很容易出现相邻重复，当然可以通用约束判断，采用二次乃至多次获得随机数直到符合约束为止，消除重复。

缺点是以牺牲时间为代价换取期望结果，牺牲的时间无法控制，这也正是对约束条件计算时“传统的随机算法”最大不足。补遗随机算法是指在处理约束的随机数问题时，利用非连续向量集合与连续向量集合的对应关系，实现的快速随机数产生算法可描述为：

① 设 A 为 L 项连续向量集合$\{X_1,X_2,\cdots X_L\}$；

② 设 B 为 P 项向量集合$\{Y_1,Y_2,\cdots Y_P\}$，其中 P＜L,B∈A，代表对集合 A 的 P 个约束；

③ 设 C = A － B 它是由属于 A 但不属于 B 的向量组成，共 L−P 项，是非连续向量集合；

④ 由集合 C 创建集合 D 为 L−P 项连续向量集合$\{Z_1,Z_2,\cdots Z_{L-P}\}$,集合 C 与集合 D 的集合元素都是 L−P 项。但集合 C 是不连续的，最大向量值等于 L，集合 D 为连续的最大向量值等于 L−P；

⑤ 对集合 D 的最大向量值 L−P 作随机运算 y=f(x)，其中 X=L−P，得到 1～L−P 之间的一个数字，通过集合 D 得到集合 C 对应元素向量值；

上述模型与待解决的具体问题密切相关。针对考生选卷问题,我们给出相应的算法步骤如下:

步骤 1 :问题定义

假设通过遗传算法，得到 L 套试卷，用连续向量集合 $A=\{X_1,X_2,\cdots X_L\}$表示；用向量集合 $B=\{Y_1,Y_2,\cdots Y_P\}$代表实现不重复相邻的 P 个约束,其中 P＜L,B∈A，假设考试教室 m 行 n 列座位排列，每一座位对应 m×n 矩阵 H 的一个元素，实现每一考生获得与前后左右都不重复的试卷。当 L=2 时，实现无相邻重复“AB”卷分配方式。此时无相邻重复进的半径为 1，当 L＞2 时，无相邻重复半径增加。

步骤 2：试卷分配

现在按照由左向右、有上之下的原则依次为每一个矩阵元素位置分配试卷。

$$H=\begin{bmatrix} a_{11}a_{12}\cdots a_{1i}\cdots a_{1n} \\ a_{21}a_{22}\cdots a_{2i}\cdots a_{2n} \\ \cdots\cdots\cdots\cdots \\ a_{m1}a_{m2}\cdots a_{mi}\cdots a_{mn} \end{bmatrix}$$

a_{11} 是矩阵 H 第一行第一列元素，显然约束为零，可从 L 套中随机得到第 i 套试卷，实现方法：依随机函数公式$y=f(x)$ 当 X=L，得到 y 是[1,L]中一个整数 i。

a_{12} 是矩阵 H 的第一行第二列元素，显然有一个约束,可随机得到 L 套试卷中除 i 以外的试卷第 j 套试卷，实现方法：C = A － B，它是不包含 i 的不大于 L 的正整数，共 L−1 项，是非连续向量集合，创建与集合 C 项数同为 L−1 项连续向量集合 D={1,2…L−1}，依随机函数公式 $y=f(x)$当 X=L−1，得到 y 是[1,L−1]中一个整数 k，通过集合 D 得到集合 C 中 k 顺序位对应元素向量值 j,第一行其它元素取值方法同上。

a_{22} 是矩阵 H 的第二行第二列元素，显然有二个约束，如果为左边 a21 分配的是 K 套试卷,上边 a_{12} 分配的是 j 套试卷,实现方法：C = A － B，它是不包含 j 与 k 的不大于 L 的正整数，共 L−2 项，是非连续向量集合，创建与集合 C 项数同为 L−2 项连续向量集合 D={1,2 … L−2}，依随机函数公式 $y=f(x)$ 当 X=L−2，得到 y 是[1,L−2]中一个整数 β，通过集合 D 得到集合 C 中 β 顺序位对应元素向量值 i，矩阵其它元素取值方法同上。

4 实验结果与分析

4.1 组卷实验设计与数据

在实验中将600道试题按要求存放于所使用的试题库中，并给出个体评价函数：

其中 Wi=(i=1,2,⋯7)。P_{zfwc} 表示各试题分数之和与用户指定的总分之差的绝对值；P_{sjwc} 表示各试题估时之和与用户指定的考试时间之差的绝对值；P_{txwc} 表示各种题型题量与用户指定的相应值的误差绝对值之各；P_{zsdwc} 是每一篇章或知识内容试题分数与用户指定的相应值的误差绝对值之和；P_{ndwc} 是每一难度的试题分数与用户指定的相应值的误差绝对值之和；P_{nlccwc} 是每一能力层次的试题分数与用户指定的相应值的误差绝对值之和；P_{qfdwc} 每一区分度试题分数与用户指定的相应值的误差绝对值之和。以下各表列出一部分实验结果。

表 1　遗传算法在不同种群及不同迭代次数中得到最小适应度的值

迭代数目(代)	最小适应度的值(种群数目=10)	最小适应度的值(种群数目=15)	最小适应度的值(种群数目=20)
10	109	91	88
50	109	88	83
100	93	86	74
150	86	86	70
200	86	83	65
250	86	83	65

表 1 是运用遗传算法在不同种群及不同迭代次数中得到的最小适应度的值。其中，交叉概率 P_c =0.8，变异概率 P_m=0.005。

表 2　遗传算法在不同种群及不同终止条件中得到的运行时间

种群数目	中止时间/s(f≤10)	中止时间/s(f≤10)
10	67	28
15	21	12
20	31	11

表 2 是应用遗传算法在不同种群不同终止条件(终止条件 1 适应度值 f 达到 16，终止条件 2 适应度值 f 达到 10)下的运行时间。其中，交叉概率 P_c =0.8，变异概率 P_m=0.005。

4.2 组卷实验结果分析

对比表 1、表 2 的实验结果可知，在不同种群下，采用遗传算法性能优于随机抽取法和回溯试探法，降低了问题的求解难度，提高了问题的求解效率。组卷是组成一份误差在可接受范围内的试卷，并非要求此试卷的整体指标一定是全局最优，适当地放大误差，可较大幅度缩短求解时间。

4.3 试卷分配实验设计与数据

假设学生在 5 行 6 列的共 30 台微机实验室进行无纸化考试，通过遗传算法，分别得到 2 套与 9 套试卷，作补遗随机算法分配试卷的实验，得到表 3、表 4 试卷分配结果。

表 3 中第一套卷命名为 A 卷，第二套卷命名为 B 卷。

表 4 是补遗随机运算结果，再次作补遗随机运算结果不同。每一单元格的内容都是去掉上一行与前一列的内容后产生的随机数字，以第二行第三列为例，结果 1 是指在 9 中去掉 4、5 之后随即产生的。

表 3　试卷分配(2 套试卷)结果

	1 列	2 列	3 列	4 列	5 列	6 列
1 行	B	A	B	A	A	A
2 行	A	B	A	B	B	B
3 行	B	A	B	A	A	A
4 行	A	B	A	B	B	B
5 行	B	A	B	A	A	A

表 4　试卷分配(9 套试卷)结果

	1 列	2 列	3 列	4 列	5 列	6 列
1 行	7	3	4	1	5	9
2 行	2	5	1	8	3	7
3 行	4	9	3	7	6	1
4 行	2	8	3	5	1	7
5 行	6	2	9	3	5	3

4.4 试卷分配实验结果分析

表 3、表 4 的实验结果可知，利用补遗随机算法可确保任何考生与相邻考生试卷(前后左右)都不相同，并具有很好的收敛速度，试卷套数越多，不相邻重复半径就越大。

5　结语

组卷问题是一个在一定约束条件下的多目标参数优化问题，而它的约束条件难以用数学形式描述，所以采用传统的数学方法求解十分困难，实践证明用遗传算法求解组卷问题有很好的效果。在试卷分配上采用补遗随机运算可以减少组卷套数，不相邻重复分配试卷，缩短命题时间。本文为求解组卷问题及类似该问题的多目标约束问题及不相邻组合问题提供一种新的方法。

参考文献

1 林莉,等.由命题综合要求自动生成试卷的软件系统的开发.计算机应用与软件, 2003,20(3):15－16.
2 胡维华.多目标选题策略研究与应用.杭州电子工业学院学报, 1999,(2):36－41.
3 杨青.基于遗传算法的试题库自动组卷问题的研究.济南大学学报, 2004,18(3):228－231.
4 孙勇,柏云.基于遗传算法的试题组卷策略.淄博学院学报, 2002,(3):27－28.
5 闭应洲,等,基于矩阵编码的遗传算法及其在自动组卷中的应用.计算机工程, 2003,(7):73－75.