基于知识点约束的遗传算法组卷策略的研究

耿 霞
(天津市信息中心,天津 300201)

摘 要：基于遗传算法的自动组卷策略采用分段实数编码，具有自适应型的交叉和变异遗传算子,基于知识点约束的分题型的组卷算法思想,为自动组卷算法的进一步发展提供了研究方向。

关键词：遗传算法;自动组卷;知识点约束;自适应
中图分类号：TP312 **文献标识码**：A

Research of genetic algorithm automatic test paper composition based on knowledge points constraint

GENG Xia
(Tianjing Information Centor, Tianjing 300201, China)

Abstract: The automatic test paper composition strategy based genetic algorithm adopts the discrete real number coding. The crossover operator and the mutation operator are adaptive. The algorithm based on knowledge points constraint and classification of question type show the research direction of the further development in this field.

Key words: genetic algorithm; automatic test paper; knowledge points constraint; adaptive

遗传算法 GA(Genetic Algorithm)是一种模拟自然界生物进化过程的随机搜索、优化方法。它是模拟达尔文的遗传选择和自然淘汰生物进化过程的计算模型[1],采用简单的编码技术来表示各种复杂的结构,并通过一组编码表示简单的遗传操作和优胜劣汰的自然选择来指导学习和确定搜索的方向。由于遗传算法采用种群的方式组织搜索,这使得它可以同时搜索解空间内的多个区域,而且用种群组织搜索方式使得其特别适合大规模并行。目前,该算法已渗透到许多领域,并成为解决各领域复杂问题的有力工具。

1 遗传算法用于组卷的优势

作为一种优化与搜索算法,遗传算法相比于其他算法应用于组卷系统所具有的优势在于[2]:

(1)遗传算法的操作对象是一组可行解,而非单个可行解,搜索轨道有多条,而非单条,因而具有良好的并行性。

(2)遗传算法只需要利用目标的取值信息,而无需梯度等高价值信息,因而适用于任何大规模、高度非线性的不连续多峰函数的优化以及解析式的目标函数的优化,具有很强的通用性。

(3)遗传算法择优机制是一种“软”选择,加上其良好的并行性,使它具有良好的全局优化性和稳健性。

(4)遗传算法操作的可行解是经过编码化的,目标函数解释为编码化个体的适应值,因而具有良好的可操作性和简单性。

2 基于遗传算法的组卷策略

组卷中决定一道试题,即是决定 1 个包含有试题唯一标识(ID)、题型、难度、区分度、考核点、考核点类型、能力层次、建议分值的 8 维向量($a1$,$a2$,$a3$,$a4$,$a5$,$a6$,$a7$,$a8$),决定一份试卷 n 道题,就决定了 1 个 $n\times6$ 的矩阵 S:

$$S=\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} & a_{1,6} & a_{1,7} & a_{1,8} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} & a_{2,5} & a_{2,6} & a_{2,7} & a_{2,8} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & a_{n,4} & a_{n,5} & a_{n,6} & a_{n,7} & a_{n,8} \end{bmatrix}$$

这就是问题求解中的目标状态矩阵。建立问题的目标矩阵后,依据遗传算法的基本流程,对本研究的组卷策略进行详细阐述。

2.1 染色体编码及群体的初始化

用遗传算法求解问题,首先要将问题的解空间映射

成一组代码串[3]。有文献用二进制编码，用 1 表示该题被选中，0 表示未被选中。这种编码简单明了，但是进行交换等遗传操作时，各题型的题目数难以精确控制。当题库中题量很大时，编码冗长。已有大量实验表明，在解决数值优化问题时，采用实数编码的遗传算法的效率要好得多，因此，本研究采用实数编码。在组卷中所得的可行解就为一份试卷，所以本研究将一份试卷映射为 1 个染色体，组成试卷的各个试题映射为基因，基因的值直接用试题的 ID 表示，这样染色体的编码可表示为：$(G_1, G_2, G_3\cdots, G_n)$，其中 $G_i(i=1\sim n, n$ 为试卷的总题目数$)$为试题的 ID。编码时应将同一题型的试题放在一起，并保证每条染色体上的基因不重复，即每套试卷中不能出现重复试题。

试题的难度分为 4 档，本研究采用离散型随机变量的二项分布函数 $B(n,p)$建立 1 个由试卷的期望平均分 P 计算难度分布的模型。

离散型随机变量的二项分布函数 $B(n,p)$：

$$P(k)=C_n^k p^k q^{n-k} \tag{1}$$

其中，$k=0,1,2,\cdots,n$，n 为整数，$p>0$，$q>0$，$p+q=1$。

另外，$q=\dfrac{\text{期望的平均分值}}{\text{满分值}}p$（即用 q 表示期望的平均得分率，也就是试卷的平均难度系数），$p=1-q$，$n=v+1$（v 为难度级别数，即若试题难度采用 4 级分档，就取 $n=5$），代入式(1)，求得 $P(k)$。因为一般而言 $q>0.6$，所以 $P(n)$、$P(n-1)$通常很小，可以将它们加到 $P(n-2)$上，将处理后得到的 $P(k)(k=0,1,2,\cdots,n-2)$作为难度级别为 $k+1$ 的分数比例，再将其乘以满分值，便得到难度的分值分布。

2.2 适应度函数

在遗传算法中，以适应度大小来区分群体中个体的优劣。一般而言，适应值越大的个体越好，越容易被保留而繁衍下一代；适应值越小的个体越差，更容易被淘汰。适应值的选取是遗传算法设计的关键，它直接决定算法的优劣以及该组卷策略的科学性。本研究提出的自动组卷模型是基于知识点分布的，采用如下方法设定适应度函数 F，F 分别由 f_1，f_2 和 f_3 3 个子函数组成。

f_1 表示章节分数分布适应函数。设 C_i、X_i、$e_i(i=1,2,\cdots,m, m$ 为章的数目$)$分别表示用户要求的各章应占的分数、实际生成试卷中各章所占的分数、用户允许各章的分数误差。生成的试卷满足用户关于内容分数分布要求的程度可以用式(2)值的大小来评价：

$$f_1=\sqrt{\sum_{i=1}^{m} d_i^2} \quad d_i=\begin{cases}0, & \text{当}|X_i-C_i|\leqslant e_i\\ |X_i-C_i|-e_i, & \text{当}|X_i-C_i|>e_i\end{cases} \tag{2}$$

其中 m 为总章数，X_i 为实际分配的分值，C_i 为预期值，e_i 为允许误差。

式(2)采用方差来统计章节分数分布的偏差，而不用 d_i 的差的绝对值来表示，是因为 f_1 是用来评价某份试卷对每一章的适应度的误差，只是简单地将误差值累加，不能充分表现该试卷每一章的偏差。

f_2 表示知识点的覆盖率和考核点类型的分配比例的适应值函数。本研究以考核点的覆盖率作为一项“软”约束条件，即考核点覆盖率越大。该试卷的适应度越大；同时根据命题的经验，如果一份试卷的考核点类型分布比例越接近 5:3:2，那么该套试卷的命题比例越科学，其试卷的考后成绩越容易呈正态分布，故本研究将这 2 个参数也作为适应度函数之一：

$$f_2=\sqrt{(\frac{a}{n}-0.5)^2+(\frac{b}{n}-0.3)^2+(\frac{c}{n}-0.2)^2}+(1-\frac{tc}{tm}) \tag{3}$$

其中 n 为本章总分数，a 是考核点为重点的分值分配，b 是考核点为次重点的分配分值，c 是考核点为一般的分配分值，t_c 为该套试卷所占的不重复考核点数，t_m 为该门课程总考核点数。

式(3)的前半部分用于衡量一份试卷的考核点类型比例与预期比例（即 5:3:2）的波动误差，值越小，误差越小，比例越接近；后半部分 t_c/t_m 得到考核点的覆盖率，用 $1-\dfrac{tc}{tm}$，则考核点的覆盖率越大，该值越小，故 f_2 的值越小，其适应度越好。

f_3 表示难度分布适应度函数。设 A_i、S_i、$e_i(i=1,2,\cdots,n, n$ 为难度等级数$)$ 分别表示用户期望的每个难度等级应占的分数、实际试卷中各等级所占的分数、允许误差。生成的试卷满足用户关于难度分数分布要求的程度可以用式(4)值的大小来评价：

$$f_3=\sqrt{\sum_{i=1}^{n} d_i^2} \quad d_i=\begin{cases}0, & \text{当}|A_i-S_i|\leqslant e_i\\ |A_i-S_i|-e_i, & \text{当}|A_i-S_i|>e_i\end{cases} \tag{4}$$

同理，该式的方差值越小，说明该试卷的难度分布越接近用户预期要求，适应度越好。

因此，该试卷的总体适应度值 $f_{min}=f_1+f_2+f_3$，f_{min} 越小越好，是最小化问题。本研究采用如下方法将目标函数 f_{min} 转化为适应度函数 f_{max}：

$$f_{max}=\begin{cases}100-f_{min}, & \text{当 } f_{min}<100\\ 0, & \text{当 } f_{min}\geqslant 100\end{cases}$$

因为指数比例既可以让非常好的个体保持多的复制机会，同时又限制了其复制数目以免其很快控制整个群体，提高了相近个体间的竞争，所以对上述适应度函数 f_{max} 采用如下指数比例变换方法转换为适应度函数 F：

$$F=\exp(\beta f_{max}) \tag{5}$$

式中，$\beta=0.06$。

2.3 选择算子

本研究的选择算子策略采用期望值模型选择机制，即先计算群体中各个个体期望值被选中的次数 $N_i=M\cdot F_i$ $(i=1,2,\cdots,M, M$ 为群体规模，F_i 为第 i 个个体的适应值$)$，用 N_i 的整数部分 $\lfloor N_i\rfloor$ 安排个体 i 被选中的次数，这

样其选出 $\sum_{i=1}^{M}\lfloor N_i \rfloor$ 个个体，然后以 N_i 的小数部分作为概率进行贝努利实验，若试验成功，则该个体被选中，不断重复，直至选满为止。个体适应值越高，被选中的概率越大。但是，适应值小的个体也有可能被选中，这样有助于增加下一代群体的多样性。

2.4 交叉算子

将以上选出的个体进行两两随机配对，对每一对相互配对的个体采用有条件的“均匀交叉”，即 2 个配对个体的每一个基因座上的基因都按一定的交叉概率 P_c 和一定的条件进行交换，产生 2 个新个体[4]。

本研究对于交叉概率 P_c 的确定采用自适应的交叉概率。简单遗传算法中，交叉率是个常数，而实际上，优良的交叉率与遗传代数的关系较大。在迭代初期，交叉率选择得大一些可以造成足够的扰动，从而增强遗传算法的搜索能力，而在迭代后期，交叉率选得小一些可以避免破坏优良基因，从而加快收敛速度。因此，本研究选择的交叉概率是个能随着演化不断调整的函数，称为变交叉概率。交叉概率计算公式为：

$$P_c' = \max\left\{\frac{P_{c,\max}}{1+t/t_{\max}}, P_{c,\min}\right\} \quad (6)$$

P_c' 是第 t 代的交叉概率，$P_{c,\max}$ 为最大交叉概率，取 0.7，$P_{c,\min}$ 是最小交叉概率，取值为 0.5，t 为遗传代数，$t_{\max}$ 是最大遗传代数。由于遗传代数 t 是变化的，所以交叉概率 Pc' 是随代数 t 而改变，除非 P_c' 总是小于 $P_{\min}$。每次交叉根据选择概率判定当前是否进行交叉，如果要交叉，则随机选出一对个体，在 2 个体中分别随机选择 1 个交叉位进行交叉。对 2 个配对个体的每一个基因座上的基因，先随机产生 1 个 0~1 的实数 $r1$，如果 $r1<Pc$ 并且满足交换条件(即交换后个体的各个基因不重复)，则交换该基因座上的基因，否则不交换。

2.5 变异算子

由于普通的变异操作可能会使用户指定范围外的题目出现在染色体中，也会使各题型的题目数难以保证，本研究采用有条件的变异算子，即每个个体的每一个基因座上的基因都按一定的变异概率 P_m 在一定的范围内进行变异。

同样，本研究的变异概率也采用自适应变异概率。在简单遗传算法中，变异率是个常数。通常对于交叉率是常数的情况，群体的素质会趋于一致，这样就形成了近亲繁殖。群体基因的多样性变差不仅会减慢进化历程，也可能会导致进化停滞，过早收敛于局部最优解。因此，变异概率也能随着演化不断调整，由于概率表达式中含有遗传代数 t，这个概率称为变动变异概率。变异概率计算公式为：

$$P_m' = \max\{P_{m,\max}\cdot\exp(-\lambda\cdot t/t_{\max}), P_{c,\min}\} \quad (7)$$

t 为遗传代数，$t_{\max}$ 为最大遗传代数，$P_{m,\max}$ 为最大变异概率，取值为 0.15，$P_{m,\min}$ 是最小变异概率，取值为 0.01，λ 为常数，取值为 10。每次变异通过交叉概率判断当前是否变异。对于个体的每一个基因座上的基因，先随机产生 1 个 0~1 的实数 $r1$，如果 $r1<P_m$，则根据一定的变异条件，即从备选题库中抽取一道同类型同分值的试题，同时保证该题不存在于该份试卷中，替换该基因座上的基因，否则不变异。

2.6 保存最优策略

为了保证优良个体在选择的过程中不被淘汰，对当前代进行了选择、交叉、变异操作产生新一代后，比较新一代的最好个体与其父代的最好个体的适应值，如果下降，则以父代最好个体替换新一代的最差个体。此策略可以保证迄今为止的最优个体不会被交叉、变异等遗传运算所破坏，它是遗传算法收敛性的一个重要保证条件。

3 试验结果及分析

为了验证上述算法的可行性和有效性，利用一门计算机课程的题库进行研究，该题库中有 1000 道题，其题库结构如下。

(1)组卷 10 套，总分为 100 分，题型分配如表 1 所示。

表 1 题型分配表

题型	分值	分配方式	备注
单项选择题	30	统一分配	每题 1 分
多项选择题	30	统一分配	每题 2 分
填空题	10	统一分配	每题 1 分
简答题	12	独立分配	3 分，4 分，5 分
论述题	8	独立分配	3 分，5 分

(2)预期平均分为 70 分，经过计算得出预期难度分配为：

易:中等偏易:中等偏难:难=20:36:31:13

(3)要求考核点覆盖率达到 70%以上，平均区分度在 0.3~0.7 之间，平均难度为中等偏易或中等偏难等级，考核点覆盖率(即重点:次重点:一般)接近 5:3:2。

(4)章节分值分布如表 2 所示。

表 2 章节分值分布表

第一章	第二章	第三章	第四章	第五章	第六章	第七章	第八章	第九章	第十章
15	7	20	13	0	12	5	6	10	12

实验设置最大迭带代数 GenMax=100，允许误差 e_i=2 分，群体规模 GenSize=100，结果各章的分布基本满足预期要求，难度和考核点类型比例基本接近正态分布，平均难度维持在第 2~3 等级之间，区分度在 0.4~0.7 之间，考核点覆盖率基本到达 70%以上。目前该算法已在自学考试命题中试用，以便今后进一步推广。

参考文献

[1] 余胜泉，姚顾波，何克抗.通用试题库组卷策略算法[R].

2000.

[2] 曾一,冉忠,郭永林.试题库中自动组卷的算法及试卷测评策略[J].计算机工程与设计,2006(8):3024-3027.

[3] 张爱文,樊红莲.自适应遗传算法用于自动组卷中的数学模型设计[J].哈尔滨理工大学学报,2006(11):18-20.

[4] MEHMET Y. Heuristic optimization methods for generating test from a question bank [M]. MICAI 2007: Advances in Artificial Intelligence, Springer Berlin/Heidelberg, 2007: 1218-1229.

(收稿日期:2009-11-13)

作者简介:

耿霞,女,1983 年生,硕士,助理工程师,主要研究方向:计算机智能辅助决策及应用。