

基于 PBIL 算法的高校自动排考系统^①

袁利永¹ 金炳尧² 刘日仙³ (1.浙江师范大学 数理与信息工程学院 浙江 金华 321004;

2.浙江师范大学 教师教育学院 浙江 金华 321004; 3.金华职业技术学院 浙江 金华 321017)

摘要: 提出了一种基于 PBIL 的高校自动排考算法,重点论述了如何优化目标函数与排考约束条件之间的关系,并对 PBIL 基因选择算法提出了改进。通过实际的测试应用,基于 PBIL 算法的自动排考系统能够较好地满足学分制下的自动排考需求,对附加约束条件具有较强的适应性,能够满足各个学校的不同排考需求。

关键词: PBIL; 进化计算; 自动排考; 时间表问题; 基因学习算法

University Automatic Examination Timetabling Based on the PBIL Algorithm

YUAN Li-Yong¹, JIN Bing-Yao², LIU Ri-Xian³ (1.College of Information Science and Engineering, Zhejiang Normal University, Jinhua 321004, China; 2.School of Teacher Education, Zhejiang Normal University, Jinhua 321004, China; 3.Jinhua College of Profession & Technology, Jinhua 321017, China)

Abstract: In this paper, an algorithm for examination timetabling based on the PBIL (Population-Based Incremental Learning) algorithm is proposed. The relationships between the optimal function and schedule constraint are emphasized. It is proved that the PBIL algorithm is a suitable method in solving this problem, and it is flexible to attach schedule constraints. Finally, improvement of genetic selection of PBIL algorithm is proposed.

Keywords: PBIL; evolutionary computing; automatic examination timetabling; timetable problems(TTP); gene learning algorithm (GL)

期末排考工作是高校教务管理人员经常面临的艰巨任务,每次安排考试课程大约在上千门左右。学分制系统的引入进一步增加了期末排考工作的复杂性,较为自由的选课制度使得课程教学班(简称教学班)中包含了各类学生。然而面对这一繁重的资源安排和处理工作,长期以来高校教务部门一直主要靠手工完成,使排考效果既不科学也耗费了大量人力。随着高等教育的不断发展,专业在不断更新,课程在急速增加,国内各所高校都面临着教学资源严重不足的情况,因此寻找最优组合的考试资源,合理高效的解决排考问题愈益突显,急需找到有效的排考算法,通过计算机来辅助教务部门完成学期末的排考工作。

事实上人们对自动排考系统已经做过一些研究,

但由于未能很好解决排考中诸多相关因素的算法问题,因此这些系统往往不能满足不同学校的需求,这也就是目前仍有许多的高校采用手工排考的原因。本文提出了一种基于 PBIL 的高校自动排考算法,通过实际的测试应用,该系统能较好地满足学分制下的自动排考需求,对附加约束条件具有较强的适应性。

1 PBIL算法概述

PBIL 是由美国卡内基梅龙大学 Baluja,S.提出进化学习算法^[1],它将进化过程视为学习过程,用学习所获取的知识—学习概率来指导产生后代这种概率是整个进化过程的信息积累,用它指导产生的后代将会更优生(比起 GA 的双亲基因重组及 EP、ES 的单个父

^① 收稿时间:2009-09-02;收到修改稿时间:2009-10-13

代变异 Gaussian), 因而能在许多应用问题中获得更快的收敛速度及更优的结果。金炳尧等对 PBIL 算法提出了改进, 并对 PBIL 算法在几种典型问题上的应用进行了研究^[2-4]。

PBIL 算法将原优化问题的解空间 S 映射到基因学习算法能够操作的基因解空间 S' 。可行解 $s' \in S'$ 由 l 个基因位组成, 其中第 i 位的值可以是集合 $\{a_{i1}, a_{i2}, \dots, a_{ini}\}$ 中的任意一个或多个。集合 $\{a_{i1}, a_{i2}, \dots, a_{ini}\}$ 称为第 i 位的等位基因。如果第 i 位只能取 $\{a_{i1}, a_{i2}, \dots, a_{ini}\}$ 中的一个值, 则称第 i 位为单等位基因位。对于有 l 个基因位、每个基因位都可以从 n 个等位基因中取值的优化问题, 其基因解可以用一个布尔矩阵 $S^{l \times n}$ 来表示, 如果矩阵元素 $s_{ij} = 1$, 就表示第 i 个基因位的取值为该基因位的第 j 个等位基因值。

矢量 $P^i = (P_{i1}, P_{i2}, \dots, P_{ini})$ 用来表示在解空间 S' 的 l 个基因位中每个基因位取与该基因位对应的等位基因值的概率。其中 $P_i = (p_{i1}, p_{i2}, \dots, p_{ini})$ 表示第 i 个基因位取与该基因位对应的 n_i 个等位基因值的概率。 p_{ij} 为第 i 个基因位取等位基因集合 $\{a_{i1}, a_{i2}, \dots, a_{ini}\}$ 中第 j 个值的概率, $j = 1, 2, \dots, n_i$, 且每个基因位取等位基因的概率之和都为 1。

PBIL 算法把优化选择问题描述为 $\min f(s), s \in$ 可行解空间, f 为优化目标函数。

典型的 PBIL 进化学习过程描述如下:

- (1) 初始化学习概率 P : $P_{ij} = 1/n_i, 1 \leq j \leq n_i, P_{ij} \in P_i, 1 \leq i \leq l$;
 - (2) 由学习概率 P 指导产生 n 个解 s_1, s_2, \dots, s_n ;
 - (3) 计算 n 个解的目标函数值 $f(s_1), f(s_2), \dots, f(s_n)$, 并找出最优解 s' ;
 - (4) 用最优解向量 s' 修正学习概率 $P_{ij} = P_{ij} + \psi \times s'_{ij}, 1 \leq i \leq l$;
 - (5) 返回步骤(2), 直至满足一定的结束条件为止。
- 参数 ψ 是一个经验常数, 在不同规模和约束条件下, 它的取值对于问题求解速度和成功率都有较大的影响, 一般介于 0.01 ~ 0.1。

2 应用PBIL算法求解自动排考问题

2.1 自动排考问题描述^[5-7]

在学分制中, 考试安排工作主要涉及以下四个对象: 考试课程及其对应的教学班、考场、监考教师、

可用考试时间(一天分为上午和下午两个考试时间)。自动排考算法需要解决的主要问题是: 在教室资源、教师资源、时间资源相对固定的情况下无冲突地安排好所有的考试课程, 并使考试安排尽可能合理。

考试安排的冲突情况一般包括:

- (1) 学生多门考试时间冲突;
- (2) 多门考试教室和时间同时冲突;
- (3) 监考教师多项监考任务时间冲突。

以上 3 条约束条件称为硬约束条件, 这是在考试安排时必须满足的条件。除此之外, 算法还应满足以下几方面的需求:

- (1) 尽量不安排学生在连续的两个考试时间参加考试, 有利于考生休息和充分发挥水平;
- (2) 教师监考次数应可灵活控制, 在安排监考教师时一方面要考虑监考次数均衡以平衡教师工作量, 另一方面也要考虑个别教师由于某种原因不参加某个特定考试时间的监考及减少监考次数的要求;
- (3) 尽量不安排教师在连续的考试时间监考, 以便教师适当休息。

以上的约束条件成为软约束条件, 这是在排考时尽量满足的条件。

排考问题所涉及的资源描述如下:

$KC = \{\text{考试课程全体}\}$ 每门课程有参加考试学生、教室需求、监考教师需求。

$XS = \{\text{参加考试对象全体}\}$ 所有考试课程学生的并集。

$JS = \{\text{可用于排考的教室全体}\}$ 有考试座位数和教室类型等指标。

$SJ = \{\text{考试时间集合}\}$ 能够排考的时间资源。

$JK = \{\text{监考教师的集合}\}$ 能够排考的监考教师, 包含每一教师排考时间约束。

2.2 自动排考算法设计

自动排考问题是多资源组合优化问题, 它是属于一类 NP 难问题。在学分制系统下, 考试时间安排是算法最复杂的部分, 其次是监考教师和考场的安排。为了提高算法效率, 降低算法复杂度, 可将自动排考算法分解为以下几个步骤: 第一步, 在满足教室和监考教师总体约束条件下, 求得满足考生时间需求的排考问题最优解 PK^{SJ} ; 第二步, 基于 PK^{SJ} 求解满足监考教师具体约束的最优解 PK^{JK} ; 第三步, 基于 PK^{JK} 求解满足监考教师具体约束的最优解 PK^{JS} , 最后将

PK^{SJ}、PK^{JK}、PK^{JS} 合并为 PK 即可。由于 PK^{SJ} 已经满足教室总体约束要求，为考试课程分配教室则变得较为简单，因此，本文只重点介绍求解 PK^{SJ}、PK^{JK} 的算法设计。

下面介绍求解 PKXS 的算法设计。

KC={kc₁,kc₂,kc₃,...,kc_n} 为考试课程集，PK^{SJ}={PK^{SJ}₁,PK^{SJ}₂,PK^{SJ}₃,...,PK^{SJ}_n} 构成长度为 n 的基因序列，其中 PK^{SJ}_i 表示 kc_i 的考试时间，PK^{SJ}_i 取 SJ={s_{j1},s_{j2},s_{j3},...,s_{jm}} 中的值，显然 PK^{JK}_i 是单等位基因。概率矩阵 P^{SJ}_n × m 表示某门课程安排在某一监考时间的概率，如 P^{SJ}_{ij} 表示考试课程 kc_i 安排在考试时间 s_{jj} 的概率。对于某些课程对考试时间的特殊约束，可通过对概率矩阵 P^{SJ}_n × m 的初始化中实现。例如，将 kc_i 固定在考试时间 s_{jj} 上，只需将 P^{SJ}_{ij} 设置为 1，将所有 P^{SJ}_{ix} (x ≠ j) 设置为 0 即可；再如，希望 kc_i 不安排在考试时间集 SJ'_i 上，只需将这些 P^{SJ}_{ix} (s_{jx} ∈ SJ'_i) 设置为 0 即可。

求解 PKXS 的目标函数表示为 f^{SJ}=w₁*f^{SX}+w₂*f^{ZJS}+w₃*f^{ZJK}+w₄*f^{FB}，其中 f^{SX} 对应于学生考试冲突约束的罚函数，f^{ZJS} 对应于教室总体约束的罚函数，f^{ZJK} 对应于监考教师总体约束的罚函数，f^{FB} 对应于考生考试时间均匀分布约束的罚函数。w₁,w₂,w₃ 分别对应三条硬约束，而 w₄ 对应软约束，因此 w₁,w₂,w₃ 应 >>w₄。w₁,w₂,w₃ 的值则根据约束条件的倚重程度进行设置。

$f^{SX} = \sum_{k=1}^N ct(XS_k) * |XS_k|$ 。设 {XS₁,XS₂,...,XS_k,...,XS_N} 是由等价关系 R={拥有相同选课的学生} 在 XS 上的一个划分，N=|XS/R| 是 XS/R 的基。ct(XS_k)=|KC_k|-|KC_k/U|，其中 KC_k={kc_x|kc_x 表示 XS_k 需参加考试的课程}，等价关系 U={拥有相同考试时间的课程}。

$f^{ZJS} = \sum_{i=1}^m shortJS(KC_i)$ 。其中 KC_i={kc_x|kc_x 表示考试时间为 s_{ji} 的课程}，shortJS 函数用于计算考试课程集 KC_i 在考试时间 s_{ji} 的教室短缺数量。

$f^{ZJK} = \sum_{i=1}^m shortJK(KC_i)$ 。其中 shortJK 函数用于计算考试课程集 KC_i 在考试时间 s_{ji} 的监考教师短缺数量。

$f^{FB} = \sum_{k=1}^N fb(XS_k) * |XS_k|$ 。其中 fb(XS_k)= $\sum_{i=1}^{M-1} ((PK^{SJ}_{k,kc_{i+1}} - PK^{SJ}_{k,kc_i}) < \epsilon ? 1 : 0)$ ，其中 kc_k,kc_{k+1} ∈ KC_k,KC_k={kc_x|kc_x 表示 XS_k 需参加考试的课程}，并使 KC_k 中的课程按考试时间升序排列，M=|KC_k|，ε 为学生考试时间间隔控制参数。

下面介绍基于 PK^{SJ} 求解 PK^{JK} 的算法设计。

KC_i 表示考试时间为 s_{ji} 的课程集合，jks_i 表示 KC_i 所需的监考教师数，其中 1 ≤ i ≤ m。PK^{JK}={PK^{JK}₁,PK^{JK}₂,...,PK^{JK}_m} 构成长度为 m 的基因序列，其中 PK^{JK}_i 表示安排在考试时间 s_{ji} 上的监考教师，PK^{JK}_i 取 JK={jk₁,jk₂,jk₃,...,jk_c} 中 jks_i 个值，显示 PK^{JK}_i 是复等位基因。概率矩阵 P^{JK}_m × c 表示某监考时间安排某教师监考的概率，如 P^{JK}_{ij} 表示监考时间 s_{ji} 安排 jk_j 监考的概率。对于某些教师对监考时间的特殊约束，可通过对概率矩阵 P^{JK}_m × c 的初始化中实现。例如 jk_i 不希望在考试时间 s_{jj} 上安排监考，只需将 P^{JK}_{ij} 设置为 0 即可。

求解 PK^{JK} 的目标函数表示为：f^{JK}=w₅*f^{CC}+w₆*f^{KFB}，其中 f^{CC} 表示场次均衡约束构成的罚函数，f^{KFB} 表示监考分布约束构成的罚函数，w₅,w₆ 的值则根据约束条件的倚重程度进行设置。

$$f^{CC} = \sum_{i=1}^C exist(limit_i) * (jks(jk_i) \in limit_i ? 0 : \beta) * (jks(jk_i) - bz)^2$$

其中 exist(limit_i) 表示 jk_i 是否存在监考场次限制 limit_i，jks(jk_i) 表示教师 jk_i 的监考场次，bz 表示平均监考次数，β 表示不符合监考场次限制 limit_i 时的罚值，一般设置比较大的常数。

$f^{KFB} = \sum_{i=1}^L \sum_{k=1}^{L-1} ((PK^{JK}_{i,s_{j_{k+1}}} - PK^{JK}_{i,s_{j_k}}) < \phi ? 1 : 0)$ ，s_{jk},s_{jk+1} ∈ SJ_i,SJ_i={s_{jx}|s_{jx} 表示 jk_i 需监考的时间}，并使 SJ_i 中的考试时间按升序排列，L=|SJ_i|，φ 为监考时间间隔控制参数。

2.3 排考算法测试结果

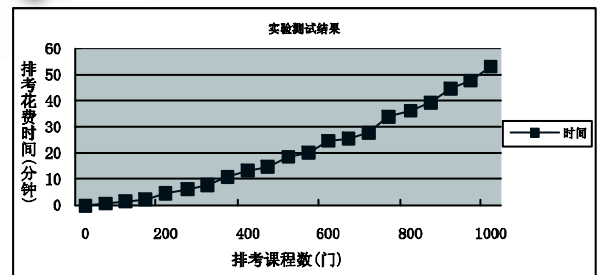


图 1 基于 PBIL 的自动排考算法测试结果

为了测试算法的时间性能，采用 VB 语言对本文提出的算法进行了实现，采用本校 2008-2009 学年第二学期的考试课程作为测试数据进行运行测试。在可用教室数固定(100)、可用时间段固定(30)的情况

下,对 50 至 1000 门课程的排考任务进行了测试。测试环境:操作系统 Windows XP, CUPP4 3.2G, 内存 2G。在测试中对一些次要约束条件(如教室类型约束、监考教师特殊要求等)进行了简化,测试结果如图 1 所示。

2.4 PBIL 基因选择算法改进

PBIL 算法在生成每一代个体种群时,一般采用概率采样实现基因重组。经典的 PBIL 在基因重组算法一般采用简单的赌盘选择算法:把第 i 个基因位的 n_i 个等位基因的概率看成是赌盘上的 n_i 个不同区域,区域的大小为其概率值。这 n_i 个区域恰好组成一个完整的赌盘(n_i 个概率是归一化的)。每次采样时用 $0 \sim 1$ 均匀分布随机数来选择每个基因位的等位基因。其过程可描述如下(k 为最后的采样所得的等位基因序号):

- (1) 产生一个 $0 \sim 1$ 均匀分布的随机数 r
- (2) $k = 1$; $\text{sum} = 0$;
- (3) While($\text{sum} < r$ 且 $k < n_i$)
- (4) $\text{sum} = \text{sum} + p_{ik}$
- (5) $k = k + 1$
- (6) End

设基因序列有 N 个基因位组成,每代个体种群数量为 M ,根据上述算法可知,生成每一代个体种群所需基因选择时间为 $\sum_{i=1}^N (n_i/2) * M$ 。

改进后的算法增加矢量 $P^i = (P^i_1, P^i_2, \dots, P^i_{n_i})$, 其中 $P^i = (p^i_{11}, p^i_{12}, \dots, p^i_{1n_i})$ 表示第 i 个基因位 n_i 个等位基因的取值区间。其中 P^i_{ij} 为第 i 个基因位取等位基因集合 $\{a_{i1}, a_{i2}, \dots, a_{in_i}\}$ 中第 j 个值的取值区间上限,表示为 $P^i_{ij} = \sum_{k=1}^j p_{ik}$, 显然 $P^i_{ij} \leq P^i_{i(j+1)}$, 且有 $P^i_{i1} = P_{i1}$, $P^i_{in_i} = 1$ 。改进后的基因选择算法虽然多了根据 P 计算 P^i 的时间开

销,但在选择基因位的等位基因时可采用二分查找,在相同条件下生成每一代个体种群所需基因选择时间为 $\sum_{i=1}^N n_i + \sum_{i=1}^N \log_2(n_i) * M$, 在 n_i 较大时优于原有算法。

3 结语

基于学分制系统的期末排考是一个复杂的系统工程问题,本文首次提出采用 PBIL 算法对高校排考问题的求解进行了尝试,并对 PBIL 算法中的基因选择算法提出了改进。计算经验表明,PBIL 算法在排考问题中收敛速度较快且稳定,对考试安排复杂约束条件的增减具有较好的适应性。

参考文献

- 1 Baluja S. Population-Based incremental learning. Technical Report, CMU-CS-94-163, CarnegieMellon University, 1994.
- 2 金炳尧,蔚承建,何振亚.一个用于优化搜索的学习算法.软件学报,2001,12(3):448-453.
- 3 金炳尧,马永进,骆红波.基于 PBIL 算法的自动组卷系统研究.计算机工程与科学,2005,27(10):48-49.
- 4 胡琨元,崔建江,郑秉霖,汪定伟,庞哈利.基于信息熵的自适应 PBIL 算法及其应用.系统仿真学报,2003,15(8):1175-1178.
- 5 田岭.大学自动排考算法设计与实现.计算机工程与设计,2007,28(10):2443-2445.
- 6 黄勇,苏守宝.一种新的高校自动排考算法.计算机技术与发展,2007,17(12):210-212.
- 7 张华,王秀坤,孙焘.蚁群算法在考试安排中的应用.计算机工程与设计,2003,24(12):62-64.