

# 基于Optuna优化的遗传算法智能组卷模型

常 宸, 胡安波, 高 鹏

(中央军委政治工作部 军事人力资源保障中心, 北京 100034)

**摘要:** 为了提升智能组卷性能, 同时解决常规的基于遗传算法优化的智能组卷模型在实际应用时参数难以确定, 在面对不同规模和特征分布的题库时性能不稳定的问题, 提出了基于Optuna优化的遗传算法智能组卷模型。通过设计分层格雷编码来克服传统二进制、十进制编码引发的汉明悬崖问题, 通过Optuna优化自反馈确定遗传算法的种群规模、迭代次数及其他参数, 动态调整遗传算法的交叉、变异速率, 实现对组卷搜索空间的自适应调整。实验结果表明, 所提模型能够有效确定参数并实现动态调整, 组卷质量优于其他基于随机和启发式算法的智能组卷模型。

**关键词:** 智能组卷; 遗传算法; Optuna

**中图分类号:** TP311

**文献标识码:** A

**doi:** 10.62756/jnuc.issn.1673-3193.2023.11.0009

**引用格式:** 常宸, 胡安波, 高鹏. 基于Optuna优化的遗传算法智能组卷模型[J]. 中北大学学报(自然科学版), 2025, 46(2): 208-218.

CHANG Chen, HU Anbo, GAO Peng. Intelligent exam paper generation model based on optuna optimized genetic algorithm[J]. Journal of North University of China(Natural Science Edition), 2025, 46(2): 208-218.

## Intelligent Exam Paper Generation Model Based on Optuna Optimized Genetic Algorithm

CHANG Chen, HU Anbo, GAO Peng

(Military Human Resource Support Center, Political Work Department of Central Military Commission, Beijing 100034, China)

**Abstract:** In order to improve the performance of intelligent exam paper generation, and solve the problem that the parameters of the genetic algorithm based model are difficult to determine in practice, and the performance is unstable in the face of question banks with different sizes and feature distributions, an intelligent exam paper generation model based on Optuna optimized genetic algorithm was proposed. By designing hierarchical gray coding, the Hamming cliff problem caused by traditional binary and decimal encoding methods was overcome. The population size, number of iterations, and other parameters of the genetic algorithm were determined by Optuna optimization self-feedback model, and the crossover and mutation rate of the genetic algorithm were dynamically adjusted to achieve adaptive adjustment of the exam paper generation search space. Experimental results show that the proposed algorithm can effectively determine the parameters, and realize dynamic adjustment. Final exam paper generation quality is better than other models based on random and heuristic algorithms.

**Key words:** intelligent exam paper generation; genetic algorithm; Optuna

收稿日期: 2023-11-13

基金项目: 国家社会科学基金重点项目(2022-SKJJ-B-072); 国防科技战略先导计划(21-ZLXD-02-00-02-006-38)

作者简介: 常宸(1994-), 男, 工程师, 硕士, 主要从事军事人力资源管理的研究。E-mail: en.ang@outlook.com。

## 0 引言

教育公平是社会公平的重要组成部分,而考试公平更是教育公平的核心内容。考试试卷是否科学直接关系到考试公平,也影响着教育质量评估<sup>[1]</sup>。长期以来,考试组织者需要花费大量时间和精力组成一份合理、科学、公平、完备的考试试卷,且随着题库规模不断扩大,组卷要求和约束条件不断多样,组卷难度也不断提升,依靠信息化智能化手段组卷成为了重要的研究课题。当前,自动化组卷主要包括基于随机、回溯试探和启发式算法的 3 类组卷模型。

基于随机算法的组卷方法<sup>[2-4]</sup>使用固定规则,从题库中随机抽取一定数量试题,在满足组卷约束条件的情况下组成试卷,具有逻辑简单、思路清晰、容易实现等优点,是当前各类组卷系统普遍使用的组卷算法。但该类算法不可回溯,必须按照前期指定的组卷目标反复尝试,题库规模大、条件约束多的情况下极易导致反复抽题和组卷失败的情况。

基于回溯试探算法的组卷方法<sup>[5-6]</sup>在随机算法基础上引入了验证过程。当本次抽取试题无法满足组卷要求且未完成组卷目标时,返回上次状态并重新抽题,直到完成组卷。回溯试探能够对所有状态结果进行遍历,对于小规模试题库组卷成功率高,能够有效避免反复搜索的问题,但是在实际组卷应用中,针对庞大的题库,该类算法组卷的时间和空间复杂度会非常高,若添加多个组卷目标,回溯试探过程设计将会非常复杂甚至无法实现。

基于启发式算法的组卷方法将组卷问题转化为优化问题来求解<sup>[7]</sup>,Yang<sup>[8]</sup>、Bui<sup>[9]</sup>和 Nguyen<sup>[10]</sup>等分别提出了基于模拟退火算法<sup>[11]</sup>和粒子群优化算法<sup>[12]</sup>的智能组卷方法。遗传算法<sup>[13]</sup>作为一种模仿生物进化演变过程的优化算法,在智能组卷问题中不断发展并得到更广泛应用。Jia 等<sup>[14]</sup>在遗传算法中引入精英策略,改进了交叉和变异算子,证明了遗传算法应用于智能组卷问题时的有效性。Han 等<sup>[15]</sup>提出了基于单目标优化的遗传算法智能组卷方法。Nie 等<sup>[16]</sup>通过优化遗传算法种群初始化、编码、交叉、变异等过程,有效提高了组卷成功率和效率。Song<sup>[17]</sup>和 Zhang<sup>[18]</sup>等使用实数编码而不是传统的二进制编码来提高方法的收敛速度。Tang 等<sup>[19]</sup>利用动态参数的自适应调整方法

和精英策略,有效提高了试卷生成的效率。Anastasiu 等<sup>[20]</sup>将许多具有代表性的关键字与测试相关联,用户通过设置许多关键字来表达组卷要求和限制条件,这些关键字接近于想要测试的主题。Teo<sup>[21]</sup>、Langi<sup>[22]</sup>、Shanthi<sup>[23]</sup>和 Abd<sup>[24]</sup>等不断拓展智能组卷的场景,引入试题难度要求、时间要求、考察方向等约束条件,丰富了基于遗传算法的智能组卷方法。Song<sup>[25]</sup>、Zhang<sup>[26]</sup>和 Chen<sup>[27]</sup>等使用遗传算法在线生成试卷,可以根据难度、知识覆盖面和题目比例等条件自动合成测试的内容。

基于启发式算法的组卷方法有效提高了组卷性能,但存在以下问题需要研究突破:

1) 应用场景受限。虽然部分学者提出了不同场景下的智能组卷方法,但大多数方法只关注特殊学校或机构面临的组卷问题,难以满足大规模考试智能组卷的要求,需要研究设计包含难度、时间、知识点覆盖度等各方面目标和时间、题型、分数等各方面约束,更加贴合实际考试场景的智能组卷方法。

2) 性能不稳定。启发式算法的超参数影响方法性能,当前基于启发式算法的智能组卷方法取得了相对比较好的效果,但在实际应用时受参数影响大,而考试工作者由于不具备算法专业知识,难以直接确定方法最优参数甚至无法确定大致范围,导致实际应用时,方法的性能无法达到论证时有关学者提出的最优性能。

3) 无法自动组卷。启发式算法作用的发挥需要考试工作者花费大量时间学习方法原理,或者反复尝试多组参数组合方案,造成时间浪费,且最终组卷质量不高。当前有众多启发式智能组卷方法,但开发实际组卷系统时,几乎全部的系统仍然是基于随机或回溯方法,未能实现智能化组卷的真正落地应用或落地效果不尽如人意。

本文提出一种基于 Optuna 优化的遗传算法智能组卷模型。首先将多目标智能组卷问题转化为多约束单目标搜索优化问题,然后以格雷编码的染色体表达种群组卷方案,基于时间、难度、知识点等要求构造适应度函数,设计自适应遗传算法搜索可能的组卷空间,搜索出的最优解即为模型得出的最终组卷方案。本文的主要贡献总结如下:①采用格雷编码表达组卷方案,有效克服了二进制、十进制编码容易产生的汉明悬崖问题,提高了组卷质量;②采用 Optuna 确定种群规模、迭代次数等遗传算法参数初始值,组卷人员无需掌握算法专业知识便可合理确定参数,也不需要反复尝试大量参数组合方案,

保证相关模型能够发挥最大性能,有效降低了模型使用门槛,更贴近实际场景,更有益于应用实践;③对比其他已有智能组卷方法,所提模型在智能组卷时具有一定的优势。

## 1 问题描述

智能组卷旨在从海量试题库中根据考试需求智能选取试题并组成高质量试卷,通常被定义为单目标多约束的组合优化问题。试题库中的每道试题通常包括编号、分数、难度、预计答题时间和对应的知识点等5个属性。假设试题库为矩阵 $A_{N \times 5}$ 。

$$A_{N \times 5} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{15} \\ a_{21} & a_{22} & \cdots & a_{25} \\ \vdots & \vdots & \vdots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{N5} \end{bmatrix}, \quad (1)$$

式中: $N$ 为试题库中试题的个数; $a_{n1}(1 \leq n \leq N)$ 到 $a_{n5}(1 \leq n \leq N)$ 分别为第 $n$ 道试题的5个属性。具体地, $a_{n1}(1 \leq n \leq N)$ 为第 $n$ 道试题的编号; $a_{n2}(1 \leq n \leq N)$ 为第 $n$ 道试题的分值,根据不同题型具体确定; $a_{n3}(1 \leq n \leq N)$ 为第 $n$ 道试题的难度,分为难、中、易等,可根据具体场景、题库特征和考试要求灵活调整; $a_{n4}(1 \leq n \leq N)$ 为第 $n$ 道试题的预计答题用时; $a_{n5}(1 \leq n \leq N)$ 为第 $n$ 道试题考察的知识点。

组卷问题转化为针对试题库矩阵 $A$ 的优化问题,通过指定目标和约束自动选取其中的部分试题,得到试卷矩阵 $B_{M \times 5}$ 。

$$B_{M \times 5} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{15} \\ b_{21} & b_{22} & \cdots & b_{25} \\ \vdots & \vdots & \vdots & \vdots \\ b_{M1} & b_{M2} & \cdots & b_{M5} \end{bmatrix}, \quad (2)$$

式中: $M$ 为试卷题目的数量; $b_{m1}(1 \leq m \leq M)$ 到 $b_{m5}(1 \leq m \leq M)$ 同样为组卷结果中第 $m$ 道试题的5个属性。

在此基础上,组卷过程应满足以下约束条件:

1) 总分约束。组卷结果所有试题得分的和等于组卷总分目标

$$S_z = \sum_{m=1}^M b_{m2}, \quad (3)$$

式中: $S_z$ 为组卷总分目标; $b_{m2}$ 为第 $m$ 道试题的分数。

2) 题目数量约束。组卷结果中各题型题目的数量等于预设目标数量

$$N_j = \sum_{m=1}^M c_m,$$

$$c_m = \begin{cases} 1, & \text{第}m\text{道题为第}j\text{个题型,} \\ 0, & \text{第}m\text{道题不为第}j\text{个题型,} \end{cases} \quad (4)$$

式中: $N_j$ 为组卷结果中第 $j$ 个题型的目标数量; $j$ 为试题库中包含的题型种类数, $j=1,2,\dots,J$ 。

3) 试题唯一性约束。组卷结果中每道试题只出现1次。

除此之外,组卷过程还包括时间、难度和知识点覆盖情况等约束,但在实际组卷中这些约束条件难以全部满足,将其定义为目标函数,以求最大限度地满足相关约束,组成高质量试卷。

1) 难度目标。组卷结果各难度分数与预设组卷难度分数最大限度地一致。

$$f_1 = \sum_{d=1}^D \frac{S_d - \sum_{m=1}^M c_m b_{m2}}{S_d} / D,$$

$$c_m = \begin{cases} 1, & \text{第}m\text{个试题为第}d\text{个难度,} \\ 0, & \text{其他,} \end{cases} \quad (5)$$

式中: $d$ 为难度层次的个数, $d=1,2,\dots,D$ ; $S_d$ 为第 $d$ 个难度层次预设的分数目标; $b_{m2}$ 为第 $m$ 道试题的分数。 $f_1 \geq 0$ ,当 $f_1=0$ 时,组卷结果完全满足预设难度目标。

2) 时间目标。组卷结果所有试题的用时总和与预设组卷总用时最大限度地一致。

$$f_2 = 1 - \frac{|t - \sum_{m=1}^M b_{m3}|}{t}, \quad (6)$$

式中: $t$ 为预设组卷总用时; $b_{m3}$ 为第 $m$ 道试题的预计用时。 $f_2 \geq 0$ ,当 $f_2=0$ 时,组卷结果完全满足预设时间目标。

3) 知识点覆盖目标。组卷结果中不同知识点的分数与预设各知识点分数最大限度地一致。

$$f_3 = \sum_{k=1}^K \frac{|S_k - \sum_{m=1}^M c_m b_{m2}|}{S_k} / K,$$

$$c_m = \begin{cases} 1, & \text{第}m\text{个试题考察知识点}k, \\ 0, & \text{其他,} \end{cases} \quad (7)$$

式中: $k$ 为涉及知识点的总个数, $k=1,2,\dots,K$ ; $S_k$ 为第 $k$ 个知识点预设的分数目标; $b_{m2}$ 为第 $m$ 道试题的分数。 $f_3 \geq 0$ ,当 $f_3=0$ 时,组卷结果完全满足知识点覆盖考察的要求。

为了便于求解,采用权重系数法将多目标优化问题转化为单目标优化问题,模型最终的目标函数为

$$f = \omega_1 f_1 + \omega_2 f_2 + \omega_3 f_3, \quad (8)$$

式中:  $\omega_1=0.3, \omega_2=0.4, \omega_3=0.3$  为权值系数,表示组卷对不同目标的侧重。 $f \geq 0$ , 当  $f=0$  时,组卷结果完全满足 3 个目标的要求。

## 2 模型设计

本文基于 Optuna 优化的遗传算法智能组卷模型,首先将智能组卷问题转化为多条件单目标的组合优化问题,然后在传统基于遗传算法智能组

卷模型的基础上引入格雷编码、Optuna 优化等过程,从而提高了组卷性能。主要步骤如下:

Step1: 初始化参数。确定种群规模、迭代次数、遗传概率、交叉概率的取值范围及精度,以及不同题型题目编号的上限和下限。

Step2: 初始化种群。基于格雷编码,随机产生符合组卷目标和约束的多个染色体,用于表示初始的组卷方案。

Step3: 智能组卷。基于 Optuna 优化遗传算法参数后,遗传算法通过选择、交叉、变异等操作优化组卷结果,经过多轮迭代后输出方法的最优参数和最终组卷结果。流程如图 1 所示。

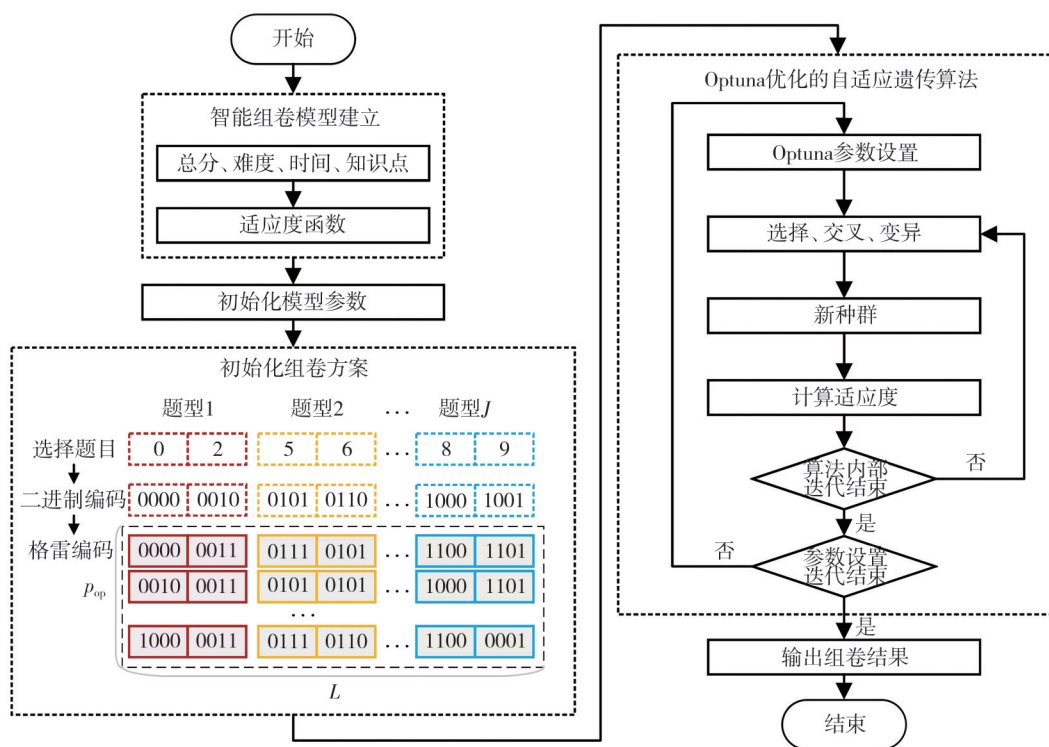


图 1 基于 Optuna 优化的遗传算法智能组卷模型流程图

Fig. 1 Flowchart of intelligent exam paper generation model based on Optuna optimized genetic algorithm

### 2.1 初始化种群及编码

初始化过程随机产生种群规模的大小,将符合约束条件的染色体空间作为优化问题解空间。不同编码方式决定染色体的组合方式,且对后续选择、交叉、变异算子产生影响。二进制编码方式简单,但计算量大、编码解码频繁。十进制编码使用题目变量的真实值为编码,染色体每个基因值用某一范围内的整数表示,个体的编码长度等于试卷题目数量,全局搜索能力强,不易陷入局部极值,收敛速度快。本文采用格雷编码,有效克服了二进制、十进

制编码产生的汉明悬崖问题。

编码过程首先计算第  $t$  个题型在二进制编码时的染色体长度  $L$ 。

$$L = \sum_{j=1}^J L_j = \sum_{j=1}^J \lceil \log(b_{u_j} - b_{l_j} + 1) \rceil, \quad (9)$$

式中:  $L_j$  为第  $j$  个题型二进制编码的长度;  $b_{u_j}$  为第  $j$  个题型编号的上限;  $b_{l_j}$  为第  $j$  个题型编号的下限。假设得到的染色体二进制编码为  $X = \langle x_L, x_{L-1}, \dots, x_0 \rangle$ , 其对应的格雷编码  $Y = \langle y_L, y_{L-1}, \dots, y_0 \rangle$  为

$$y_i = \begin{cases} x_1 & i=1, \\ x_{i-1} \oplus x_i & i>1. \end{cases} \quad (10)$$

最后,随机生成行数为 $p_{op}$ ,列数为 $L$ ,元素为0/1的二维矩阵作为染色体初始种群。每个染色体为一组基因,由各个题型的基因合并组成。

## 2.2 选择算子

采用轮盘赌算子,根据个体的适应度值计算个体在子代中出现的概率,按照概率随机选择个体构成子代种群。

## 2.3 交叉算子

设计分层次两点交叉策略,在各个题型的基因型内部随机设置交叉点,并以此为间隔,以交叉概率 $p_c \in [0, 1]$ 交换基因型。通过分层设计,能够保证交叉前后题目仍属于当前题型。

## 2.4 变异算子

交叉完成后,设计分层多点变异策略,设定变异概率 $P_m \in [0, 0.05]$ ,每次变异对每一位产生随机数 $N_{rand} \in [0, 1]$ ,当 $N_{rand} \leq P_m$ 时,该位取反。当变异后的题型编号超过当前题型的范围时,重新执行变异操作。当交叉、变异导致组卷方案产生重复题目时,从题库中未被选择的对应题型的题目中随机选择相同数目的题目替换重复题目,直至题目不重复。

## 2.5 Optuna参数初始化

选择、交叉、变异完成后,计算遗传算法本轮组卷方案适应度,并不断迭代,重复以上过程。本轮遗传算法迭代结束后,为了降低方法使用门槛,同时确保遗传算法发挥最优性能,在种群规模、交叉概率、变异概率这3个关键参数取值范围及优化精度控制下,引入Optuna框架<sup>[28]</sup>对相关超参数进行自适应调整优化。向Optuna框架返回当前参数选择方

案下的组卷性能,并调整参数,重新进入遗传算法组卷过程,直至方法收敛或Optuna框架达到最大迭代次数,输出最终组卷方案。

## 3 实验分析

### 3.1 实验设计

考虑保密原因,本文随机生成仿真题库进行实验,采用目标函数作为评价指标,将所提方法(O-GA)分别与近年来研究学者提出的基于粒子群算法(PSO-2018<sup>[10]</sup>)、模拟退火算法(SA-2015<sup>[8]</sup>)、实数编码遗传算法(RCGA-2018<sup>[26]</sup>)和差分进化算法(DE-2019<sup>[29]</sup>)的智能组卷方法进行对比。为体现一般性,所有实验重复执行5次,并计算均值和标准差,综合比较5次结果。

粒子群算法:模拟鸟群觅食过程中的迁移和群聚行为,与遗传算法类似,通过系统初始化随机解,追随最优粒子迁移并不断迭代,最终找到最优解,包含惯性权重、个体记忆和集体记忆等3个超参数。

模拟退火算法:从某一较高初温出发,伴随温度参数的不断下降,结合一定的概率突跳特性在解空间中随机寻找目标函数的全局最优解,包含初始温度、链长、冷却耗时等3个超参数。

差分进化算法:来源于遗传退火算法,使用浮点数替换位串编码并用算术运算替换逻辑运算,过程中引入差分变异操作,具有原理简单、超参数少、鲁棒性强的优点。

### 3.2 实验数据与组卷目标

为全面测试所提方法的性能,涵盖不同规模的应用场景,本文模拟产生不同规模,包含不同题型、难度、答题时间和知识点的仿真题库,题库的各项信息如表1所示。

对于不同题库,设置不同的组卷目标,具体如下如表2所示。

表1 仿真题库信息

Tab.1 Simulation question bank information

题库名称	题库规模	包含题型个数	包含难度个数	包含时间个数	包含知识点个数
QB1、QB2、QB3	400	4	3	10	10
QB4、QB5、QB6	1000	5	4	20	10

表2 不同仿真题库对应的组卷目标

Tab.2 The target of the paper generation to the different simulation question banks

题库名称	题型数量约束	难度目标	知识点目标	时间目标/min	总分约束
QB1、QB2、QB3	10/10/10/4	30/40/30	每个知识点10分	100	100
QB4、QB5、QB6	10/10/10/4/2	20/30/30/20	每个知识点5分	100	100

以仿真题库 QB1 为例, 模拟产生题目总数为 400 的题库, 并希望组成总分为 100 的试卷。题库包含单选、多选、判断、简答等 4 种题型, 组卷对各题型数量的约束分别设置为 10, 10, 10, 4 道题; 包含难、中、易等 3 种题目难度, 组卷对各难度分数的目标分别为 30, 40, 30 分; 包含 10 个不同的知识点, 希望组卷结果在各个知识点上的分数均为 10 分。组卷过程自动产生符合总分、题目数量、各题型数量约束, 并在难度、知识点覆盖度和答题时间上尽量符合要求的试题集合。

### 3.3 实验结果

#### 3.3.1 对比实验结果

为了发挥所有对比算法的最优性能, 同时确保公平, 对于 QB1~QB3, 设置所有方法的种群规模均为 50, 迭代次数均为 100, 对于 QB4~QB6, 设置所有方法的种群规模均为 100, 迭代次数均为 200。设置 PSO-2021 方法的惯性权重、个体记忆和集体记忆这 3 个参数分别为 0.5/1/1.5/2, 0.3/0.6/0.9, 0.3/0.6/0.9, 进行 36 组实验。设置 SA-2015 方法的初始温度、链长和冷却耗时这 3 个参数分别为 100/1 000/10 000, 200/300/400, 50/100/150, 进行 27 组实验。设置 RCGA-2018 方法的变异概率和交叉概率这 2 个参数分别为 0.1/0.3/0.6/0.9 和 0.1/0.3/0.6/0.9, 进行 16 组实验。为保证对比公平, 设置所提方法优化的种群规模范围为 40~50(精度为 2), 迭代次数为 80~100(精度为 2), 规模和迭代次数小于或等于其他对比算法。将对比算法各组最优结果作为各对比方法的最终结果, 与本文提出的 O-GA 智能组卷方法进行对比, 结果如表 3 所示。

从实验结果中可以得出以下结论:

1) 方法性能方面, 对于不同规模, 或者不同数据分布的题库, 不同方法表现有所差异, 参数设置影响算法性能, 全部对比方法在不同参数设置情况下的表现存在巨大差异, 总体上, 所提方法在全部仿真题库上的表现优于其他对比方法。对于组卷工作者, 在没有先验知识的情况下, 确定启发式算法合适的种群规模和迭代次数是重要且困难的。当题库规模扩大时, 由于拥有更多不同类型、不同属性的题目, 所提方法能够找到质量更优、符合组卷目标、考察能力更全面的组卷方案。

2) 方法效率方面, 对于全部基于启发式算法

的方法, 题库规模直接影响种群大小和迭代次数, 方法运行时间与种群规模呈正向相关, 总体上, 设置同等种群、同等迭代次数的情况下, DE-2019 是速度最快的, 其次是 RCGA-2018 和 O-GA, 最后是 SA-2015 和 PSO-2021, 但 DE-2019 和 RCGA-2018 的组卷质量远远低于其他 3 类方法, RCGA-2018 在所有数据集上均在变异率和交叉率最低的情况下取得最优结果, 表明该算法的优化算子并没有发挥很大作用, 推测这与实数编码产生的汉明悬崖问题有关。

表 3 对比实验结果

Tab. 3 Results of comparative experiments

题库名称	算法名称	目标函数均值	目标函数标准差	运行时间/s	总用时/s
QB1	DE-2019	0.195	0.020	8.480	8.480
	SA-2015	0.141	0.018	27.580	580.534
	RCGA-2018	0.150	0.028	4.410	80.479
	PSO-2021	0.142	0.038	3.894	171.051
	<b>O-GA</b>	<b>0.069</b>	<b>0.025</b>	<b>75.809</b>	<b>75.809</b>
QB2	DE-2019	0.219	0.019	8.530	8.530
	SA-2015	0.127	0.015	22.187	495.181
	RCGA-2018	0.172	0.022	4.612	81.105
	PSO-2021	0.155	0.015	4.566	176.070
	<b>O-GA</b>	<b>0.071</b>	<b>0.011</b>	<b>73.178</b>	<b>73.178</b>
QB3	DE-2019	0.195	0.020	8.480	8.480
	SA-2015	0.125	0.024	17.173	519.336
	RCGA-2018	0.161	0.037	4.125	72.539
	PSO-2021	0.180	0.021	5.049	176.641
	<b>O-GA</b>	<b>0.065</b>	<b>0.005</b>	<b>74.852</b>	<b>74.852</b>
QB4	DE-2019	0.153	0.020	7.263	7.263
	SA-2015	0.111	0.050	27.557	612.374
	RCGA-2018	0.137	0.017	3.945	73.138
	PSO-2021	0.086	0.035	15.847	880.565
	<b>O-GA</b>	<b>0.067</b>	<b>0.012</b>	<b>83.370</b>	<b>83.370</b>
QB5	DE-2019	0.152	0.014	7.309	7.309
	SA-2015	0.138	0.042	26.972	607.572
	RCGA-2018	0.146	0.049	20.403	67.681
	PSO-2021	0.113	0.045	17.548	974.971
	<b>O-GA</b>	<b>0.054</b>	<b>0.003</b>	<b>84.003</b>	<b>84.003</b>
QB6	DE-2019	0.025	0.154	7.082	7.082
	SA-2015	0.136	0.008	27.656	623.021
	RCGA-2018	0.135	0.029	3.970	72.398
	PSO-2021	0.082	0.015	17.615	976.696
	<b>O-GA</b>	<b>0.056</b>	<b>0.008</b>	<b>85.871</b>	<b>85.871</b>

3) 方法稳定性方面, 所有方法独立运行 5 次的标准差均在 0.03 以内, 由于使用 Optuna 进行参数自适应优化, 除题库 QB1 外, 所提方法的稳定性都优于其他所有对比方法, 这也保证了所提方法能够稳定应用于不同规模不同数据分布的题库和考试场景, 有效解决了相关类型算法在落地使用时需要人工调整参数以确保性能最优的问题。

相较于其他启发式算法存在的收敛速度慢、

参数敏感、收敛精度低等问题,遗传算法的选择操作有效排除了劣质组卷方案,保留了优秀方案,交叉和变异算子保证了遗传算法具备更优的全局搜索能力,且不容易陷入局部最优,使搜索过程向更优的组卷方向进化。通过引入格雷编码和Optuna自适应性参数优化保证了遗传算法能够发挥最优性能,一定程度上改善了收敛速度和精度的问题,进一步提升了组卷算法整体的性能、效率和稳定性。

### 3.3.2 Optuna优化过程

以QB1题库1次运行结果为例(全部题库独立运行5次的结果规律类似,不作展示),图2所示为Optuna框架优化过程中方法性能的变化。图3和图4所示分别为Optuna框架得出的超参数重要性及对方法目标函数损失值的影响。

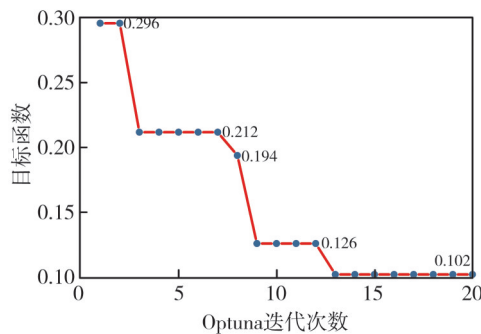


图2 Optuna框架优化过程中方法性能的变化

Fig. 2 Changes in method performance during Optuna framework optimization

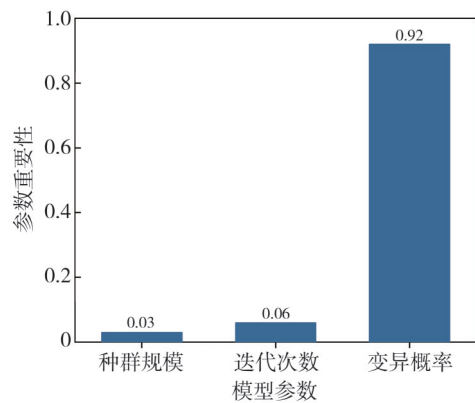


图3 Optuna框架得出的超参数重要性

Fig. 3 Hyperparameter importance derived from the Optuna framework

通过Optuna不断优化遗传算法的种群规模、迭代次数和变异概率这3个参数,经过20次迭代,组卷目标函数由0.296下降至0.102,方法性能得到显著提升,过程中不需要工作人员根据先验知识设置方法参数,算法整体运行时间约1 min,与同类型算

法相比,速度也具备优势,远小于人工调试参数使其发挥最优性能的时间(其他所有算法调整参数运行时间合计均在半天左右)。由图3和图4可以看出,3个参数对方法性能均有不同程度的影响,且规律难以直接掌握,变异概率对方法性能的影响相对较大,此项参数设置更加重要。

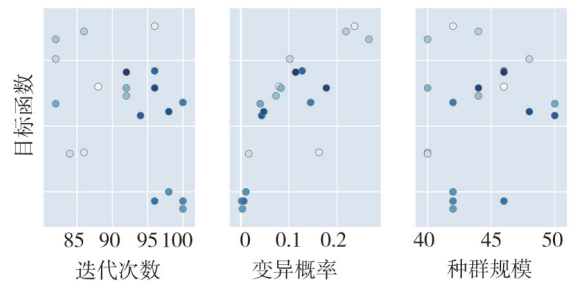


图4 Optuna框架得出的超参数对损失值的影响

Fig. 4 The effect of the hyperparameters derived by the Optuna framework on the loss value

### 3.3.3 基于格雷编码的遗传算法优化过程

使用独立运行的5次实验中,经过Optuna优化后方法产生最优目标函数的参数进行单独的遗传算法优化,并跟踪优化过程中目标函数的变化。表4所示为不同题库,在方法产生的最优参数作用下,组卷产生的目标函数值及对应的组卷结果。

以QB1为例,图5所示为不同题库方法优化过程中种群内染色体目标函数和方法总体目标函数的变化,对于其他题库,呈现类似的规律,不作展示。

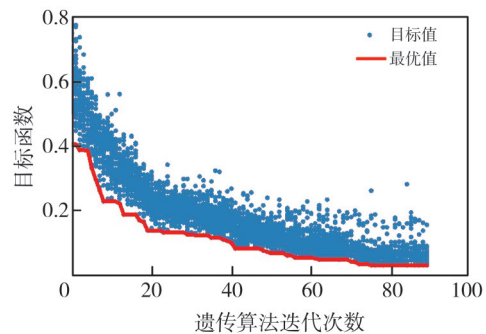


图5 遗传算法优化过程

Fig. 5 Genetic algorithm optimization process

图5中,不同蓝色散点分别表示遗传算法迭代过程中种群中染色体对应目标值的变化,红色折线表示算法整体最优值的变化。可以看到,随着遗传算法不断迭代,目标函数值不断降低,对于所有方法,由0.4左右下降至0.1以下,优化过程是有效的。

### 3.3.4 遗传算法参数敏感性分析

为验证遗传算法参数对组卷结果的影响,以题库QB1为例,设置遗传算法的种群规模为10~

190(以 20 为间隔), 迭代次数为 10~190(以 20 为间隔), 变异概率为 0.001~0.19(以 0.002 为间隔), 进行 1 000 组实验, 每组实验重复 5 次并取 5 000 次实验的平均值, 此处仅展示分别固定变异

概率为 0.009、种群规模为 90、迭代次数为 90 时, 本文方法得到的目标函数均值、标准差和运行时间结果, 如图 6~图 8 所示。其他组的实验结果结论与此一致, 不作展示。

表 4 使用最优参数的遗传算法优化组卷结果

Tab. 4 Optimize paper generation results using genetic algorithm with the optimal parameters

题库名称	种群规模	迭代次数	变异概率	目标函数	组卷结果(题目编号)
QB1	50	90	0.002	0.03	题型 1: 20/21/24/32/41/57/63/80/91/99 题型 2: 111/114/116/121/123/151/157/167/196/197 题型 3: 202/208/248/268/272/276/280/282/292/299 题型 4: 314/319/375/377
QB2	42	84	0.002	0.076	题型 1: 17/20/23/35/51/52/66/77/83/99 题型 2: 101/111/112/116/133/141/157/196/197/199 题型 3: 205/208/213/217/236/259/266/285/291/299 题型 4: 309/319/372/379
QB3	46	88	0.006	0.032	题型 1: 1/46/55/63/77/78/85/86/95/96 题型 2: 104/107/108/110/118/126/152/166/181/199 题型 3: 202/213/236/246/256/281/283/288/291/299 题型 4: 331/365/375/399
QB4	40	100	0.005	0.024	题型 1: 38/39/66/92/125/129/138/140/141/183 题型 2: 215/231/240/247/312/315/331/360/374/399 题型 3: 400/414/421/422/423/445/482/504/522/559 题型 4: 649/668/678/771 题型 5: 859/965
QB5	48	80	0.003	0.018	题型 1: 28/43/65/67/91/112/119/153/194/199 题型 2: 200/218/225/226/336/370/371/372/381/394 题型 3: 413/428/437/445/469/473/509/529/537/566 题型 4: 751/759/765/779 题型 5: 805/999
QB6	48	84	0.002	0.012	题型 1: 19/69/127/133/136/150/168/187/188/195 题型 2: 231/261/288/296/302/309/310/369/376/399 题型 3: 451/454/494/534/560/568/576/578/586/599 题型 4: 617/689/759/772 题型 5: 834/900

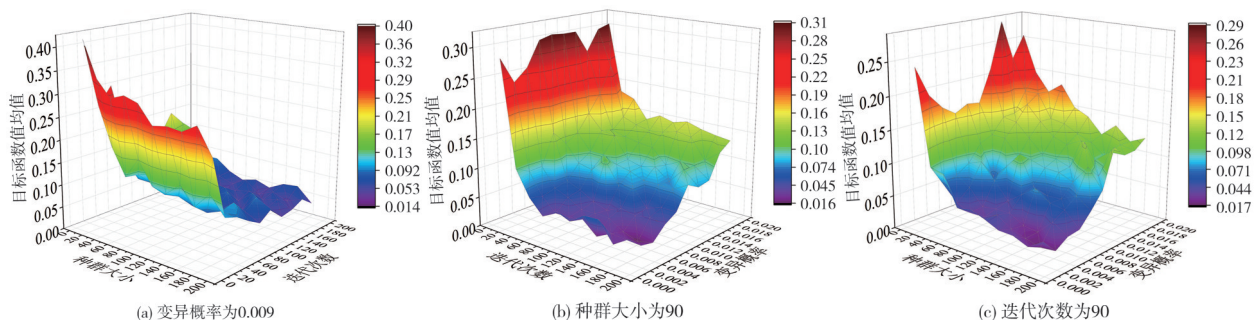


图 6 遗传算法参数与目标函数值均值(以题库 QB1 为例)

Fig. 6 Mean values of genetic algorithm parameters and objective function (taking QB1 as an example)

由图 6~图 8 可知, 3 个参数综合影响方法的性能, 目标函数均值在 0.447 2~0.008 范围波动, 目标函数值标准差在 0.01~0.1 范围波动, 在没有有效的参数选择手段或具备一定专业知识可以快速调试方法所需参数的情况下, 相关方法在落地

应用时可能效果不理想。时间方面, 随着种群规模和迭代次数的增加, 方法运行时间增加, 这与一般想法一致, 参数选择的关键是找到相对合适的, 与题库特征相适应的种群规模、迭代次数和变异概率, 以发挥最优性能。就 QB1 而言, 通过

1 000组实验,找到最优的目标函数是0.008,而所提方法找到的最优的目标函数是0.069,位于1 000组实验结果的第123位,相比较将近10 h的运行时间,算法利用约1 min的时间将参数定位到

相对合适的位置,结果也相对理想。当然,在时间允许的情况下,通过设置多组实验遍历所有参数组合,的确能找到更优的结果,但时间代价是非常巨大的。

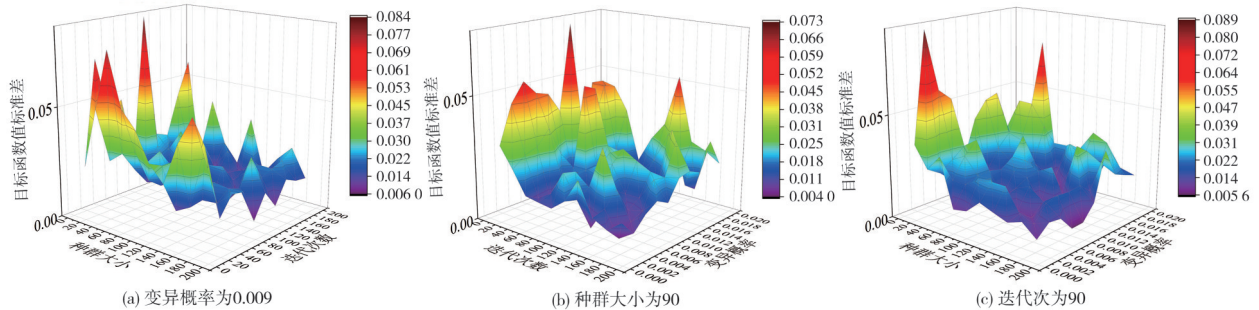


图7 遗传算法参数与目标函数值标准差(以题库QB1为例)

Fig. 7 Standard deviation of genetic algorithm parameters and objective function (Taking QB1 as an example)

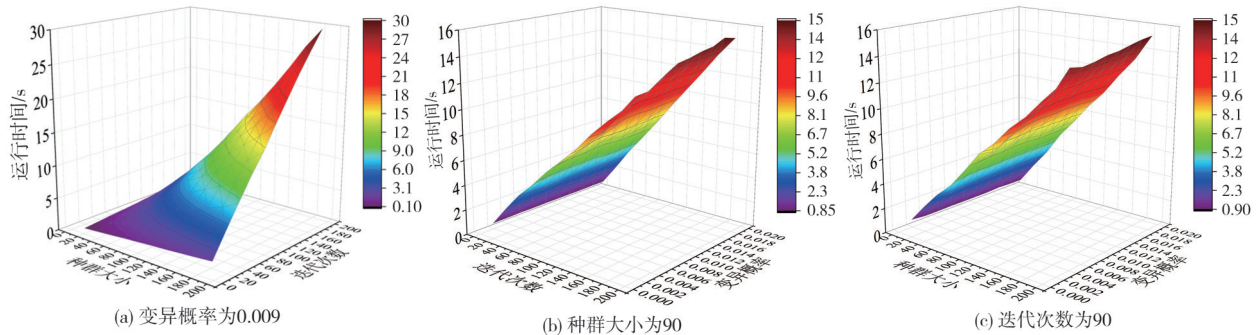


图8 遗传算法参数与运行时间(以题库QB1为例)

Fig. 8 Run time of genetic algorithm parameters and objective function (Taking QB1 as an example)

### 3.3.5 方法鲁棒性分析

为验证方法的鲁棒性,以题库QB1为例,设置种群规模为50,迭代次数为90,变异概率为0.002,重复进行100次组卷实验,变化统计箱形图如图9所示。

坏的情况下,方法表现与其他对比算法基本相当,在绝大多数情况下,本文方法远优于其他对比算法。

## 4 结论

本文设计了基于Optuna优化的自适应遗传算法智能组卷方法。将智能组卷问题转化为多条件多目标的组合优化问题,在传统遗传算法基础上引入格雷编码、Optuna优化等手段,提高组卷性能的同时优化了组卷速度,实现了启发式算法参数的自适应调整,保证了相关方法能够落地应用且发挥较优性能,形成了真正智能化可落地开发应用的智能组卷方法。实验结果表明,所提模型在性能、效率和稳定性方面优于其他对比算法,且引入的各个手段有效。

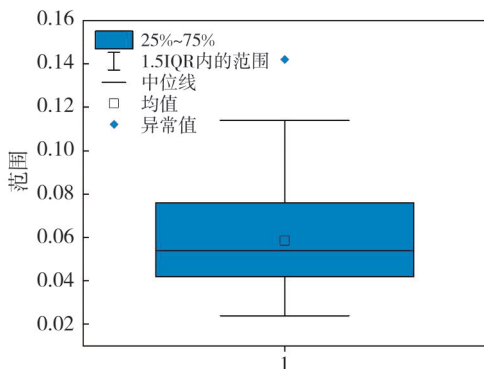


图9 方法鲁棒性测试实验结果(以题库QB1为例)

Fig. 9 Experimental results of method robustness test(Taking QB1 as an example)

由图9可知,100次组卷实验中方法组卷结果的目标函数波动范围约为0.02~0.14,大部分结果落在0.04~0.08之间,均值为0.05左右,在最

### 参考文献:

[1] 中国高考报告学术委员会. 高考评价体系解读(2023)[M]. 北京: 现代教育出版社, 2022.  
 [2] NAIK K, SULE S, JADHAV S, et al. Automatic

- question paper generation system using randomization algorithm[J]. *International Journal of Engineering and Technical Research*, 2014, 2(12): 192-194.
- [ 3 ] KIRAN F, GOPAL H, DALVI A. Automatic question paper generator system[J]. *International Journal of Computer Applications*, 2017, 166(10): 42-47.
- [ 4 ] RAO V S, SAI V C, SANDEEP S S, et al. Automated exam paper process based on schedule and authenticity [C]//The International Conference of Advance Research & Innovation, 2020.
- [ 5 ] EL-RAHMAN S A, ZOLAIT A H. Automated test paper generation using utility based agent and shuffling algorithm [J]. *International Journal of Web-Based Learning and Teaching Technologies*, 2019, 14(1): 69-83.
- [ 6 ] BANKAR M P, BHOR P, BHALERAO P, et al. Automated generation of question paper for online MCQ test by using shuffling algorithm [J]. *International Journal for Research Trends and Innovation*, 2018, 3(4): 2456-3315.
- [ 7 ] CHATTOPADHYAY S, MARIK A, PRAMANIK R. A brief overview of physics-inspired metaheuristic optimization techniques[DB/OL]. (2022-01-30)[2023-11-13]. <https://arxiv.org/abs/2201.12810>.
- [ 8 ] YANG J. Design on generating test paper based on simulated annealing algorithm [C]//The 2nd International Conference on Civil, Materials and Environmental Sciences. London, UK: Atlantis Press, 2015: 689-693.
- [ 9 ] BUI T, NGUYEN T, VO B, et al. Application of particle swarm optimization to create multiple-choice Tests[J]. *Journal of Information Science & Engineering*, 2018, 34(6): 1405-1423.
- [10] NGUYEN T, NGUYEN L T T, BUI T, et al. Multi-swarm optimization for extracting multiple-choice tests from question banks [J]. *IEEE Access*, 2021, 9: 32131-32148.
- [11] GELATT M P, VECCHI S, KIRKPATRICK C D. Optimization by simulated annealing [J]. *Science*, 1983, 220(4598): 671-680.
- [12] KENNEDY J, EBERHART R. Particle swarm optimization [C]//The ICNN' 95-International Conference on Neural Networks, 1995: 1942-1948.
- [13] HOLLAND J H. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*[M]. Cambridge: MIT Press, 1992.
- [14] JIA Z H, ZHANG C E, FANG H S. The research and application of general item bank automatic test paper generation based on improved genetic algorithms [C]//The 2011 IEEE 2nd International Conference on Computing, Control and Industrial Engineering, 2011: 14-18.
- [15] HAN L, LI X. The analysis of exam paper component based on genetic algorithm [C]//The 2014 Fourth International Conference on Communication Systems and Network Technologies. New York, USA: IEEE, 2014: 561-564.
- [16] NIE J. Research on optimizing intelligent test paper forming strategy based on improved genetic algorithms [C]//The Proceedings of the 2019 International Conference on Artificial Intelligence and Computer Science, 2019: 117-120.
- [17] SONG H F, YANG W W. The research of auto-composing test paper technology based on genetic algorithm [J]. *Applied Mechanics and Materials*, 2014, 519: 1188-1192.
- [18] ZHANG K, ZHU L. Application of improved genetic algorithm in automatic test paper generation [C]//The 2015 Chinese Automation Congress, 2015: 495-499.
- [19] TANG H T. Strategy for test paper composition based on genetic algorithm [J]. *Applied Mechanics and Materials*, 2014, 513: 1688-1691.
- [20] ANASTASIU D P, BOLD N, NIJLOVEANU D. A method based on genetic algorithms for generating assessment tests used for learning [J]. *Polibits*, 2016 (54): 53-60.
- [21] TEO N H, BAKAR N A, ABD R M. Representing examination question knowledge into genetic algorithm [C]//The 2014 IEEE Global Engineering Education Conference, 2014: 900-904.
- [22] LANGI A Z, ROSMANSYAH Y. Generation quiz with genetic algorithm based on bloom's taxonomy classification in serious game based virtual environments [C]//The 2016 International Conference on ICT For Smart Society, 2016: 42-48.
- [23] SHANTHI B S, HARSHITHA L J, MANASA K. Automated exam question generator using genetic algorithm [J]. *International Research Journal of Engineering and Technology*, 2019(6): 1687-1691.
- [24] ABD R T N T, ABD A Z, AB R H, et al. Automated exam question generator using genetic algorithm [C]//The 2017 IEEE Conference on e-Learning, e-Management and e-Services, 2017: 12-17.
- [25] SONG W. Online test paper composition based on genetic algorithm [C]//The 2018 3rd International

- Conference on Modelling, Simulation and Applied Mathematics, 2018: 158-161.
- [26] ZHANG Y, ZHANG J, WANG P. Research and implementation of intelligent test paper composition based on genetic algorithm[C]//The 2018 9th International Conference on Information Technology in Medicine and Education. , 2018: 552-556.
- [27] CHEN X, ZHONG D, LIU Y, et al. Auto-generating examination paper based on genetic algorithms [C]//The 14th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, 2020: 887-893.
- [28] AKIBA T, SANO S, YANASE T, et al. Optuna: A next-generation hyperparameter optimization framework [C]//The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019: 2623-2631.
- [29] 蔡佳. 差分进化算法在大学英语考试智能组卷中的研究应用[J]. 微型电脑应用, 2019, 35(7): 25-27.
- CAI Jia. Research and application of differential evolution algorithm in the intelligent test paper of college english test [J]. Microcomputer Applications, 2019, 35(7): 25-27. (in Chinese).