

一种改进的无死锁试卷生成算法

An Improved Algorithm of Forming Test Papers without Deadlock

刘玉龙 赵立江

(徐州师范大学计算机科学系 徐州221009)

Abstract This paper proposes an improved algorithm of forming a test paper. The algorithm can avoid any deadlock when several test papers must be formed simultaneously, and lead the least rate of reuse.

Keywords Algorithm, Multitasking, CBE, CAI courseware

1. 引言

作为计算机远程教育(网上教育)中一个必不可少的组成部分,教学评价系统(即考试系统)设计成败的关键在于网络环境下的题库管理系统的设计,这一设计主要应考虑以下几方面的问题^[1]:

(1)多卷并发试卷的时间冲突。一方面,由于网络用户数目的事先不确定性和题库中试题总量有限,可能产生多份试卷争夺试题资源的现象,从而形成时间冲突;另一方面,由于网络考试平台的开放性,对考试时间不能作统一要求,为保证试卷的保密,需对试题复用时间间隔加以一定的限制,从而也形成时间冲突。即某试题一旦被选用,该试题必须间隔一段时间后方可再次使用。

解决多卷并发试卷的时间冲突问题,仅靠简单的随机选择算法已不能解决问题,无法完成网络环境下多任务试卷的并发要求,而单纯靠增加题库题量的方法会增加开发成本,也不足取。

(2)多卷并发试卷的内容冲突。指的是在同一份试卷中,各种题型之间出现的试题内容重复。解决内容冲突是一个较为复杂的问题。可以建立一种冲突串解决机制^[2]以及对生成的试卷采用人工干预方法进行调整。

(3)多任务环境下的任务死锁。是指当同时有两个或两个以上试卷生成任务时,由于题库已空,任意一个试卷都无法生成,每一任务均需等待其它任务释放试题,从而形成死锁。关于多卷并发试卷的时间冲突和多卷并发试卷的内容冲突的解决,文^[2]介绍了几种解决多卷并发试卷的内容冲突的具体实现方案,包括增加试题库中试题的有效数量,以及使用人工干预和撤销较长时间内不能完成组卷的任务等。这几种方案对打破死锁的形成也有一定的效果。

从本质上讲,多卷并发试卷生成时的时间冲突、内容冲突及多任务死锁均由各任务对题库资源的争夺而产生,在题库容量不可能无限制增加的前提下,时间冲突、内容冲突与多任务死锁是不可能同时避免的。如何既能避免多任务死锁又能在解决时间冲突、内容冲突时使得试题复用率较低,就成为衡量此类算法性能优劣的标准之一。

文^[3]中给出了一种算法,只要试卷的容量 sm 不大于试题库的总容量 tm ,即使并发试卷的份数 n 超过试题库的总容量 tm ,也不会发生死锁现象,但试题复用率较高。

本文给出一种改进的无死锁试卷生成算法,既保持了无死锁特性,又大大降低了试题的复用率。

2. 改进的低重复率无死锁试卷生成算法

现行题库多按知识点、题型、难度的层次模式构造,所以在讨论多卷并发试卷生成算法时关键在于如何考虑好知识点、题型、难度系数均相同时的情形,即如何在同一类型的试题中抽取试题,而如何将抽取出的各种类型的试题组合成试卷,则相对简单,本文主要讨论前者,而且仍然采用文^[3]中使用的方法,在考虑试卷生成算法时仅对题库中的题号进行处理。现在,先分析一下文^[3]中提供的算法。

算法一(无死锁试卷生成算法):

- ①接考试1请求之后,清试卷题号库 i ;
- ②随机生成题号 tn ;
- ③判试卷题号库 i 中是否已有此题号,若有,转②;否则,转④;
- ④判试题题号库中是否有此题号,若有,则将该题号移入试卷题号库 i 后转⑤;若无,则将该题号复制到试卷题号库 i 后转⑤;
- ⑤判断新试卷题号库 i 是否已满,若否,转②;若是,

转⑫;

⑫生成试卷 i ;

⑬释放试卷题号库 i 中的题号并移回试题题号库内;

⑭第 i 次考试试卷生成结束。

算法一之所以产生较高试题复用率的原因在于:当试卷 i (即试卷题号库 i)需要题号 tn 而试卷库(即试题题号库)中已无此题号(该题号已被其它试卷使用)时,无论题库中是否还有其它可用试题,该算法一律将题号 tn 复制到试卷题号库 i 中去,从而引起对题号为 tn 的试题的重复使用,下面的算法对此进行了改进,当试卷 i 需要题号 tn 而试题库中已无此题号时,先检查试题库中是否还有其它试题,若有则重新抽题,若无,再将题号 tn 复制到试卷题号库 i 中去。算法如下:

算法二(改进的无死锁试卷生成算法):

①接受考试申请 i ;

②清卷库 i ;

③生成题号 tn ;

④判题库中是否有此题号 tn ,若有,将题号 tn 移入卷库 i 后转⑧;若无,转⑤;

⑤判卷库 i 中是否有题号 tn ,若有,转③;若无,转⑥;

⑥判题库是否为空,若不空,转③;若为空,转⑦;

⑦将题号 tn 复制到卷库 i 中,并给题号 tn 作复用标记;

⑧判卷库 i 是否已满,若不满,转③;若已满,转⑨;

⑨按照卷库 i 中的题号生成试卷 i ;

⑩释放卷库 i 中无复用标记的题号,清卷库 i 后等待新的考试申请。

算法二采取的策略是,只要试题库内还有未用试题,就不使用已经被其它试卷使用的试题,这样,只要试题库不空,就不会产生试题复用现象,从而大大降低了试题复用率。

若在算法二的基础上再考虑一下复用试题的优选,则可以进一步降低试题复用率。下面的算法三采用的策略是,当试题库为空时选择复用次数较少(复用次数最少的试题可能有多个)的试题。

算法三(低复用率无死锁试卷生成算法):

①接受考试申请 i ;

②清卷库 i ;

③生成题号 tn ;

④判题库中是否有此题号 tn ,若有,将题号 tn 移入卷库 i 后转⑧;若无,转⑤;

⑤判卷库 i 中是否有题号 tn ,若有,转③;若无,转⑥;

⑥判题库是否为空,若不空,转③;若为空,转⑦;

⑦判题号 tn 的复用次数是否最低,若最低,则将题号 tn 复制到卷库 i 中,并为题号 tn 作复用次数计数,然后转⑧;否则转③;

⑧判卷库 i 是否已满,若不满,转③;若已满,转⑨;

⑨按照卷库 i 中的题号生成试卷 i ;

⑩释放卷库 i 中无复用标记的题号,清卷库 i 后等待新的考试申请。

算法三的框图如图1所示。

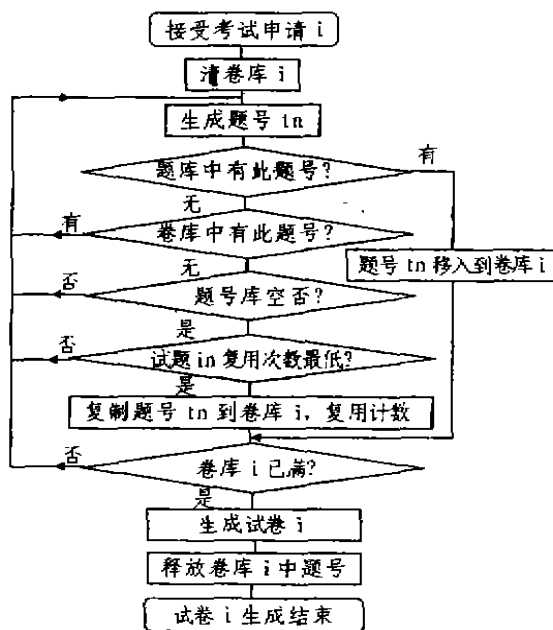


图1 低复用率无死锁试卷生成算法的框图

参考文献

- 1 马继红, 寒枫. 试题库管理系统的设计与实现. 计算机系统应用, 1994, 1
- 2 陆虹. 多任务并发在试题库管理系统中的应用. 计算机工程, 1999, 4
- 3 刘玉龙, 赵立江. 网络型题库开发过程中的若干问题. 计算机与教育, 重庆, 1999, 11