****

硕士学位论文

|  |
| --- |
| 基于协同候鸟前以算法的全自动免疫分析 |
| 检测设备的分批优化调度问题研究 |

|  |  |
| --- | --- |
| 作者姓名 |  |
| 学科专业 | 控制理论与控制工程 |
| 指导教师 |  |
| 所在学院 | 自动化科学与工程学院 |
| 论文提交日期 | 2019年4月 |

一、绪论

# 1.1 课题背景和意义

# 1.2 免疫分析检测设备调度问题的国内外研究现状

# 1.3 论文主要研究内容及其章节安排

二、免疫分析检测设备分批优化调度问题分析及建模

# 2.1 免疫分析检测设备相关知识

## 2.1.1 免疫检验基本原理

## 2.1.2 免疫检验设备基本组成

## 2.1.3 免疫检验流程

# 2.2 分批调度问题特点分析

## 2.2.1 问题描述

## 2.2.2 问题特点分析

# 2.3 柔性作业车间分批调度问题

## 2.3.1 柔性作业车间分批调度问题背景

## 2.3.2 柔性作业车间分批调度问题分类

# 2.4 免疫检测设备的柔性作业车间分批调度模型

# 2.5 柔性作业车间调度问题求解方法

三、候鸟迁移算法理论分析

# 3.1 基本候鸟迁移算法

# 3.2 候鸟迁移算法研究现状

# 3.3 候鸟迁移算法性能分析

## 3.3.1 算法优点

## 3.3.2 算法缺点

四、针对分批调度问题的约束处理策略

# 4.1 编码解码方案设计

## 4.1.1 免修补双矩阵编码

## 4.1.2 基于机器柔性灵活指数的解码方式

## 4.1.3 实验分析

# 4.2 搜索算子的设计

## 4.2.1 粗细粒度配合搜索

## 4.2.2 粗粒度交叉搜索算子

## 4.2.3 细粒度邻域搜索算子

4.2.3.1 单一邻域

4.2.3.2 启发式邻域

4.2.3.3 细粒度全邻域

4.2.3.4 策略邻域

4.2.3.5 实验分析

五、基于竞争式协同候鸟迁移算法的优化调度

# 5.1 引言

# 5.2 竞争式协同候鸟迁移算法

## 5.2.1 改进的V字飞行阶段

在迁移的鸟群里，前鸟对跟随它的跟随鸟有一种利益机制，使得跟随鸟可以得益于前鸟的抬升力而节省飞行体力。在基本MBO算法中，这种利益机制体现为，前鸟把自己优秀的邻域解与其跟随鸟共享，称为“邻域共享”。如果前鸟分享的邻域解很优秀，就会直接整个替换跟随鸟。在这种共享模式下，跟随鸟整体虽然不及前鸟的邻域解优秀，但是它或许会带有优秀的代码片段，如果直接被整个替换掉了，它所包含的优秀代码片段也会被丢弃。同时，如果跟随鸟被前鸟的优秀邻域整个替换了，跟随鸟来到了前鸟的邻域，它之前所在的邻域范围就少了一个个体，因此被挖掘的机会也减少了，如果这只跟随鸟是它所在邻域范围内唯一的一只鸟个体，那么该邻域范围就废弃了，就失去了被挖掘的机会。因此这种“邻域共享”会加剧种群多样性的减少，使搜索更容易陷入局部最优。

为了缓解“邻域共享”所带来的多样性急剧减少，也为了适当保留跟随鸟自身的优秀代码片段，本章提出使用“代码片共享”以取代“邻域共享”。具体来说，让跟随鸟与前鸟通过交叉，用二者的代码片重新构成一个新的解。新的解里面带有前鸟的代码片，也带有跟随鸟的代码片，体现“代码片共享”。这个新的解处于前鸟的邻域范围内，同时也处于跟随鸟的邻域范围内，即两个鸟邻域范围的交集。这种方式既能够体现利前鸟对于跟随鸟的利益机制，又能避免跟随鸟的信息被完全丢弃，使种群保持一定的多样性，保持一定活力，同时还避免跟随鸟所在的邻域被完全废弃，保持对解域的全局搜索，防止早熟收敛。

在CCMBO的V字飞行阶段里，每一只跟随鸟首先与前鸟进行次代码片共享，即交叉。每次交叉得到两个子代，选择较优的子代作为代码片共享的结果。次代码片共享之后得到个新的解，放入候选集里面待跟随鸟选择。与基本MBO相同，接下来跟随鸟自己做次的邻域搜索，得到的个邻域解也放入候选集里面。此时候选集里就有了个供当前跟随鸟选择的解，如果这个解之中最优解优于当前跟随鸟，那么就用最优解来更新当前跟随鸟。领头鸟的更新方式与基本MBO里的一样，都是把次邻域搜索得到的邻域解放入候选集，然后择优更新。一只跟随鸟更新的具体过程如下，以左翼队列中某一只跟随鸟为例：

step1：令当前鸟的候选集；

step2：令当前鸟与其前鸟交叉，得到两个子代，较优的子代放入候选集；

step3：如果候选集里解个体的数量少于，那么回到step2，否则进行step4；

step4：对当前鸟进行一次邻域搜索，得到一个邻域解，放入候选集；

step5：如果候选集里解个体的数量少于，那么回到setp4，否则进行step6；

step6：找出候选集 最优的解，如果比更优，那么用替换。

值得注意的是，对于本阶段的更新来说，同步更新和异步更新是有区别的。同步更新是指，所有跟随鸟，无论排在队伍中的第几个位置，都是在同一时刻，根据其前鸟来生成各自的候选集，同时使用各自的候选集更新当前跟随鸟。而异步更新指的是，对于V字型的左翼队列或者右翼队列，都是从第一个跟随鸟开始，按照队形顺序，前面一只鸟生成候选集并更新之后，后面一只鸟才能根据已更新的前鸟来生成自己的候选集并更新，按顺序直到排末尾的跟随鸟更新，整个种群的更新才完成。因为异步更新是在前鸟更新之后才去更新的，所以在一次种群的更新过程，前鸟的信息不仅可以传给紧跟着它的第一只鸟，还可能传给它的第二只、第三只跟随鸟，具有“多级传播”的特点。特别是当前鸟明显优于其后面连续几只跟随鸟的时候，这种“多级传播”会加剧种群多样性损失。而对于同步更新来说，在一次种群的更新过程中，前鸟的信息是不会被“多级传播”的，前鸟只对紧跟其后的一只跟随鸟具有影响力，不会加剧种群多样性损失，因此此处的V字飞行阶段采用同步更新的方式。V字飞行阶段需要进行次的迭代更新，整个种群采用同步更新一代的具体过程如下：

step1：领头鸟生成候选集，择优更新；

step2：对左翼队列所有鸟， 使用它们各自的前鸟生成各自的候选集

step3：使用各自的候选集对更新；

step4：对右翼队列所有鸟，用它们各自的前鸟生成各自的候选集，

step5：使用各自的候选集对更新。

## 5.2.2 减速调整阶段

当鸟群在领头鸟的带领下加速或者匀速飞行时，鸟群呈现出标准的V字队形。但是当领头鸟更替过程中导致鸟群减速的时候，鸟群的V字队形会被打乱。当领头鸟感到疲劳的时候，它会减慢飞行速度，退到鸟群里面，不再飞在鸟群的最前方。跟随鸟也各自调整自己的速度，由于每只鸟速度不一致而无法保持原来的V字队形。根据这一个鸟类飞行现象，本章提出减速飞行调整阶段，来模拟V字队形被打乱后，鸟个体在散乱队形中自由试探自由飞行的行为。

在这个阶段里，鸟个体两两之间原本比较固定的跟随与被跟随的关系被打破，每只鸟都会不断试探，跟随不同的鸟，调整自己飞行的方向和速度。与V字飞行阶段类似，减速飞行调整阶段也需要几个循环才能完成，循环次数设为。在减速飞行调整阶段的每个循环里，每只鸟都会选择另一只鸟去跟随，同一只鸟在不同的循环里会跟随不同的鸟，以体现鸟个体在调整速度和试探方向的过程中，比较无序的行为特点和散乱的队形结构。在这个阶段，一只鸟跟随另一只鸟所代表的搜索与V字飞行阶段是一样的，都是进行信息共享。只不过减速飞行调整阶段中，跟随与被跟随的关系的动态变化的，在V字飞行阶段，跟随与被跟随的关系是固定不变的。减速调整阶段需要进行代的迭代更新，整个种群每次更新的过程如下：

step1：令；

step2：令鸟个体的候选集，为随机选择另一只鸟，与交叉得到两个子代，选择较好的子代放入候选集；

step3：如果候选集里面个体的数量小于，那么回到step2，否则进行step4；

step4：鸟个体进行一次邻域搜索，把邻域解放入候选集；

step5：如果候选集里面个体的数量小于，那么回到step4，否则进行step6；

step6：如果，那么，否则结束本次迭代。

## 5.2.3 竞争阶段

经过减速飞行调整阶段之后，每只鸟都适应了自己新的飞行方向和速度，整个鸟群的速度开始趋于稳定。那么接下来，每只鸟就能够确定自己在新一轮飞行队伍的位置，并且新的领头鸟也能被确定下来，鸟群就能形成新一轮飞行的队形了。

在基本MBO中，领头鸟的替换是轮替式的，跟随鸟按照V字型的前后顺序，轮流替换旧的领头鸟，成为新的领头鸟。在领头鸟疲劳了之后，它会退到V字型队伍的队尾，此时只有紧跟领头鸟的两只跟随鸟有机会成为新的领头鸟。无论排在队伍中间和队伍后面的鸟有多优秀，无论它们是否比新的领头鸟优秀，它们在本轮领头鸟替换过程中是没有机会成为领头鸟的，排在中间的优秀个体需要等待好多轮才能有机会成为领头鸟。这种轮替式的领头鸟更换机制并没有让优秀个体在种群中充分发挥影响作用。

同时，在这种轮替制下，在领头鸟更替之前和更替之后，几乎所有鸟在V字型队伍里的顺序是没有变化的。对于大多数鸟来说，它们各自的前鸟，在很多轮飞行里面都是不变的，这就导致了鸟与鸟个体之间交互协同的拓扑结构比较固定，每只鸟的眼光都只能长期局限于某一只前鸟。如果不打破这种单一的拓扑结构，难以充分发挥群体的协同潜力。（考虑插图示意拓扑怎么固定）

为了让优秀的个体在种群中充分发挥个体优势的影响作用，同时为了打破鸟群比较固定的协同拓扑，本章提出给MBO引入竞争阶段。鸟群在减速飞行调整阶段结束之后，鸟群的队形依然是散乱的，经过竞争阶段之后，鸟群可以按照实力优劣竞争领头鸟的位置，确定下一轮飞行的V字型队形，可以形成与上一轮飞行完全不一样的协同拓扑，丰富鸟个体之间互相协作的关系。

在竞争阶段的过程中，所有鸟凭借自身实力角逐领头鸟的位置，越优秀的鸟越有可能成为新的领头鸟。同时个体之间也会互相竞争队伍中间的位置，越优秀的鸟越有可能排在队伍的前面，越差的鸟越有可能排在队伍后面。经过这样的竞争之后，新形成的V字型队形具有一定程度的优劣梯度，这样可以让更优秀的个体有更多机会发挥引领作用，提高寻优的效率，而不是像基本MBO中轮替式的机会均等，优秀个体可能要等待好久才能成为领头鸟。

在大自然的复杂环境下，偶然因素不可避免，竞争阶段也会受到偶然随机因素的影响，具体表现为，最优秀的个体并不一定能成为下一轮飞行的领头鸟，最差的个体也不一定是排在队伍末尾的那个。竞争阶段的准则是，更优秀的个体排在前面的可能性更大，越优秀的个体成为新领头鸟的可能性越大，而不是按照严格适应度排序来构成队形。这种偶然因素的加入可以防止领头鸟的位置被某一个陷入局部最优的个体长期垄断，提高跳出局部最优的能力。即使不是最优秀的个体，只要足够优秀了，也有机会成为领头鸟，这样一来，除了最优秀的个体所在的区域，其他优秀个体所在的区域的潜力，在不同领头鸟的带领作用下，也能得到挖掘。这种加入偶然因素的排序被称为模糊排序。竞争阶段鸟群竞争形成新的V字队形具体过程如下：

step1：对所有鸟进行模糊排序，得到；

step2：成为新的领头鸟，；

step3：如果为偶数，那么加入左翼队列的末尾，即成为，否则加入右翼队列的末尾，即成为；

step4：如果，那么回到step3，否则新的V字队形构建完成。

## 5.2.4 算法流程及复杂度分析

CCMBO主要由四个阶段组成，分别是：

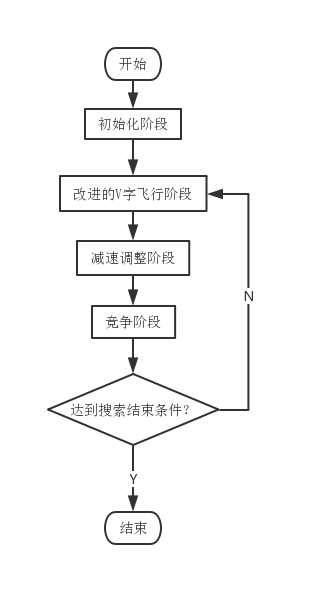
（1）初始化阶段。为了体现算法的鲁棒性，使用随机初始化，不使用加入启发式的初始化。随机生成所有个体之后，使用与5.2.3相同的方法选出领头鸟，构建V字队形。

（2）改进的V字飞行阶段。保持V字队形不变，所有鸟使用5.2.1的更新方法更新代。

（3）减速调整阶段。打破原有的V字队形，所有鸟使用5.2.2的更新方法在散乱的鸟群中更新代。

（4）竞争阶段。每只鸟的速度和方向趋于稳定，使用5.2.3的方法选出新的领头鸟，并重新构成V字队形，准备下一轮飞行。

算法流程图如下：



初始化阶段结束之后，整个种群进入一轮飞行，每一轮飞行都包含次迭代的V字飞行阶段，和次迭代的减速调整阶段，最后再进行一次竞争阶段，竞争阶段只是构建下一轮飞行的V字队形，并不进行种群更新。因此在一轮飞行中，整个种群一共进行次的更新。完成一轮飞行之后，如果还未达到搜索结束条件，就使用新的V字队形进行下一轮的飞行。

令鸟群大小为，染色体维度为，在最坏的情况下，CCMBO的时间复杂度为：

（1）初始化阶段，首先是对个个体做随机初始化，时间复杂度是，然后是对个个体排序从而构建V字队形，时间复杂度是，因此本阶段的时间复杂度是；

（2）改进的V字飞行阶段，个个体通过交叉或者邻域搜索的方法各自生成个候选解，如此更新代，本阶段的时间复杂度是；

（3）减速调整阶段，个个体通过邻域搜索各自生成个候选解，如此更新代，本阶段的时间复杂度是；

（4）竞争阶段，个个体进行排序，构建新的V字型队伍，本阶段时间复杂度是；

综合以上的分析，和代表着优化问题的规模，、和是与和无关的常量，因此对任何规模的优化问题来说，CCMBO算法的时间复杂度为：



# 5.3 实例仿真以及性能评价

为了验证CCMBO算法的效果，本节首先把CCMBO应用在标准的FJSP分批调度问题上，对CCMBO各个改进的部分进行实验，与基本MBO进行对比以验证改进的效果，并且与使用该标准问题的其他算法进行对比。然后把CCMBO算法应用到免疫分析检测设备分批调度问题上，验证分批调度的效果。

本节使用Python3.6编写算法，在3.20GHz，16.0GB的计算机上进行实验。

## 5.3.1 测试例子

5.3.1.1 FJSP分批调度标准测试用例

此处使用【赵】 中的测试用例，一共四个测试用例，类型都属于混排，带有准备操作，准备操作可分离，以完工时间最小化作为目标。使用来描述测试用例的规模，其中代表批的数量，代表机器数，代表每个批的工件总数。一下是四个测试用例的规模：

P1：规模是，即4个批，6台机器，每一批都包含8个工件，这属于中小规模的分批调度问题。每类工件的每个工序在不同机器上的加工时间以及准备操作时间如表所示；

P2：规模是，批数与机器数跟P1相同，每一批都包含20个工件，这属于中小规模的分批调度问题。加工时间与准备操作时间与P1也相同，如表所示；

P3：规模是，即6个批，6台机器，每一批都包含10个工件，这属于中小规模的分批调度问题。每类工件的每个工序在不同机器上的加工时间以及准备操作时间如表所示；

P2：规模是，批数与机器数跟P3相同，每一批都包含20个工件，这属于大规模的分批调度问题。加工时间与准备操作时间与P3也相同，如表所示；

（不知道要不要把具体数据列在正文）

5.3.1.2 免疫分析检测设备分批调度问题

## 5.3.2 实验结果与分析

六、基于多领头鸟分化协同候鸟迁移算法的优化调度

# 6.1 多领头鸟分化协同候鸟迁移算法

## 6.1.1 多领头鸟分化协同机制

## 6.1.2 基于年龄群体特征的邻域搜索机制

## 6.1.3 算法流程

# 6.2 实例仿真以及性能评价

七、基于多微鸟群协同候鸟迁移算法的多目标分批优化调度