# 分批调度编码方案

## 双矩阵编码

两个矩阵，一个代表分批方案(P1)，一个代表调度方案(P2)。优点是：

1、不会出现不可行解，不需要对个体进行修补

2、可以表示不等量分批/等量分批，比较灵活

## 基于机器柔性灵活指数(flexibility index, FI)的解码方式

解码的时候会衡量每一台机器的柔性灵活指数，即一台机器能加工的工件的批量总数，如果该指数较低，说明该机器比较不灵活，应该优先给它安排工序。

# 针对分批调度编码的搜索算子

由于编码是双矩阵，比一般的一维编码更复杂，所以需要重新设计搜索算子，以提高搜索效率。

分为交叉算子和邻域算子，交叉算子负责粗粒度层面的搜索，邻域算子负责细粒度的搜索，如此一来，可以从不同层次去充分配合去搜索双矩阵结构。

后面的算法会从这里选择合适的搜索算子

## 交叉搜索算子

负责粗粒度：

P1,P2----多点交叉

## 邻域搜索算子

负责细粒度：

P1基础邻域----改变子批数，改变子批量

P2基础邻域----swap, insert, inverse, POX

P1启发式邻域----改变关键子批批量

P2启发式邻域----改变关键工件在机器的优先度

P1细粒度全邻域

P2细粒度全邻域

引入了邻域搜索策略：

贪心搜索

多步搜索

贪心多步搜索

# 竞争制候鸟迁移算法求解单目标分批调度问题

原始候鸟迁移算法里，领头鸟是按顺序轮换的，不需要竞争就能当上领头鸟。为了提高鸟群寻优的效率，本文把领头鸟的轮换制改为竞争制。

整个算法框架由如下三个阶段构成。

## 原始候鸟迁移算法

算法流程介绍

分析缺点

分析优点，以及我为什么选择它

## 基于信息共享的V字飞行阶段

原始候鸟迁移算法中，前一只鸟与后一只鸟共享它的邻域个体，这称为利益机制。本文把邻域共享改为信息共享，即共享部分编码信息。

实质：后一只鸟与前一只鸟交叉得到的子代，与后一只鸟的邻域解构成候选集，给后一只鸟则有替换。按照相同的V字型进行多个循环。

## 减速飞行调整阶段

当领头鸟想从领头位置退下时，它会减速，鸟群其他鸟也纷纷减速。原本的V字型队伍被打乱，原本两两的跟随与被跟随的关系也被打破，每一只鸟都会调整自己的速度和方向，此时鸟群进入减速飞行调整阶段。

实质：每只鸟随机选择另一只鸟跟随，跟随进化的方式与V字阶段一样。如此进行多个循环。

## 模糊竞争阶段

减速飞行调整阶段结束后，各个鸟都调整好自己的速度了，此时较优的鸟会竞争排在前，交差的鸟会排在后，重新形成V字型。因为大自然的鸟类对于优劣没有定量的标准，因此不会形成精准的大小排序，所以本文为了更贴近自然规则，使用模糊的排序来构成新的V字型。

实质：按照模糊排序，重新构成V字型

## 年龄分段机制

不同年龄的鸟使用不同的邻域搜索方法，体现不同年龄个体所积累的经验丰富程度的不同

青年----单次单步

中年----贪心单步

老年----贪心多步

老年个体年龄到达一定阈值后，就会重新初始化

## 算法流程图

各个阶段的流程图

总的流程图

## 实验

要对比改进前和改进后的，即本算法

要对比本算法和其他文献的算法

要做小规模问题，即自己改编的经典FJSP问题

要做中等规模问题，即ZHAO的4个问题

要做大规模问题，即免疫检验设备调度问题

# 多领头鸟协同竞争制候鸟迁移算法求解单目标分批调度问题

在自然界中，有时候鸟的队形不是单一领头鸟的V字型，而是多个领头鸟带领的一字长蛇形。由于分批调度问题分为两个子问题，使用多领头鸟协同可以提高问题求解效率

# 多微鸟群并行候鸟迁移算法求解多目标分批调度问题