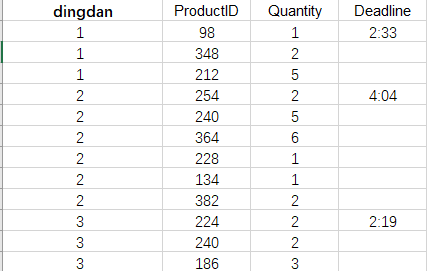
**总体目标：订单分批提升拣选效率**

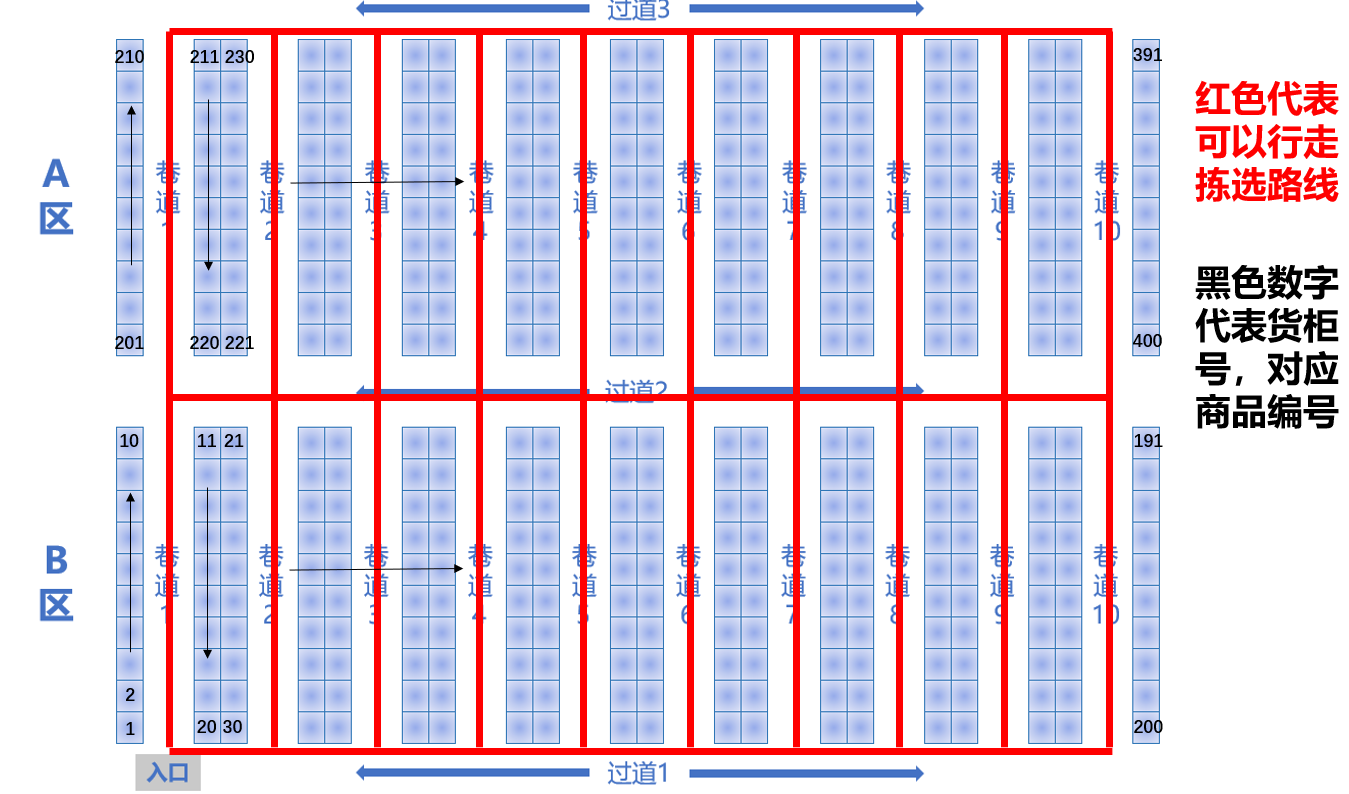
首先进行订单分批，考虑最小化服务时间，总延迟时间，路径约束，得到优化后的订单分批结果，其次进行路径对比，包括“s”型拣选策略和改进“s“型拣选策略。对比内容为不进行分批的订单(按订单号先到先得)和进行分批的订单分别采用不同的拣选策略获得的效果。**算法要求：遗传算法**。

# 数据

订单数据分为四个特征，1订单号、2商品种类（对应仓库货柜号）、3各自商品种类对应的数量、4订单的期限时间（例如2：33代表153分钟）。



仓库布局如下图所示。



# 拣选约束条件

|  |  |
| --- | --- |
| **参数** | **数值** |
| 拣选员工数量 | 2 |
| 拣选行走速度 | 40m/min |
| 拣选速度 | 5s/个 |
| 单个拣选员工最大容量 | 100个（不考虑重量和体积） |

1、同一订单不可被分配到不同批次的订单中进行拣选，但是同一批次中的订单可以由不同的拣选人员以及拣选设备完成；

2、巷道是宽巷道，不同拣选员工在同一巷道中相遇不会形成堵塞，不同拣选员工同时从同一储位拣选品项不会造成等待和拥挤；

3、货柜够大，多个拣货员可以同时提取同一货柜商品。

# 建模参考

有关模型建立的相关参数如表3-1所示。

|  |  |
| --- | --- |
| 参数符号 | 含义 |
|  | 表示订单集合 |
|  | 表示批次集合 |
|  | 表示拣选人员集合 |
|  | 完成批次时所需拣选的品项数量 |
|  | 表示在批次中，拣选位置的拣选顺序，其中, |
|  | 拣货车最大体积限制 |
|  | 拣货车最大载重量 |
|  | 订单中所有品项体积 |
|  | 订单中所有品项质量 |
|  | 批次中拣选作业时间 |
|  | 订单的交货期 |
|  | 在拣货批次中货物与的距离，由于计算较为复杂，详见后面 |
|  | 拣选批次形成的拣选路径上从出入口到第一个拣选z 货物的距离 |
|  | 拣选批次形成的拣选路径上从最后一个拣选货物到出入口的距离 |
|  | 表示在批次中，拣选位置数量和起始点总和 |
|  | 订单是否在批次中 |
|  | 在拣选批次的拣选路径上货物和是否相邻 |
|  | 表示拣选员工的总拣选时间 |
|  | 订单拣选完成时间 |
|  | 批次的拣选作业时间 |
|  | 订单的延迟时间， |

（1）最小化总服务时间

拣选作业所耗费的时间占整个仓库总作业时间的一半左右，是仓库作业环节中最耗时的部分，因此在优化订单分批时需要先考虑拣选作业，降低拣选作业时间可以促进拣选效率的提高，故本文重点研究的订单分批问题是面对一组固定数量等待拣选的订单，尽可能降低拣选作业的时间，达到提高拣选效率的目的，基于此构建总拣选作业时间最小目标函数，如式下所示。

假设第一个批次开始进行拣选的时间为,则任一批次的拣选作业的时间

则拣选完所有订单的完成时间为

（2）最小化总延迟时间

随着物流的发展，顾客对配送的时效要求不断提高，若订单在顾客可接受时间范围外送达，则会影响顾客满意度，而订单按时拣选完成是准时配送的前提，因此，最小化总延迟时间也是本文的重要优化目标。

假设订单属于第个批次，订单的完成时间为第一个批次开始的时间和订单所在批次的拣选时间加上之前批次拣选所耗费的时间之和，即第一个批次开始的时间与前个批次拣选时间之和，计算公式为，由此订单延迟时间则可以表示为

目标函数：

(3.1)

约束条件：

(3.2)

(3.3)

(3.4)

(3.5)

(3.6)

(3.7)

(3.8)

(3.9)

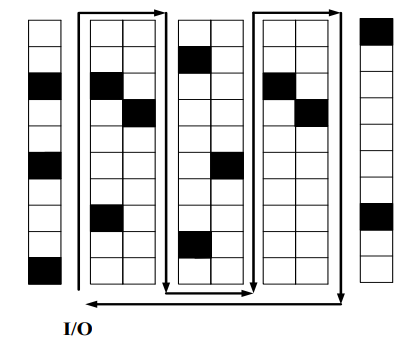
式(3.1)为目标函数，总目标是最小化总服务时间和总延迟时间，拣选人员工作量均衡。约束条件（3.2）表示任一批次包含的商品总重量不超过拣选设备的最大载重量；约束条件(3.3)表示任一批次包含的商品总体积不超过拣选设备的最大容量；约束条件(3.4)确保拣选路径形成一个闭合回路；约束条件(3.5)表示任意一张订单只能属于一个批次，订单无法分割到不同批次中；约束条件(3.6)和(3.7)表示任意一种货物只能在一个批次中且只能在一条拣选路径中被拣选；约束条件(3.8)表示决策变量的取值范围为0或1，当订单被划分在批次中时，为1，否则为0；约束条件(3.9)表示拣选完货物是否立即前往点，当拣选完货物后立即拣选货物，则为1，若拣选完货物后没有立即拣选货物，则取值为0。

# 拣选策略参考

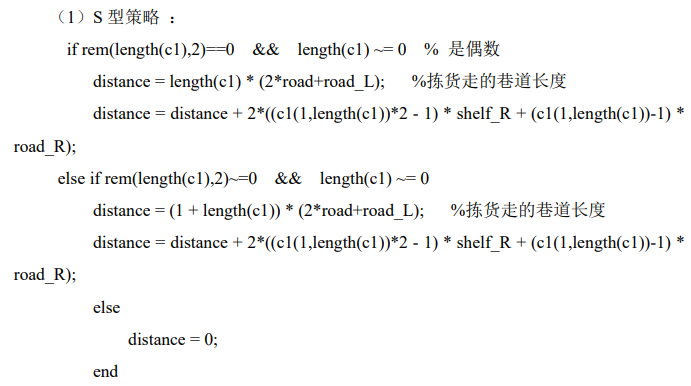
**S 型策略参考**

穿越式策略由于行走路线类似于“S”型，所以又叫做“S 型策略”。该策略是指 从一条巷道的头走到尾，再从相邻巷道的尾走到头，与此同时进行拣选作业，按 照此种路径完成拣选作业的一种策略，作业人员的拣选路径包含所有待拣选物品 的巷道，此种策略的行走路线如图 2.7 所示。穿越式拣选策略适用于货品储存位 置相对发散的情形。

S示意图



S代码示意



**改进s型策略参考**

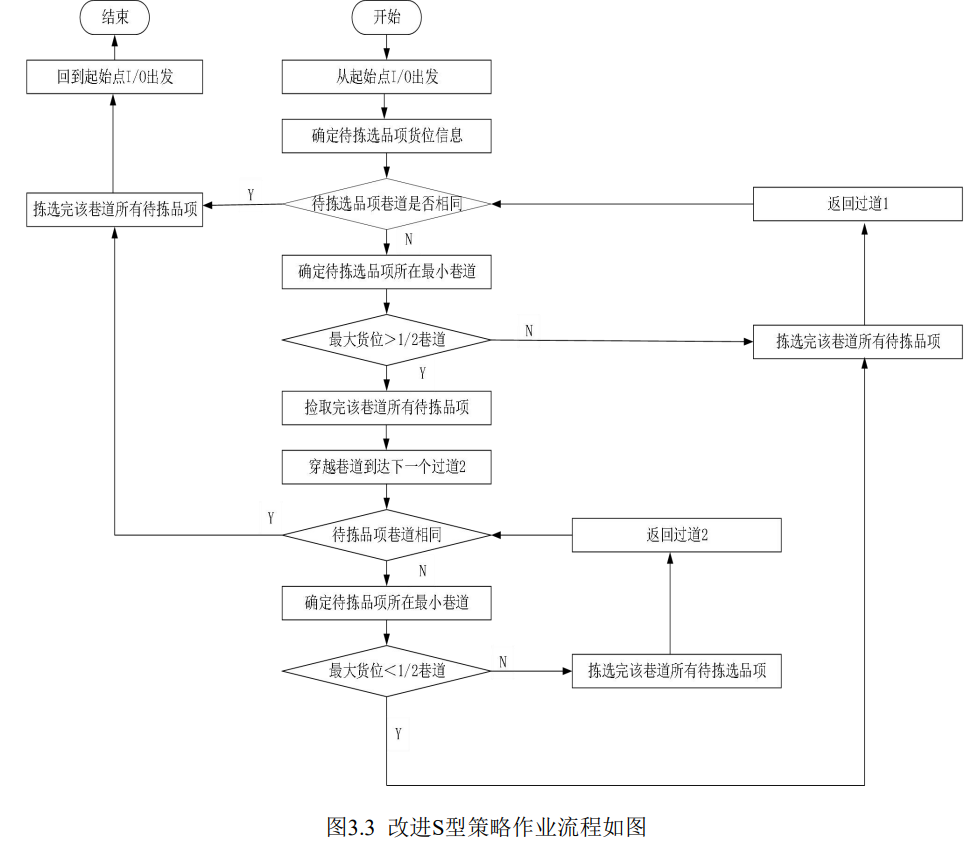
由于 S 型策略行走的 路径较为简单，且在订单拣选作业中容易被拣货员执行，所以目前“人到货”的配送中 心采用 S 型策略较多。由于 S 型策略的特点，所以它比较适合订单货物品项较多、货架 分布散且广等情况，这时使用 S 型策略可以兼顾到每一个货位。当待拣选货物品项分布 在各个巷道的两端时，可以在一定程度上减少拣货员的拣选行走总路程，所以中点策略 较适合把出入库频次较高的货物品项放在距离起始点（I/O）较近的货位上。综合考虑 S 型策略和中点策略的特点，本文在对订单拣选路径进行优化时，结合使用 S 型策略和中 点策略这两种策略，即引入中点策略对 S 型策略进行部分环节优化。

步骤 1：拣选时，如果该待拣选货物货位的位置距离该巷大于该巷道长度的 1/2 时， 拣选人员拣选完货物后无需返回，直接穿过该巷道今年入下一个货物所在巷道即可；

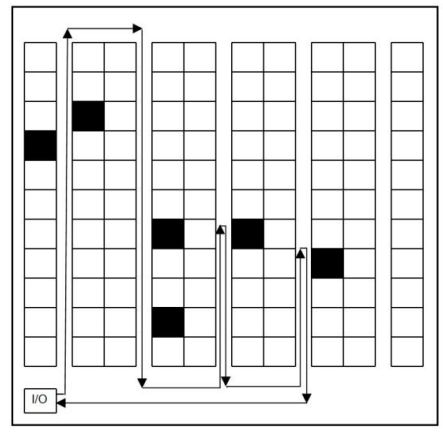
步骤 2：如果该待拣选货物货位所在位置小于该巷道总长度的 1/2 时，拣选人员拣选 完货物后拣选后无需再穿过一整条巷道，直接返回到进入该巷道的位置就可以，然后接着对下一个货物进行拣选，然后去到到下一个待拣选货物的所在巷道；

步骤 3：当某个巷道内没有需要被拣选的货物，直接跳过该巷道，进入下一个拣选巷道。

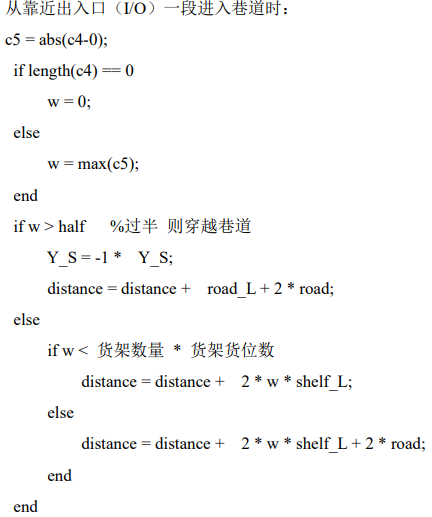
改进型s流程图

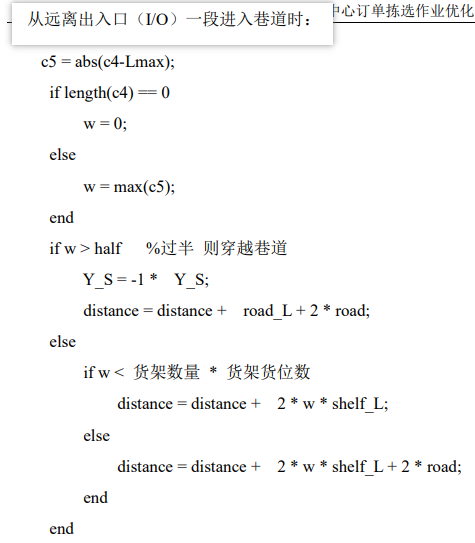


改进型s示意图



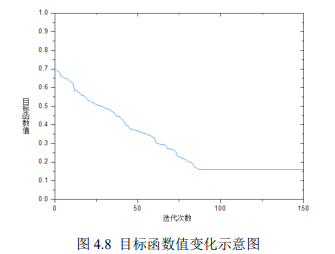
改进型s代码示例（matlab）





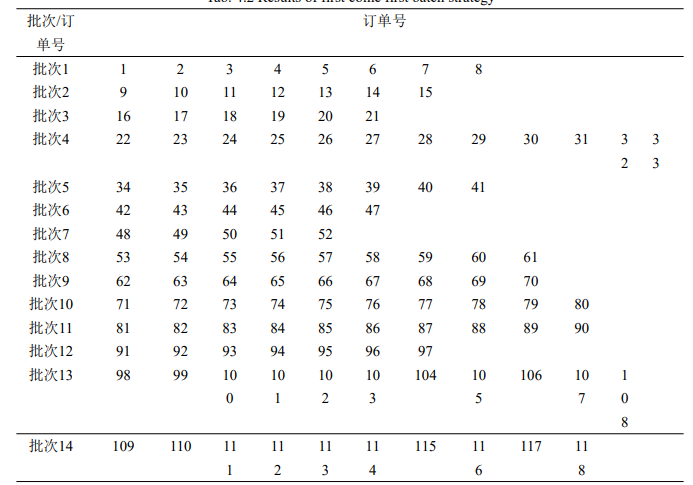
# 结果要求1：代码（包括分批和拣选策略）

# 结果要求2：目标函数收敛图



**（示例）**

# 结果要求3：订单分批优化结果



**（示例）**

# 结果要求4：拣选结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **对比类型** | **拣选时间** | **订单延迟时间** | **总拣选路径** | **目标值函数** |
| **四种对比类型** | **对比参数可以根据建模者进行调整，需要体现优化后结果** | | | |
| 先到先得和s |  |  |  |  |
| 改进分批和s |  |  |  |  |
| 先到先得和改进s |  |  |  |  |
| 改进分批和改进s |  |  |  |  |