## 分货算法模型设计

## 一、问题描述

在固定的时间窗口内，对陆续达到的车辆进行批次处理，生成对应装车清单。

在当前时间窗口下进行批次处理之前，获取实时库存。

设当前货物共有 N 件，货物（cargo） 包含属性：品种、重量、规格、优先级、所在仓库、目的地（城市、区县、详细地址）。

设当前时间窗口下达到 M 辆车，车辆（truck） 包含属性：车辆载重、期望运输品种（非必需）、期望运输目的地（非必需）。

根据实时库存数据为每辆车生成装车清单（loading） 即车辆 选取的货物集合。

* **限制条件**：

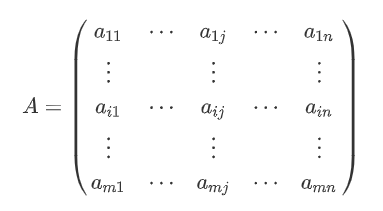
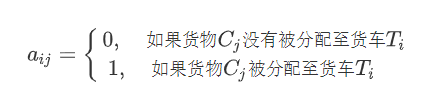
1. 单个车次货物总重量不得超过该车载重，同时不得低于车载重的80%；
2. 单个车次货物小品种 ；
3. 单个车次货物目的地为同一城市；
4. 单个车次取货仓库数目

* **优化目标**：

1. 优先发运客户催货、超期货物；

将客户催货和超期货物转化为优先级数字 ，其中客户催货对应优先级数字2，超期货物对应优先级数字1，若无优先级则默认为0，数字越大优先级越高。

1. 货物运载总重量最大；

构造车次和货物的矩阵A，同时同一件货只能被放置一辆车上。

其中 是当前车次的载重限制。

1. 对于拼货车次，当前车次下三号（发货通知单号、订单号、合同号）尽可能一致

同样将其转化为数字 ，则对于同一辆车,

## 二、框架设计

* **生成装车清单候选集**

1. 库存预处理。处理异常数据，合并货物信息；
2. 筛选库存。根据车次信息中的货物品种和运输城市，筛选同品种同城市的货物数据；
3. 在限制条件下枚举所有可能的装车清单候选集。

* **多目标优化**
  + **降维为单目标优化**

1. 线性加权法：

设 l 为每一辆车候选集长度， 分别为优先级、重量、拆单量的权重值。

此时目标函数转换为

其中

表示该候选集货物总重量， 表示该车次载重。

1. 抢占式优化：

## 数据

### 输入数据

* **可发库存数据**

实际更新库存时间：20分钟。

该表中数据：0点库存快照，每隔20分钟更新库存后插入的新数据，根据时间间隔对重量进行计算获取粗略的实时库存数据。

装车清单主表：ods\_db\_inter\_lms\_bclp\_loading\_main

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **含义** |
| car\_mark | varchar | 车牌号 |
| main\_prod\_list\_no | varchar | 主清单号 |
| prod\_list\_no | varchar | 清单号 |
| status | varchar | 业务状态 |

* **司机（车辆）数据**

调度司机表：ods\_db\_trans\_t\_plan\_driver 20190801-至今

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **含义** |
| plan\_no | VARCHAR | 调度单号 |
| order\_no | VARCHAR | 委托单号 |
| driver\_id | VARCHAR | 司机 |
| vehicle\_no | VARCHAR | 车牌号 |
| plan\_weight | DECIMAL ( 18, 6 ) | 计划重量（吨） |

司机到达时间近似看成装车清单表生成时间。

### 输出数据

* **装车清单详表**：ods\_db\_inter\_lms\_bclp\_loading\_detail

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **含义** |
| carmark | varchar | 车牌号 |
| main\_prod\_list\_no | varchar | 主清单号 |
| prod\_list\_no | varchar | 清单号 |
| notice\_num | varchar | 发货通知单号 |
| oritem\_num | varchar | 订单项次号 |
| commodity\_name | varchar | 品名 |
| standard | varchar | 规格 |
| weight | varchar | 重量 |
| outstock\_code | varchar | 出库仓库 |
| instock\_code | varchar | 入库仓库 |
| create\_time | varchar | 创建时间 |