

# 物流仿真和算法在汽车供应链中的应用

毕兴伟 步 时 王灵峰 张争光 范 武  
吉利汽车研究院(宁波)有限公司 浙江 宁波 315336

**摘要:** 本文主要阐述了Anylogic仿真和算法在汽车供应链中的应用, 首先介绍了Anylogic仿真, 其次介绍了仿真技术与算法结合的应用, 最后介绍了仿真和算法在供应链应用中的相关应用案例。

**关键词:** Anylogic; 供应链; 装箱算法; 寻路算法

## 1 汽车供应链现状分析

汽车供应链需要解决的核心问题是为了降低零部件库存而增加配送频次导致的运输成本的增加的问题。汽车制造工厂拥有很多的供应商, 所以有效的路径设计和装箱设计会有效的控制大批量/低频次配送的费用<sup>[1]</sup>。本文利用Anylogic仿真与算法结合, 对配送车辆以及配送路径作出最优化选择并建立模型, 达到节约运输成本并准时运送货物的目的。

## 2 Anylogic 仿真介绍

Anylogic是一款应用广泛的, 对离散、系统动力学、多智能体和混合系统建模和仿真的工具<sup>[2]</sup>。它以多智能体与GIS技术结合为平台, 采用离散触发机制进行供应链仿真。调用地图瓦片接口, 可实现地图可视化自定义。车辆运输仿真系统基于Java语言开发, 可以直接生成应用程序, 供客户离线使用; 也可以部署至云平台, 多终端、跨平台使用该模型。

针对汽车供应链的复杂性, 利用Anylogic软件对供应链网络进行仿真分析。将主机厂、中转仓及供应商之间的关系, 通过智能体以及流程图等进行衔接, 并将最终成果在GIS中直观地展现出来, 对其区域仓的产品流以及布局进行分析, 统计物流成本。采用装箱算法和寻路算法, 求出配送线路和装箱策略, 再使用Anylogic软件进行仿真, 验证优化结果。

## 3 物流算法设计

### 3.1 启发式装箱算法

结合汽车零部件包装属性, 对三维装箱问题的空间搜索策略进行了创新, 保证计算精度的同时降低了计算复杂度。我们将货物包装看作包含实际包装尺寸的最小长方体。如图1所示, 我们将车箱右下角设为坐标原点, 建立三维空间坐标系。货物可沿x轴、y轴、z轴摆放。

汽车行业智能工厂与数字化车间标准应用项目, 项目编号GZ06/23-096

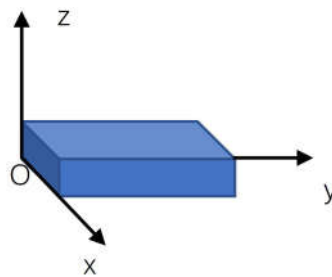


图1

3.1.1 对于特殊类型的包装, 将三维装箱问题简化为一维装箱问题提升算法性能。具体步骤是, 筛选有特定堆叠要求的货物包装(例如: 带轮料架, 该包装仅可同类型堆叠), 对于这类特殊类型的货物包装, 按照包装属性优先筛选, 并将三维装箱问题简化为依次沿z-x-y方向的一维装箱问题堆叠(这里z-x-y方向顺序考虑了后续车箱内异构性包装的堆叠以及运输稳定性, 具体可根据包装数据作为算法参数设定)。

3.1.2 对于可组托的包装类型, 由于汽车零部件包装的标准化属性, 我们首先将包装问题简化为考虑托盘承载平面的二维装箱问题, 建立线性规划模型, 调用求解器求解最优的单层可组托件数以及组托包装类型(即组托标准)。尽管该过程需要求解时间较长, 但由于汽车零部件的包装尺寸符合标准化要求, 我们可将组托标准维护到数据库中, 算法求解过程中直接调用该组托标准结果即可。具体的整数线性规划模型如下:

$$\begin{aligned} \min z &= c^T x \\ \text{s.t.} & \begin{cases} Ax \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub \end{cases} \end{aligned}$$

$c$ 、 $x$ 、 $b$ 、 $lb$ 、 $ub$ 为列向量,  $A$ 为矩阵, 其中,  $x$ 部分或全部分量为整数

3.1.3 对于其他通常的包装类型, 我们采用了一种基于重心的空间搜索策略, 该方法不需要回溯, 可减少算法计算时间。具体步骤是: 我们将车箱按照可堆叠平面

划分车箱空间,计算每个可堆叠平面对应的重心,以平面重心从低到高的顺序的开始装载,获取该平面对应空间可堆叠的货物,同样将可堆叠货物按重心从低到高放置,直到各可堆叠平面对应的车箱可用空间遍历结束位置。

### 3.2 寻路算法

首先应用基于贪心策略的扫描算法获取车辆路径规划的初始解,其次,基于自适应大邻域搜索(Adaptive Large Neighborhood Search,简记为ALNS)算法框架<sup>[3]</sup>,在较短时间内可获得较合理的循环取货模式下的运输和装载方案。

针对具体业务,在ALNS算法框架中,设计了两种类型的算子来源:a、算法内部调整;b、用户交互。基于实际业务场景,同时考虑运输距离和车辆装载率,创新性地设计了算法内部破坏算子、修复算子。

#### 3.2.1 破坏算子

3.2.1.1 统计每条车辆路线中车箱内货物的相似度,筛选相似度较低的某条车辆路径,随机移除n种类型的货物。

相似度r定义:

$$r = \sqrt{(l_1 - l_2)^2 + (w_1 - w_2)^2 + (h_1 - h_2)^2}$$

其中,  $l$ ,  $w$ ,  $h$ 分别为包装的长宽高。

3.2.1.2 统计车箱内货物相似度,筛选货物相似度较低(异构性较强)的车辆路线,随机移除该线路中n个供应商(取货点)及其对应的货物。

3.2.1.3 计算各路线的车辆装载率,以概率p筛选路线并移除该路线所有货物。

概率p定义:  $p = 1 / \sum_i (l_i \cdot w_i \cdot h_i)$

其中,  $l_i$ ,  $w_i$ ,  $h_i$ 分别为货物i的长宽高。

3.2.1.4 随机选择m条车辆路线,获取对车辆行驶路程影响最大的n个供应商,移除相应供应商及其对应的货物。

3.2.1.5 随机移除m条车辆路线中n个供应商及其对应的货物。

#### 3.2.2 修复算子:

3.2.2.1 对于待修复的供应商(破坏算子移除),获取距离供应商最近的n个供应商(n个供应商属于已规划的车辆路径)及其对应的m条车辆路径,计算待修复供应商插入m条车辆路径对应的目标函数变化量,加入随机系数,选择考虑随机系数的目标函数最小的路径及配送顺序插入。

3.2.2.2 对于已规划未满载的车辆路径,获取待修复的供应商货物中可放入剩余空间的货物,计算货物及其

对应供应商的成本增量 $\Delta C$ ,选择 $\Delta C$ 最小的路径及其配送顺序插入供应商,如果剩余货物无法加入已规划未满载车辆,则选择新的车辆加入车辆路径。

结合Anylogic软件,可开发相应影响运输成本的交互因子,例如:订单数量(货量)、中转仓位置和数量等。将用户的交互行为视为“隐型”的算法破坏算子和修复算子行为,从而可基于用户经验,对算法规划结果进行修正。算法返回给用户取货路线及装载方案规划结果后,用户可通过交互界面基于经验在结果返回界面手动调整车辆运输路线及装载方案,用户调整后,一方面,可直接调用算法接口计算调整后的运输成本;另一方面,也可调用算法,此时算法将用户操作后结果作为ALNS破坏算子、修复算子操作后的一个过程解,并基于此继续迭代直到满足迭代终止条件为止。另外,由于部分汽车零部件订单中部分通用件需求相对稳定,故算法后台会同时记录用户交互界面的常见操作,并基于此定义相关的破坏算子和修复算子。

## 4 供应链物流规划实例

### 4.1 仿真数据准备

Anylogic可以直接处理多种数据存储,包括Oracle、MySQL等,这里我们用MySQL数据库作为数据来源。

4.1.1 新建数据库map GIS,字符集ut18mb4,排序规则ut18mb4 bin;

4.1.2 在数据库map GIS,建立如下表:包括供应商、中转仓、工厂经纬度坐标(x, y);

表1 map GIS

编号	供应商代码	经度X坐标	纬度Y坐标	物料信息
1	1156	22.759354	113.499547	10993, 10992,
2	1264	31.31958	121.1186	11724,
3	1309	30.82888	121.227	12060,
4	1201	31.47872	121.1256	11232, 11231, 11229,
5	1331	29.71814	106.6312	12149,
6	1289	30.52705	104.2251	11938,
7	1269	29.67395	106.5898	11942,
8	1168	29.29075	106.3148	11755
9	1200	43.85525	125.4064	11228,
10	1184	43.90159	125.4533	11230,
11	1328	43.37653	124.8754	11942
12	1280	43.47382	125.5317	11790
13	dc_0	31.65251	119.9486	
14	dc_1	29.82156	105.7428	
15	dc_2	32.67234	127.6739	
16	dc_3	42.93856	125.45333	
17	wh_01	28.41683	121.3943	

#### 4.2 仿真建模

新建工程后，在Anylogic面板里选择数据库，将数据库添加到Main智能体中，将数据库拖到Main智能体后，对数据库链接做详细编辑，引入JDBC驱动包，编辑完数据库后，需要添加数据表，将面板中的查询按钮拖入面板。

点击“查询”，可以重命名并选择对应的数据库，选择“SQL”可以灵活的查找表里对应的数据，设置完数据库元素后，前期准备的物料已经完成，接下来用Java语言，将GIS地图渲染在地图里。

点击“Main面板”空白地方，在右侧出现智能体行动，在启动时添加函数InitGisMap()。

在Main面板中添加作业流程，设置供应商坐标为初始值，中转仓坐标为途径地，工厂坐标为目的地，其中中转仓可以作为变量从程序里计算，也可以固定经纬度，但数据必须是地图里的真实存在的，否则智能体找不到路径将沿着直线行驶，不会走GIS地图道路。

#### 4.3 调用算法接口

通过http模式调用装箱算法，输出该订单对应的装箱策略；基于到货时间和车辆装载容积以及各个供应商的位置，输出相对优解的运输策略，避免依靠物流规划人员的经验判断，导致的物流成本波动和浪费。

在满足到货时间的前提下，各个供应商取货线路最短为第一基准，选用车辆时，依据运输成本总最低为第一基准。

#### 4.4 运行模型

运输车辆依据运输规划线路，结合实时路况模块，确定该运输线路的行驶速度V。按照该线路的行驶速度V，分别从供应商A/B/C...中转仓...到制造工厂，输出该线路的运行周期和运输成本。并将各条线路的运输周期与该批次订单的到货时间进行对比，对于超出到货时间的线路，调用策略优化模块，变更配送策略，进行二次仿真。

表2 运输周期及成本

序号	车辆类型	行驶距离	行驶路径	物料种类	装载数量	运输成本	运输周期
1	17.5	445.3	[dc_0, 1156, 1264, 1309, wh_01]	[10993, 10992, 11724, 12060,]	[3,5, 14,12]	6329	5.56
2	17.5	375.4	[dc_1, 1201, 1331, wh_01]	[11232, 11231, 11229, 12149]	[5, 7, 16, 6]	5432	4.42
3	9.6	242.2	[dc_2, 1289, 1269, 1168, wh_01]	[11938, 11942,11755]	[4, 3, 9,]	3501	3.02
4	12.5	328.9	[dc_3, 1200, 1184, 1280, wh_01]	[11228, 11230, 11942]	[10,6, 8]	4759	4.1
5	7.5	43	[dc_3, 1328, wh_01]	[ 11790]	[10]	628	0.71

#### 4.5 生成运输指令

仿真验证的运输策略均满足到货时间后，仿真模型锁定策略，并一键生成运输指令。运输指令包括：运输车辆型号、取货时间、取货顺序：供应商A+物料A、供

应商B+物料B、供应商C+物料C...行驶线路，并以字符串的形式，写入运输管理系统TMS的数据库中，运输车辆按照指令自动运行。

表3 运输指令

车牌	驾驶员	行驶路径	物料种类	装载数量	交期
辽A7DU39	张三	[dc_3, 1200, 1184, 1280, wh_01]	[11228, 11230, 11942]	[10,6, 8]	2024.06.03, 13: 20

### 5 结论

本文根据Anylogic仿真与算法在供应链配送运输、装箱的研究，提高了物流配送及供应链的畅通性和效率。实现了从减少物流费用、节省运力、缩短运输时间等方面，降低物流成本，提高服务质量。为汽车供应链配送规划人员，提供了新的规划方法和思路。

#### 参考文献

[1]蓝青松,徐广抑.从传统运输迈向现代物流——入场

物流的循环取货管理模式[J].上海汽车.2003(08)

[2]何瑞春,赵敏,向万里.基于AnyLogic的系统建模与仿真[M].北京.化学工业出版社.2020年:6

[3]林宇鹏,毛宁,徐国宁,陈庆新,区乐颖.基于ALNS算法的机场客舱清洁人员调度研究[J].机电工程技术.2023, 52(11)