洋山时间矩阵模型

任务和车匹配时间估算

赵云

2019

|  |  |  |  |
| --- | --- | --- | --- |
| 版本描述 | 版本号 | 修订者 | 日期 |
| 建立时间矩阵模型 | v1.0 | 赵云 | 20190116 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[需求介绍 3](#_Toc535396224)

[模型说明 3](#_Toc535396225)

[Storm实现 3](#_Toc535396226)

[storm拓扑 4](#_Toc535396227)

[Redis 服务器 5](#_Toc535396228)

# 需求介绍

洋山TOS选车调度时，需要使用车辆和任务的时间，需要提供每辆车执行每一个待选任务的时间。

# 模型说明

需要做如下假设，以建立模型

假设1 待选AGV起点位置只用QCPB和WSTP，其他位置的AGV都可以park到pb，因此可做此假设；

假设2 任务作业位置只有岸桥和堆场，洋山还有悬臂吊，也可以等同堆场处理；

假设3 任务作业位置具体到岸桥和堆场，以其中心作为任务的终点位置；

假设4 上档PB服从正态分布；

假设5 路径只有直行和转弯，不考虑斜行、s弯；

假设6 不考虑拥堵，等待时间；

# Storm实现

这个模型本质是一个批处理案例，通常利用 Hadoop 批处理作为解决方案。与 Hadoop 不同的是，基于 Storm 的方案会实时输出结果。

我们的这个例子有三个主要组件

* 一个用户应用，用于测试系统
* 一个 Redis 服务器，用于持久化数据
* 一个 Storm 拓扑，用于分布式实时处理数据

Storm Cluster

Redis

用户应用

AGV，任务信息，岸桥，pb信息

AGV，任务，时间三元组

AGV，任务，时间三元组

图 构架概览

用户应用的目的是推送任务给storm集群，并且从数据库redis获取结果。

JedisReaderWriter

用于向Redis数据库中写入、读取pb的位置和在当前岸桥的上下档pb列表中服从的正态分布的概率值；

PbInfo

存取pb的位置和概率值；

UpQcpbReaderWriter 和DownQcpbReaderWriter

分别用于用于向Redis数据库中写入、读取QC的上下档pb位置和概率值；

AGVTaskReaderWriter

用于向Redis数据库中写入、读取AGV和任务的位置方向和编号信息；

PositionInfo

存取AGV和任务的位置方向和编号信息；

## storm拓扑

这个系统搭建的目的是解决大规划计算的算力问题。任务列表显示了一定数量的任务和待选AGV的列表，拓扑接收列表，并更新AGV执行每一个任务的时间估算结果。

AGV1-任务1

AGV1-任务n

AGVm-任务1

AGVm-任务n

AGV...-任务...

AGV1-任务...

AGVm-任务...

Storm Topology

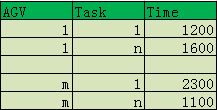


图 storm拓扑的输入和输出

我们的Storm拓扑有四个组件：一个spout向拓扑提供数据，三个bolt完成计算任务。

AGVTaskSpout

从用户处获取AGV和任务列表发送给拓扑。

，及岸桥和pb信息

GetLargeProbablityPBBolt

根据任务计算是否需要使用上下档pb，将AGV-任务信息和pb的位置使用概率信息一起发送给下一个Bolt。pb使用的概率服从正态分布（假设4）。

AGVTaskTimeBolt

完成AGV执行该任务的打分，比如是距离，将AGV-任务编号及打分发送给下一个Bolt。

NewsNotifierBolt

汇总计算结果，将结果写入数据库。

## Redis 服务器

Redis 是一种先进的、基于内存的、支持持久化的键值存储（见[http://redis.io](http://redis.io/)）。本例使用它存储以下信息：

用户浏览队列，用来为 Storm 拓扑提供数据

* Storm 拓扑的中间数据，用于拓扑发生故障时恢复数据
* Storm 拓扑的处理结果，也就是我们期望得到的结果。