表示车辆集合，表示订单集合，

表示初始点

**决策变量：**

若表示第i辆车 去处理第j个订单，若表示第i辆车不处理第j个订单

若表示第i个订单 在第j个订单之前，若表示第i个订单在第j个订单之后

当是车辆的出发点，可以看做有一个虚拟的订单，并且该订单总是排在其它订单的前面。

到达第i个任务的Origin depot的时间

到达第i个任务的Destination depot的时间

**参数：**

表示从第i个任务Destination depot到第j个任务Origin depot所需时间

表示从第i个任务Origin depot到Destination depot所需时间

**目标函数：**



目标是有效时间极大化



约束1 每个订单最多有一辆车去处理：



约束2 时间约束，若同一辆车上订单i在订单j之前处理则需满足

约束3 对于任务i的到达Origin depot的时间和Destination depot时间约束，



约束4：要么订单i在订单j之前 要么订单j在订单i之前



约束5：虚拟的订单也就是初始点始终在其它订单的前面



约束6：整数变量约束:





约束7：时间窗约束，由于时间窗约束的下限约束已经在目标函数里边考虑，此处仅需考虑时间窗约束的上限。



**1分析问题**

因为是整数变量，同时是连续变量。因此该问题属于一个混合整数规划问题。同时该问题的约束和目标函数里边没有非线性项，进一步可知该问题属于线性混合整数规划问题。一般来讲该类问题是NP-hard的。

仔细观察约束1-5可知，约束1只和有关，约束4,5只和有关，约束3只和有关，而约束2是和,,都有关系。约束2是一个耦合约束，它耦合了三类决策变量，而且这三类决策变量有离散的也有连续的。

**2求解思路**

基于以上分析，约束2是一个关键的耦合约束，其它约束则相对容易处理，因此我们采用拉格朗日松弛来进行该问题的求解。基本思路是松弛掉约束2，然后将该问题分为三个子问题。

2.1松弛约束2

松弛之后目标函数变为两项，一项是原目标函数F1，另外一项是松弛后的目标函数F2,如下所示：





其中

由此可以定义出松弛问题



对于松弛问题而言，除了约束2被松弛到目标函数上了，其它的约束依然保留。

因为本问题是求极大化问题，松弛问题的解应该为原问题的上界。从松弛问题里边应该找一个最好的上界，即最小的上界，由此可得与之对应的对偶问题



2.2拆解子问题

如前所述,,之间已经解耦了，所以分别将,,变量的约束和目标函数归并为一类子问题。可得如下三个子问题：

**子问题1（全单模的整数规划问题，可以由求解线性规划得到最优解）：**



约束1 每个订单都最多有一辆车去处理：



约束2 虚拟订单有固定的车去处理



整数变量约束：

,

**子问题2（整数规划问题）：**



约束4：要么订单i在订单j之前 要么订单j在订单i之前



约束5：起始点一定在所有订单之前





由于该整数规划问题的约束和目标函数非常简单，由约束可知和必然只能有一个为1另外一个为0，同时观察目标函数可知，由于是极大化的问题，因此在和两者中要选择系数大的那个为1。

基于以上分析不难直接得到该问题的最优解为







其中

**子问题3（线性规划问题）：**



约束3：表示从第i个任务结束点到第j个任务开始点所需时间





表示在第i个任务执行过程中所需时间,



2.3对偶问题

前面已经提到了对偶问题，如下所示



对偶问题是一个非光滑的凸优化问题，一般采用次梯度算法求解。

2.4求解流程