

空地协同智能路径规划方法

杜文博、翟若楠、陈子鸿、郭通、李宇萌

发明背景

随着电子商务行业的飞速发展，以客户为中心的快递运送服务需求逐年提高。考虑城市化进程与交通运输发展，需要更快速，更高效，更可靠的城市货物配送系统来满足日益增长的服务需求。就目前所知，传统的卡车配送服务体系无法解决交通拥堵所导致的低时效问题，其服务质量和速度大程度受制于当前路况评定以及路线规划等问题。

根据最新的研究表明，若将无人机加入到传统配送体系，与卡车协同运作，将有助于减少配送总体效率并提高整体配送机动性。与传统卡车相比，无人机不受限于静态道路，而是在三维空间纵向运动，这使得路径更趋向于直线并且不受限于交通情况，其平均运输速度将大于卡车配送。但是无人机对于卡车相比，其最大飞行距离和最大负载能力将限制它的路径规划以及配送服务对象。所以配送效率的提高取决于合理的路径规划以及协作方式。

卡车无人机配送问题具有求解规模大，约束条件多，求解难度指数增长等困难。在现实情况中，无人机有效载荷有限，大多数情况下单次只能服务一个客户，且其服务范围受制于电池能源，发射回收需要卡车司机进行操作。所以卡车无人机配送问题为组合优化问题，且目标之间存在多重约束。

现在的大多数模型都针对单卡车单无人机，而算法多采用模拟退火、爬山法等智能优化算法，依赖初始路径的选择，容易得到局部最优解。

发明内容

本发明主要针对单卡车多无人机模型，整体的算法参照 memetic 文化基因算法，对路径采用自然数编码。

具体步骤如下：

步骤一、构建单卡车多无人机路径规划的数学模型。具体包括：

构建最小化目标函数。公式为 $\min \{ \max [t_{n+1}, \max \{ \widehat{t_{n+1}^v} \}] \}$, $\forall v \in U$, 其中 t_{n+1} 为卡车最后抵达仓库的时间, $\max \{ \widehat{t_{n+1}^v} \}$ 为无人机队最后独立到达仓库的时间, U 为所有无人机的集合。当无人机和卡车一起抵达仓库时, $\widehat{t_{n+1}^v}$ 值为 0。接下来需要建立约束条件，约束条件主要分为以下几类：服务路径限制，无人机路径限制，卡车无人机交互。

服务路径限制的约束条件表达式为 $\sum_{i \in C_-} x_{ij} = \sum_{k \in C_+} x_{jk} = 1$, $\sum_{i \in C_-} x_{ij} + \sum_{i \neq j} \sum_{k \in C_+} \sum_{v \in U} y_{ijk}^v = 1$, $\forall j \in C$, 其中 x_{ij} 表示卡车路径，若卡车从节点 i 出发并直接抵达节点 j 则为 1，否则为 0。 y_{ijk}^v 表示无人机 v 的路径，若无人机从节点 i 出发，到达节点 j 服务后，回到节点 k 和卡车汇合则为 1，否则为 0。 C_- 表示出发地点的集合， C_+ 表示抵达地点的集合， C 表示所有客户节点集合， U 表示所有无人机集合。该约束条件保证卡车和无人机只访问一次客户节点，并且流量守恒，到达客户节点后会离开客户节点，不会停留。

无人机路径限制的约束条件表达式为 $y_{ijk}^v(\widehat{d}_{ij} + \widehat{d}_{jk}) \leq L, \forall i \in C_-, \forall j \in C, \forall k \in C_+, i \neq j \neq k, \forall v \in U$, \widehat{d}_{ij} 表示无人机从节点*i*飞向节点*j*所飞行的时间。*L*表示无人机的最大飞行时间限制。该约束条件保证无人机的路径安排不超过其最大飞行时间, 保证无人机电量能满足飞行条件。

卡车和无人机限制的约束条件表达式为 $0 \leq \widehat{t}_j^v + \widehat{d}_{jk} \leq l_k, \forall j \in C, \forall k \in C_+, \forall v \in U$, 其中 l_k 表示卡车离开节点*k*时候的时间。 \widehat{t}_j^v 表示无人机*v*到达节点*j*的时间。该约束条件保证卡车在无人机全部回收后再出发。同理, $0 \leq \widehat{t}_i^v \leq l_i, \forall i \in C_-, \forall v \in U$ 表明卡车需要在无人机全部起飞后才能出发, 其中 \widehat{t}_i^v 表示无人机*v*离开节点*i*所用的时间。

步骤二、构建卡车无人机运输次序编码个体。具体包括, 每一个个体包括所有客户节点的无人机卡车分配情况以及先后次序。对于任意一个个体, 以一个一辆卡车两个无人机为例, 其编码为 $\begin{pmatrix} 0 & L_1 & X & L_2 & L_3 \\ 0 & C_1 & \cdots & C_{n-3} & 0 & C_{n-2} & X & C_{n-1} & C_n \\ 0 & R_1 & X & R_2 & R_3 \end{pmatrix}$, 其中第二行存放实际服务节点, 卡车路径和无人机路径用“0”作为分隔符区分, 不同无人机之间用“X”作为分隔符区分。X的数量为 $N_u - 1$, 其中 N_u 无人机的总数量。 L_i 表示无人机出发的节点为卡车路径中第几个节点(从0开始计数), 以上图为例, 当 $L_1 = 1$ 时, 代表第一架无人机从卡车路径第一个路径 C_1 出发访问客户节点 C_{n-2} 。 R_i 与之类似, 表示无人机降落的位置在卡车中的次序。

步骤三、初始化算法基础参数。具体包括, 个体种群数量*N*, 迭代次数*Gen*, 卡车节点时间矩阵 d_{ij} , 无人机节点时间矩阵 \widehat{d}_{ij} , 无人机个数 N_u , 客户节点个数 N_c , 无人机飞行时间限制*L*, 交叉概率系数 P_c , 变异概率系数 P_m 。

步骤四、初始化个体参数。具体包括, 以0表示仓库节点, 先采用随机整数 $\in [0, N_c - 1]$ 的方式产生只有卡车服务的路径编码, 每一个节点有且只访问一次, 最后回到仓库。在卡车路径生成结束之后, 在结尾插入 $N_u - 1$ 个无人机分割符

“X”。初始化的形式为 $\begin{pmatrix} 0 & X & X \\ 0 & C_1 & \cdots & C_{n-1} & 0 & X & X \\ 0 & X & X \end{pmatrix}$ 。

步骤五、计算各个体适应度。具体为, 先判断个体是否满足约束条件, 对满足约束条件的个体, 其适应度公式为 $F = \min \{ \max [t_{n+1}, \max \{ \widehat{t}_{n+1}^v \}] \}$, 对于不满足约束条件的个体进行惩罚, 其适应度计算公式 $F = \min \{ \max [t_{n+1}, \max \{ \widehat{t}_{n+1}^v \}] \} + \alpha$, 其中 $\alpha = \frac{\sum F}{N}$, $\sum F$ 为上一代所有个体适应度之和。

步骤六、运用随机抽样选择原则进行选择。具体为, 每个个体映射到区间的连续片段, 每个所在片段长度对于适应度的倒数成比例(适应度越小越好), 用

一组等距离指针随机指向以上区间，每个指针所指个体为所选个体。指针之间的距离为 $1/N$ 。

步骤七、根据顺序交叉算子进行个体交叉。具体为，选中一个个体作为父代，按顺序取出其除去所有分隔符“0”“X”的所有路径节点（编码中的第二行）作为父代路径，运用顺序交叉算子与另一选中个体进行交叉，步骤为，随机选择父代路径一个连续区段作为子代路径相应位置基因，将剩余基因按照另一父代路径中排序依次填入到子代路径中。最后将子代路径按照父代的分隔符依次填入生成子代。

步骤八、对个体进行变异操作。变异采用领域搜索的方法，具体拥有三个算子，分别为插入分隔符、2-opt 和插入节点。依次执行三个算子，记变异前的初始解为 S_0 。

首先进行插入分隔符操作。具体为，首先将 S_0 作为最优解，取出卡车和无人机服务路径（个体编码中的第二行）并剔除分隔符（卡车与无人机分割符“0”和无人机与无人机分隔符“X”），找到所有可能的插入位置按顺序插入分隔符（分隔符 0 在 X 之前）产生领域新解，并用无人机分配算子为产生的新路径安排无人机起飞降落位置。记录历史可行的最优解并记为 S_1 。

然后进行 2-opt 操作。具体为，首先将 S_0 作为最优解，取出卡车和无人机服务路径，找到所有可能的领域进行 2opt 操作，若操作区域同时存在“0”和“X”分隔符，根据需要调换“0”和“X”的位置保证“0”始终位于“X”的前列，之后用无人机分配算子为产生的新路径安排无人机起飞降落位置。记录历史可行的最优解并记为 S_2 。

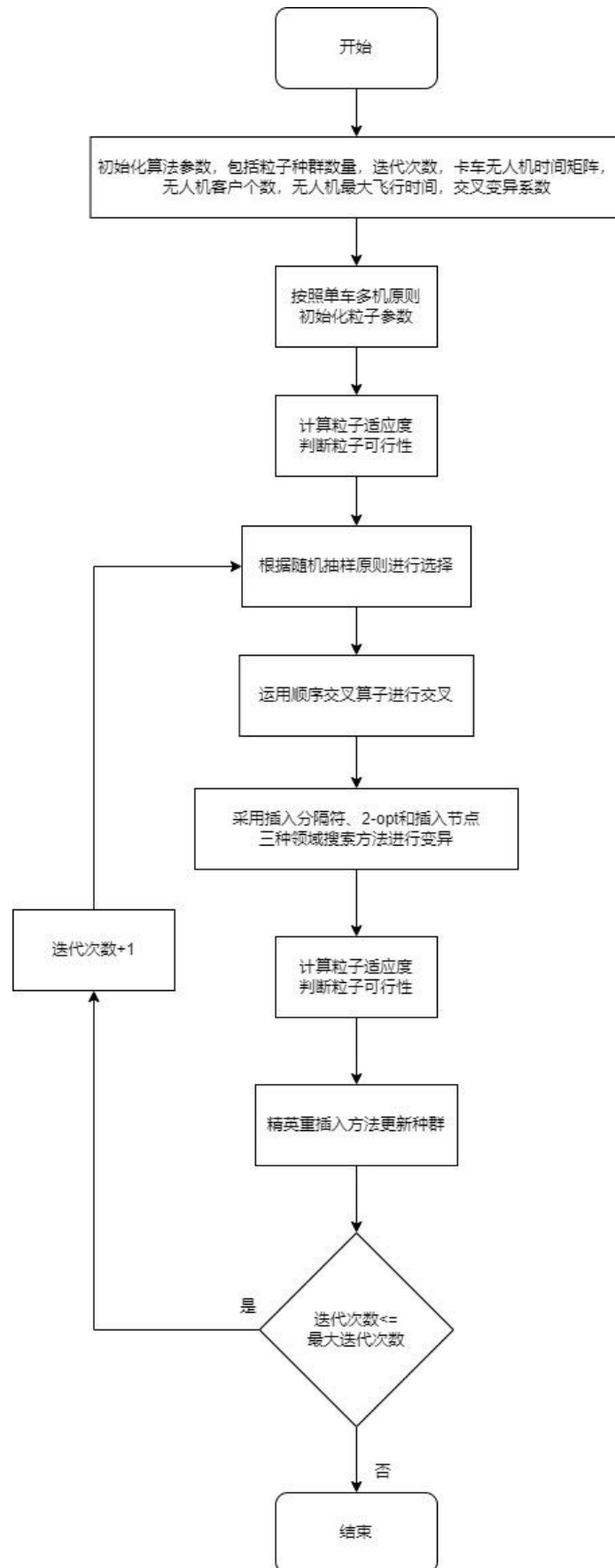
最后进行插入节点操作。具体为，首先将 S_0 作为最优解，取出卡车和无人机服务路径，随机选择一个客户服务节点，将其插入所有可能的位置并用无人机分配算子为产生的新路径安排无人机起飞降落位置。记录历史可行的最优解并记为 S_3 。

无人机分配算子具体为，根据无人机服务节点 j 生成与之对应的所有可行的起降位置，记为集合 P_j ，为无人机服务节点 j 从 P_j 随机选择一个起降组合作为其起降位置，并实时更新集合 P_j ，直到安排完所有的无人机服务节点。重复 5 次以上步骤，记录历史可行最优解。若无可行最优解，则返回初始解 S_0 。

比较 S_1 、 S_2 、 S_3 ，选择之中的最优解 $S_{New} = \min (S_1, S_2, S_3)$ 作为变异之后的解。

步骤九、精英重插入的方法更新种群。具体为，重新计算适应度，将交叉和变异产生的个体取代适应度较低的父代个体，保证种群的总数不变。

步骤十、返回步骤六，对种群进行迭代。



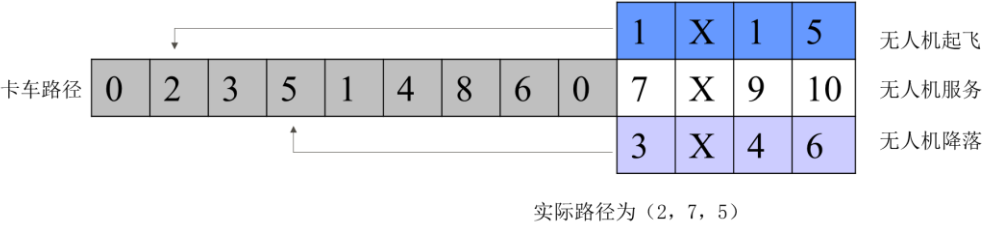


图 1 个体编码示意图

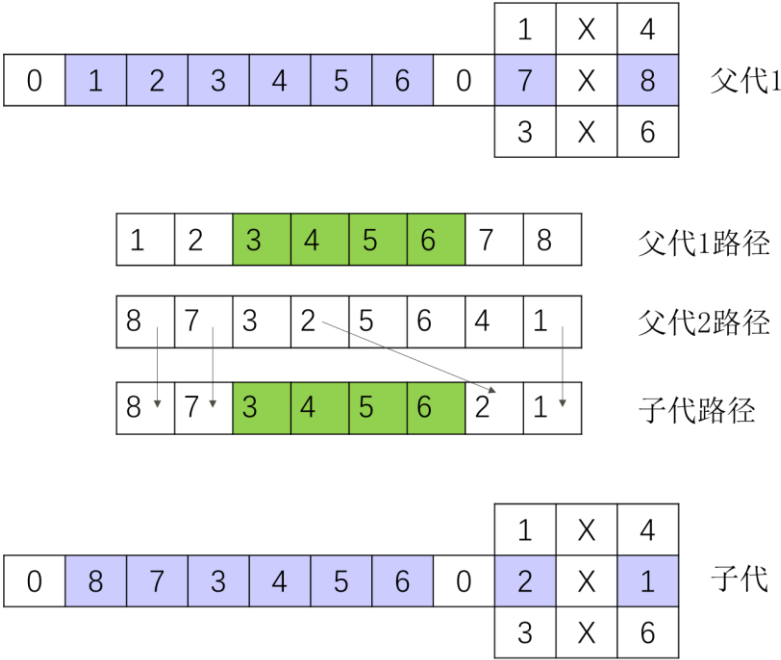


图 2 交叉示意图

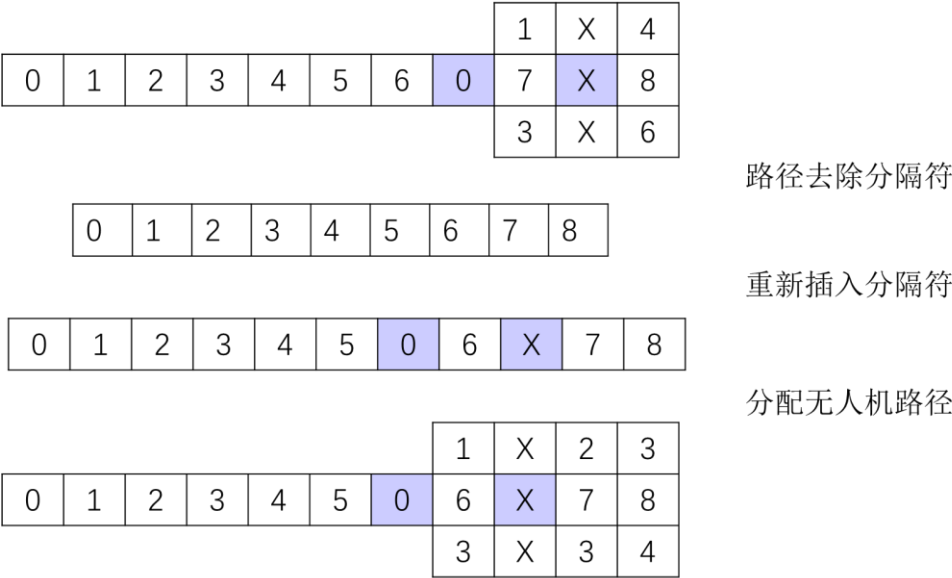


图 3 插入分隔符示意图

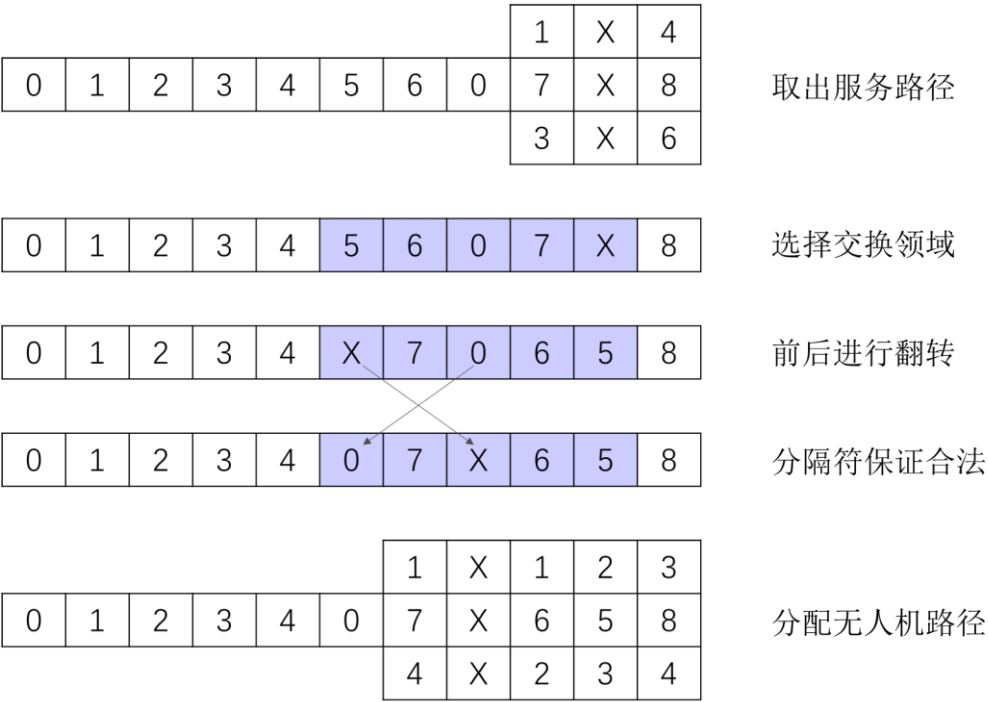


图 4 2opt 示意图