 DOI: https://doi.org/10.12345/bdai.v6i1.23361

Large scale electric vehicle routing problem solving method based on “divide and conquer” and column generation

Jianing Jiang

School of Economics and Management, China University of Mining and Technology, Xuzhou, Jiangsu, 221000, China

Abstract

In this paper, the divide-and-conquer strategy and column generation algorithm are used to solve the routing problem of large-scale electric vehicles. The large scale problem is decomposed into several small scale problems by the divide-and-conquer solution framework, and the column generation method is used to solve each small scale problem, and the obtained solutions are merged into the original solution. Based on the data set of Jingdong 2018 Global Optimization Competition, the calculation verification demonstrates the advantages of “divide and conquer” method in solving large-scale electric vehicle routing problems.

Keywords

electric vehicle routing problem; Divide and rule; Column generation; Large-scale combinatorial optimization; cluster

基于“分而治之”与列生成的大规模电动车辆路径问题求解方法

江佳宁

中国矿业大学经济管理学院，中国·江苏 徐州 221000

摘　要

本文综合运用“分而治之”的策略和列生成算法求解大规模电动车辆路径问题。采用“分治”的求解框架将大规模问题分解成多个小规模问题，并使用列生成方法求解各个小规模问题，最后将求得的解合并成原问题的解。基于京东2018全球优化竞赛的数据集进行了计算验证，说明了“分治”法在求解大规模电动车辆路径问题的优势。

关键词

电动车辆路径问题；分而治之；列生成；大规模组合优化；聚类

1 引言

近年来，电动汽车产业快速发展，电动汽车车辆路径问题（Electric Vehicle Routing Problem, EVRP）顺势被提出，且成为近年来管理运筹学领域的研究热点问题之一。

本文综合运用“分治”和列生成算法用于求解大规模 EVRP。建立了问题的基于弧的混合整数线性规划模型。对于大规模 EVRP，我们提出一种基于客户点聚类的分解策略，将大规模 EVRP 问题分解成多个小规模 EVRP。对于每个小规模 EVRP，采用基于列生成的方法进行求解。最后将多个小规模 EVRP 的解合并成原大规模 EVRP 的解。我们采用京东 2018 全球优化竞赛的数据集进行了计算验证，计算结果显示，在设置相同的算法终止条件下，采用“分治”法求得的总成本与直接求解的总成本的平均比值为 44%，说明了“分治”法在求解大规模 EVRP 的优势。

【作者简介】江佳宁（2004–），男，中国安徽桐城人，本科，从事大数据管理与应用研究。

2 问题描述

2.1 问题描述

下面对问题所涉及的要素进行详细描述。

路网：在一个城市中有多个节点，这些节点包含配送中心、客户点、充电站点。任意两个节点之间均有边（代表对应的道路）连接，边对应的距离和行驶时间均已知，且是固定的。节点以及边构成了车辆运行的路网。

车辆电动车辆对应一个固定的使用成本，电动车数量充足。服务过程中的实际装载容量、重量均不超过其最大载重量和最大装载体积。

配送中心：所有车辆均从配送中心出发，最终回到配送中心。

客户点：客户点是节点中的一个子集。每个客户点对应一个揽收需求或者一个配送需求，该需求的体积和重量已知（假设一个客户不同时有揽收或配送需求）。

充电站：充电站是节点中的一个子集。假设每个充电站都有足够数量的充电桩，也就是说，车辆一旦到达充电站

不需要排队就能立即进行充电。

问题优化的目标是最小化总运行成本。总成本是固定成本 f_k 、充电成本 g_k 、旅行成本 l_k 、等待成本 h_k 这四个成本之和。

基于上述描述，电动车辆路径问题（EVRP）可陈述为：如何在满足各种约束条件下，制定电动车的行驶路径以及充电计划，使得所有的需求都被满足，且总运行成本尽可能的小。

2.2 基本假设

本文所做的假设如下：假设车辆数量足够；充电站充电桩数量足够，车辆一旦到达充电站不需要排队就能立即进行充电；客户的服务时间窗是确定的，不变的；车辆续航里程是精确可计算的，不受天气、温度、道路情况等影响。

3 数学模型

3.1 参数

o	起点（仓库节点）
d	终点（仓库节点）
$(i,j)\in A$	弧，$i,j\in N$（节点），且 $i\neq j$
$k\in K$	车辆
t_{ij}	弧 (i,j) 的行驶时间
$[s_j,e_j]$	客户 $j\in J$ 的时间窗
u_j	在节点 j 处所需服务时间，如果 $j\in J$，则 u_j 表示的是服务时间；如果 $j\in C$，则 u_j 表示的是所需的充电时间。
q_j 、m_j 、a_{ij}	分别为节点 k 对应的负载重量，负载体积、续航里程
R_k 、Q_k 、M_k	分别为车辆 j 的续航里程、最大载重量、最大载货体积
l_k 、f_k 、g_k 、h_k	车辆 k 的单位距离成本、固定使用成本、充电成本、单位时间等待成本

3.2 模型

式（1）表示目标函数，即最小化总成本。式（2）表示每辆车 $k\in K$ 均从仓库出发，即所选择的流出节点 O 的弧的总数为 $|K|$ 和每辆车 $k\in K$ 最终均流入仓库。式（3）表示所选择的流入中间节点（包括客户节点和充电站节点）的弧数量等于流出的弧数量和选择流入每个客户的弧数量等于 1，表示每个客户被访问 1 次。式（4）是时间顺序约。式（5）表示访问用户的时间窗约束和展示了服务用户 $j\in J$ 时所需要的等待时间约束。式（6）展示了决策变量 y_k 与决策变量 x_{ij}^k 之间的关联关系。式（7）、（8）分别表示了车辆 $k\in K$ 沿一条弧行驶时在两个节点上的累积负载重量、累积负载体积所需满足的关系，这与时间顺序约束（4）具有类似的形式。式（9）分别表示车辆 $k\in K$ 在访问任意一个节点时，其累积的负载重量、累积负载体积都不能超过其最大负载重量、最大负载体积。式（10）表示了车辆 $k\in K$ 沿一条弧行驶时在对应两个节点上的剩余续航里程所需满足的关系。式（11）表示如果车辆 $k\in K$ 行驶至一个充电站，则剩余续航里程会得到补充。式（12）表示车辆 $k\in K$ 在访问任意一个节点时的剩余续航里程均需要大于等于零，且小于等于最大续航里程。式（13）给出了其余决策变量的取值范围。

$\min z=\sum_{k\in K}f_k y_k+\sum_{k\in K}\sum_{(i,j)\in A}l_k x_{ij}^k+\sum_{k\in K}\sum_{j\in J}h_k\cdot p_j+\sum_{k\in K}\sum_{i\in N}\sum_{j\in C}g_k x_{ij}^k$	(1)
$\sum_{j\in N^+(o)}x_{o,j}^k=1,\ \forall k\in K$ ； $\sum_{j\in N^-(d)}x_{j,d}^k=1,\ \forall k\in K$	(2)
$\sum_{k\in K}\sum_{i\in N}x_{ij}^k-\sum_{k\in K}\sum_{i\in N}x_{ji}^k=0,\ \forall j\in J\cup C$ ； $\sum_{k\in K}\sum_{i\in N}x_{ij}^k=1,\ \forall j\in J$	(3)
$b_i+u_i+t_{ij}-(1-x_{ij}^k)M\le b_j,\forall(i,j)\in A,k\in K$	(4)
$b_j\le e_j,\forall j\in J$ ； $p_j\ge s_j-b_j,j\in J$	(5)
$y_k\ge x_{ij}^k,i\in N,j\in J\cup C,k\in K$	(6)
$w_{ik}+q_j-(1-x_{ij}^k)Q_k\le w_{jk},\forall(i,j)\in A,k\in K$	(7)
$v_{ik}+m_j-(1-x_{ij}^k)M_k\le v_{jk},\forall(i,j)\in A,k\in K$	(8)
$q_i\le w_{ik}\le Q_k,\forall i\in N,\forall k\in K$； $m_i\le v_{ik}\le M_k,\forall i\in N,\forall k\in K$	(9)
$r_{ik}-a_{ij}-(1-x_{ij}^k)M_k\le r_{jk},\forall(i,j)\in A^r,k\in K$	(10)
$r_{ik}+(1-x_{ij}^k)M\ge R_k,(i,j)\in A^C,k\in K$	(11)
$0\le r_{ik}\le R_k,\forall i\in N,\forall k\in K$ ；	(12)
$x_{ij}^k\in\{0,1\},\forall(i,j)\in A,k\in K$ ； $y_k\in\{0,1\},k\in K$ ； $b_j\ge 0,p_j\ge 0,j\in J$	(13)

4 算法设计

4.1 基于聚类的分治求解策略

为了取得更好的求解效果，应当尽可能地将距离接近的节点划分到同一个小规模问题中去我们采用 K-means 聚类实现对节点的划分。其主要思路如下：①首先确定需要划分的小规模问题的数量 θ，然后计算出每个小规模问题所包含的客户点的数量 $n=\left\lceil\frac{N}{\theta}\right\rceil$（N 为全部客户点的数量）。②然后，从全部的客户点中，采用 K-means 聚类的方法，聚成一个包含 n 个客户点的节点子集 G，然后将配送中心和全部的充电站节点添加到这个节点子集 G 中。这样就形成了一个小规模问题。③将节点子集 G 中的客户点从所有客户点集合中删除。④重复上述过程，直到所有 r 个小规模问题全部生成。

在上述过程中，最后一个小规模问题中，其客户点的数量可能少于 n 个。

4.2 列生成方法求解线性松弛问题

对于每一个小规模问题，使用基于列生成 [1][2] 的求解方法。约束（2）~（13）给出了问题的解决空间，其中，对于车辆 $k\in K$ 来说，其可行的路径是从配送中心出发，访问若干个客户点后，返回配送中心。使用 Ω_k 表示车辆 $k\in K$ 的所有可行路径的集合，使用 $\Omega=\{\Omega_1,\Omega_2,\text{L}\ \Omega_k\text{L}\}$ 表示全部

可行路径的集合，则原问题转变为从集合Ω中选择一部分路径，使得每个客户都被访问一次。

给定可行路径$\phi \in \Omega$，其对应的总成本记为θ_ϕ，使用$\sigma_{\phi j} \in \{0,1\}$表示路径$\phi \in \Omega$是否访问客户$j \in J$，即$\sigma_{\phi j}=1$表示路径$\phi$访问了客户$j \in J$，$\sigma_{\phi j}=0$表示路径$\phi$不访问客户$j \in J$。

使用新的决策变量$\lambda_\phi \in \{0,1\}$表示是否选择路径$\phi \in \Omega$，则原问题可转换为如下问题：

$\min z=\sum_{\phi\in\Omega}\theta_\phi \cdot \lambda_\phi$	(14)
$\sum_{\phi\in\Omega}\sigma_{\phi j}\cdot\lambda_\phi=1,\forall j\in J$；$\sum_{\phi\in\Omega_k}\lambda_\phi=1,\forall k\in K$	(15)
$\lambda_\phi \in \{0,1\},\phi\in\Omega$	(16)
$\min z^k=\theta_\phi-\sum_{\forall j\in J}\left(\sigma_{j\phi}\cdot\xi_j\right)-\delta_k$；$\phi\in\Omega_k$	(17)

式（14）~（16）所定义的问题记为Ψ，我们称为主问题。由于集合Ω所包含的路径数量是非常多的，现实中不太可能完全列举出来。一种比较经典的做法是采用动态生成的方法，即先生成少量的路径，得到Ω的子集$\Omega' \subset \Omega$，基于Ω'求解Ψ的线性松弛问题（去除λ_ϕ的整数限制），然后根据求解的情况动态的添加路径到Ω'，不断地重复该操作直到满足某些条件时止。

将式（14）~（16）中的路径集合Ω和Ω_k分别使用子集Ω'和Ω'_k替换，则得到限制主问题Ψ'。记约束（15）对应的最优对偶解分别为ξ_j和δ_k，根据单纯形法最优性判定条件，可得车辆$k \in K$对应的子问题由式（17）表示。

列生成过程如下：先基于部分路径求解限制主问题Ψ'的线性松弛问题，得到约束（15）的最优对偶解。然后针对车辆$k \in K$求解子问题（17），得到新的路径ϕ以及子问题目标函数值z^k。如果$z^k<0$，则将路径ϕ添加到Ω'_k和Ω'。重复上述过程，直到$z^k \geq 0$。如果$z^k \geq 0$，说明Ψ的线性松弛问题已求得最优。

子问题对应一个“资源约束的最短路径”问题，我们采用基于标签的动态规划算法 [3][4]。

4.3 计算整数可行解

在 4.2 节，通过列生成方法可以求得Ψ的线性松弛问题的解，即所求得的λ_ϕ值可能取分数值。为了求得整数可行解，我们将列生成过程嵌入到分支定界的框架中，即在分支定界的每一个节点上进行一次列生成求解。我们采用如下的二元分支决策。①强制访问分支：客户j必须被车辆$k \in K$服务；②强制不访问分支：客户j必然不被车辆$k \in K$服务。通过这些方法，我们成功地找到整数可行解。

5 计算验证

5.1 实验数据

我们采用京东 2018 全球优化物流大赛 B 榜的数据集。表 1 列出了该数据集的统计信息。

表 1 数据集的统计信息

算例	客户数	配送客户数	揽收客户数
1	1500	1300	200
2	1400	1200	200
3	1300	1100	200
4	1200	1000	200
5	1100	900	200

实验数据由两部分数据组成，分别是距离时间数据和节点数据。表 2 展示了前 11 个节点数据。其第二列为节点的类型：1 代表配送中心，2 代表收货商家，3 代表发货商家，4 代表充电站；最后两列为对应客户的时间窗。

表 2 算例数据格式

ID	Type	Longitude	Latitude	Total weight	Total volume	Start time	Stop time
0	1	116.227105	39.761300	-	-	08:00	00:00
10001	2	116.437355	39.989739	0.0056	0.0099	09:30	10:00
10002	2	116.382306	39.960325	0.2076	0.3666	09:30	11:30
10003	2	116.623784	40.034688	0.00473	0.032	09:00	11:00
10004	2	116.321390	39.811570	0.05863	0.1687	08:00	09:30
10005	2	116.455510	39.944381	0.03645	0.0745	10:00	11:00
10006	2	116.546946	39.864005	0.02595	0.0542	08:30	09:30
10007	2	116.192636	39.828248	0.0198	0.1117	10:00	11:00
10008	2	116.390727	39.893730	0.02653	0.0409	08:00	09:30
10009	2	116.124924	40.046550	0.06097	0.1057	09:00	10:30
10010	2	116.388077	39.902642	0.01075	0.0455	10:30	11:30
……	……	……	……	……	……	……	……

我们选择了 Visual Studio 2015 作为开发平台，C++ 作为编程语言进行算法编程。本文的实验测试均在处理器为 Intel(R) Core(TM) i5-10210U CPU 1.60GHz 2.11 GHz、内存为 16GB 以及 64 位的 Windows10 操作系统的 PC 上运行。

5.2 实验结果

我们选择了不同的分组数量进行实验，分组的数量分别是 10,20，……60 组，记录下不同分组情形下求得的各种成本、使用的车辆数以及消耗的计算时间（Cpu time）。此外，我们也进行了不分组情形下的实验，

算例的“分治”法对应的成本最小的是 295077.8，最大的是 454918.3，这些数据普遍要比直接求解的成本要小很多。我们列出“分治”法对应的成本的平均值，并进一步计算这些平均值与直接求解的成本之间的比值，发现最小比值为 32%，最大比值为 59%。观察这些数据可以得出结论，“分治”法求得的目标函数值要明显小于直接求解法。

从求解值可以看出，说明了“分治”法的优势。

6 结论

本文运用“分治”和列生成算法求解大规模 EVRP。总体上采用“分治”的求解框架，基于客户点聚类的方式将大规模 EVRP 分解成多个小规模 EVRP，并使用列生成方法对求解各个小规模 EVRP，最后将多个小规模 EVRP 的解合并成原大规模 EVRP 的解。采用京东 2018 全球优化竞赛的数据集进行了计算验证，该数据集由 5 个超 900 个客户点的算例组成。将大规模 EVRP 分解成不同数量的小规模 EVRP，结果均显示在相同的算法终止条件下，采用“分治”法求得的总成本均小于直接求解的总成本，采用“分治”法求得的总成本与直接求解的总成本的平均比值为 44%，说明了“分治”法在求解大规模 EVRP 的优势。本文的局限性在于以下方面，在使用列生成求解各小规模问题时，没有使用精确的标签算法求得子问题的最优解。在后续研究中，可以改进子问题的求解策略。

参考文献

[1] Tahir A, Desaulniers G, El Hallaoui I. Integral column generation for set partitioning problems with side constraints[J]. INFORMS Journal on Computing, 2022, 34(4): 2313-2331.

[2] Martin-Iradi B, Ropke S. A column-generation-based matheuristic for periodic and symmetric train timetabling with integrated passenger routing[J]. European Journal of Operational Research, 2022, 297(2): 511-531.

[3] Duman E N, Taş D, Çatay B. Branch-and-price-and-cut methods for the electric vehicle routing problem with time windows[J]. International Journal of Production Research, 2022, 60(17): 5332-5353.

[4] Xia Y, Zeng W, Zhang C, et al. A branch-and-price-and-cut algorithm for the vehicle routing problem with load-dependent drones[J]. Transportation Research Part B: Methodological, 2023, 171: 80-110.