Research Article

A Combinatorial Benders’ Cuts Algorithm for the Local Container Drayage Problem

Zhaojie Xue, 1 Canrong Zhang, 2, 3 Peng Yang, 2, 3 and Lixin Miao 2, 3

1 Department of Transportation Engineering, College of Civil Engineering, Shenzhen University, Shenzhen 518060, China
2 Research Center for Modern Logistics, Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China
3 Shenzhen Logistics Engineering & Simulation Laboratory, Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China

Correspondence should be addressed to Peng Yang; yangpeng.lotus@gmail.com

Received 6 November 2014; Accepted 15 February 2015

Copyright © 2015 Zhaojie Xue et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper examines the local container drayage problem under a special operation mode in which tractors and trailers can be separated; that is, tractors can be assigned to a new task at another location while trailers with containers are waiting for packing or unpacking. Meanwhile, the strategy of sharing empty containers between different customers is also considered to improve the efficiency and lower the operation cost. The problem is formulated as a vehicle routing and scheduling problem with temporal constraints. We adopt combinatorial benders’ cuts algorithm to solve this problem. Numerical experiments are performed on a group of randomly generated instances to test the performance of the proposed algorithm.

1. Introduction

In recent decades container transportation has played an increasingly important role in international logistics. Container terminal is a crucial node for the departure and arrival of containers from/to the hinterland. For the container terminal, inland container transportation, that is, collections and distributions of containers from/to customers, make a great impact on the whole operational efficiency of the terminal. The inland container transportation mainly involves two transportation modes, road and rail, between which the road transportation has advantages in providing door-to-door services for customers. However, the road transportation covers relatively short distance but high cost per TEU (20 ft equivalent unit) per kilometer. Moreover, road transportation generates a lot of congestions and environmental pollutions in the local area of the terminal. Hence it is significant to do operational research on the problem of the container road transportation.

The container road transportation between a terminal and customers, providing container pickup and delivery services to the terminal, is usually referred to as drayage operation. Here we study the drayage operation in the local area near a terminal. Without loss of generality, the terminal can be a seaport, a railway station, a depot, or a warehouse. The appropriate local area makes that the traveling time between the terminal and customers is short relatively to the container packing or unpacking time at the customers. This problem is referred to as Local Container Drayage Problem (LCDP). In LCDP a container truck consists of a tractor and a trailer, and normally they can be separated. For a pickup or a delivery task, if the packing or unpacking time is long enough relatively to the time that spent on the traveling from the terminal to the customer location, when the truck arrives at the customer location, the tractor, which can be assigned to handle other tasks, would be separated with the trailer, leaving the trailer and the container at the customer location for packing or unpacking. It is intuitive that the utilization of the container trucks will increase and the cost will decrease with this tractor and trailer separated operation mode.

The rest of paper is organized as follows: Section 2 gives a literature review of related research. In Section 3 the problem is described in detail and the mathematical formulation is given. In Section 4 a solution algorithm,
combinatorial benders’ cuts algorithm, is presented, followed by the numerical experiments shown in Section 5 in order to testify the efficiency of the algorithms. In the last section the conclusions are delivered.

2. Literature Review

Container drayage problem has been extensively studied in recent years. The authors of [1] modeled the local truck-load pickup and delivery problem as a multiple traveling salesman problem with time windows. Overconstrained and underconstrained problems were iteratively solved to obtain a solution. And a specific time window partitioning scheme was used. The authors of [2] addressed a problem of vehicle routing that arises in picking up and delivering full container load from/to an intermodal terminal. Delivery and pickup trips can be merged in one route, if the time of tasks and the suitability of containers for cargo allow. Potential cost and time saving are expected. A subgradient heuristic based on lagrangian relaxation was developed. The authors of [3] extended the problem in [2] by considering time windows at customer locations and the terminal. A two-phase insertion heuristic was proposed to construct an initial solution, which was then improved with a local search heuristic. The authors of [4] studied the cross-border drayage problem. They developed an attribute-decision model for the problem, and an adaptive labeling algorithm was proposed to solve it. Various regulatory policies were evaluated through numerical experiments. The authors of [5] studied the local container drayage problem given a port access appointment system, imposing a limitation on the number of trucks that can access the port during each time slot. The problem was formulated as a truck routing and scheduling problem with port access capacity constraints. The authors of [6] focused on a fleet management problem for a container company. The operating cost was divided into three parts: the routing cost, the resource assignment cost, and the container repositioning cost. The authors considered the dynamic character of this problem. The proposed solution approach was based on the decomposition of this problem into three simpler subproblems associated with each of the three cost components. Random and real-world cases were used to conduct evaluation experiments. The authors of [7] examined a container drayage problem in a one-terminal and one-depot local area. The problem included three types of container movements as inbound full, outbound full, and inbound empty movements. Empty containers were considered as separate transportation resources besides trucks. It also dealt with the time windows at the terminal and customers places. The authors of [8, 9] extended this problem to an environment of one-terminal with multidepot and multiterminal with multidepot local area, respectively. They solved the problem using different methods, reactive tabu search algorithm, and window partition based heuristic, respectively. The drayage systems always involve multiple resources, such as tractors, trailers, and drivers. The author of [10] modeled the drayage problem as a multiresource routing problem with flexible tasks. The solution approach developed to handle this problem included column generation embedded in a branch-and-bound framework.

The previous research of our team, [11], has studied the LCDP under a special operation mode with tractor and trailer separation. Another essential difference between [11] and the above reviewed research is that we specially consider the needed container (un)packing time at each customer. We formulate the problem as a variation of Vehicle Routing Problem (VRP), in which the container processing time at each customer location can be treated as a set of temporal constraints between different tasks. Temporal constraint is a kind of Synchronization Constraints (SC). Vehicles are independent of one another in the classical VRP. In the VRP with SC, by contrast, vehicles are interdependent with each other, that is, a change in one route may affect other routes because of the SC between different tasks on different routes. The paper [12] is a recommended survey on the VRP with SC. It presented a classification of different types of synchronization and discussed the central issues related to the VRPs with multiple SC from the perspectives of modeling and algorithm. Because of the SC, for the algorithms developed for the classical VRP some additional solution efforts are necessary to solve such kind of variations of VRP in most cases. For example, when the exact approach Branch-Cut-Price is applied on the VRP with SC, the pricing subproblem becomes a shortest path problem with resource constraints, which still has no successful exact solution method [12]. When adopting local search heuristics for such kind of problems, a local change within one route may result infeasibility because of the SC between different tasks. The paper [13] presented a mathematical model for the combined vehicle routing and scheduling problem with time windows and additional temporal constraints. The research proposed an optimization based heuristic to solve real size instances and made a computational study by comparing a direct use of a commercial solver against the proposed heuristic. The paper [14] studied the VRP with time windows and temporal dependencies. Four different formulations were introduced and ranked according to the tightness with which they describe the solution space. The Dantzig-Wolfe decomposition of the time-indexed formulation and a column generation-based solution approach were adopted.

In this paper, we study the same problem as our previous work, [11]. In [11], we solved the problem using a tabu search heuristic. As we know, although the heuristic can handle real-size problems, it cannot always obtain the optimal solution, even for small-size problems. In this paper, we adopt a combinatorial benders’ cuts algorithm to exactly solve the problem. This is the innovation of our research from the perspective of algorithm in the area of the VRP with SC.

3. Problem Modeling

We state the problem in [11] as follows. The LCDP assumes that there is a certain number of pickup and delivery full-container-load tasks that need to be fulfilled by a set of container trucks in a local terminal area within a fixed time period. We can mathematically represent the LCDP using a graph \( G = (N, A) \).
Table 1: Calculation of service setup time $t_{ij}$.

<table>
<thead>
<tr>
<th>$i$</th>
<th>0</th>
<th>$P$</th>
<th>$D$</th>
<th>$P'$</th>
<th>$D'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>—</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
</tr>
<tr>
<td>$P$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0} + \tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td></td>
</tr>
<tr>
<td>$D$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0} + \tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td></td>
</tr>
<tr>
<td>$P'$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td></td>
</tr>
<tr>
<td>$D'$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td>$\tau_{i0}$</td>
<td></td>
</tr>
</tbody>
</table>

Here the notations of $\theta(i) = i'$ and $\theta(i') = i$ for $\forall i \in C_1$ are introduced to describe the correspondence between $C_1$ and $C_2$.

The arc set is defined as $A = \{(i, j) \mid i, j \in N, i \neq j\}$. All divided task nodes are related to their corresponding customer nodes in space. A physical travel time $\tau_{ij}$, which is the direct vehicle travel time from the location of node $i$ to the location of node $j$, is associated with each arc $(i, j)$ of $A$. In reality, $\tau_{ij}$ is often calculated as the travel time on the shortest path between nodes $i$ and $j$ of the road network. Based on $\tau_{ij}$, we define an attribute $t_{ij}$, the service setup time, for each arc $(i, j)$ of $A$, as the vehicle travel time from node $i$ to node $j$ if a tractor serves the nodes $i$ and $j$ consecutively. $t_{ij}$ can be calculated with the information given in Table 1.

In Table 1, the service setup times between the terminal and task nodes and vice versa can be calculated easily. If a tractor serves a task node in $C_1$ and a task node in $C_2$ consecutively, it can go directly from the former task node to the latter one. So the service setup time between a task node in $C_1$ and a task node in $C_2$ is the same as the direct vehicle travel time between the two nodes. The service setup time between a task node in $D'$ and a task node in $P$ is equal to the direct vehicle travel time between the two of them. The reason is that in this paper we consider the strategy of sharing empty containers between the delivery customers and the pickup ones, that is, the empty container which is available after unpacking at a delivery customer can be used directly by a pickup customer. For other combinations of $i$ and $j$ in Table 1, the tractor needs to go back to the terminal from node $i$ and then to the location of node $j$. So under this situation, the service setup time equals the summation of the travel time of the two processes.

To solve the LCDP, we need to construct some routes, each of which is conducted by one tractor and starts from the terminal node, consecutively serves some task nodes, and returns to the terminal node in the end. Note that each task node must be served by one tractor exactly once. And the two task nodes corresponding to the same pickup or delivery customer do not have to be served by the same tractor. At each customer’s location it needs some time to finish the (un)packing work. So the service beginning time of the task nodes should satisfy not only the sequence constrains in one route, but also the (un)packing time of the customers requirement. The operational objective of the LCDP is to minimize the total cost, including the fixed cost of tractors and the variable cost of total vehicle travelling.
3.1. Mathematical Model

3.1.1. Denotations

\( \rho_1 \): the fixed cost of a tractor;
\( \rho_2 \): the unit variable cost of vehicle traveling;
\( p_i \): (un)packing time needed by customer \( i, \ i \in C_1 \);
\( T \): the whole time period;
\( K \): a sufficient large constant.

3.1.2. Decision Variables

\( X_{ij} \): it equals 1 if node \( i \) and \( j \) are served consecutively by the same tractor, 0 else; \( i, j \in N, i \neq j \);
\( S_i \): the service beginning time of the tractor at node \( i, i \in C \).

3.1.3. Mathematical Formulation. Consider [LCDP]

\[
\begin{align*}
\text{Min} & \quad \rho_1 \sum_{j \in C} X_{0j} + \rho_2 \sum_{i \in N \setminus \{i\}} t_{ij} X_{ij}, \\
& \quad \sum_{i \in N \setminus \{j\}} X_{ij} = 1, \quad \forall j \in C, \\
& \quad \sum_{i \in N \setminus \{j\}} X_{ij} - \sum_{i \in N \setminus \{j\}} X_{ji} = 0, \quad \forall j \in C, \\
& \quad S_i + t_{ij} \leq K \left( 1 - X_{ij} \right) + S_p, \quad \forall i, j \in C, \ i \neq j, \\
& \quad S_i + p_i \leq S_{0i}, \quad \forall i \in C_1, \\
& \quad S_i \geq t_{0i}, \quad \forall i \in C_1, \\
& \quad S_i + t_{0i} \leq T, \quad \forall i \in C_2, \\
& \quad X_{ij} \in \{0, 1\}, \quad \forall i, j \in N, \ i \neq j.
\end{align*}
\]

The objective function is minimizing the total cost, which consists of the fixed tractor cost and the operating variable cost. Constraints (3) ensure that each task is processed exactly once. Equations (4) enforce the flow conservation constraints. Constraints (5) represent the relationship of the service beginning times between two consecutive tasks served by the same tractor. Constraints (6) ensure the (un)packing time requirements of customers, which are treated as temporal constraints between different task nodes. Constraints (7) and (8) implicitly make sure all tasks should be finished within the whole time period, which put duration constraints on each route. Constraints (9) define the decision variables.

4. Combinatorial Benders’ Cuts Algorithm

If we ignore the temporal constraints between different tasks and treat the duration constraints on each route as vehicles’ capacity constraint, the LCDP can be simplified to classical vehicle routing and scheduling problem, which was shown to be NP-hard in [15]. In this section we develop a combinatorial benders’ cuts (CBC) algorithm to solve this problem. CBC was firstly developed by [16], and [17] extended it to solve mixed-integer linear programming with special structures. CBC algorithm derives benders’ cuts from the minimal set of inconsistencies of the subproblem, nevertheless traditional benders’ cuts method is from the dual information of the subproblem.

In the LCDP formulation, there are both binary variables \( X \) : \( \{X_{ij}\}_{i \in N \setminus j} \) and continuous variables \( S \) : \( \{S_i\}_{i \in C} \). The objective function is only dependent on the binary variables. We can split the [LCDP] into a [MASTER] subproblem and a [SLAVE] linear system:

[MASTER]

\[
\begin{align*}
\text{Min} & \quad \rho_1 \sum_{j \in C} X_{0j} + \rho_2 \sum_{i \in N \setminus \{j\}} t_{ij} X_{ij}, \\
& \quad \sum_{i \in N \setminus \{j\}} X_{ij} = 1, \quad \forall j \in C, \\
& \quad \sum_{i \in N \setminus \{j\}} X_{ij} - \sum_{i \in N \setminus \{j\}} X_{ji} = 0, \quad \forall j \in C,
\end{align*}
\]

[SLAVE](X)

\[
\begin{align*}
S_i + t_{ij} & \leq K \left( 1 - X_{ij} \right) + S_p, \quad \forall i, j \in C, \ i \neq j, \quad (11) \\
S_i + p_i & \leq S_{0i}, \quad \forall i \in C_1, \quad (12) \\
S_i & \geq t_{0i}, \quad \forall i \in C_1, \quad (13) \\
S_i + t_{0i} & \leq T, \quad \forall i \in C_2, \quad (14)
\end{align*}
\]

We solve the [MASTER] problem at integrality. It is assumed that the [MASTER] problem is feasible and bounded. Let \( X^* \) be an optimal solution of the [MASTER] problem. Then if the linear system [SLAVE](\( X^* \)) is feasible, that is, it has a solution \( S^* \), \( (X^*, S^*) \) is an optimal solution of the [LCDP]. If [SLAVE](\( X^* \)) is infeasible, we look for a Minimal Infeasibility System (MIS) of [SLAVE](\( X^* \)). Here a subsystem is called an MIS if it is infeasible and it has no infeasible proper subsystem.

We adopt the fast MIS search method proposed in [17]. This method requires that the [SLAVE](\( X^* \)) should be decomposed into two parts: conditional constraints (II) which include \( X^* \) and unconditional constraints (12)–(14) which only involve the continuous variable \( S \). The detailed description of fast MIS search method is referred to [17]. We denote the row indices of conditional constraints (II) which are involved in the MIS of [SLAVE](\( X^* \)) as \( \Psi \). From the definition of the MIS, we can observe that at least one binary variable \( X_{ij} \) in \( \Psi \) has to be changed to break the infeasibility, which can be translated to the following CBC cut:

\[
\sum_{X_{ij} \in \Psi} X_{ij} \leq |\Psi| - 1.
\]

One or more CBC cuts of the above type are derived and are added back to the [MASTER] problem, and iteratively we can
solve the original problem exactly. It is advisable to embed the cuts generation process into a branch-and-cut scheme. In our computational experiments, we use commercial software Cplex's standard branch-and-cut framework to make comparisons.

Additionally, we derive three kinds of valid inequalities of the [LCDP], which can be included into the [MASTER] subproblem to help to strengthen the [MASTER] subproblem and to increase the feasibility of the [SLAVE] subproblem.

(1) Lower bound on the number of tractors (CUT1):

\[
\text{CUT}_1: \sum_{j \in C} X_{0j} \geq \left( \sum_{i \in N} \sum_{j \in N, j \neq i, j \neq \theta(i)} t_{ij} X_{ij} + \sum_{i \in C_1} p_i X_{i\theta(i)} \right) / T.
\]  

(16)

This lower bound on the number of tractors takes into account not only the travelling time, but also the (un)packing time at customers.

(2) Subtour elimination inequalities (CUT2):

\[
\text{CUT}_2: X_{ij} + X_{ji} \leq 1, \quad \forall i, j \in C.
\]  

(17)

Subtour needs to be forbidden in the VRP. To make a trade-off between accuracy and tractability, here we only consider the two nodes' subtour elimination.

(3) Deadlock elimination inequalities (CUT3):

\[
\text{CUT}_3: X_{\theta(i)j} + X_{\theta(j)i} \leq 1, \quad \forall i, j \in C_1.
\]  

(18)

Deadlock needs to be forbidden in scheduling problem. Here we only consider the situation of two pairs of nodes \((i, \theta(i)), (j, \theta(j))\) for simplicity.

In sum, the procedure of the proposed CBC algorithm is shown in Figure 2.

### 5. Computational Experiments

In this section computational experiments are conducted to testify the performance of proposed algorithms. First we describe the test instances generation process. Then the computational results are listed and some analyses are conducted. All procedures and proposed algorithms are coded in Java and embedded with ILOG Cplex concert technology (version 12.4). All experiments are executed on a PC with a CORE i5 CPU of 2.50 GHz and an RAM of size equal to 4.0 GB.

#### 5.1. Test Instances

We generate test instances randomly to testify the performance of proposed algorithms. The local area is set to be a square with side length equal to 200 km. The terminal is located at the center of the area. And all customers' locations are uniformly distributed in the region. The physical traveling time between two task nodes are calculated simply by the Euclidean distance between the two locations divided by the speed of vehicle which is set as 70 km/h. As described in Table 1 the service setup time between different task nodes can be obtained. Other parameters are set as below:

- 16 hours: whole time period;
- \(U(3, 5)\) hours: (un)packing time of customer \(i, i \in C_1\);
- 10: fixed cost for each tractor;
- 1: traveling cost per unit of time.

#### 5.2. Computational Results

In this part, we show the computational results of the proposed CBC algorithm for a group of randomly generated instances. The CBC cuts generation process is embedded into a branch-and-cut scheme (Cplex), and hence comparisons between the proposed CBC algorithm and a conventional branch-and-cut algorithm are also conducted. In Cplex, there are two methods, conventional branch-and-cut and dynamic search, to solve general MIP models. We directly use Cplex, setting MIP search parameter to branch-and-cut, as the comparison code with the CBC algorithm. In addition, we also compute the results using the CBC algorithm without valid inequalities (CUT1, 2, 3) to testify the utility of the valid inequalities.

Table 2 shows the computational results. Column number means the index of the instance. \(P/D\) represents the number...
<table>
<thead>
<tr>
<th>Number</th>
<th>P/D</th>
<th>CBC</th>
<th>Branch-and-cut (CPLEX)</th>
<th>CBC without CUT 1, 2, 3</th>
<th>Gap*</th>
<th>Gap**</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1/2</td>
<td>20.5</td>
<td>3.21E+02</td>
<td>153</td>
<td>12</td>
<td>1.27E+03</td>
</tr>
<tr>
<td></td>
<td>2/2</td>
<td>21.4</td>
<td>1.77E+03</td>
<td>871</td>
<td>24</td>
<td>4.1</td>
</tr>
<tr>
<td></td>
<td>3/2</td>
<td>23.6</td>
<td>2.03E+04</td>
<td>2165</td>
<td>23</td>
<td>32.2</td>
</tr>
<tr>
<td></td>
<td>4/2</td>
<td>22.8</td>
<td>4.51E+04</td>
<td>807</td>
<td>6</td>
<td>23.6</td>
</tr>
<tr>
<td></td>
<td>5/2</td>
<td>38.7</td>
<td>1310.1</td>
<td>25889</td>
<td>5</td>
<td>3326.0</td>
</tr>
<tr>
<td></td>
<td>6/2</td>
<td>36.8</td>
<td>1289.5</td>
<td>20167</td>
<td>11</td>
<td>4058.2</td>
</tr>
<tr>
<td></td>
<td>7/2</td>
<td>37.9</td>
<td>12173.2</td>
<td>45063</td>
<td>1</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>4/3</td>
<td>39.2</td>
<td>18000</td>
<td>101859</td>
<td>0</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>5/3</td>
<td>40.8</td>
<td>18000</td>
<td>142137</td>
<td>10</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>6/3</td>
<td>39.2</td>
<td>18000</td>
<td>143221</td>
<td>1</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>7/3</td>
<td>39.7</td>
<td>18000</td>
<td>159763</td>
<td>1</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>8/3</td>
<td>36.0</td>
<td>18000</td>
<td>137129</td>
<td>2</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>9/3</td>
<td>40.6</td>
<td>18000</td>
<td>130077</td>
<td>0</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>10/3</td>
<td>38.1</td>
<td>18000</td>
<td>121977</td>
<td>4</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>11/3</td>
<td>39.7</td>
<td>18000</td>
<td>159195</td>
<td>2</td>
<td>18000</td>
</tr>
<tr>
<td></td>
<td>12/3</td>
<td>39.0</td>
<td>18000</td>
<td>12764</td>
<td>20</td>
<td>18000</td>
</tr>
</tbody>
</table>

*Gap = (Best<sup>2</sup> − Best<sup>1</sup>)/Best<sup>1</sup> (%).
** Gap = (Best<sup>3</sup> − Best<sup>1</sup>)/Best<sup>1</sup> (%).
of pickup and delivery customers. There are kinds of algorithms which are tested: the CBC algorithm, conventional branch-and-cut in Cplex, and the algorithm without the valid inequalities (CUT, 1, 2, 3). We list the best solutions (Best) obtained within the computation time (CPU), the number of nodes in the branch-and-bound tree (Nodes), and the number of cuts added (Cuts). The results show that CBC can always get better (or the same) solutions in shorter CPU time than the branch-and-cut algorithm in Cplex. The average gap between the solutions of two kinds of algorithms is 6.08%. It is investigated that the number of cuts added during the computation for CBC is larger than in Cplex. Consequently, the number of nodes in the branch-and-bound tree becomes less with the effects of the cuts. When comparing the two methods of the CBC algorithm with and without three categories of valid inequalities, it is shown that the valid inequalities would averagely improve the performance by 1.67%.

6. Conclusions

Inland container drayage plays an important role in the whole chain of the international transportation. In this paper we study the local container drayage problem under the operation mode of tractor and trailer can be separated. The problem is formulated as a vehicle routing and scheduling problem with temporal constraints. A CBC algorithm is developed to solve this problem. Comparing with standard branch-and-cut algorithm in Cplex, the CBC algorithm averagely improves the performance by 6.08%. From the computational results, we can see that although the CBC algorithm improves the performance comparing with the commercial software Cplex, it still cannot handle the natural combinatorial complexity of the problem. This is a challenging direction for future research.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the two anonymous referees for their valuable comments and suggestions which improved the quality of this paper. This work is supported by the National Nature Science Foundation of China under Grant nos. 71472108 and 71102011.

References

Submit your manuscripts at
http://www.hindawi.com