A hierarchical relaxations lower bound for the capacitated arc routing problem

Anita Amberg\(^1\) and Stefan Voß\(^2\)

\(^1\)Darmstadt University of Technology, FG Operations Research, Hochschulstraße 1, D-64289 Darmstadt, Germany
\(^2\)Braunschweig University of Technology, Institut für Wirtschaftswissenschaften, Abt-Jerusalem-Straße 7, D-38106 Braunschweig, Germany; (Corresponding Author, e-Mail: stefan.voss@tu-bs.de)

Abstract

Amongst the most important features of supply chain management are those related to interconnecting various companies within a supply network. While the performance of these interconnections is greatly influenced by information technology and information management we still face another important issue regarding logistics performance which is transportation, routing, and network design.

In this paper we consider an important problem within this frame, the capacitated arc routing problem (CARP), and provide a new lower bounding scheme to obtain bounds on the quality of heuristically obtained solutions. Given a network with demands on the arcs the CARP is to find a set of minimum cost routes for a set of vehicles to service all arcs with positive demands without violating the capacities of the vehicles. Based on benchmark instances from the literature as well as real-world problem instances we can show that our bounds are more effective than other bounds available from the literature.

1 Introduction

Arc routing problems arise in several areas of distribution management. Interest in these problems is increasing due to deregulation policies influencing public companies formerly responsible for most of these problem areas. In this paper we consider the capacitated arc routing problem (CARP) which belongs to an important class of vehicle routing problems with a great variety of real-world applications.

In general, the CARP can be characterized as follows. A set of customers has to be served by a fleet of vehicles operating from one or more depots. Each vehicle starts and ends its route at the depot it is assigned to. Furthermore, it has given capacities with respect to time (i.e., upper bounds on the maximum allowed time duration) and quantity (e.g., a maximum allowed capacity for demand to be served). Additional side constraints may exist, e.g., restricting the assignment of customers to vehicles. The objective is to find routes for the vehicles where each route satisfies the capacity constraints of the respective vehicle, each customer is served, and all side constraints are fulfilled such that the total transportation costs are minimized. Often, instead of optimizing transportation costs, the objective is to minimize transportation time or distance.

Arc routing applications like the CARP refer to problems where the distribution or collection of goods is bound up with traversing a distance. Typical applications that have been considered over the last decades refer to mail delivery, snow removal and winter gritting, garbage disposal, street sweeping and police patrols. That is, customers are modelled as arcs or edges, whereas in node routing problems customers correspond to the nodes as, e.g., in the traveling salesman problem.

The purpose of this paper is to investigate a lower bounding scheme for the CARP on undirected graphs. First some basic properties on the CARP will be presented to understand the concepts developed later throughout the paper. Then we shall give a survey on some important lower bounds for the CARP taken from the literature. This includes the exposition of some new ideas and development of cuts around depots which are used to determine a hierarchy of lower bounds. Results are presented in Section 4 and explained in detail for some benchmark instances within a comprehensive appendix. Finally some conclusions and ideas for future research are given in Section 5.

2 The capacitated arc routing problem

Based on the above description of the CARP customers, depots and streets may be represented by a graph $G = (V, E, c, d)$ with node set $V$ and edge set $E$. Usually, one considers a subset $R$ of $E$ of so-called required or productive edges that need to be served. By $g(i)$ we denote the degree of node $i$, i.e., the number of edges incident to $i$. For each edge $e = [i, j] \in E$ we assume a nonnegative (transportation) cost $c_e(= c_{ij})$; for each $e \in R$ we assume a posi-
tive demand \( d_e(= d_{ij}) \). \( M \) denotes the number of vehicles.\(^1\) Each vehicle has a given demand capacity; for ease of exposition we assume an identical capacity \( Q \). We call the depot or center node where all routes start and end \( v_{\text{depot}} \). Edges incident with \( v_{\text{depot}} \) are called central edges. For simplicity, if not stated otherwise, we assume all edges \( e \in E \) to be productive, i.e., having \( d_e > 0 \). Thus the total cost of the required edges is \( c(E) = \sum_{e \in R} c_e; d(R) = \sum_{e \in R} d_e \) gives the total demand.

Now, the CARP is to find routes starting and ending at the depot, satisfying the capacity constraints, covering all required edges and having minimum total cost. One may distinguish the 1-CARP (CARP for short) where all vehicles are stationed in the same node (the depot or center) and have identical capacity constraints, and the \( m \)-CARP with vehicles starting from multiple depots and/or having different capacity requirements. The underlying graph of a CARP may be directed, undirected or mixed, and the subgraph of required arcs that need serving may be connected but need not be. We distinguish unit weight problems where all productive edges have the same demand and general weight problems where the demand may be different for all edges.

Within the literature a great variety of special cases of the NP-hard CARP has been intensively studied. The well-known Chinese postman problem (CPP) is the basic arc routing problem named after the Chinese scientist Kwan [14] who was the first to publish on this problem. The CPP may be used as a relaxation for capacitated arc routing problems. Though polynomial algorithms for the CPP in undirected or directed graphs are available (cf. [11]) already the CPP in mixed graphs (containing directed arcs as well as undirected edges) with some nodes having odd degree is NP-hard as shown by [16]. Introducing additional constraints even in undirected or directed graphs usually yields NP-hard problems such as the rural postman problem (RPP) where the set of required arcs (i.e., those arcs which need serving) need not be connected and has to be linked using non-required arcs, or the capacitated Chinese postman problem (CCPP), where the capacity of the postman is restricted (used throughout the literature as basic model of the 1-CARP described above). For an excellent up-to-date review of arc routing problems including an extensive treatment of the CARP see [2]. A more recent survey may be found in some of the contributions in [9].

When developing solution methods, it is important to note that capacitated arc routing problems consist of two interdependent subproblems: The assignment problem which forms subsets or clusters of required arcs served by the same vehicle and the sequencing or routing problem which determines the sequence of serving the arcs. Most heuristics for the CARP are based on ideas to solve these subproblems sequentially. Recently also some meta-heuristics have been considered; see e.g. [1, 13] for different tabu search implementations and [15] for a genetic algorithm.

3 Lower bounds

Lower bounds for the CARP have been investigated throughout the literature with increasing success although large scale real-world problem instances have not yet been successfully investigated to a full extent. Usually lower bounds for the CARP are initially described for the undirected CCPP and then extended to the CARP as necessary (i.e., considering problem extensions such as multiple depots or different types of vehicles with different capacities).

In this section we first provide insights into some of the efforts undertaken in the literature and then provide the concept of the hierarchical relaxations lower bound (HRLB).

A first but usually not very good lower bound for the CARP is obtained solving the CPP. Other lower bounds are the matching lower bound MLB [12], the node scanning lower bound [3], the matching path lower bound [17], the successive cuts lower bounds [18] and the node duplicated lower bounds ND1, ND2 and ND3 [6]. Additional references on polyhedral results and valid inequalities for cutting plane algorithms are [4, 5].

In the (undirected) CPP relaxation of the CARP (Section 3.1), one giant route has to be found. To guarantee continuity of this route, each node must have even degree. Thus, we refer to the respective constraints as even degree constraints. If capacity is restricted, smaller routes starting and ending at the depot have to be constructed and capacity degree constraints can be derived. These constraints force the number of edges in so-called critical cuts (where edges are missing) to be sufficient for the required number of vehicles to spread all over the graph. The MLB, e.g., considers one capacity degree constraint – the one for the cut separating the depot from the rest of the nodes (Section 3.2). The above mentioned bounds from [18, 6] also use the notion of cuts, however, combining only single cuts with the even degree constraints. In Section 3.3 we show how our HRLB is able to identify critical cuts and solve the arising hierarchical relaxations of the CARP that integrate the even degree constraints and the “critical” capacity degree constraints.

3.1 The Chinese postman lower bound

A first lower bound CPPLB for the CARP is obtained by solving the undirected CPP. This bound consists of the length \( c(R) \) of the productive edges and the length \( c_{\text{Euler}} \) of those edges that need to be added to make the graph Eulerian, i.e. to fulfill the even degree constraints. To exemplify let us consider the graph \( G = (V,E = R,c = d) \).
on the left hand side of Fig. 1. All edges are productive, edge weights denote cost values $c_{ij}$ which are assumed to equal the demands $d_{ij}$. Nodes 3, 4, 5 and 6 have odd degree. We construct a CPP-matching-graph with node set $V_U = \{3, 4, 5, 6\}$ and shortest distances $s_{ij}$ as weights between nodes $i$ and $j$. An optimal solution chooses edges $[3,4]$ and $[5,6]$ with a weight of $s_{34} + s_{56} = 22$ (given in bold on the right hand side of Fig. 1). CPPLB = $c(R) + c_{Euler} = 71 + 22 = 93$ is derived.

![Figure 1. Graph G with its matching graph](image)

3.2 The matching lower bound

Christofides [7] proposed to make all node degrees even by solving the CPP matching problem and then adding as many cheapest edges incident to the depot as necessary to ensure that each vehicle leaves and enters the depot in order to fulfill the capacity degree constraint for the depot. However, these two ideas may not be applied independently. This is sketched in the example in Fig. 2. On the left hand side the odd nodes 1 and 2 are connected with an additional edge at cost 4. Assume that two vehicles leaving and entering the depot and thus two additional edges in the central cut (the cut through all edges incident with the depot 0) are needed. Adding two copies of the cheapest edge $[0, 1]$ leaving the depot raises the cost by 4 leading to overall additional costs of 8 (Fig. 2, middle). However, a feasible lower bound has total additional cost of 5. It is obtained by substituting $[1, 2]$ (that had been added to make the odd nodes 1 and 2 even) by the two edges $[1, 0]$ and $[0, 2]$ (Fig. 2, right).

MLB is based on these ideas [12]. The bound aims at enlarging the given graph such that the depot has a degree of $2M$ and all other nodes have an even degree without neglecting the above mentioned interdependencies.

As the CPP-matching-graph, the MLB-matching-graph includes the nodes having odd degree. Furthermore, it contains copies of the depot $v_{depot}$ and its closest node $v_{min}$. To achieve the required number $p = 2 \cdot M - g(v_{depot})$ additional central edges, we need $p$ copies of the depot and of $v_{min}$, if the depot has even degree, and $p - 1$ such copies, if the depot has odd degree (because we already have one copy of the depot among the odd nodes).

The depot copies now can be connected either to odd nodes by means of shortest paths or to the closest node $v_{min}$ with cost of $c_{min} = c(v_{depot}, v_{min})$ but not to each other (neither to the depot copy among the odd nodes, if it exists). Copies $v_{min}$ can be matched with each other with cost 0 (which is necessary in cases where some central edges are added by matching some depot copies with odd nodes).

For our example in Fig. 1 above we have $c(R) = d(R) = 71$. Assume a vehicle capacity of $Q = 20$. Then the number of routes is $M = [d(R)/Q] = 4$. That is, $p = 2 \cdot M - g(v_{depot} = 0) = 6$ central edges need to be added. The MLB-matching-graph in Fig. 3 includes the nodes with odd degree (nodes 3, 4, 5 and 6). The depot has an even degree, i.e., $p = 6$ copies of the depot (each one shown as a bullet in a box marked with “depot 0”) are added together with shortest distance edges to all nodes with odd degree in the original graph (the numbers $s_{01} = 7, s_{05} = 10$ and $s_{06} = 30$).

The depot node copies are not connected with each other but with the corresponding number of copies of their closest node $v_{min}$ (shown as a bullet in a box marked with “node $v_{min}$”) in the central box. In our example $v_{min} = 2$ with the cheapest central edge $[0, v_{min}]$ having a cost value of $c_{min} = s_{02} = 2$. Note, that in Fig. 3 for each edge leading to the border of the depot box or the $v_{min}$ box an identical edge is leading to each of the node copies, e.g., $s_{62} = 3$.

![Figure 2. Interdependencies between adding matching edges and edges in the cut (added edges are shown with broken lines)](image)

3.3 The hierarchical relaxations lower bound

HRLB is a new lower bounding procedure for the CARP which iteratively builds and solves tightening relaxations of...
Capacity degree constraints CDCs: For any noderset $W$ with $v_{\text{terpol}} \in W$ define $S(W)$ as the set of productive edges $e = [i,j]$ with $i \in W$ and $j \notin W$. $S(W)$ is denoted as cutset or cut and $W$ as central set. The number of productive edges within $S(W)$ is equal to $|S(W)|$.

Due to restricted vehicle capacity $Q$, a minimum number of vehicles leaving and entering the central set $W$ is calculated as $M(W) = [(d(R) - d(W))/Q]$ with $d(W)$ being the total demand of edges $e \in W \times W$ inside the central set $W$; $d(R) - d(W)$ is the total demand of edges outside the central set $W$. Thus, for any cutset the smallest number of edges (two for each vehicle) necessary to satisfy the demand outside of $W$ is $2 \cdot M(W)$. If $p(W) = 2 \cdot M(W) - |S(W)| > 0$, then we are $p(W)$ edges short within $S(W)$. In that case, call $p(W)$ the number of missing edges. With integer variables $y_e$ as defined above, the capacity degree constraint with respect to $W$ is as follows:

$$\text{CDC}_W : \sum_{e \in E, e \in S(W)} y_e \geq p(W) = 2 \cdot M(W) - |S(W)|$$

Critical cuts: For a given central set $W$ with $p(W) \leq 0$ $\text{CDC}_W$ is redundant and need not be considered. For $p(W) = 1$ the constraint is redundant, too, due to the even degree constraints. Therefore, we define $W$ as critical (or critical set) and $S(W)$ as critical (or critical cut) whenever $p(W) \geq 2$. For HRLB we only consider critical cuts.

The HRLB procedure: The HRLB procedure generates and iteratively solves a hierarchy of relaxations $R_1, R_2, \ldots, R_i$ etc. The first relaxation $R_1$ is similar to the CPP with the even degree constraints and considers a single capacity degree constraint for the central noderset $W_1 = \{v_{\text{terpol}}\}$. The second relaxation $R_2$ considers two capacity degree constraints, one with respect to $W_1$ and a second with respect to a central noderset $W_2$. Relaxation $R_i$ considers all constraints from relaxation $R_{i-1}$ while adding one more capacity degree constraint.

Fig. 4 shows the development of the HRLB for our example from Fig. 1 above. The first cut $S_1 = S(W_1)$ separates $W_1 = \{v_{\text{terpol}} = 0\}$. Subsequently, we consider cuts $S_2$ up to $S_5$ with the central nodsets $W_2 = \{0, 2\}, W_3 = \{0, 2, 1\}, W_4 = \{0, 2, 1, 3, 4\}$ and $W_5 = \{0, 2, 1, 3, 4, 5\}$.

The solution process for $R_i$ has two phases, the capacity degree constraint phase (C-phase for short) and the matching phase (M-phase for short). Within the C-phase we add minimum cost edges satisfying the most recent capacity degree constraint $\text{CDC}_i$ while taking the capacity degree constraints of all previous relaxations into account (leading to overall additional capacity costs called $cc(i)$). The respective minimum cost value is the shadow price for capacity degree constraint $\text{CDC}_i$. To also satisfy the even degree constraints the M-phase now solves a special CPP, the opportunity cost CPP (\(\delta\)-CPP). The opportunity costs are based...
on the shadow prices of the C-phase and reflect the fact that previously added edges might need to be replaced by others. This is explained for our small example (see Fig. 4).

In the first iteration we consider $S_1$ where six edges need to be added. From edges $[0,1]$ and $[0,2]$ the minimum cost one is chosen six times with overall cost of 12. (For reason of clarity we leave the consideration of the M-phase for a later paragraph.) In the second iteration we have $S_2$ where four edges need to be added. $S_2$ consists of edges $[0,1], [1,2], [2,4]$ and $[2,5]$ with $[1,2]$ being the minimum cost (3) edge. If we add this edge for four times we would have an overall additional cost of 24. However, if we add four copies of $[0,1]$ (edge cost 4), then we can remove four of those copies of $[0,2]$ (edge cost 2) added in the first iteration leading to an overall additional cost of 20. This is possible because edge $[0,1]$ is part of both cuts $S_1$ and $S_2$.

To identify capacity degree edges that have to be replaced lateron by other capacity or by even degree edges we define opportunity cost values $\hat{c}_e$ for edges in critical cuts. These are modified or reduced cost values $\hat{c}$ taking into account prospective shadow prices for reducing the original cost values of edges within the cut. That is, based on the considered capacity degree constraints, we may add a (dual) variable $sp(W)$ to any central set $W$. In our small example the shadow price for the first CDC is $sp_1 = \alpha_{22} = 2$. Computing opportunity costs $\hat{c}$ for the edges $e \in S_1$, we get $\hat{c}_{22} = 2 - 2 = 0$ and $\hat{c}_{01} = 4 - 2 = 2$. Now, in the second iteration, edge $[0,1]$ that leads to the minimum cost completion in the subsequent cut $S_2$ as shown above can easily be identified as the minimum opportunity cost edge in cut $S_2$.

In the same way cost reduced edges may be used for matching the odd nodes. Thus, the M-phase uses reduced cost values $\hat{c}$ calculated from the dual variables $sp(W)$ of the C-phase in a modified C-CPP. Edges in this problem consist of paths in the original graph with cost values modified according to the reduced cost values. Again, any time an edge is used that includes a cost reduction the capacity degree edge corresponding to this cost reduction is removed from the solution. Fig. 5 (left) shows our small example after the C-phase of the third iteration: The graph shows the edges $2 \times [0,2], 4 \times [0,1]$ and $4 \times [1,3]$ added according to cuts $S_1, S_2$ and $S_3$ as broken lines as well as the modified cost values according to the performed iterations (indicated as numbers above the struck out cost values). The graph has odd degree nodes 3, 4, 5 and 6. The right hand side of Fig. 5 shows the matching graph based on the reduced cost values (and the paths corresponding to given matching costs).

The optimal matching with respect to the reduced costs consists of matching edges $[3,4]$ and $[5,6]$ corresponding to paths $[3,1,0,2,4]$ and $[5,6]$. Performing edge corrections according to the included cost reductions leads to additional even degree edges giving the combined solution $3 \times [0,2], 3 \times [0,1], 3 \times [1,3], 1 \times [2,4]$ and $1 \times [5,6]$ as an optimal solution for relaxation $R_3$.

We now give a survey of the iterative HRLB scheme that integrates the introduced components and shows some additional features of the algorithm.

Initialize opportunity cost values $\hat{c}_e = c_e$ for $e \in E$, primal variables for capacity degree edges $y^C = 0$ and shadow prices $sp = 0$, iteration counter $i = 1$ and the first central set $W_1 = \{v_{ij} \in p\}$.

repeat Iteration $i$

Consider new capacity degree constraint

$CDC_i : \sum_{e \in e : e \in e(W)} y_e \geq p_e = p(W_i) - 2 \times M(W_i) - |S(W_i)|$

if (CDC is critical, i.e., $p_i \geq 2$) then

C-Phase: (capacity degree solution $y^C$ with capacity cost $c(e)$ and shadow prices $sp$)

a) Compute the minimum cost edge $e_{min}(i)$ in cut $S(W_i)$ (productive or non-productive, minimum with respect to $\hat{c}$)

b) Set shadow price $sp_{ij} = \hat{c}(e_{min}(i))$

$\hat{c}(e_{min}(i))$ might contain cost reductions $sp_{ij}, j = 1, \ldots, i - 1$. At this cost, $e_{min}(i)$ is available only $r(e_{min}(i)) = \min \{p_{ij} | sp_{ij} \}$ is included in $\hat{c}(e_{min}(i))$ times.

c) Check availability

if ($e_{min}(i)$ is not available in the required number $p_i$ then

Rearrange the opportunity cost structure

repeat

$e_{min}(i)$ is not available in the required number $p_i$.

Set one of the restricting $sp_{ij} = 0$.

Remove the respective capacity degree edges of iteration $j$.

Compute a new minimum cost edge $e_{min}(i)$ in $S(W_i)$.

until a new $e_{min}(i)$ with $r(e_{min}(i)) \geq p_i$ is found.

Recompute minimum cost edges and shadow prices in the concerned cuts $S(W_i)$.

endif

Figure 4. W -cuts of the HRLB

Figure 5. Resulting graph after the C-Phase of the third iteration and the matching-graph
d) Update capacity degree edges
Add \(p_\ell\) edges \(e_{\min}(\ell)\) to the capacity degree solution \(y^C\).
Observe that for each cost reduction \(sp_\ell\) included in minimum cost edges \(e_{\min}(\ell)\) respective edge corrections are executed.

e) Compute the capacity cost \(ce(i)\) as the cost sum of the primal capacity degree solution (edge corrections included) or as the opportunity cost sum of the minimum cost edges (edge corrections not included) corresponding to the dual solution:

\[ ce(i) = \sum_{e \in H} c_e \cdot y_e^C = \sum_{i=1}^k p_i \cdot e_{\min}(j) + \sum_{j=1}^i p_j \cdot sp_j \]

M-Phase: even degree solution
a) Solve \(\ell\)-CPP: based on actual shadow prices \(sp_1, \ldots, sp_k\) with matching cost \(mc(i)\), matching edges (including cost reductions).
b) Derive the even degree solution as matching edges plus edge corrections.

Combined solution: \(y = y^C + y^M\) (Note that edge corrections are included.)

New lower bound: \(LB_1 = ce(i) + mc(i)\)

Extension of the central set
Extend \(W_i\) by the endnode(s) \(v_{\min}\) of the minimum cost edge(s) \(e_{\min}(i) : W_{i+1} = W_i \cup v_{\min}\)
else
 Extend \(W_i\) by an arbitrary node \(v_i : W_{i+1} = W_i \cup v_i\)
endif until \(W_{i+1} = V\):
result: \(HRLB = c(R) + \max\{LB_1, LB_2, \ldots, LB_i\}\) is a valid lower bound for the CARP.

### 3.4 Mathematical programming formulations

While usually the LP-relaxation of a mathematical programming formulation is considered and valid inequalities are successively added we proceed as follows. In the initial lower bound CPPLB the capacity degree constraints are relaxed and then some of them are added as needed.

To provide a formal mathematical formulation for relaxation \(R_i\) we use the notation introduced above. In addition \(W_i\) denotes the set of all central sets for which a capacity degree constraint is considered in \(R_i\). These models may be used to explain in detail the investigation performed in the previous section. By means of detailed examples given in the appendix we provide examples for the successive evaluation of corresponding constraints and central cuts.

#### Relaxation \(R_i\) (primal)

Minimize

\[ \sum_{e \in E} c_e \cdot y_e \]

subject to

\[ \sum_{e \in E, e \in S(U)} y_e \geq 1 \quad \forall U \in \mathcal{U} \]

\[ \sum_{e \in E, e \in S(W)} y_e \geq p(W) \quad \forall W \in W_i \]

\[ y_e \geq 0 \quad \text{and integer} \quad \forall e \in E \]

Dualizing the linear relaxation of this model using variables \(u(U)\) and \(sp(W)\) leads to the following model:

Maximize

\[ \sum_{U \in \mathcal{U}} u(U) + \sum_{W \in \mathcal{W}_i} sp(W) \]

subject to

\[ \sum_{U \in \mathcal{U}, e \in S(U)} u(U) + \sum_{W \in \mathcal{W}_i, e \in S(W)} sp(W) \leq c_e \quad \forall e \in E \]

\[ u(U), sp(W) \geq 0 \quad \forall U \in \mathcal{U} \text{ and } W \in \mathcal{W}_i \]

#### 4 Numerical results

Current state of the art with respect to the CARP reveals that even small sized benchmark problem instances are currently not fully accessible with respect to finding optimal solutions; see, e.g., [13]. In Table 1 we consider the well-known problem instances from de Armon [8] as they have been intensively studied within the literature. Most of these problem instances had been solved to optimality by various algorithms throughout the 1990s. The instances we consider in Table 1 are those that have not yet been solved to optimality (proven) or been reported to be solved only in [13]. The columns of the table provide the problem instance number, the number of nodes and edges as well as the lower bounds MLB, ND2, ND3 and HRLB. In column LB we provide those bounds reported in [13] although their detailed working has not been verified.\(^2\)

Our bound HRLB is able to achieve all best known lower bounds indicating that it is competitive with respect to solution quality. In one case we are able to improve on the best known lower bound. Some detailed analysis is given in the appendix. Besides the small example used to exemplify concepts throughout the paper the appendix includes two real-world instances that we have already used in an earlier study (cf. [1, 10]) as well as some of the instances considered in Table 1. The real-world instances reveal the fact that we can easily adapt our approach to the \(m\)-CARP incorporating different capacities and depots.

| No | \(|V|\) | \(|E|\) | MLB | ND2 | ND3 | HRLB | LB | UB |
|----|-----|-----|-----|-----|-----|------|----|----|
| 1  | 12  | 22  | 530 | 310 | 316 | 316  | 316| 316|
| 4  | 11  | 19  | 274 | 274 | 287 | 287  | 287| 287|
| 5  | 13  | 26  | 370 | 376 | 376 | 377  | 377| 377|
| 6  | 12  | 22  | 295 | 295 | 298 | 298  | 298| 298|
| 7  | 12  | 22  | 312 | 312 | 325 | 325  | 325| 325|
| 10 | 27  | 46  | 260 | 330 | 334 | 344  | 344| 348|
| 11 | 27  | 51  | 259 | 277 | 285 | 303  | 303| 311|
| 14 | 13  | 23  | 424 | 428 | 428 | 450  | 448| 458|
| 15 | 10  | 28  | 536 | 536 | 536 | 536  | 536| 544|

**Table 1. Lower bounds for the de Armon-data**

#### 5 Conclusions and future research

In this paper we have considered the capacitated arc routing problem, an important problem with respect to

\(^2\)It should be noted that the purpose of [13] is mainly in providing a tabu search procedure which is performed in a concise way
real-world routing problems in various application settings. Based on the idea that each cut around a depot needs to be traversed a certain number of times (to be determined along logical tests) we have developed the concept of hierarchically expanded lower bounds. Results on benchmark problem instances from the literature show that our results are in line or even better than the best existing lower bounds.

Future research has to deal with an efficient implementation of the hierarchical relaxations lower bound which includes an automatic separation of capacity degree constraints belonging to critical cuts. Furthermore, it should be embedded into an exact branch and bound or branch and cut routine. Regarding lower bounding schemes we may extend our research to provide even more improved schemes. Based on preliminary ideas one may try to find feasible solutions based on our hierarchical relaxations lower bound as we have shown for some problem instances where we were able to construct optimal solutions based on the bound.

It may be shown that, if it is not possible to construct a feasible solution based on a certain bound that no such solution exists and the lower bound may be increased. For instance, we conjecture that the lower bound of 450 for de Armon 14 needs to be increased by at least 1 unit.

Acknowledgements: The helpful comments of the referees are greatly appreciated.

References


A Appendix

In this appendix we provide a detailed analysis and calculation with respect to some CARP instances. Besides the small example used to exemplify concepts throughout the paper these include two real-world instances that we have already used in an earlier study (cf. [1] and [10]) and two of the well-known de Armon [8] benchmark instances.

For each instance we give a table with critical cuts $S_i$, number $p_i$ of edges missing in $S_i$, the respective capacity degree constraints CDC$_i$, shadow prices $sp_i$ (dual variables corresponding to the CDCs – so-called dual capacity degree solution) and capacity degree edges (primal variables satisfying the CDCs but not the even degree constraints – so-called primal capacity degree solution). If a shadow price $sp_i$ contains cost reductions from shadow prices of other iterations, the associated correction of the respective capacity degree edge is given in brackets.

The dual capacity degree solution is given by the shadow prices. Its objective function value is the sum of products of shadow price times number of missing edges for all generated cut constraints. The primal capacity degree solution is given by $p_i$ times the respective capacity degree edge (including corrections). Its objective function value is the sum of products of $p_i$ times the cost value of the capacity degree edge of $S_i$ (including corrections) for all generated cut constraints.
Using the shadow prices we compute reduced opportunity costs for the involved edges (i.e., for each edge that is part of at least one of the critical cuts with strictly positive shadow price) yielding an opportunity cost CPP. The minimum cost matching of this CPP gives the (primal) even degree solution, if for each cost reduction used by the matching edges respective edge corrections are executed.

A.1 A small example

Our small example is a general weight problem. The total cost of the required edges is 71, the CPPLB is 93 = 71 + 22 from non-productive matching paths (3,4) and (5,6). Node 0 is the depot. Edge demand equals edge cost. Thus, total demand is 71. Vehicle capacity is 20. Thus, at least \( M = 4 \) routes are required. The MLB is 105, ND2\(=\)ND3=117, the HRLB is 129 (58 for non-productive edges). Fig. 6 shows the graph with subsequent cuts \( S_1, \ldots, S_8 \). Capacity degree constraints generated for the cuts \( S_1, \ldots, S_4 \) are shown below.\(^3\) Cut \( S_5 \) is not critical.

The objective function value of the dual capacity degree solution given by the shadow prices \( sp_i \) is \( \sum_i p_i \cdot sp_i = 38 \). The primal capacity degree solution is \( 6 \times \{0, 2\}, 4 \times \{0, 1\}, \ldots \) and has total opportunity cost of 20. The primal even degree solution includes these matching edges plus edge corrections (given in brackets) for each cost reduction used in the matching edges, i.e., edges \( [3,1] = [1,3], [1,0] = [1,0] \), etc., and has total opportunity cost of 20. The opportunity cost CPP consists of the edges [3,1], [1,0], [0,2], [2,4], [5,6] and has total opportunity cost of 20. The HRLB is 93 \( = \) 71 + 22. Thus, the combined solution consists of the edges [3,1], [1,0], [0,2], [2,4], [5,6] and has total opportunity cost of 20. The HRLB for our example is 58 = 38 + 20 for non-productive edges and 129 = 71 + 58 in total. One may easily verify that a feasible solution with the same objective exists, i.e., the bound is optimal.

\(^3\)Corrections refer to eliminating previously added edges (indicated by \(-\)) and to adding new edges (indicated by \(+\)).

A.2 Königstein

The first real world problem is dealing with planning routes for winter gritting in the area of Königstein.\(^4\) It is a general weight problem with 65 nodes and 94 productive edges. There also exist some non-required, so-called short cut edges that need not be served. Thus, they are not counted when computing the number of edges already available for crossing a cut, but they can be used as capacity degree edges or for matching the nodes with odd degree. The total length (cost) of the required edges is 202 km, the CPPLB is 261.5 km = 202 + 30.5.

Edge demand is proportional to edge length (cost) and also given in km. Thus, total demand is 202 km. There are six vehicles, each stationed in the depot node. The capacities are 65 km (vehicle 1), 50 km (vehicles 2 and 3), 25 km (vehicles 4 and 5) and 15 km (vehicle 6). Two side constraints have to be relaxed when computing the lower bound: Vehicle 1 has to pass a refilling station and one of the edges can only be served by the smaller vehicles 4, 5 or 6. With respect to the differing capacities, we have to extend our ideas to the \( m \)-CARP. For the computation of the capacity degree constraints we need to know the maximum demand that can be satisfied by 1, 2, \ldots, 6 vehicles. Of course, we suggest to use the vehicles in decreasing order of capacity. The total capacity of the five biggest vehicles 1 to 5 is sufficient to satisfy total demand.

MLB=ND2=ND3=270.5 km (68.5 km for non-productive edges), HRLB=273.5 km (71.5 km for non-productive edges). Capacity degree constraints were generated for the cuts \( S_1, S_2 \) and \( S_4 \).

Combining the capacity and the even degree solution while making appropriate modifications each time a cost reduced edge is used, we obtain a solution that fulfills capacity and even degree constraints and has minimal cost. With

\(^4\)Figures showing the data of the instances Königstein – and Wennigsen of the next section – are available from the authors upon request.
these edges, a feasible and hence optimal solution of the CARP with five routes can be constructed without adding further non-productive edges. (The relaxed side constraints can be fulfilled, too!) Having 71.5 km of non-productive edges, it improves over the previously best known solution having 79.5 km of non-productive edges.

A.3 Wennigen

The second real world problem is dealing with planning routes for winter gritting in the area of Wennigen. It is a general weight problem with 41 nodes and 55 productive edges. The total length (cost) of the required edges is 229.4 kilometer (km), the CPPLB is 294.0 km. Edge demand is proportional to edge length (cost). There are six vehicles. Five of them are stationed in the depot node, each having a capacity of 45, one vehicle is stationed in a second depot node and has a capacity of 30. To satisfy total demand six vehicles have to be used. To compute the minimum demand that has to be satisfied by routes starting from the first depot node, we subtract the capacity of the sixth vehicle (starting from the second depot) from total demand getting a demand value of 199.4. Given this, we have MLB = 297.0 (67.6 for non-productive edges), ND2 = 310.2 (80.8), ND3 = 330.8 (91.4) and HRLB = 325.8 (96.4).

Similarly as for the previous instance we can construct a feasible and hence optimal solution of the CARP based on the HRLB calculations. With 96.4 km non-productive edges, this solution is a significant improvement of the best known solution having 102.4 km non-productive edges.

A.4 De Armon

Instance de Armon 5 is a unit weight problem with 13 nodes and 26 edges. The total cost of the required edges is 316, the CPPLB is 346 = 316 + 30 from the unproductive matching paths (8,10,9), (5,6) and (1,12). Node 1 is the depot. Total demand is 26 (each edge has a demand of 1), vehicle capacity is 5. Thus, at least $M = 6$ routes are required. MLB=370, ND2=ND3=376, HRLB=377 (61 for non-productive edges). Fig. 7 shows the graph of de Armon 5 with subsequent cuts $S_1, \ldots, S_4$.

The dual capacity degree solution given by the shadow prices is $s_{p_1} = 4$, $s_{p_2} = 2$ and $s_{p_4} = 7$ with an objective function value 53. The primal capacity degree solution consists of $7 \times [1, 12], 2 \times \{(1, 2), (1, 12), (6, 5)\}, 3 \times [6, 5] = 5 \times [1, 12], 2 \times [1, 2], [6, 5]$ also with total cost of 53.

The minimum opportunity cost matching in the resulting $\epsilon$-CPP consists of the edges $[1, 2], (s_{p_1}, s_{p_2}, s_{p_4}), [2, 9], [6, 5]$, $[9, 6]$, $[12, 6]$, $[6, 7]$, $[7, 8]$ having total (opportunity) cost of 8 (included cost reductions in brackets). Thus, the even degree solution is $[1, 2], (s_{p_1}, s_{p_2}, s_{p_4}), [2, 9], [6, 5]$, $[9, 6], [12, 6], [1, 12], [6, 7], [7, 8]$ having total (opportunity) cost of 8 (included cost reductions in brackets).

Combining capacity and even degree solution including the corrections gives the minimum cost solution satisfying the three degree constraints and all even degree constraints. It consists of the following edges: $6 \times [1, 12], [1, 2], [2, 9], [6, 5], [12, 6], [6, 7], [7, 8]$. Thus, the HRLB for de Armon 5 is 61 for non-productive edges and 377 = 316 + 61 in total.

Note, that shadow price and capacity degree edge of $S_3$ has been changed during the computation process: First, when considering $S_1$ and then $S_2$, we had $s_{p_2} = 3 = c_{12,6}$ from (minimum cost) capacity degree edge $[12, 6]$ of $S_2$. Integrating $S_4$ where $s_{p_3} = 3$ additional edges are needed, with the given opportunity cost structure $(s_{p_1} = 4$ and $s_{p_2} = 3$) the minimum cost edge in $S_4$ is edge $[1, 2]$ with opportunity cost $6 = c_{1, 2} - s_{p_1} - s_{p_2}$. However, for this opportunity cost of 6, edge $[1, 2]$ can be used only twice because the included cost reduction (corresponding to removing edge $12, 6$ that had been added to satisfy CDC$_2$) is only available $p_2 = 2$ times. Thus, we have to change the opportunity cost structure: We temporarily set $s_{p_2} = 0$, remove the capacity degree edges of $S_2$, compute the new minimum cost edge of $S_4$ as edge $[6, 5]$ and $s_{p_4} = c_{6,5} = 7$ and then with the new opportunity costs (from $s_{p_1} = 4$ and $s_{p_4} = 7$) we recompute the minimum cost edge in $S_2$ as edge $[1, 2], [12, 6]$ and $s_{p_2} = 2 = c(1, 2) = c(1, 2) - s_{p_1} - s_{p_4} = c(1, 2) - (c(1, 2) - c(6, 5))$. Integrating CDC$_4$ leads to a reduction of shadow price $s_{p_2}$ of cut $S_2$ (because part of the cost for adding an edge in cut $S_2$ is “paid” when adding an edge in cut $S_4$) and changes the capacity degree edges of $S_2$. The main result is when
computing a capacity degree solution the capacity degree constraints CDC_i have to be considered in order of non-increasing number p_i of missing edges.

A.5 De Armon 14

De Armon 14 is a general weight problem with 13 nodes and 23 edges. The total length of the required edges is 336. The CPPLB is 364 = 336 + 48 from matching paths (5,1,3), (2,8) and (9,13). Node 1 is the depot. Total demand is 212, vehicle capacity is 35. Thus, at least M = 7 routes are needed. The MLB is 424 (88 for non-productive edges), ND2 and ND3 are 428 (92), and HRLB is 450 (114). With a modification the lower bound may even be increased above 450. The length of the best known feasible solution is 458 (cf. [13]). Fig. 8 gives the graph of de Armon 14 after iteration 2 of the HRLB. It shows node set S, the set of required edges e ∈ E as full lines, assigned with edge demand d_e and edge cost c_e (in some cases crossed out where cost reductions have been performed).

For the HRLB calculation for de Armon 14 we set the opportunity cost c_e = c_e for all edges e ∈ E and shadow prices s_p = 0. The following table gives a survey of the results. As a modification we do not consider critical cuts (even if two or more edges are missing) if matching the odd nodes would add a sufficient number of additional edges (as for S_4). Fig. 9 shows the instance after iteration 5.