ROUTING PROBLEMS

Ulrich Derigs
University of Cologne, Cologne, Germany

Keywords: arc routing, capacitated arc routing problem, Chinese postman problem, node routing, traveling salesman problem, vehicle routing

Contents

Summary

The area of routing is one of the most vital areas within Optimization and Operations Research. From a theoretical/mathematical programming point of view the concept of "routing" means basically to determine an optimal set of cycles within a graph or network. In practice routing problems arise in distribution and transportation planning; in service planning, as for instance waste collection; and in operations management or manufacturing, when optimal sequences in automatic drilling and plotting have to be determined. In this article we give an overview of the main developments in

deterministic routing. Here a distinction is made between node-oriented problems and arc-oriented problems. The term "arc routing" refers to problems where the service activity is associated with the arcs of the graph, while "node routing" refers to problems where the service is associated with the nodes of the graph. In this article we introduce standard models and outline related exact and "approximative" approaches for problems within both classes of routing.

1. Introduction

The area of routing is one of the most vital areas within Optimization and Operations Research. From a theoretical/mathematical programming point of view the concept of "routing" means basically to determine an optimal set of cycles within a graph or network. Here different configurations of the underlying graphical structure, different objectives as well as different constraints have lead to a great variety of challenging and well-studied standard optimization problems. Under practical aspects routing problems include transportation planning problems for the delivery of goods from a depot to a set of customer locations, like the delivery of newspapers or the scheduling of fuel deliveries, the routing of waste collection vehicles, and so on, as well as optimization problems in manufacturing, as for instance to determine optimal sequences in automatic drilling and plotting. The theories, models, and algorithms that have been developed are implemented on an operational level for daily use as well as on a strategic level when service systems - like communication or transportation networks - are designed.

The transportation sector has always been of significant economic importance. In every economy transportation accounts for a substantial percentage of the gross national product. Under the phenomenon of globalization, deregulation, and time-based competition the pressure for cost reduction and productivity improvement is ever-increasing and a must for private companies as well as for the economy as a whole. Under aspects of increasing consumption of natural resources and increasing pollution through transportation the problem of "optimal routing" gets an ecological dimension and objective.

Developments in this problem domain have always been among the major success stories of Optimization and Operations Research, and due to an increasing need for better decision support, routing has found a renewed interest in the last decade leading to significant progress in theory, models, algorithms, and routing software.

From a mathematical as well as from a practical point of view, we can distinguish two classes of routing problems. The term "arc routing" refers to problems where the service activity is associated with the arcs of the graph - as for instance when modeling the snowplowing activities in street networks. On the other hand the term "node routing" refers to problems where the service is associated with the nodes of the graph - as for instance when modeling the daily delivery of newspapers to newspaper stands. Here the arcs deliver the basis for paths connecting the locations to be served. Note that the distinction between these concepts is a matter of modeling and not an objective attribute of the planning situation. Within optimization these two models are studied separately due to their intrinsic combinatorial properties and we follow this separation here. Within the literature as well as in practice, node routing has always played a dominant role, and it is only recently that arc routing has found similar attention.

In the following we briefly introduce the mathematical notation used in this article. A *graph* $G = (V, E \cup A)$ is specified by a *node* set $V = \{1,...,n\}$, undirected *edges* E, and directed *arcs* A. G is called *directed* if E is empty, *undirected* if A is empty, and *mixed* if both E and A are nonempty. In a directed graph, arc (i, j) is directed from node i to node j. The undirected edge e between nodes i and j is denoted by $e = e(i, j)$ without regard to the order of i and j. We assume that a cost $c_{i,j} \geq 0$ is associated with all edges $e(i, j) \in E$ and arcs $(i, j) \in A$. This definition does not allow for multiple edges/arcs. A graphical structure with several edges/arcs between nodes is called a *multigraph*. An undirected graph G is called a *complete* graph if every pair of nodes is met by an edge in E.

Given a node i of a mixed graph $G = (V, E \cup A)$ let $\delta^+(i) = \{ j \in V \mid (i, j) \in A\}$ and $\delta^-(i) = \{ j \in V \mid (j,i) \in A\}$ and define $d^+(i) = |\delta^+(i)|$ the *out-degree* of node i, and $d^-(i) = |\delta^-(i)|$ the *in-degree* of node i in G, respectively. Similarly, let $\delta(i) = \{ j \in V \mid e(i, j) \in E\}$ and $d(i) = |\delta(i)|$ the *degree* of node i in G. For a subset S $\subseteq$ V we define $\gamma(S)$ as the set of edges/arcs in G with both end nodes in S and $\delta(S)$ as the set of edges/arcs in G with exactly one end node in S. For a subset $F \subseteq E$ we define $c(F) := \sum_{e \in F} c_e$.

A *path* in a graph $G = (V, E \cup A)$ is a sequence $(i_1, e_1, i_2, e_2, \ldots, e_p, i_{p+1})$ of nodes and edges/arcs such that e_k is either an arc directed from i_k to i_{k+1} or an edge joining these nodes. Two distinct nodes i and j of a graph G are said to be *connected* if there exists a path in G from i to j. A graph G is called a *connected graph* if all nodes are connected. Note that in the sequel of this article on routing problems we assume all graphs to be connected, since for unconnected graphs the routing problem decomposes into smaller subproblems. A *spanning tree S* in G is a subset of the edges such that $|S| = |V| - 1$ and S does not contain a cycle (subtour) in G. From these properties it follows that the graph which is induced by S is connected and spans the set V.

A *tour*, or a *cycle*, is a path such that $i_1 = i_{p+1}$. This definition allows for multiple incidences of the same edge in a tour. A tour is called a *Hamiltonian tour* or *a traveling salesman* tour if it contains every node exactly once. It is called an *Eulerian tour* or an *Euler tour* if it contains every edge of G exactly once, and a *postman tour* if each edge occurs at least once in the tour. A graph allowing an Euler tour is called an *Eulerian graph* or an *Euler graph*. The *length* of a path or tour T is defined as the sum of the cost of the arcs and edges contained in T.

The organization of this article is as follows. In Sections 2 and 3 we introduce the two fundamental routing problems: the Chinese postman problem and the traveling salesman problem. These problems represent already the essentials of the two different classes (i.e., they contain the core structures, motivate general algorithmic ideas, etc.) There is one striking difference between these standard problems: while the Chinese postman problem is efficiently solvable, the traveling salesman problem is *NP*-hard. This difference vanishes for the general and more complicated arc routing problems and node routing problems, which are *NP*-hard in general. In Sections 4 and 5 we then discuss two more general routing problems: the vehicle routing problem and the capacitated arc routing problem.

2. The Chinese Postman Problem

The Chinese postman problem (CPP) is the fundamental arc routing problem. It is to find the minimum length postman tour in an undirected graph G. This problem was first solved in 1962 by Mei Ko Kwan, a Chinese mathematician. He considered this problem on the practical background of a postman delivering the daily mail for a certain district of streets - hence, the problem is referred to as the "Chinese" postman problem. In its basic form CPP has been defined on an undirected graph and in the following we refer to this problem as undirected postman problem (UPP). Later this problem was extended to directed and mixed graphs. Edmonds and Johnson - who related this problem to matching theory - published the first polynomial algorithm for UPP in 1973.

2.1. The Undirected Postman Problem

It is obvious that whenever there is an Eulerian tour in the graph, then this tour solves UPP. A necessary and sufficient condition for the existence of such a tour was already given in 1736 by Leonhard Euler: "For a graph to be Eulerian each node must be incident to an even number of edges." It follows that if this condition is fulfilled, UPP is reduced to the problem of identifying and tracing an Eulerian tour. If the original graph does not satisfy this condition, some edges have to be traversed more than once by the postman; in other words, the graph has to be balanced (i.e., copies of certain edges have to be added). This leads to the formulation of the following mathematical program.

Given a postman tour every edge of G is traversed at least once. So let $1 + t_e$ be the number of times that edge e is contained in the tour. Now we construct a (multi)graph G' from G introducing t_e additional copies of edge e into G. Then the postman tour in G becomes an Eulerian tour in G'. Thus UPP can be reformulated as an "augmentation problem," (i.e., the problem of determining the Eulerian supergraph of minimum cost) and this can be formulated as a mathematical program in the following way:

***Undirected postman problem* (*UPP*)**

$$\text{minimize} \quad \sum_{e \in E} t_e \cdot c_e \qquad \text{subject to} \tag{1}$$

$$\sum_{e \in \delta(v)} (t_e + 1) \equiv 0 \mod 2 \quad \text{for } v \in V \tag{2}$$

$$t_e \geq 0, \text{ integer for } e \in E, \tag{3}$$

that is, find values t_e for $e \in E$ such that after adding t_e copies of e, G is balanced (i.e., every node has even degree) and such that the sum of the cost of the additional copies is minimized.

Now if node v has an odd degree in G then an odd number of incident edges has to be added such that v gets even degree in G'. If node v is an even degree node in G, then an even number of incident edges or no incident edge has to be added, respectively. Therefore, the process of duplicating edges leads to a collection of paths starting and ending at odd degree vertices, and one has to decide which pairs of odd degree nodes are to be joined together by a path of duplicated edges. Note that in an undirected graph there is always an even number of nodes with odd degree.

The optimal augmentation (i.e., the optimal set of paths) can be determined by solving a perfect matching problem on a related auxiliary graph. Here a *perfect matching* in a graph is a subset M of edges such that every node is met by exactly one edge from M. This leads to the following procedure:

Undirected postman algorithm

1. Determine for every pair $i, j \in V$ of odd degree nodes in G the shortest path $P_{i,j}$ joining these two nodes and define $d_{i,j}$ to be the length of path $P_{i,j}$.
2. Construct the complete graph $G = (\bar{V}, \bar{E})$ where $\bar{V}$ denotes the set of all odd degree nodes in G and associate with the edge joining i and j in $\bar{V}$ the cost $d_{i,j}$.
3. Determine a perfect matching M of minimum cost in $\bar{G}$.
4. Duplicate in G the edges of those paths $P_{i,j}$ that correspond to the edges in the optimal perfect matching M to obtain an Eulerian (multi)graph G'.
5. Identify and trace an Eulerian tour in G'.

Steps 1 - 4 of the UPP algorithm can be implemented using a labeling technique that enables us to solve the shortest path and the matching problem simultaneously. Consequently, this labeling technique combines elements from shortest path computation and elements from matching algorithms. Then Step 5 requires one to construct a traversal in G'. This is a rather simple problem that can be solved by a straightforward list procedure.

2.2. The Directed Postman Problem

In the undirected postman problem the key idea is to add edges such that every node gets an even degree since when traversing a postman tour, one must be able to leave every node that the tour visits. In a directed graph, we have to obey the direction of arcs, and a necessary and sufficient condition for a directed graph to allow an Eulerian tour is that the number of arcs leading into a given node must equal the number of arcs directed out of that node. Again, if the original graph does not satisfy this condition, the graph has to be augmented (i.e., copies of certain arcs have to be added). Here, an optimal augmentation can be determined by solving a network flow problem on an auxiliary graph.

For that purpose we compute the so-called imbalance $b(i) = d^-(i) - d^+(i)$ for all nodes i in V and define decision variables $x_{i,j}$ as the number of copies of arc (i,j) to be added to

G. Then the optimal set of arcs to be added to make G an Eulerian graph can be determined by solving the following network flow problem:

$$\text{mimimize} \quad \sum_{(i,j)\in A} c_{i,j} x_{i,j} \quad \text{subject to} \tag{4}$$

$$\sum_{j\in\delta^+(i)} x_{i,j} - \sum_{j\in\delta^-(i)} x_{j,i} = b(i) \quad \text{for } i \in V \tag{5}$$

$$x_{i,j} \geq 0 \quad \text{and integer} \qquad \text{for } (i,j) \in A. \tag{6}$$

The optimal flow (i.e., the set of arcs with $x_{i,j} > 0$) can be decomposed into a collection of paths from nodes with $b(i) > 0$ to nodes with $b(i) < 0$, and these arcs form the optimal augmentation. Introducing copies of the arcs of these paths, G is balanced with minimal cost.

The network flow problem can be transformed into an equivalent classical transportation problem. For that purpose let $I = \{i \in V \mid b(i) > 0\}$ and $J = \{j \in V \mid b(j) < 0\}$ and determine for each pair (i,j) with $i \in I$ and $j \in J$ the length $d_{i,j}$ of the shortest path connecting *i* and *j*. Now, using these cost-values, solve the transportation problem with the set *I* as supply nodes with supply $b(i)$ and the nodes in *J* as demand nodes with demand $-b(i)$. For a discussion of the network flow problem and the classical transportation problem (see *Fundamentals of Optimization and Operations Research* and *Graph and Network Optimization*).

2.3. The Mixed Postman Problem

As long ago as 1962, Ford and Fulkerson proposed necessary and sufficient conditions for a mixed graph to be Eulerian. Here, every node has to be incident with an even number of arcs and edges, and for every nonempty subset *S* of *V*, the difference between the number of arcs from *S* to *V\S* and the number of arcs from *V\S* to *S* must be less than or equal to the number of edges between *S* and *V\S*. Yet verifying these conditions or solving the augmentation problem to balance the mixed graph appropriately is *NP*-hard. The augmentation or balancing problem can again be formulated as an integer programming problem in which the variables represent the number of copies of each arc or edge that must be added to make the graph Eulerian. These programs may then be solved by “branch & bound” or “branch & cut” approaches. Once the Eulerian supergraph has been determined the mixed graph can be transformed into an equivalent Eulerian directed graph and then any algorithm for the directed case can be applied to identify the postman tour.

Due to the intrinsic complexity of the postman problem in mixed graphs, several heuristics have been developed, with the best-known heuristics having a worst-case ratio of 2. (For a discussion of performance criteria for heuristics, see *Approximative Algorithms*.)

2.4. Some Variants of the Classical Postman Problem

Quite a number of variants of the classical postman problem/model have been proposed in literature. The "windy postman problem" consists of determining a least-cost postman tour of an undirected graph $G = (V,E)$ with the cost of traversing an edge depending on the direction of the traversal (i.e., travel with or against the wind). Except for some cases which can be solved in polynomial time this problem is again *NP*-hard. In the "rural postman problem" we consider a graph $G = (V, E \cup A)$ and a set $R \subseteq E \cup A$ of so called required arcs and edges. Then we have to determine a traversal of minimal length that covers all the edges and arcs in R. In general the rural postman problem is *NP*-hard for undirected and directed as well as for mixed graphs.

3. The Traveling Salesman Problem

The traveling salesman problem (TSP) is one of the most celebrated problems in Operations Research in general and in combinatorial optimization, specifically. Here TSP, because of its simple conception on one side and its intrinsic algorithmic complexity on the other hand, plays the role of a standard reference: All major algorithmic techniques in combinatorial optimization, like polyhedral combinatorics, enumeration and relaxation techniques, and heuristic schemata, have been developed with the TSP as one of the motivating benchmark problems. The literature related to TSP is enormous, and development in the TSP domain mirrors the progress of combinatorial optimization in general. TSP has practical relevance as such, and occurs as a subproblem in several more complex routing problems. Except for a few special cases that are more or less academic, traveling salesman problems are hard. Note that the problem to determine whether a graph contains a Hamiltonian tour is *NP*-hard already.

The generic traveling salesman problem is defined on a directed complete graph $G = (V,A)$ and with every arc (i,j) is associated a nonnegative cost $c_{i,j}$ representing a distance, a travel cost, or a travel time. Then we want to find a tour of minimal cost that meets every node exactly once. If $c_{i,j} = c_{j,i}$ for all $i, j \in V$ the TSP is said to be a symmetric TSP. Another special case occurs if the triangle condition holds i.e., $c_{i,j} + c_{j,k} \geq c_{i,k}$ for all triples $i, j, k \in V$. For the following discussion we define $\boldsymbol{T}$ as the set of all traveling salesman tours in G and z_{OPT} as the length of the optimal traveling salesman tour in G.

3.1. Mathematical Programming-Based Formulations and Algorithms

3.1.1. The Asymmetric Case

Assume a directed graph $G = (V,A)$ with $V = \{1,...,n\}$ and for every arc $(i,j) \in E$, let $c_{i,j}$ be the cost of the arc. Without loss of generality we can assume that G is a complete graph by defining $c_{i,j}$ to be sufficiently large for those (i,j) not contained in A. Define decision variables $x_{i,j}$ that take on the value 1 if the optimal tour includes the arc (i,j) and the value zero, otherwise. Then the traveling salesman problem can be formulated as follows:

$$\text{minimize} \quad \sum_{(i,j)\in A} c_{i,j}\, x_{i,j} \qquad \text{subject to} \tag{7}$$

$$\sum_{j=1}^{n} x_{i,j} = 1 \qquad \text{for i} = 1, 2, \ldots, n \tag{8}$$

$$\sum_{i=1}^{n} x_{i,j} = 1 \qquad \text{for } j = 1, 2, \ldots, n \tag{9}$$

$$\sum_{(i,j)\in S\times S} x_{i,j} \le |S| - 1 \quad \text{for } S \subseteq V,\ 2 \le |S| \le n-2 \tag{10}$$

$$x_{i,j} \in \{0,1\} \qquad \text{for } i, j = 1,\ldots,n;\ i \neq j. \tag{11}$$

In this formulation, the so called *degree constraints* (8) and (9) ensure that for each node i there is exactly one arc leading into i and exactly one arc leaving i. Constraints (10) are called subtour elimination constraints, and together with the degree constraints they exclude (0,1)-solutions representing a set of disjoint cycles on subsets of V. When relaxing the subtour elimination constraints we obtain a linear assignment problem that can be solved rather efficiently. This assignment relaxation has been used in several branch & bound approaches for solving asymmetric TSP. (For a discussion of the assignment problem and the branch & bound approach see *Fundamentals of Optimization and Operations Research*, and *Graph and Network Optimization*.)

-
-
-

Bibliography

Ball M.O., Magnanti T.L., Monma C.L., and Nemhauser G.L. (eds.) (1995). *Network Routing. Handbooks of Operations Research and Management Science* 8, 785 pp. Amsterdam: North-Holland. [This book contains nine chapters written by leading experts in the field and provides a broad coverage of the models and algorithms developed for routing.]

Bodin L., Golden B., Assad A., and Ball M.O. (1983). Routing and scheduling of vehicles and crews: the state of the art. *Computers and Operations Research* **10**, 63–212. [Valuable source with a description of the classical approaches to vehicle routing.]

Dror M. (ed.) (2000). *Arc Routing. Theory, Solutions and Application*, 483 pp. Boston: Kluwer Academic Publishers. [State of the art compendium on the three aspects of arc routing given in the title with 12 articles written by leading experts in the field.]

Edmonds J. and Johnson E.L. (1973). Matching, Euler tours and the Chinese postman. *Mathematical Programming* **5**, 88–124. [First description of the polynomial matching based approach for solving CPP.]

Eilon S., Watson-Gandy C.D.T., and Christofides N. (1971). *Distribution Management*: *Mathematical*

Modeling and Practical Analysis, 240 pp. London: Griffin. [A classical application driven description of standard models and algorithms for routing.]

Eiselt H.A., Gendreau M., and Laporte G. (1995a). Arc routing problems. Part I: the Chinese postman problem. *Operations Research* **43**, 231–242. [Extensive discussion of basic arc routing problems.]

Eiselt H.A., Gendreau M., and Laporte G. (1995b). Arc routing problems. Part II: the rural postman problem. *Operations Research* **43**, 399–414. [Further, extensive discussion of basic arc routing problems.]

Euler L. (1736). Solutio problematis ad geometrica situs pertinentis. *Commentarii Academiae Scientarum Petropolitana* **8**, 128–140. [First scientific discussion of an arc routing problem.]

Kwan M.-K. (1962). Graphic programming using odd and even points. *Chinese Mathematics* **1**, 273–277. [Birth of the Chinese postman problem.]

Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G., and Shmoys D.B. (ed.) (1985). *The Traveling Salesman Problem*, 465 pp. New York: John Wiley & Sons. [The compendium on all aspects of the traveling salesman problem and its variants.]

Savelsbergh M.W.P. (1985). *Computer Aided Routing*, 134 pp. Doctoral thesis. Amsterdam: Center for Mathematics and Computer Science.[This work shows how mathematical programming based algorithms and heuristics can be integrated in a decision support system for vehicle routing.]

Biographical Sketch

Ulrich Derigs is Director of the Department of Information Systems and Operations Research (WINFORS) at the University of Cologne, Germany. He received a Masters degree (1975) and a Doctoral degree (1979) in Mathematics and a Doctoral degree in Economics (1981) from the University of Cologne. In 1985 he completed his habilitation at the University of Bonn. Ulrich Derigs was Research Assistant at the Mathematical Institute (1976–1979) and at the Seminar for Industrial Engineering (1979–1981) at the University of Cologne. From 1981 to 1985 he was Research Assistant at the Institute for Econometrics and Operations Research at the University of Bonn and a member of a Sonderforschungsbereich (DFG). From 1985 to 1990 Ulrich Derigs was Professor of Information Systems and Operations Research at the University of Bayreuth and since 1990 he has been Professor at the University of Cologne.

Ulrich Derigs has done extensive research in combinatorial optimization with an emphasis on the design, analysis, and evaluation of efficient algorithms and industrial applications. Today his interest lies in the interface between Operations Research and information systems. His focus is the design and implementation of model based decision support concepts and systems in different application areas, like routing and scheduling, production planning, logistics, finance, and telecommunication.

Ulrich Derigs is on the editorial board of several journals. From 1988 to 1992 he was Editor in chief of *OR-Spectrum*, the journal of DGOR, the German OR society. From 1992 to 1998 he was member of the board of DGOR and from 1996 to1998 president of the DGOR.