Chapter 1

An Overview of Vehicle Routing Problems

Paolo Toth
Daniele Vigo

1.1 Introduction

The last decades have seen an increasing utilization of *optimization packages*, based on *Operations Research* and *Mathematical Programming* techniques, for the effective management of the provision of goods and services in distribution systems. The large number of real-world applications, both in North America and in Europe, have widely shown that the use of computerized procedures for the distribution process planning produces substantial savings (generally from 5% to 20%) in the global transportation costs. It is easy to see that the impact of these savings on the global economic system is significant. Indeed, the transportation process involves all stages of the production and distribution systems and represents a relevant component (generally from 10% to 20%) of the final cost of the goods.

The success of the utilization of Operations Research techniques is due to the development of computer systems, from both the hardware and the software points of view, and to the increasing integration of information systems into the productive and commercial processes.

A different factor of success, as important as the others, is the development of modeling and algorithmic tools implemented in recent years. Indeed, the proposed *models* take into account all the characteristics of the distribution problems arising in real-world applications, and the corresponding *algorithms* and computer implementations find good solutions for real-world instances within acceptable computing times.

In this book, we consider only the problems concerning the distribution of goods between depots and final users (*customers*). These problems are generally known as *Vehicle Routing Problems* (VRPs) or Vehicle Scheduling Problems. The models and algorithms proposed for the solution of vehicle and scheduling problems, presented in detail in this book, can be used effectively not only for the solution of problems concerning the *delivery* or *collection* of goods but for the solution of different real-world applications arising in transportation systems as well. Typical applications of this type are, for instance, solid waste collection, street cleaning, school bus routing, dial-a-ride systems, transportation of handicapped persons, routing of salespeople, and of maintenance units.

The distribution of goods concerns the service, in a given time period, of a set of customers by a set of *vehicles*, which are located in one or more *depots*, are operated by a set of crews (*drivers*), and perform their movements by using an appropriate *road network*. In particular, the solution of a VRP calls for the determination of a set of *routes*, each performed by a single vehicle that starts and ends at its own depot, such that all the requirements of the customers are fulfilled, all the *operational constraints* are satisfied, and the global *transportation cost* is minimized. In this section, we describe the typical characteristics of the routing and scheduling problems by considering their main components (road network, customers, depots, vehicles, and drivers), the different operational constraints that can be imposed on the construction of the routes, and the possible objectives to be achieved in the optimization process.

The road network, used for the transportation of goods, is generally described through a *graph*, whose *arcs* represent the road sections and whose *vertices* correspond to the road junctions and to the depot and customer locations. The arcs (and consequently the corresponding graphs) can be *directed* or *undirected*, depending on whether they can be traversed in only one direction (for instance, because of the presence of one-way streets, typical of urban or motorway networks) or in both directions, respectively. Each arc is associated with a *cost*, which generally represents its length, and a *travel time*, which is possibly dependent on the vehicle type or on the period during which the arc is traversed.

Typical characteristics of customers are

- vertex of the road graph in which the customer is located;
- amount of goods (*demand*), possibly of different types, which must be delivered or collected at the customer;
- periods of the day (*time windows*) during which the customer can be served (for instance, because of specific periods during which the customer is open or the location can be reached, due to traffic limitations);
- times required to deliver or collect the goods at the customer location (*unloading* or *loading times*, respectively), possibly dependent on the vehicle type; and
- subset of the available vehicles that can be used to serve the customer (for instance, because of possible access limitations or loading and unloading requirements).

Sometimes, it is not possible to fully satisfy the demand of each customer. In these cases, the amounts to be delivered or collected can be reduced, or a subset of customers can be left unserved. To deal with these situations, different *priorities*, or *penalties* associated with the partial or total lack of service, can be assigned to the customers.

The routes performed to serve customers start and end at one or more depots, located at the vertices of the road graph. Each depot is characterized by the number and types of vehicles associated with it and by the global amount of goods it can deal with. In some real-world applications, the customers are a priori partitioned among the depots, and the vehicles have to return to their home depot at the end of each route. In these cases, the overall VRP can be decomposed into several independent problems, each associated with a different depot.

Transportation of goods is performed by using a *fleet* of vehicles whose composition and size can be fixed or can be defined according to the requirements of the customers. Typical characteristics of the vehicles are

- *home depot* of the vehicle, and the possibility to end service at a depot other than the home one;
- *capacity* of the vehicle, expressed as the maximum weight, or volume, or number of pallets, the vehicle can load;
- possible subdivision of the vehicle into *compartments*, each characterized by its capacity and by the types of goods that can be carried;
- devices available for the loading and unloading operations;
- subset of arcs of the road graph which can be traversed by the vehicle; and
- *costs* associated with utilization of the vehicle (per distance unit, per time unit, per route, etc.).

Drivers operating the vehicles must satisfy several constraints laid down by union contracts and company regulations (for instance, working periods during the day, number and duration of breaks during service, maximum duration of driving periods, overtime). In the following, the constraints imposed on drivers are imbedded in those associated with the corresponding vehicles.

The routes must satisfy several operational constraints, which depend on the nature of the transported goods, on the quality of the service level, and on the characteristics of the customers and the vehicles. Some typical operational constraints are the following: along each route, the current load of the associated vehicle cannot exceed the vehicle capacity; the customers served in a route can require only the delivery or the collection of goods, or both possibilities can exist; and customers can be served only within their time windows and the working periods of the drivers associated with the vehicles visiting them. *Precedence constraints* can be imposed on the order in which the customers served in a route are visited. One type of precedence constraint requires that a given customer be served in the same route serving a given subset of other customers and that the customer must be visited before (or after) the customers belonging to the associated subset. This is the case, for instance, of the so-called *pickup and delivery problems*, wherein the routes can perform both the collection and the delivery of goods, and the goods collected from the pickup customers must be carried to the corresponding delivery customers by the same vehicle. Another type of precedence constraint imposes that if customers of different types are served in the same route, the order in which the customers are visited is fixed. This situation arises, for instance, for the so-called *VRP with Backhauls*, wherein again, the routes can perform both the collection and the delivery of goods, but constraints associated with the loading and unloading operations, and the difficulty in rearranging the load of the vehicle along the route, mean that all deliveries must be performed before the collections.

Evaluation of the global cost of the routes, and the check of the operational constraints imposed on them, requires knowledge of the *travel cost* and the *travel time* between each pair of customers and between the depots and the customers. To this end, the original road graph (which often is very sparse) is generally transformed into a *complete graph*, whose vertices are the vertices of the road graph corresponding to the customers and the depots. For each pair of vertices i and j of the complete graph, an arc (i, j) is defined whose cost c_{ij} is given by the cost of the *shortest path* starting from vertex i and arriving at vertex j in the road graph. The travel time t_{ij}, associated with each arc (i, j) of the complete graph, is computed as the sum of the travel times of the arcs belonging to the shortest path from i to j in the road graph. In the following, instead of the original road graph, we consider the associated complete graph, which can be directed or undirected, depending on the property of the corresponding cost and travel-time matrices to be *asymmetric* or *symmetric*, respectively.

Several, and often contrasting, objectives can be considered for the vehicle routing problems. Typical objectives are

- minimization of the *global transportation cost*, dependent on the global distance traveled (or on the global travel time) and on the fixed costs associated with the used vehicles (and with the corresponding drivers);
- minimization of the number of vehicles (or drivers) required to serve all the customers;
- balancing of the routes, for travel time and vehicle load;
- minimization of the penalties associated with partial service of the customers;

or any weighted combination of these objectives.

In some applications, each vehicle can operate more than one route in the considered time period, or the routes can last for more than 1 day. In addition, sometimes it is necessary to consider *stochastic* or *time-dependent dynamic* versions of the problem, i.e., problems for which, a priori, there is only partial knowledge of the demands of the customers or of the costs (and the travel times) associated with the arcs of the road network.

More than 40 years have elapsed since Dantzig and Ramser [11] introduced the VRP. In their paper, the authors described a real-world application (concerning the delivery of gasoline to gas stations) and proposed the first *mathematical programming formulation* and algorithmic approach for the solution of the problem. A few years later, Clarke and Wright [9] proposed an effective greedy heuristic that improved on the Dantzig–Ramser approach. Following these two seminal papers, many models and exact and heuristic algorithms were proposed for the optimal and approximate solution of the different versions of the VRP. The most important and most effective models and algorithms are described in the various chapters of this book.

There are several main survey papers on the subject of VRPs. A classification scheme was given in Desrochers, Lenstra, and Savelsbergh [13]. Laporte and Nobert [32] presented an extensive survey that was entirely devoted to exact methods for the VRP, and they gave a complete and detailed analysis of the state of the art up to the late 1980s. Other surveys covering exact algorithms, but often mainly devoted to heuristic methods, were presented by Christofides, Mingozzi, and Toth [7], Magnanti [36], Bodin et al. [4], Christofides [5], Laporte [30], Fisher [19], Toth and Vigo [41, 42], and Golden et al. [26].

An annotated bibliography was proposed by Laporte [31], and an extensive bibliography was presented by Laporte and Osman [33]. A book on the subject was edited by Golden and Assad [25].

Models and algorithms for the solution of the so-called *Arc Routing Problem*, i.e., the variant of the problem arising when the customers are located not at the vertices but along the arcs of the road network, are described in the recent book edited by Dror [14]. The particular case of the VRP arising when only one vehicle is available at the depot and no additional operational constraints are imposed, i.e., the well-known *Traveling Salesman Problem*, is extensively described in the classic book edited by Lawler et al. [34].

1.2 Problem Definition and Basic Notation

In this section we give a formal definition, as graph theoretic models, of the basic problems of the vehicle routing class. These problems, which have received the greatest attention in the scientific literature, are examined in detail in the first two parts of the book. We first describe the Capacitated VRP, which is the simplest and most studied member of the family, then we introduce the Distance-Constrained VRP, the VRP with Time Windows, the VRP with Backhauls, and the VRP with Pickup and Delivery.

For each of these problems, several minor variants have been proposed and examined in the literature, and often different problems are given the same name. Although in many cases the solution methods, particularly the heuristic ones, may be adapted to incorporate additional features, this indeterminacy in problem definition generally causes much confusion. Therefore, for each problem we first describe the basic version, i.e., the one that in this book is denoted by the corresponding acronym, and then we discuss the variants. In addition, we make an explicit distinction between the symmetric and asymmetric versions of a problem only if models and solution approaches proposed in the literature make use of this distinction.

Also in this section, we introduce all the relevant notation and terminology used throughout the book. Additional notation and definitions required to describe particular variants and practical VRP problems are given in the appropriate chapters. Figure 1.1 summarizes the main problems described in this section and illustrates their connections. In the figure, an arrow moving from problem A to problem B means that B is an extension of A.

1.2.1 Capacitated and Distance-Constrained VRP

The first part of this book (Chapters 2–6) concentrates on the basic version of the VRP, the *Capacitated VRP* (CVRP). In the CVRP, all the customers correspond to deliveries and the demands are deterministic, known in advance, and may not be split. The vehicles are identical and based at a single central depot, and only the capacity restrictions for the vehicles are imposed. The objective is to minimize the total cost (i.e., a weighted function of the number of routes and their length or travel time) to serve all the customers.

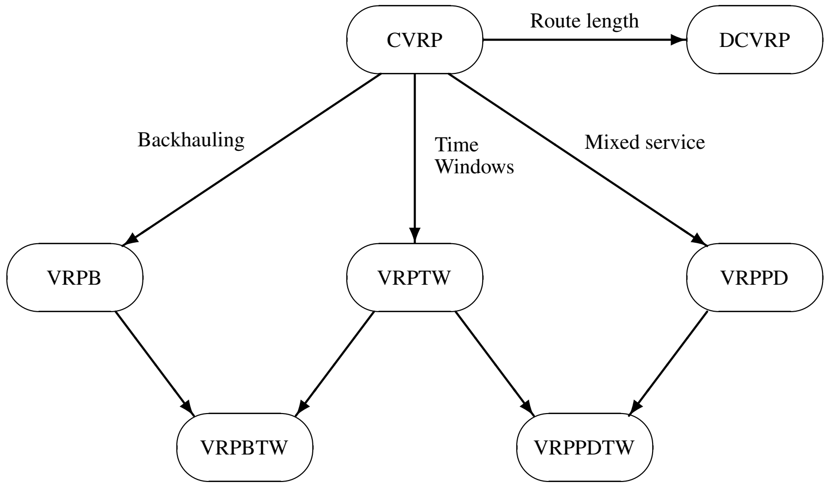

Figure 1.1. *The basic problems of the VRP class and their interconnections.*

The CVRP may be described as the following graph theoretic problem. Let $G = (V, A)$ be a complete graph, where $V = \{0, \dots, n\}$ is the vertex set and A is the arc set. Vertices $i = 1, \dots, n$ correspond to the customers, whereas vertex 0 corresponds to the depot. Sometimes the depot is associated with vertex $n + 1$.

A nonnegative cost, c_{ij}, is associated with each arc $(i, j) \in A$ and represents the travel cost spent to go from vertex i to vertex j. Generally, the use of the loop arcs, (i, i), is not allowed and this is imposed by defining $c_{ii} = +\infty$ for all $i \in V$. If G is a directed graph, the cost matrix c is asymmetric, and the corresponding problem is called asymmetric CVRP (ACVRP). Otherwise, we have $c_{ij} = c_{ji}$ for all $(i, j) \in A$, the problem is called symmetric CVRP (SCVRP), and the arc set A is generally replaced by a set of undirected edges, E. Given an edge $e \in E$, let $\alpha(e)$ and $\beta(e)$ denote its endpoint vertices. In the following we denote the edge set of the undirected graph G by A when edges are indicated by means of their endpoints (i, j), $i, j \in V$, and by E when edges are indicated through a single index e.

Graph G must be strongly connected and is generally assumed to be complete. Given a vertex i, let $\Delta^+(i)$ denote the so-called *forward star* of i, defined as the set of vertices j such that arc $(i, j) \in A$, i.e., the vertices that are directly reachable from i. Analogously, let $\Delta^-(i)$ denote the *backward star* of vertex i, defined as the set of vertices j such that arc $(j, i) \in A$, i.e., the vertices from which i is directly reachable. Given a vertex set $S \subseteq V$, let $\delta(S)$ and $E(S)$ denote the set of edges $e \in E$ that have only one or both endpoints in S, respectively. As usual, when a single vertex $i \in V$ is considered, we write $\delta(i)$ rather than $\delta(\{i\})$.

In several practical cases, the cost matrix satisfies the *triangle inequality*,

$$c_{ik} + c_{kj} \geq c_{ij} \quad \text{for all } i, j, k \in V. \tag{1.1}$$

In other words, it is not convenient to deviate from the direct link between two vertices i and j. The presence of the triangle inequality is sometimes required by the algorithms for CVRP, and this may be obtained in a simple way by adding a suitably large positive quantity M to the cost of each arc. However, the drastic distortion of the metric induced by this operation may produce very bad lower and upper bounds with respect to those corresponding to the original costs. Note that when the cost of each arc of the graph is equal to the cost of the shortest path between its endpoints, the corresponding cost matrix satisfies the triangle inequality.

In some instances the vertices are associated with points of the plane having given coordinates, and the cost c_{ij}, for each arc $(i, j) \in A$, is defined as the Euclidean distance between the two points corresponding to vertices i and j. In this case the cost matrix is symmetric and satisfies the triangle inequality, and the resulting problem called *Euclidean* SCVRP. Observe that the frequently performed rounding to the nearest integer of the real-valued Euclidean arc costs may cause a violation of the triangle inequality, whereas this does not happen if the costs are rounded up.

Each customer i $(i = 1, \dots, n)$ is associated with a known nonnegative demand, d_i, to be delivered, and the depot has a fictitious demand $d_0 = 0$. Given a vertex set $S \subseteq V$, let $d(S) = \sum_{i \in S} d_i$ denote the total demand of the set.

A set of K identical vehicles, each with capacity C, is available at the depot. To ensure feasibility we assume that $d_i \leq C$ for each $i = 1, \dots, n$. Each vehicle may perform at most one route, and we assume that K is not smaller than $K_{\min}$, where $K_{\min}$ is the minimum number of vehicles needed to serve all the customers. The value of $K_{\min}$ may be determined by solving the *Bin Packing Problem* (BPP) associated with the CVRP, which calls for the determination of the minimum number of bins, each with capacity C, required to load all the n items, each with nonnegative weight d_i, $i = 1, \dots, n$. Although BPP is NP-hard in the strong sense, instances with hundreds of items can be optimally solved very effectively (see, e.g., Martello and Toth [37]).

Given a set $S \subseteq V \setminus \{0\}$, we denote by $r(S)$ the minimum number of vehicles needed to serve all customers in S, i.e., the optimal solution value of the BPP associated with item set S. Note that $r(V \setminus \{0\}) = K_{\min}$. Often, $r(S)$ is replaced by the trivial BPP lower bound

$$\lceil d(S)/C \rceil . \tag{1.2}$$

The CVRP consists of finding a collection of exactly K simple *circuits* (each corresponding to a vehicle route) with minimum cost, defined as the sum of the costs of the arcs belonging to the circuits, and such that

(i) each circuit visits the depot vertex;

(ii) each customer vertex is visited by exactly one circuit; and

(iii) the sum of the demands of the vertices visited by a circuit does not exceed the vehicle capacity, C.

Several variants of the basic versions of CVRP have been considered in the literature. First, when the number K of available vehicles is greater than $K_{\min}$, it may be possible to leave some vehicles unused, and thus at most K circuits must be determined. In this case, fixed costs are often associated with the use of the vehicles, and the additional objective requiring minimization of the number of circuits (i.e., of the vehicles used) is added to that requiring minimization of the total cost. Another frequently considered variant arises when the available vehicles are different, i.e., have different capacities C_k, $k = 1, \dots, K$. Finally, routes containing only one customer may not be allowed. In the next section we discuss how models for the basic CVRP can be adapted to take these additional features into account.

The CVRP is known to be NP-hard (in the strong sense) and generalizes the well-known *Traveling Salesman Problem* (TSP), calling for the determination of a minimum-cost simple circuit visiting all the vertices of G (Hamiltonian circuit) and arising when $C \geq d(V)$ and $K = 1$. Therefore, all the relaxations proposed for the TSP are valid for the CVRP.

The first variant of CVRP we consider is the so-called *Distance-Constrained VRP* (DVRP), where for each route the capacity constraint is replaced by a maximum length (or time) constraint. In particular, a nonnegative *length*, t_{ij} (or t_e) is associated with each arc $(i, j) \in A$ (or edge $e \in E$), and the total length of the arcs of each route cannot exceed the maximum route length, T. If the vehicles are different, then the maximum route lengths are T_k, $k = 1, \dots, K$. Moreover, when arc lengths represent travel times, a *service time*, s_i, may be associated with each customer i, denoting the time period for which the vehicle must stop at its location. Alternatively, the service times can be added to the travel times of the arcs, i.e., by defining, for each arc (i, j), $t_{ij} = t'_{ij} + s_i/2 + s_j/2$, where t'_{ij} is the original travel time of arc (i, j).

Generally, the cost and the length matrices coincide, i.e., $c_{ij} = t_{ij}$ for all $(i, j) \in A$ (or $c_e = t_e$ for all $e \in E$). Hence, the objective of the problem is to minimize the total length of the routes or of their duration, when the service time is included in the travel time of the arcs. The case in which both the vehicle capacity and the maximum distance constraints are present is called *Distance-Constrained CVRP* (DCVRP).

Exact and heuristic algorithms for CVRP and DCVRP are described in Chapters 2–4 and 5 and 6, respectively.

1.2.2 VRP with Time Windows

The *VRP with Time Windows* (VRPTW) is the extension of the CVRP in which capacity constraints are imposed and each customer i is associated with a time interval $[a_i, b_i]$, called a *time window*. The time instant in which the vehicles leave the depot, the travel time, t_{ij}, for each arc $(i, j) \in A$ (or t_e for each $e \in E$) and an additional service time s_i for each customer i are also given. The service of each customer must start within the associated time window, and the vehicle must stop at the customer location for s_i time instants. Moreover, in case of early arrival at the location of customer i, the vehicle generally is allowed to wait until time instant a_i, i.e., until the service may start.

Normally, the cost and travel-time matrices coincide, and the time windows are defined by assuming that all vehicles leave the depot at time instant 0. Moreover, observe that the time window requirements induce an implicit orientation of each route even if the original matrices are symmetric. Therefore, VRPTW normally is modeled as an asymmetric problem.

VRPTW consists of finding a collection of exactly K simple circuits with minimum cost, and such that

(i) each circuit visits the depot vertex;

(ii) each customer vertex is visited by exactly one circuit;

(iii) the sum of the demands of the vertices visited by a circuit does not exceed the vehicle capacity, C; and

(iv) for each customer i, the service starts within the time window, $[a_i, b_i]$, and the vehicle stops for s_i time instants.

VRPTW is NP-hard in the strong sense, since it generalizes the CVRP, arising when $a_i = 0$, $b_i = +\infty$, for each $i \in V \setminus \{0\}$. Moreover, the so-called *TSP with Time Windows* (TSPTW) is the special case of VRPTW in which $C \geq d(V)$ and $K = 1$.

Exact and heuristic algorithms for VRPTW are described in Chapter 7.

1.2.3 VRP with Backhauls

The *VRP with Backhauls* (VRPB)is the extension of the CVRP in which the customer set $V \setminus \{0\}$ is partitioned into two subsets. The first subset, L, contains n *Linehaul customers*, each requiring a given quantity of product to be delivered. The second subset, B, contains m *Backhaul customers*, where a given quantity of inbound product must be picked up. Customers are numbered so that $L = \{1, \ldots, n\}$ and $B = \{n + 1, \ldots, n + m\}$.

In the VRPB, a precedence constraint between linehaul and backhaul customers exists: whenever a route serves both types of customer, all the linehaul customers must be served before any backhaul customer may be served. A nonnegative demand, d_i, to be delivered or collected depending on its type, is associated with each customer i, and the depot is associated with a fictitious demand $d_0 = 0$. When the cost matrix is asymmetric, the problem is called *Asymmetric VRP with Backhauls* (AVRPB). VRPB (and AVRPB as well) consists of finding a collection of exactly K simple circuits with minimum cost, and such that

(i) each circuit visits the depot vertex;

(ii) each customer vertex is visited by exactly one circuit;

(iii) the total demands of the linehaul and backhaul customers visited by a circuit do not exceed, separately, the vehicle capacity C; and

(iv) in each circuit all the linehaul customers precede the backhaul customers, if any.

Circuits containing only backhaul customers generally are not allowed. Moreover, observe that precedence constraint (iv) introduces an implicit orientation of the "mixed" vehicle routes, i.e., the routes that visit both linehaul and backhaul vertices.

Let K_L and K_B denote the minimum number of vehicles needed to serve all the linehaul and backhaul customers, respectively. These values can be obtained by solving the BPP instances associated with the corresponding customer subsets. To ensure feasibility, we assume that K is not smaller than the minimum number of vehicles needed to serve all the customers, i.e., $K \geq \max\{K_L, K_B\}$.

VRPB and AVRPB are NP-hard in the strong sense, since they generalize the basic versions of SCVRP and ACVRP, respectively, arising when $B = \emptyset$. Moreover, the so-called

TSP with Backhauls (TSPB) is the special case of VRPB in which $C \geq \max\{d(L), d(B)\}$ and $K = 1$. The case of VRPB in which time windows are present has been studied in the literature and is called the *VRP with Backhauls and Time Windows* (VRPBTW).

Exact and heuristic algorithms for VRPB and AVRPB are described in Chapter 8.

1.2.4 VRP with Pickup and Delivery

In the basic version of the *VRP with Pickup and Delivery* (VRPPD), each customer i is associated with two quantities d_i and p_i, representing the demand of homogeneous commodities to be delivered and picked up at customer i, respectively. Sometimes, only one demand quantity $d_i = d_i - p_i$ is used for each customer i, indicating the net difference between the delivery and the pickup demands (thus being possibly negative). For each customer i, O_i denotes the vertex that is the origin of the delivery demand, and D_i denotes the vertex that is the destination of the pickup demand.

It is assumed that, at each customer location, the delivery is performed before the pickup; therefore, the current load of a vehicle before arriving at a given location is defined by the initial load minus all the demands already delivered plus all the demands already picked up.

The VRPPD consists of finding a collection of exactly K simple circuits with minimum cost, and such that

(i) each circuit visits the depot vertex;

(ii) each customer vertex is visited by exactly one circuit;

(iii) the current load of the vehicle along the circuit must be nonnegative and may never exceed the vehicle capacity C;

(iv) for each customer i, the customer O_i, when different from the depot, must be served in the same circuit and before customer i; and

(v) for each customer i, the customer D_i, when different from the depot, must be served in the same circuit and after customer i.

Often the origin or the destination of the demands are common (for example they are associated with the depot, as in CVRP and VRPB), and hence there is no need to explicitly indicate them. This problem is known as the *VRP with Simultaneous Pickup and Delivery* (VRPSPD).

VRPPD and VRPSPD are NP-hard in the strong sense, since they generalize the CVRP arising when $O_i = D_i = 0$ and $p_i = 0$ for each $i \in V$. Moreover, the so-called *TSP with Pickup and Delivery* (TSPPD) is the special case of VRPSPD in which $K = 1$. The case of VRPPD in which time windows are present has been studied in the literature and is called the *VRP with Pickup and Deliveries and Time Windows* (VRPPDTW). Exact and heuristic algorithms for an extended version of VRPPD are described in Chapter 9.

1.3 Basic Models for the VRP

In this section we present the main mathematical programming formulations that can be used to model the basic VRPs presented in the previous section. In general, we give the models for the CVRP and discuss how they may be extended to incorporate additional constraints or different objective functions. Additional formulations can be found in Laporte and Nobert [32].

Three different basic modeling approaches have been proposed for the VRP in the literature. The models of the first type, known as *vehicle flow formulations*, use integer variables, associated with each arc or edge of the graph, which count the number of times the arc or edge is traversed by a vehicle. These are the more frequently used models for the basic versions of VRP. They are particularly suited for cases in which the cost of the solution can be expressed as the sum of the costs associated with the arcs, and when the most relevant constraints concern the direct transition between the customers within the route, so they can be effectively modeled through an appropriate definition of the arc set and of the arc costs. On the other hand, vehicle flow models cannot be used to handle many practical issues, e.g., when the cost of a solution depends on the overall vertex sequence or on the type of vehicle assigned to a route. Moreover, the linear programming relaxation of vehicle flow models can be very weak when the additional operational constraints are tight.

The second family of models is based on the so-called *commodity flow formulation*. In this type of model, additional integer variables are associated with the arcs or edges and represent the flow of the commodities along the paths traveled by the vehicles. Only recently have models of this type been used as a basis for the exact solution of CVRP.

The models of the last type have an exponential number of binary variables, each associated with a different feasible circuit. The VRP is then formulated as a *Set-Partitioning Problem* (SPP) calling for the determination of a collection of circuits with minimum cost, which serves each customer once and, possibly, satisfies additional constraints. A main advantage of this type of model is that it allows for extremely general route costs, e.g., depending on the whole sequence of the arcs and on the vehicle type. Moreover, the additional side constraints need not take into account restrictions concerning the feasibility of a single route. As a result, they often can be replaced with a compact set of inequalities. This produces a formulation whose linear programming relaxation is typically much tighter than that in the previous models. Note, however, that these models generally require dealing with a very large number of variables.

To simplify the notation, unless explicitly stated, in the following we assume that the graph $G(V, A)$ (or $G(V, E)$) is complete.

1.3.1 Vehicle Flow Models

We start by describing an integer linear programming formulation for ACVRP, which is later adapted to SCVRP. The model is a *two-index vehicle flow formulation* that uses $O(n^2)$ binary variables x to indicate if a vehicle traverses an arc in the optimal solution. In other words, variable x_{ij} takes value 1 if arc $(i, j) \in A$ belongs to the optimal solution and takes value 0 otherwise.

$$\text{(1.3)} \qquad \text{(VRP1)} \quad \min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

subject to

$$\text{(1.4)} \qquad \sum_{i \in V} x_{ij} = 1 \qquad \forall\, j \in V \setminus \{0\},$$

$$\text{(1.5)} \qquad \sum_{j \in V} x_{ij} = 1 \qquad \forall\, i \in V \setminus \{0\},$$

$$\text{(1.6)} \qquad \sum_{i \in V} x_{i0} = K,$$

$$\text{(1.7)} \qquad \sum_{j \in V} x_{0j} = K,$$

$$\text{(1.8)} \qquad \sum_{i \notin S} \sum_{j \in S} x_{ij} \geq r(S) \qquad \forall\, S \subseteq V \setminus \{0\},\, S \neq \emptyset,$$

$$\text{(1.9)} \qquad x_{ij} \in \{0, 1\} \qquad \forall\, i, j \in V.$$

The *indegree* and *outdegree* constraints (1.4) and (1.5) impose that exactly one arc enters and leaves each vertex associated with a customer, respectively. Analogously, constraints (1.6) and (1.7) impose the degree requirements for the depot vertex. Note that one arbitrary constraint among the $2|V|$ constraints (1.4)–(1.7) is actually implied by the remaining $2|V| - 1$ ones; hence it can be removed.

The so-called *capacity-cut* constraints (CCCs) of (1.8) impose both the connectivity of the solution and the vehicle capacity requirements. In fact, they stipulate that each cut $(V \setminus S, S)$ defined by a customer set S is crossed by a number of arcs not smaller than $r(S)$ (minimum number of vehicles needed to serve set S). The CCCs remain valid also if $r(S)$ is replaced by the trivial BPP lower bound (1.2); see, e.g., Cornuéjols and Harche [10].

Observe that when $|S| = 1$ or $S = V \setminus \{0\}$ the CCCs (1.8) are weakened forms of the corresponding degree constraints (1.4)–(1.7). Note also that, because of the degree constraints (1.4)–(1.7), we have

$$\text{(1.10)} \qquad \sum_{i \notin S} \sum_{j \in S} x_{ij} = \sum_{i \in S} \sum_{j \notin S} x_{ij} \qquad \forall\, S \subseteq V \setminus \{0\},\, S \neq \emptyset.$$

In other words, each cut $(V \setminus S, S)$ is crossed in both directions the same number of times. From (1.10) we may also restate (1.8) as

$$\text{(1.11)} \qquad \sum_{i \notin S} \sum_{j \in S} x_{ij} \geq r(V \setminus S) \qquad \forall\, S \subset V,\, 0 \in S.$$

An alternative formulation may be obtained by transforming the CCCs (1.8), by means of the degree constraints (1.4)–(1.7), into the well-known *generalized subtour elimination* constraints (GSECs):

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \le |S| - r(S) \qquad \forall\, S \subseteq V \setminus \{0\},\, S \ne \emptyset, \tag{1.12}$$

which impose that at least $r(S)$ arcs leave each customer set S.

Both families of constraints (1.8) and (1.12) have a cardinality growing exponentially with n. This means that it is practically impossible to solve directly the linear programming relaxation of problem (1.3)–(1.9). A possible way to partially overcome this drawback is to consider only a limited subset of these constraints and to add the remaining ones only if needed, by using appropriate *separation procedures*. The considered constraints can be relaxed in a Lagrangian fashion, as done by Fisher [18] and Miller [39] (see Chapter 2), or they can be explicitly included in the linear programming relaxation, as done in branch-and-cut approaches (see Chapter 3). Alternatively, a family of constraints equivalent to (1.8) and (1.12) and having a polynomial cardinality may be obtained by considering the subtour elimination constraints proposed for the TSP by Miller, Tucker, and Zemlin in [38] and extending them to ACVRP (see, e.g., Christofides, Mingozzi, and Toth [7] and Desrochers and Laporte [12]):

$$u_i - u_j + Cx_{ij} \le C - d_j \qquad \forall\, i, j \in V \setminus \{0\}, i \ne j, \text{ such that } d_i + d_j \le C, \tag{1.13}$$

$$d_i \le u_i \le C \qquad \forall\, i \in V \setminus \{0\}, \tag{1.14}$$

where u_i, $i \in V \setminus \{0\}$, is an additional continuous variable representing the load of the vehicle after visiting customer i. It is easy to see that constraints (1.13)–(1.14) impose both the capacity and the connectivity requirements of ACVRP. Indeed, when $x_{ij} = 0$, constraint (1.13) is not binding since $u_i \le C$ and $u_j \ge d_j$, whereas when $x_{ij} = 1$, they impose that $u_j \ge u_i + d_j$. (Note that isolated subtours are eliminated as well.)

It is worth noting that the linear programming relaxation of formulation (1.3)–(1.7), (1.13), (1.14), and (1.9) generally is much weaker than that of formulation (1.3)–(1.9). Tightening constraints were proposed by Desrochers and Laporte [12].

Model VRP1 can be easily adapted to the symmetric problem. To this end it should be noted that in SCVRP the routes are not oriented (i.e., the customers along a route may be visited indifferently clockwise or counterclockwise). Therefore, it is not necessary to know in which direction edges are traversed by the vehicles, and for each undirected edge $(i, j) \in A$, $i, j \ne 0$, only one of the two variables x_{ij} and x_{ji} must be used, for example, that with $i < j$. Note that when single-customer routes are not allowed, the edges incident to the depot can be traversed at most once. When, instead, a single-customer route is allowed for customer j, one may either include in the model both binary variables x_{0j} and x_{j0} or use a single integer variable, which may take value $\{0, 1, 2\}$. In this latter case, if $x_{0j} = 2$, then a route including the single customer j is selected in the solution. In the following models we assume that single-customer routes are allowed. The symmetric version of model VRP1 then reads

$$\text{(VRP2)} \quad \min \sum_{i \in V \setminus \{n\}} \sum_{j>i} c_{ij} x_{ij} \tag{1.15}$$

subject to

$$\sum_{h<i} x_{hi} + \sum_{j>i} x_{ij} = 2 \qquad \forall\, i \in V \setminus \{0\}, \tag{1.16}$$

$$\sum_{j \in V \setminus \{0\}} x_{0j} = 2K, \tag{1.17}$$

$$\sum_{i \in S} \sum_{\substack{h < i \\ h \notin S}} x_{hi} + \sum_{i \in S} \sum_{\substack{j > i \\ j \notin S}} x_{ij} \geq 2r(S) \qquad \forall\, S \subseteq V \setminus \{0\},\, S \neq \emptyset, \tag{1.18}$$

$$x_{ij} \in \{0, 1\} \qquad \forall\, i, j \in V \setminus \{0\},\, i < j, \tag{1.19}$$

$$x_{0j} \in \{0, 1, 2\} \qquad \forall\, j \in V \setminus \{0\}. \tag{1.20}$$

The degree constraints (1.16) and (1.17) impose that exactly two edges are incident into each vertex associated with a customer and that $2K$ edges are incident into the depot vertex, respectively. The CCCs (1.18) impose both the connectivity of the solution and the vehicle capacity requirements by forcing that a sufficient number of edges enter each subset of vertices. Constraints (1.10)–(1.12) may be adapted to SCVRP in a similar way.

The symmetric version of the two-index models is more frequently defined by using variables with a single index e associated with the undirected edges $e \in E$. If single-customer routes are not allowed, all used variables are binary; otherwise, if $e \notin \delta(0)$, then $x_e \in \{0, 1\}$, whereas if $x_e \in \delta(0)$, then $x_e \in \{0, 1, 2\}$.

$$\text{(VRP3)} \quad \min \sum_{e \in E} c_e x_e \tag{1.21}$$

subject to

$$\sum_{e \in \delta(i)} x_e = 2 \qquad \forall\, i \in V \setminus \{0\}, \tag{1.22}$$

$$\sum_{e \in \delta(0)} x_e = 2K, \tag{1.23}$$

$$\sum_{e \in \delta(S)} x_e \geq 2r(S) \qquad \forall\, S \subseteq V \setminus \{0\},\, S \neq \emptyset, \tag{1.24}$$

$$x_e \in \{0, 1\} \qquad \forall\, e \notin \delta(0), \tag{1.25}$$

$$x_e \in \{0, 1, 2\} \qquad \forall\, e \in \delta(0). \tag{1.26}$$

Also in this case, due to (1.22), the CCCs (1.24) may be rewritten as the generalized subtour elimination constraints:

$$\sum_{e \in E(S)} x_e \leq |S| - r(S) \qquad \forall\, S \subset V \setminus \{0\}, S \neq \emptyset, \tag{1.27}$$

where $r(S)$ may be replaced by the trivial BPP lower bound.

Two-index vehicle flow models have been extensively used to model the basic versions of SCVRP and ACVRP and some other variants, such as the VRPB, but they generally are inadequate for more complex versions of VRP. In fact, as mentioned, they can be used only when the cost of the solution can be expressed as the sum of the costs associated with the traversed arcs. In addition, it is not possible to directly know which vehicle traverses an arc used in the solution. Hence, these models are not suited for the cases where the cost (or the feasibility) of a circuit depends on the overall vertex sequence or on the type of vehicle allocated to the route.

A possible way to partially overcome some of the drawbacks associated with the two-index models is to explicitly indicate the vehicle that traverses an arc, so that more involved constraints may be imposed on the routes. In this way one obtains the so-called *three-index vehicle flow formulation* of SCVRP and ACVRP, which uses $O(n^2K)$ binary variables x: variable x_{ijk} counts the number of times arc $(i, j) \in A$ is traversed by vehicle k $(k = 1, \ldots, K)$ in the optimal solution. In addition, there are $O(nK)$ binary variables y: variable y_{ik} $(i \in V; k = 1, \ldots, K)$ takes value 1 if customer i is served by vehicle k in the optimal solution and takes value 0 otherwise. The three-index model for ACVRP is given in the following.

$$\text{(VRP4)} \quad \min \sum_{i \in V} \sum_{j \in V} c_{ij} \sum_{k=1}^{K} x_{ijk} \tag{1.28}$$

subject to

$$\sum_{k=1}^{K} y_{ik} = 1 \qquad \forall\, i \in V \setminus \{0\}, \tag{1.29}$$

$$\sum_{k=1}^{K} y_{0k} = K, \tag{1.30}$$

$$\sum_{j \in V} x_{ijk} = \sum_{j \in V} x_{jik} = y_{ik} \qquad \forall\, i \in V, k = 1, \ldots, K, \tag{1.31}$$

$$\sum_{i \in V} d_i y_{ik} \leq C \qquad \forall\, k = 1, \ldots, K, \tag{1.32}$$

$$\sum_{i \in S} \sum_{j \notin S} x_{ijk} \geq y_{hk} \qquad \begin{array}{l} \forall\, S \subseteq V \setminus \{0\}, h \in S, \\ k = 1, \ldots, K, \end{array} \tag{1.33}$$

$$y_{ik} \in \{0, 1\} \qquad \forall\, i \in V, k = 1, \ldots, K, \tag{1.34}$$

$$x_{ijk} \in \{0, 1\} \qquad \forall\, i, j \in V, k = 1, \ldots, K. \tag{1.35}$$

Constraints (1.29)–(1.31) impose that each customer is visited exactly once, that K vehicles leave the depot, and that the same vehicle enters and leaves a given customer, respectively. Constraints (1.32) are the capacity restriction for each vehicle k, whereas constraints (1.33) impose the connectivity of the route performed by k. These latter constraints may be replaced by *subtour elimination constraints* (SECs) (see Fisher and Jaikumar [20]):

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq |S| - 1 \qquad \forall\, S \subseteq V \setminus \{0\},\ |S| \geq 2,\ k = 1, \dots, K, \tag{1.36}$$

which impose that for each vehicle k at least 1 arc leaves each vertex set S visited by k and not containing the depot. Alternatively, the three-index version of the generalized Miller–Tucker–Zemlin subtour elimination constraints (1.13) can be used.

$$u_{ik} - u_{jk} + Cx_{ijk} \leq C - d_j \qquad \forall\, i, j \in V \setminus \{0\},\ i \neq j, \text{ such that } d_i + d_j \leq C,\ k = 1, \dots, K, \tag{1.37}$$

$$d_i \leq u_{ik} \leq C \qquad \forall\, i \in V \setminus \{0\},\ k = 1, \dots, K. \tag{1.38}$$

Note that these constraints replace also the capacity requirements (1.32).

The undirected version of the above model can be obtained easily by using binary variables x_{ek}, $e \in E$ and $k = 1, \dots, K$.

$$\text{(VRP5)} \quad \min \sum_{e \in E} c_e \sum_{k=1}^{K} x_{ek} \tag{1.39}$$

subject to

$$\sum_{k=1}^{K} y_{ik} = 1 \qquad \forall\, i \in V \setminus \{0\}, \tag{1.40}$$

$$\sum_{k=1}^{K} y_{0k} = K, \tag{1.41}$$

$$\sum_{e \in \delta(i)} x_{ek} = 2y_{ik} \qquad \forall\, i \in V,\ k = 1, \dots, K, \tag{1.42}$$

$$\sum_{i \in V} d_i y_{ik} \leq C \qquad \forall\, k = 1, \dots, K, \tag{1.43}$$

$$\sum_{e \in \delta(S)} x_{ek} \geq 2y_{hk} \qquad \forall\, S \subseteq V \setminus \{0\},\ h \in S,\ k = 1, \dots, K, \tag{1.44}$$

$$y_{ik} \in \{0, 1\} \qquad \forall\, i \in V,\ k = 1, \dots, K, \tag{1.45}$$

$$x_{ek} \in \{0, 1\} \qquad \forall\, e \notin \delta(0),\ k = 1, \dots, K, \tag{1.46}$$

$$x_{ek} \in \{0, 1, 2\} \qquad \forall\, e \in \delta(0),\ k = 1, \dots, K. \tag{1.47}$$

Three-index vehicle flow models have been extensively used to model more constrained versions of the VRP, such as the VRPTW, due to their greater flexibility in incorporating additional features (see the next section). The main drawback of these models is represented by the increased number of variables. On the other hand, they generalize the two-index models, which may be obtained by simply defining $x_{ij} = \sum_{k=1}^{K} x_{ijk}$ for all $(i, j) \in A$ or $x_e = \sum_{k=1}^{K} x_{ek}$ for all $e \in E$, thus allowing both the direct use of all the inequalities proposed for two-index models and the development of additional and stronger formulations.

1.3.2 Extensions of Vehicle Flow Models

Vehicle flow formulations, particularly the more flexible three-index ones, may be adapted to model some variants of the basic versions of SCVRP and ACVRP. In the following we discuss some of them by describing only the modifications required by the asymmetric models VRP1 and VRP4. Models VRP2, VRP3, and VRP5 can be adapted in a similar way. The adaptations required to model VRPB, VRPTW, and VRPPD are described in Chapters 7, 8, and 9, respectively.

First, we consider the case in which the graph is not complete, arising when some of the arcs are missing. This may be immediately incorporated into the considered models by defining the cost of the missing arcs as a suitably large positive value (practically equivalent to $+\infty$). When the number of missing arcs is large, i.e., when $|A| = m \ll n^2$, the models may be modified to take advantage of the graph sparsity by explicitly using the forward and backward stars of the vertices. As an example, model VRP1 becomes

$$\text{(VRP6)} \quad \min \sum_{(i,j)\in A} c_{ij} x_{ij} \tag{1.48}$$

subject to

$$\sum_{i\in\Delta^-(j)} x_{ij} = 1 \qquad \forall\, j \in V \setminus \{0\}, \tag{1.49}$$

$$\sum_{j\in\Delta^+(i)} x_{ij} = 1 \qquad \forall\, i \in V \setminus \{0\}, \tag{1.50}$$

$$\sum_{i\in\Delta^-(0)} x_{i0} = K, \tag{1.51}$$

$$\sum_{j\in\Delta^+(0)} x_{0j} = K, \tag{1.52}$$

$$\sum_{j\in S} \sum_{i\in\Delta^-(j)\setminus S} x_{ij} \ge r(S) \qquad \forall\, S \subseteq V \setminus \{0\},\, S \neq \emptyset, \tag{1.53}$$

$$x_{ij} \in \{0, 1\} \qquad \forall\, (i, j) \in A. \tag{1.54}$$

A frequent modification of the models we consider is obtained by replacing the single depot vertex with K vertices, one for each available vehicle. For the asymmetric case, this is obtained by defining an extended complete digraph $G' = (V', A')$, where $V' :=$

$V \cup \{n+1, \ldots, n+K-1\}$ contains $K-1$ additional copies of vertex 0, and the cost c'_{ij} of each arc in A' is defined as follows:

$$c'_{ij} := \begin{cases} c_{ij} & \text{for } i, j \in V \setminus \{0\}, \\ c_{i0} & \text{for } i \in V \setminus \{0\},\ j \in W, \\ c_{0j} & \text{for } i \in W,\ j \in V \setminus \{0\}, \\ \lambda & \text{for } i, j \in W, \end{cases} \tag{1.55}$$

where $W := \{0\} \cup \{n+1, \ldots, n+K-1\}$ is the set of the K vertices of G' associated with the depot, and λ is a proper value. After this transformation, constraint (1.6) may be replaced by K constraints of type (1.4), one for each copy of the depot. Analogously, constraint (1.7) may be replaced by K constraints of type (1.5). This extension was originally proposed by Lenstra and Rinnooy Kan [35] to transform into an ordinary TSP the m-TSP, which calls for the determination of a collection of m circuits visiting m times a distinguished vertex (i.e., the depot) and one time each for the remaining vertices. Observe that, by appropriately defining λ, we may obtain different effects. In particular, when $\lambda = M$, where M is a very large positive number, the model requires use of all the K available vehicles, i.e., leads to the min-cost solution performing exactly K routes. Defining $\lambda = 0$ leads to the min-cost solution using at most K routes, whereas defining $\lambda = -M$ leads to the min-cost solution using $K_{\min}$ routes. Different values of λ can take into account possible fixed costs associated with the use of the vehicles.

An alternative way to model the case in which some vehicles may be left unused may be obtained by replacing constraints (1.6) and (1.7) in model VRP1 with

$$\sum_{i \in V} x_{i0} \le K, \tag{1.56}$$

$$\sum_{j \in V} x_{0j} = \sum_{i \in V} x_{i0}, \tag{1.57}$$

whereas in model VRP4 constraint (1.30) may be replaced with

$$\sum_{k=1}^{K} y_{0k} \le K. \tag{1.58}$$

Generally, the possibility of leaving some vehicles unused is associated with the presence of fixed costs for their use and, possibly, the additional objective requiring the minimization of the number of vehicles used, and then of the total routing costs associated with the use of vehicles. There are different ways to take this requirement into account. When considering models that impose the use of all the K available vehicles, one may first compute $K_{\min}$, by solving the BPP associated with ACVRP or SCVRP, and then define $K = K_{\min}$. Otherwise, the instance may be extended, as described above, by adding multiple copies of the depot and the parameter λ is set to $-M$.

When the model allows for the determination of solutions using a number of vehicles smaller than K, this objective may be easily included by adding a large constant value to the cost of the arcs leaving the depot. Thus, the optimal solution first minimizes the number of arcs leaving the depot (hence the number of circuits) then minimizes the cost of the other used arcs. In three-index models, where the use of each vehicle may be individually determined, the fixed costs may be different, and they can be directly included into an extended objective function rather than being added to the cost of the arcs leaving the depot.

Three-index vehicle flow models may easily take into account the case of a nonhomogeneous fleet, where each vehicle may have a different capacity C_k, $k = 1, \dots, K$. This is obtained by replacing C with C_k in the capacity constraints (1.32).

Finally, in some cases, as in Fisher [18], routes serving a single customer are not allowed. In the models for the ACVRP, this can be imposed by adding the following additional constraints:

$$x_{0j} + x_{j0} \leq 1 j \in V \setminus \{0\}. \tag{1.59}$$

In the models for SCVRP, the infeasibility of the single customer routes can be easily imposed, as discussed in the previous section, by imposing that each variable associated with an edge incident into the depot-vertex does not take value 2. In this case, constraints (1.19) and (1.20) may be replaced by

$$x_{ij} \in \{0, 1\} i, j \in V, i < j. \tag{1.60}$$

It should be noted that in many practical cases the above assumption is not constraining. Indeed, customer j can be served alone in a route if and only if on the remaining $K - 1$ vehicles there is enough space to load the demand of the other customers, i.e., if $r(V \setminus \{j\}) \leq K - 1$. By replacing $r(\cdot)$ with the trivial BPP lower bound we may restate the above condition as

$$d_j \geq C_{\min} = d(V) - (K - 1)C. \tag{1.61}$$

If, given an SCVRP (or ACVRP) instance, condition (1.61) is satisfied by no customer j, then in any feasible solution no customer may be served alone in a route (hence the constraints preventing single-customer routes are superfluous).

1.3.3 Commodity Flow Models

Commodity flow models were first introduced by Garvin et al. [21] for an oil delivery problem and later extended by Gavish and Graves [23, 24] to variants of TSP and VRP. These formulations, in addition to the variables used by the two-index vehicle flow formulations of section 1.3.1, require a new set of (continuous) variables, associated with the arcs, which represent the amounts of demand that flow along them. The reader is referred to Laporte and Nobert [32] for a presentation and a discussion of early commodity flow models. However, no such model was used to develop exact approaches to VRP.

Baldacci, Mingozzi, and Hadjiconstantinou [2] presented an exact approach to SCVRP, based on the extension to SCVRP of the *two-commodity flow* formulation for the TSP introduced by Finke, Claus, and Gunn [16]. (See also Langevin et al. [29] for an extension of the model for the TSP with Time Windows.) Since commodity flow formulations require arc orientation, we define the model on a directed graph equivalent to the undirected one.

The formulation requires the extended graph $G' = (V', A')$ obtained from G by adding vertex $n+1$, which is a copy of the depot node, as explained in section 1.3.2. Routes are now paths from vertex 0 to vertex $n+1$. Two nonnegative *flow variables*, y_{ij} and y_{ji}, are associated with each arc $(i, j) \in A'$. If a vehicle travels from i to j, then y_{ij} and y_{ji} give the vehicle load and the vehicle residual capacity, respectively, along the arc, i.e., $y_{ji} = C - y_{ij}$. The roles are reversed if the vehicle travels from j to i. Therefore, the equation $y_{ij} + y_{ji} = C$ holds for each arc $(i, j) \in A'$.

For any route of a feasible solution, the flow variables define two directed paths, one from vertex 0 to $n+1$, whose variables represent the vehicle load, and another from vertex $n+1$ to vertex 0, whose variables represent the residual capacity on the vehicle. In other words, think of this as one vehicle going from 0 to $n+1$, leaving vertex 0 with just enough product, delivering at every customer an amount equal to its demand, and arriving empty at vertex $n+1$; and think of another vehicle leaving vertex $n+1$ empty and picking up at every customer an amount equal to its demand. An example with four clients and $C = 25$ is shown in Figure 1.2, where the demands are shown next to each vertex.

As in two-index vehicle flow models, for each arc $(i, j) \in A'$, let x_{ij} be equal to 1 if the arc is in the solution and be equal to 0 otherwise. Then, an integer formulation of SCVRP is as follows:

$$\text{(VRP7)} \quad \min \sum_{(i,j)\in A'} c_{ij}x_{ij} \tag{1.62}$$

subject to

$$\sum_{j\in V'}(y_{ji} - y_{ij}) = 2d_i \qquad \forall\, i \in V' \setminus \{0, n+1\}, \tag{1.63}$$

$$\sum_{j\in V'\setminus\{0,n+1\}} y_{0j} = d(V \setminus \{0, n+1\}), \tag{1.64}$$

$$\sum_{j\in V'\setminus\{0,n+1\}} y_{j0} = KC - d(V \setminus \{0, n+1\}), \tag{1.65}$$

$$\sum_{j\in V'\setminus\{0,n+1\}} y_{n+1\,j} = KC, \tag{1.66}$$

$$y_{ij} + y_{ji} = Cx_{ij} \qquad \forall\, (i, j) \in A', \tag{1.67}$$

$$\sum_{j\in V'}(x_{ij} + x_{ji}) = 2 \qquad \forall\, i \in V' \setminus \{0, n+1\}, \tag{1.68}$$

$$y_{ij} \geq 0 \qquad \forall\, (i, j) \in A', \tag{1.69}$$

$$x_{ij} \in \{0, 1\} \qquad \forall\, (i, j) \in A'. \tag{1.70}$$

Flow conservation constraints (1.63) impose that the difference between the sum of the commodity flow variables associated with arcs entering and leaving each vertex i is equal to twice the demand of i. Constraints (1.64)–(1.66) impose the correct values for the commodity flow variables incident into the depot vertices. Finally, constraints (1.67) and (1.68) impose the relation between vehicle flow and commodity flow variables and the vertex degree, respectively.

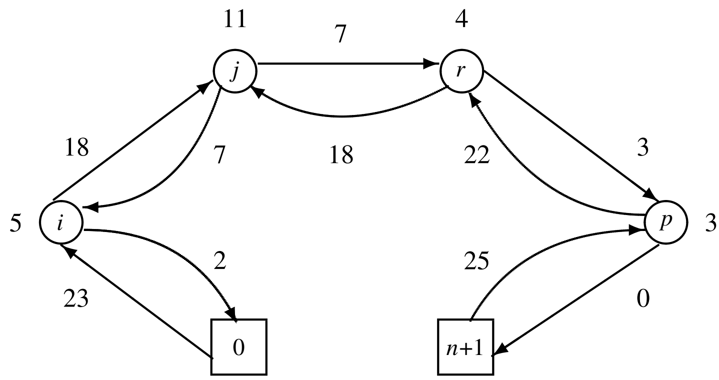

Figure 1.2. *Example of flow paths on a route ($C = 25$).*

Baldacci, Mingozzi, and Hadjiconstantinou [2] showed that the linear relaxation of this mixed integer program dominates that of model VRP1 when the CCCs (1.8) are dropped. The elimination of these inequalities, of course, weakens formulation VRP1 to a great extent, and thus the result is not so surprising.

1.3.4 Set-Partitioning Models

The *set-partitioning* (SP) formulation of the VRP was originally proposed by Balinski and Quandt [3] and uses a possibly exponential number of binary variables, each associated with a different feasible circuit of G. More specifically, let $\mathcal{H} = \{H_1, \dots, H_q\}$ denote the collection of all the circuits of G, each corresponding to a feasible route, with $q = |\mathcal{H}|$. Each circuit H_j has an associated cost c_j. In addition, let a_{ij} be a binary coefficient that takes value 1 if vertex i is visited (or *covered*, in the set partitioning jargon) by route H_j and takes value 0 otherwise. The binary variable x_j, $j = 1, \dots, q$, is equal to 1 if and only if circuit H_j is selected in the optimal solution. The model is

$$\text{(VRP8)} \quad \min \sum_{j=1}^{q} c_j x_j \tag{1.71}$$

subject to

$$\sum_{j=1}^{q} a_{ij} x_j = 1 \qquad \forall\, i \in V \setminus \{0\}, \tag{1.72}$$

$$\sum_{j=1}^{q} x_j = K, \tag{1.73}$$

$$x_j \in \{0, 1\} \qquad \forall\, j = 1, \dots, q. \tag{1.74}$$

Constraints (1.72) impose that each customer i is covered by exactly one of the selected circuits, and (1.73) requires that K circuits are selected. This is a very general model that may easily take into account several constraints as, for example, time windows, since route feasibility is implicitly considered in the definition of set $\mathcal{H}$. Moreover, the linear programming relaxation of this formulation typically is very tight.

Observe that if the cost matrix satisfies the triangle inequality, then the set partitioning model may be transformed into an equivalent *set-covering* (SC) model VRP8′ by writing (1.72) as

$$\sum_{j=1}^{q} a_{ij} x_j \geq 1 \qquad \forall\, i \in V \setminus \{0\}. \tag{1.75}$$

Any feasible solution to model VRP8 is also feasible for VRP8′, and any feasible solution to VRP8′ may be transformed into a feasible solution of VRP8 of not greater cost. Indeed, if the VRP8′ solution is infeasible for VRP8, this means that one or more customers are visited more than once. Then, these customers may be removed, by applying shortcuts, from all but one of the routes where they are included. Since the triangle inequality holds, each such shortcut would not increase the cost of the solution. The main advantage of using the VRP8′ formulation with respect to the VRP8 one is that in the former only inclusion-maximal feasible circuits, among those with the same cost, need be considered in the definition of $\mathcal{H}$. This considerably reduces the number q of variables. In addition, when using the VRP8′ formulation the dual solution space is considerably reduced since dual variables are restricted to nonnegative values only.

One of the main drawbacks of the VRP8 and VRP8′ models is represented by the huge number of variables, which, in non-tightly-constrained instances with tens of customers, may easily run into the billions. The explicit generation of all the feasible circuits (columns) is thus normally impractical, and one has to resort to a *column generation* approach to solve the linear programming relaxation of models VRP8 and VRP8′ (see Chapter 4).

1.4 Test Instances for the CVRP and Other VRPs

Despite the interest in VRPs by the scientific community and by practitioners, the computational testing of the solution methods for the VRP generally has been carried out by considering only a limited set of Euclidean test instances, which were proposed by Christofides and Eilon [6] and by Christofides, Mingozzi, and Toth [7]. These instances are identified with a variety of names by the various authors who used them in their papers and this may cause some confusion. Therefore, in this book we adopted the unified naming scheme described by Vigo [43] to identify the test instances used for CVRP and DCVRP.

The naming scheme for the instance data and solutions is an extension of that adopted by Augerat et al. [1]. The name of each instance should allow one to determine quickly its characteristics. In particular, the names have the form `tnnnvkkp` and are made up of five positional fields. The first field, `t`, is one alphabetical character that identifies the problem type and is equal to

- `E` for Euclidean SCVRP instances,
- `S` for non-Euclidean SCVRP instances,

- `A` for ACVRP instances, and
- `D` for symmetric DCVRP instances.

The second field of the name, `nnn`, is a three-digit integer that denotes the number of vertices of the problem graph, i.e., including the depot vertex. The third field, `v`, is normally equal to "-", but it may be an alphabetical character used to distinguish several instances that are characterized by the same number of vertices and available vehicles. The fourth field, `kk`, is a two-digit integer that denotes the number of available vehicles. Finally, the last field of the name, `p`, is an alphabetical character that identifies the paper where the problem data are first given or an alternative source for them, as follows:

- `a` Hays [28] and Eilon, Watson-Gandy, and Christofides [15],
- `c` Christofides, Mingozzi, and Toth [7],
- `d` Dantzig and Ramser [11] and Eilon, Watson-Gandy, and Christofides [15],
- `e` Christofides and Eilon [6],
- `f` Fisher [18],
- `g` Gaskell [22] and Eilon, Watson-Gandy, and Christofides [15],
- `h` Hadjiconstantinou, Christofides, and Mingozzi [27],
- `m` Christofides, Mingozzi, and Toth [8],
- `n` Noon, Mittenthal, and Pillai [40],
- `v` Fischetti, Toth, and Vigo [17], and
- `w` Clarke and Wright [9] and Eilon, Watson-Gandy, and Christofides [15].

For example, according to this naming scheme, E051-05e identifies the classical 50-customers Euclidean instance with 5 available vehicles proposed by Christofides and Eilon [6], and A073-03v identifies the 72-customers ACVRP instance with 3 vehicles described by Fischetti, Toth, and Vigo [17].

Bibliography

[1] P. Augerat, J.M. Belenguer, E. Benavent, A. Corberán, D. Naddef, and G. Rinaldi. Computational results with a branch and cut code for the capacitated vehicle routing problem. Technical Report RR 949-M, Université Joseph Fourier, Grenoble, 1995.

[2] R. Baldacci, E. Hadjiconstantinou and A. Mingozzi. An exact algorithm for the capacitated vehicle routing problem based on a two-commodity network flow formulation. *Operations Research* to appear, 2004.

[3] M. Balinski and R. Quandt. On an integer program for a delivery problem. *Operations Research*, 12:300–304, 1964.

[4] L.D. Bodin, B.L. Golden, A.A. Assad, and M. Ball. Routing and scheduling of vehicles and crews, the state of the art. *Computers and Operations Research*, 10(2):63–212, 1983.

[5] N. Christofides. Vehicle routing. In E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys, editors, *The Traveling Salesman Problem*, Wiley, Chichester, UK, 1985, pp. 431–448.

[6] N. Christofides and S. Eilon. An algorithm for the vehicle dispatching problem. *Operational Research Quarterly*, 20:309–318, 1969.

[7] N. Christofides, A. Mingozzi, and P. Toth. The vehicle routing problem. In N. Christofides, A. Mingozzi, P. Toth, and C. Sandi, editors, *Combinatorial Optimization*, Wiley, Chichester, UK, 1979, pp. 315–338.

[8] N. Christofides, A. Mingozzi, and P. Toth. Exact algorithms for the vehicle routing problem based on the spanning tree and shortest path relaxations. *Mathematical Programming*, 20:255–282, 1981.

[9] G. Clarke and J.V. Wright. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12:568–581, 1964.

[10] G. Cornuéjols and F. Harche. Polyhedral study of the capacitated vehicle routing problem. *Mathematical Programming*, 60:21–52, 1993.

[11] G.B. Dantzig and J.H. Ramser. The truck dispatching problem. *Management Science*, 6:80, 1959.

[12] M. Desrochers and G. Laporte. Improvements and extensions to the Miller–Tucker–Zemlin subtour elimination constraints. *Operations Research Letters*, 10:27–36, 1991.

[13] M. Desrochers, J.K. Lenstra, and M.W.P. Savelsbergh. A classification scheme for vehicle routing and scheduling problems. *Journal of Operational Research Society*, 46:322–332, 1990.

[14] M. Dror, editor. *Arc Routing: Theory, Solutions and Applications*. Kluwer, Boston, MA, 2000.

[15] S. Eilon, C. Watson-Gandy, and N. Christofides. *Distribution Management, Mathematical Modeling and Practical Analysis*. Griffin, London, 1971.

[16] G. Finke, A. Claus, and E. Gunn. A two-commodity network flow approach to the traveling salesman problem. *Congressus Numernatium*, 41:167–178, 1984.

[17] M. Fischetti, P. Toth, and D. Vigo. A branch-and-bound algorithm for the capacitated vehicle routing problem on directed graphs. *Operations Research*, 42:846–859, 1994.

[18] M.L. Fisher. Optimal solution of vehicle routing problems using minimum k-trees. *Operations Research*, 42:626–642, 1994.

[19] M.L. Fisher. Vehicle routing. In M.O. Ball, T.L. Magnanti, C.L. Monma, and G.L. Nemhauser, editors, *Network Routing*, *Handbooks in Operations Research and Management Science* 8, North-Holland, Amsterdam, 1995, pp. 1–33.

[20] M.L. Fisher and R. Jaikumar. A generalized assignment heuristic for the vehicle routing problem. *Networks*, 11:109–124, 1981.

[21] W.M. Garvin, H.W. Crandall, J.B. John, and R.A. Spellman. Applications of linear programming in the oil industry. *Management Science*, 3:407–430, 1957.

[22] T.J. Gaskell. Bases for vehicle fleet scheduling. *Operational Research Quarterly*, 18:281–295, 1967.

[23] B. Gavish and S. Graves. The travelling salesman problem and related problems. Working Paper 7905, Graduate School of Management, University of Rochester, Rochester, NY, 1979.

[24] B. Gavish and S. Graves. Scheduling and routing in transportation and distributions systems: Formulations and new relaxations. Working paper, Graduate School of Management, University of Rochester, Rochester, NY, 1982.

[25] B.L. Golden and A.A. Assad. *Vehicle Routing: Methods and Studies*. North-Holland, Amsterdam, 1988.

[26] B.L. Golden, E.A. Wasil, J.P. Kelly, and I.M. Chao. Metaheuristics in vehicle routing. In T.G Crainic and G. Laporte, editors, *Fleet Management and Logistics*, Kluwer, Boston, MA, 1998, pp. 33–56.

[27] E. Hadjiconstantinou, N. Christofides, and A. Mingozzi. A new exact algorithm for the vehicle routing problem based on q-paths and k-shortest paths relaxations. *Annals of Operations Research*, 61:21–43, 1995.

[28] R. Hayes. The delivery problem. Management Science Research Report 106, Carnegie Institute of Technology, Pittsburgh, PA, 1967.

[29] A. Langevin, M. Desrochers, J. Desrosiers, S. Gèlinas, and F. Soumis. A two-commodity formulation for the traveling salesman and the makespan problems with time windows. *Networks*, 23:631–640, 1993.

[30] G. Laporte. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59:345–358, 1992.

[31] G. Laporte. Vehicle routing. In M. Dell'Amico, F. Maffioli, and S. Martello, editors, *Annotated Bibliographies in Combinatorial Optimization*, Wiley, Chichester, UK, 1997.

[32] G. Laporte and Y. Nobert. Exact algorithms for the vehicle routing problem. *Annals of Discrete Mathematics*, 31:147–184, 1987.

[33] G. Laporte and I.H. Osman. Routing problems: A bibliography. *Annals of Operations Research*, 61:227–262, 1995.

[34] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys, *The Traveling Salesman Problem*. Wiley, Chichester, UK, 1985.

[35] J.K. Lenstra and A.H.G. Rinnooy Kan. Some simple applications of the traveling salesman problem. *Operational Research Quarterly*, 26:717–734, 1975.

[36] T.L. Magnanti. Combinatorial optimization and vehicle fleet planning: Perspectives and prospects. *Networks*, 11:179–214, 1981.

[37] S. Martello and P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*, Wiley, Chichester, UK, 1990.

[38] C.E. Miller, A.W. Tucker, and R.A. Zemlin. Integer programming formulations and traveling salesman problems. *Journal of the ACM*, 7:326–329, 1960.

[39] D.L. Miller. A matching based exact algorithm for capacitated vehicle routing problems. *ORSA Journal on Computing*, 7(1):1–9, 1995.

[40] C.E. Noon, J. Mittenthal, and R. Pillai. A TSSP+1 decomposition strategy for the vehicle routing problem. *European Journal of Operational Research*, 79:524–536, 1994.

[41] P. Toth and D. Vigo. Exact algorithms for vehicle routing. In T. Crainic and G. Laporte, editors, *Fleet Management and Logistics*, Kluwer, Boston, MA, 1998, pp. 1–31.

[42] P. Toth and D. Vigo. Models, relaxations and exact approaches for the capacitated vehicle routing problem. *Discrete Applied Mathematics*, to appear.

[43] D. Vigo. VRPLIB: A vehicle routing problem instances library. Technical Report OR/00/3, Università di Bologna, Italy, 2000.