

2. Background

In this chapter, we describe the planning process of a public transport company (Section 2.1) and, especially, the crew rostering problem in Section 2.2. In Section 2.3 we discuss some well-known optimization problems and techniques, which will be applied in the remainder of this thesis.

2.1. Planning Process of Public Bus Transport Companies

The complete planning process in public bus transit is very complex, therefore, it is traditionally divided into three phases: the strategic, tactical, and operational phases. Each phase can be further divided into several sub-problems, which are solved sequentially. The details about each phase can be found in Desaulniers & Hickman (2007). Figure 2.1, derived from Steinzen (2007), illustrates the relationship between the different problems in the planning process. In the *strategic planning* process, network design and line planning problems are solved based on a given original-destination (O-D) matrix. Each entry in the matrix represents the number of passengers travelling between any two points in the network by the time of day. The *network design* problem involves minimizing construction costs for determining the links in the network, while the line routes and their frequencies are determined in the *line planning* problem for the given transportation network. In both problems, the demand from passengers should be satisfied. The planning horizon of this stage is typically several years.

In the *tactical planning* process, line routes and their frequencies are given as input for the *timetabling* problem. In addition, the travel times along the lines and any potential layover times at stations are assumed to be known. The planning horizon for this problem is normally seasonal. In timetabling, timetables are determined, and a set of service trips with start and end locations and times are given for the next step: *operational planning*.

In *vehicle scheduling*, vehicles are assigned to service trips, resulting in vehicle blocks. After defining a sequence of tasks in each vehicle block, each task must be assigned to one duty that can be performed by a single anonymous crew (driver) for one day. This process is defined as *crew scheduling*. The planning horizon for both vehicle and crew scheduling is usually one day. Each generated duty must satisfy some work regulations. Examples of work regulations are maximum length of a duty, maximum driving time without a break in a duty, minimum break time during a duty, etc. Each generated duty has the following properties:

- a start time, when it begins,
- an end time, when it ends,
- the duration, which is equal to end time minus start time,

- a depot, where it begins and ends,
- a vehicle type it belongs to,
- the paid time, which is possibly not equal to the duration — this is calculated depending on the duty type,
- a shift type, which depends on its start and end time; for example, one duty beginning between 3 am and 6 am is an early shift, and
- the calendar date or day of the week that the duty belongs to.

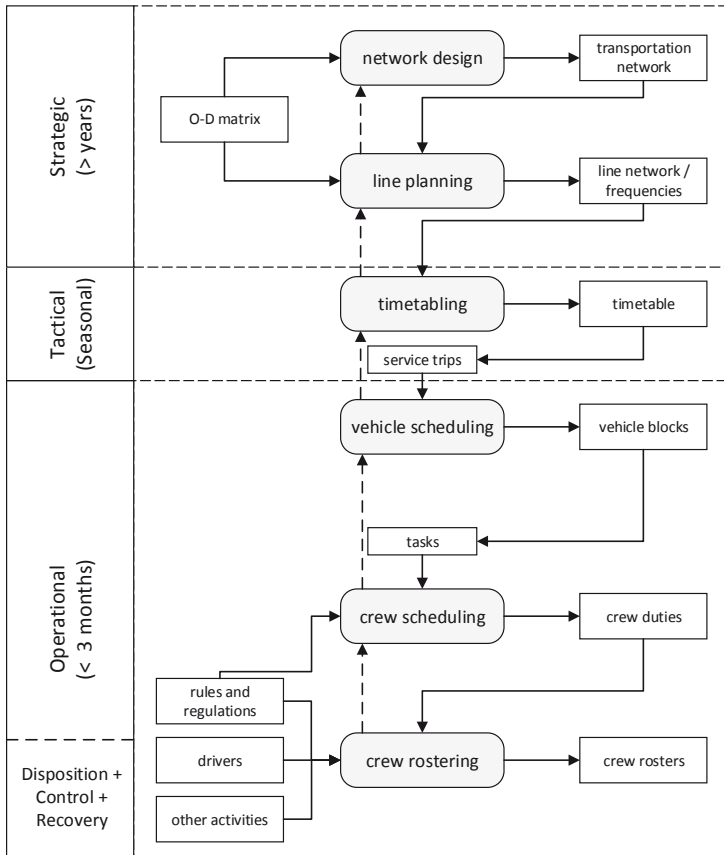


Figure 2.1. The sequential planning process in public bus transit.

The generated duties, as well as some other given activities, such as standbys, are required to be covered by drivers. Driver assignment occurs in the last step of the operational planning process in the *crew rostering problem*, which is solved on a monthly or semi-monthly basis. Law and labor union rules, as well as personal preferences of drivers, are considered during the assignment. The resulting schedule for each driver or group of drivers is called a *roster*. The details of the input information about activities and drivers, and the rules and regulations of the crew rostering problem, can be found in Section 2.2.1.

On the day of operations, the schedules generated in the operational planning process are seldom operated exactly as planned due to frequent disruptions, such as incidents, variations in traffic conditions, vehicle breakdowns, and absences of drivers. Therefore, operations control or recovery techniques should be applied for these problems.

Crew scheduling and rostering problems are becoming more and more important in the planning process at a public bus company since crew costs mostly outweigh vehicle costs (see Bodin et al. (1983) and Leuthardt (1998)). Compared with crew scheduling problems in the planning process, crew rostering has received much less attention in the academic literature. One reason is that most of the cost benefits can be achieved in crew scheduling for generating duties. However, the generation of duties in crew scheduling does not include any information about drivers (anonymous duties). The assumption made in solving this problem is that all crews are equal. Therefore, it causes difficulties generating schedules of drivers in the crew rostering problem, since the drivers are individual people with different qualifications and preferences. It is possible that some duties cannot be covered on some days, while some drivers do not get any jobs on some other days. In crew rostering, the minimization of operational costs is still important; moreover, drivers' preferences are considered as well. Rosters which are generated by considering the desires of drivers bring higher acceptance than rosters that ignore individual wishes (see Hanne et al. (2009)). That means fewer exchanges, and less absence in operational days. Therefore, fewer recovery activities are expected, which implies lower operational costs. Besides that, the even distribution of workload is considered an important objective, which induces fewer payments for overtimes. For these reasons, the crew rostering problem becomes very difficult to solve, and more complex when dealing with large real-world problems.

2.2. Crew Rostering Problem

In this section, we first give details about the input information for solving the crew rostering problem. After a definition of cyclic and non-cyclic crew rostering problems, a short description of sequential and integrated approaches follows.

2.2.1. Input Information

Different kinds of input information for solving the crew rostering problem, including information on duties, other activities, drivers, and rules and regulations, are shown below.

Activities

Not only must the duties and shifts, which are described in Section 2.1, be assigned to drivers, but also some other activities, such as standbys, days off, leaves, and training

periods. *Days off* consist of a couple of rest days between working days. *Standby* activities are planned to cover the absences of drivers, while *leaves* are vacations. The training periods and leaves are preassigned and fixed for each driver and cannot be changed in the optimization. Their distribution is decided by bus companies and drivers.

Driver information

The data about drivers includes not only the depot (where a driver begins and ends) and vehicle types (depending on the capacity, speed, or equipment of each vehicle), but also the number of days off, the target working hours for the current planning period, the target number of standby activities, and the required training periods and leaves. All of these depend on the work contracts and the drivers' current work-accounts. The work-account of each driver includes their current overtime and their number of days off from previous periods. It describes the driver's credit, e.g. a driver with more overtime in previous periods can get more jobs with shorter working hours in order to reduce overtimes gradually. Additionally, the drivers can express their preferences, including their daily desired activities and their possible combination of activities. The daily desires of a driver mean that the driver wishes to get one activity on one day, while the possible combination of activities can be, for instance, similar shifts within a working week, and/or not to get an early shift after a day off. Hanne et al. (2009) explain the importance of considering employees' preferences in scheduling their working time; in particular, the advantages of assigning flexible working times to employees. They mention that the employees can better manage their work-life balances, which brings increased morale and reduced absenteeism. That means fewer exchanges and less absence in operational days, which implies lower operational costs, since fewer recovery activities are expected. Moreover, they assume that "happy staff means happy customers". Better services are expected.

Rules and regulations

The rules and regulations can be labor rules, some of which are imposed by the bus company, and others are due to the agreement between the bus company and employee unions. Every bus company can also define its own internal constraints that are stricter than those required by law. We consider three types of rules: horizontal rules, vertical rules, and quality rules. Horizontal rules are rules that depend only on one roster, while vertical rules combine the information among all rosters. Quality rules are horizontal rules — without these rules the generated rosters are still legal — but they affect the quality of the rosters. Compared with the rules in the airline sector discussed in Kohl & Karisch (2004), we have more complex horizontal rules to generate a feasible roster, but less complex vertical rules (since drivers work mostly alone).

Horizontal rules These rules consist of compatibility, working block, and days off block. The *incompatible* connections of activities are gathered in a list of forbidden sequences according to, for example, the working regulations. It is forbidden to follow an early shift with a late shift because of the short rest time between them. The forbidden sequences are provided by bus companies, and it is possible that a combination of activities that drivers

find undesirable is regarded as forbidden as well. The length of forbidden sequences is not limited to two; for instance, a late shift one day before a standby and an early shift one day later may be forbidden if they appear simultaneously on three consecutive days. Additionally, two other situations are also considered: *double-off* is defined as at least two consecutive days off, while a *single-off* is a single day off between two work-related activities. Many of the restrictions in this class can be implicitly considered during the network generation (see Section 5.1). The maximum consecutive numbers of working days and days off (*working/days off block*) are restricted based on working regulations. For example, the number of consecutive working days is limited to five, while the number of consecutive days off is restricted to three.

Vertical rules These rules consider restricted resources among all rosters. Not only are the limits on the assignment of the amount of duty/shift during each day considered, but also the limits on the available numbers of days off and standbys. The limit depends on the number of drivers and duties available on that day. The upper bound can be easily computed before optimization.

The horizontal and vertical rules described above are formulated as hard constraints in Section 5.2, while the following quality rules are formulated as soft constraints. They can be suspended by applying specific penalty costs in the objective to restrict the number of violations.

Quality rules These rules include preferences of the bus company and drivers, real operational costs, fairness among all drivers, and the robustness of the rosters.

Preferences: In this category, the preferences of the bus company as well as of the drivers are considered. As mentioned before, the daily desired activities and the possible combination of activities are given by drivers. Some alternatives can be provided by the bus company if the primary desires of the drivers cannot be satisfied. The number of alternatives and the penalty costs resulting from them can vary from one bus company to another (see Example 1).

Example 1. Bus companies A and B have four types of shifts (early1, early2, midday, late shift). Early2 (ES2) begins later than early1 (ES1) but earlier than midday. In both bus companies, 50% of drivers want to get early shifts (regardless of early1 or early2) and 40% of them want to get midday shifts (MS). Late shifts are unpopular. However, some drivers are required to cover the late shifts. Company A wants to maximize the number of satisfied drivers, while company B wants to minimize the total distance between the desired and assigned shifts for all drivers. Figure 2.2 shows the different functions for selecting alternatives (i.e. ES1, ES2, and MS) to cover late shifts. Company A will select the function on the left to make sure most of the drivers desiring MS get their wish fulfilled, while company B will select the function on the right to guarantee that most of late shifts are carried by the drivers desiring MS.

Additionally, a linear function as shown in Day & Ryan (1997) is also possible for calculating the penalty costs of alternatives. Due to the limited number of drivers that may simultaneously take a day off, some desired days off (on weekends) can not be satisfied, but can be moved to within a couple of days earlier or later. These are called *moved days off* and

moved days off on weekends. The numbers of these are limited due to their unpopularity. The number of single-off activities is often restricted due to drivers' preferences. The distance between two double-off activities is defined in the company rules to make sure, for example, that a driver gets at least two doubles-offs within 16 days.

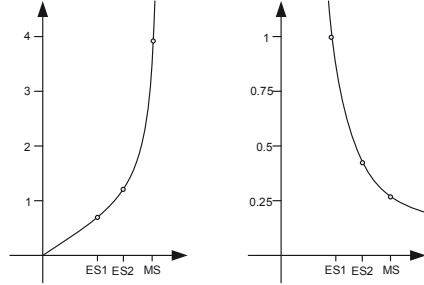


Figure 2.2. An example of different functions of the penalty costs for the alternatives of late shifts.

Real costs: The insufficient assignment of standbys and duties/shifts might cause additional personnel costs or additional costs for the overtimes of all planned drivers. The same is true for open days, on which drivers do not get any activities. Therefore, the number of open days should be minimized. An additional situation that should be avoided is one where, for example, some drivers work overtime while others work substantially less than the regular working time. This can be avoided by minimizing the maximum overtime for all drivers, which results in less payment for overtime.

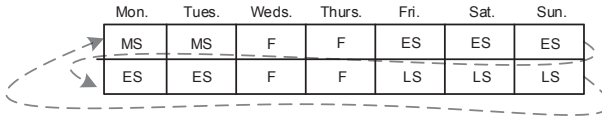
Fairness: Besides the above mentioned fair distribution of working times, insufficient days off for each driver will impact the fairness of the days off distribution for all drivers. Moreover, the distribution of unpopular activities is embedded in the preferences of drivers (i.e. the desirability of an activity on one particular day).

Robustness: Sometimes in operational days disruptions occur, such as illness. Therefore, a set of standbys should be planned to cover for absences, but the number of these should be minimized to avoid unused standbys. In this work, the number of standbys per day is defined by the bus company, based on experience. The optimal planning of the assignment of standbys based on historical statistics of illness is discussed in Xie et al. (2012b).

Moreover, we need to consider the previous planning period, which is assumed to be fixed for the current planning period. Some other fixed activities are also considered in the current planning period, such as a training period and annual leave. We assume that all fixed activities comply with the horizontal and vertical rules. However, the horizontal rules should be checked between fixed and unfixed activities. The details will be shown in Section 5.1 together with the example shown in Figure 2.4 in the next sub-section.

2.2.2. Cyclic and Non-Cyclic Crew Rostering

There are several ways to generate a roster. A *cyclic roster* is generated for a group of drivers who have the same qualifications and similar preferences. Such a roster includes several rows of duties from Monday to Sunday. The number of rows is equal to the number of drivers in the group. All drivers within a group use the same roster but begin with different rows. An example is shown in Figure 2.3, in which different activities, ES (early shift), MS (midday shift), LS (late shift), and F (day off), are assigned to two drivers.



Mon.	Tues.	Weds.	Thurs.	Fri.	Sat.	Sun.
MS	MS	F	F	ES	ES	ES
ES	ES	F	F	LS	LS	LS

Figure 2.3. An example of a cyclic roster.

In cyclic rostering, the number of days in the planning period is equal to the number of drivers multiplied by seven. Each week is assigned to each driver in such a way that each weekly pattern is worked in parallel with a person. Cyclic rostering is a rather simple way to generate rosters, because instead of generating a roster for each driver a roster is generated for a group of drivers. Furthermore, the roster is fair since drivers within a group have the same duties, including unpopular duties, and the days off and weekends off are evenly distributed. However, this fairness is only achieved if absences and other occurrences do not arise in real life, otherwise the cycles are destroyed. Moreover, a cyclic roster considers duties from days of the week, not calendar dates. Therefore, it is not flexible enough to respond to changes in traffic, such as an increase in traffic on holidays. Moreover, a new employee can not be easily added to an existing group, since the planning horizon will be changed and one more week must be planned.

The shortcomings of cyclic rostering can be avoided if an individual roster is generated for each driver within a given time period (e.g. two months). Such a *non-cyclic roster* may consider wishes with respect to a special single day off or vacation periods. Non-cyclic rostering provides more freedom to take holidays and special events into account. Figure 2.4 shows an example of a two-week, non-cyclic roster for two drivers, in which the first five days are in the previous period. As mentioned before, the horizontal rules between fixed and unfixed activities must be enforced. We assume that the maximum consecutive number of working days is restricted to five. The five fixed activities in the last period for driver *d2* are work-related activities. Therefore, a day off activity should be assigned to the first day of the planning period (day 6 in the example). Additionally, we assume the maximum length of a days-off period is three days. A working day is therefore required for *d1* on day 6. We assume that the distance between two double-off activities from the previous period to the current one is defined to be seven. The last double-off in the previous period of *d1* is on 5th day, and a double-off is required within the next

seven days, such as on days 10 and 11. Such restrictions have been used to reduce the complexity of our network design introduced in Section 5.1. Note that the previous period is implicitly considered in the cyclic roster, if the roster is not disrupted.

	Day1	Day2	Day3	Day4	Day5	Day6	Day7	Day8	Day9	Day10	Day11	Day12	Day13	Day14
d1	MS	MS	F	F	F	ES	ES	ES	ES	F	F	LS	LS	LS
d2	ES	ES	MS	MS	MS	F	F	MS	MS	MS	MS	ES	F	F

Figure 2.4. An example of a non-cyclic roster.

The non-cyclic rostering approach is widely used in the airline industry, especially in Europe (see Kohl & Karisch (2004), Capanera & Gallo (2004) and Maenhout & Vanhoucke (2010)). In recent years, this way of generating rosters has also been investigated in railways (see De Pont (2006) and Hanne et al. (2009)). In *bidline rostering*, which is used mostly in North American airlines, a roster is generated subsequently for each individual crew member in decreasing order of seniority in the crew rank (see Kohl & Karisch (2004) for an overview). In the rest of this thesis, the cyclic crew rostering is called CCR and non-cyclic crew rostering NCCR.

The crew rostering problem in transportation is different from the ones in other applications, since it deals with trips/flights. The rostering of call centers, police services, or airport ground staffs deals with flexible demand, while the rostering of nurses and ambulance services deals with shift-based demand. More details can be found in Ernst et al. (2004).

2.2.3. Sequential vs. Integrated Approach

The crew rostering problem is usually divided into several sequential sub-problems due to its high complexity (see Moz & Pato (2007) for the nurse rostering problem). A roster is considered as a schedule of combinations of work-related activities (standby, shift/duty) and day off activities (single-off, double-off). The sub-problems are *days off scheduling* (the optimal distribution of days off by considering fair distribution of days off), *shift assignment* (the allocation of shifts to drivers), and *duty sequencing* (according to the result of shift assignment, duties are assigned to drivers). The problem of integrated days off scheduling and shift assignment is named *rota scheduling* in Emden-Weinert et al. (2000) and Xie et al. (2012b). In this thesis, the integrated shift assignment and duty sequencing problem is called *shift-duty assignment*.

In order to understand the drawbacks of the sequential crew rostering problems, for example the approach we use in this thesis, with rota scheduling first and duty sequencing second, two examples will be shown below. An *early shift* (ES) means a duty begins between 0:01 am and 11:59 am, while a duty with the *midday shift* (MS) begins between 12:00 noon and 8:59 pm. Each *late shift* (LS) duty begins between 9:00 pm and 12:00 midnight. The minimum rest period is defined as 12 hours.

Example 2. We assume that the sequence (MS, ES) is selected in the rota scheduling problem. The rush hours in the morning are, for example, between 6:00 and 8:00 am, and in the evening between 4:00 and 7:00 pm. However, a duty with MS that begins at 3:00 pm (shortly before rush hour in the evening) and ends at 10:00 pm cannot be followed by any duties in the next day with ES, which begins before 10:00 am. That means, if we allow the sequence (MS, ES) in the rota scheduling problem, however, many MS duties that begin after the rush hour in the evening can not be followed by any ES during the rush hour in the morning. Therefore, the selected sequences (MS, ES) in the rota scheduling can not be assigned by most duties in duty sequencing. This results in many empty days for the employees, and more duties remain unassigned. However, some of them could be assigned, if we consider the duties in rota scheduling. Such an approach is called *integrated crew rostering*.

Example 3. We assume that the sequence (LS, MS) is forbidden in the rota scheduling problem, due to violating overnight rest time. However, some duties of LS, such as ending after 4:00 am, can take the duties of MS on the day beginning after 4:00 pm. Therefore, they are not needed in the list of forbidden sequences of shifts. The check for a violation can occur directly with the duties in the integrated crew rostering.

In order to avoid the drawbacks of the sequential approach in practice, it is standard practice to define more and more shift types, such as defining 14 shift types instead of 4. However, this brings more complexity, and some duties are still left unassigned. Details will be shown in the results chapter (see Chapter 6).

2.3. Mathematical Background

The crew rostering problem addressed in this thesis is often formulated using mathematical optimization models. In this work the crew rostering problem is formulated as a mixed-integer problem and a network flow problem. Therefore, the necessary background of the both problems is provided, including the relevant combinatorial optimization techniques.

2.3.1. Linear and Mixed-Integer Programming

A *mixed-integer program* (MIP) is formulated as follows.

$$\min \quad c^T x \quad (2.1)$$

$$s.t. \quad Ax \geq b \quad (2.2)$$

$$x_i \in \mathbb{Z}, \forall i \in \mathcal{I} \quad (2.3)$$

$$x_j \in \mathbb{R}, \forall j \notin \mathcal{I} \quad (2.4)$$

where x is a vector of n decision variables, while $c \in \mathbb{R}^n$ is a vector of objective coefficients. Let $b \in \mathbb{R}^m$ be a vector of upper bounds on constraints. A is defined as an $m \times n$ matrix of constraint coefficients. The set $\mathcal{I}$ defines the set of variables that take integer values. If $\mathcal{I} = \emptyset$, then the problem is a *linear program* (LP). However, if $\mathcal{I} = \{1, 2, \dots, n\}$, then the problem is an *integer program* (IP). A special case, a 0-1 integer program, is called a *binary program*. Unlike LPs, which can be solved in polynomial time, MIPs and IPs are

NP-hard (see Wolsey (1998)). There are several efficient algorithms to solve LP problems, such as the interior point method described in Karmarkar (1984), and the simplex method described in Dantzig et al. (1955).

The crew rostering problem in this thesis is formulated as an MIP problem (see Section 5). Solution approaches for MIPs are well studied; a number of solvers are commercially available to solve MIPs, such as the ones used in this thesis, the CPLEX and Guorbi solvers (see IBM Corporation (2014) and Gurobi Optimization (2014)).

2.3.2. Network Flow Problem

A network is defined as an acyclic graph with only one source and sink. Network flow problems are widely used in operations research, which use the models and methods of optimization together with graph theory. In this section two related network flow problems are described, which will be used in the remainder of the thesis, namely the minimum cost flow, and multicommodity cost flow problems.

Minimum cost flow problem The *minimum cost flow problem* (MCFP) is a very fundamental network flow problem, since many other well-known network problems are special cases of this problem, such as the multicommodity flow problem, which will be described later in this subsection. The MCFP is also known as the transshipment problem. The goal of this problem is to find a minimum cost shipment of one commodity through a network, such that the demands at certain nodes are satisfied from available supplies at other nodes (see Ahuja et al. (1993)).

Define $\mathcal{G}=(\mathcal{N},\mathcal{A})$ as a directed graph with $\mathcal{N}$ as the set of nodes and $\mathcal{A}$ as the set of directed arcs. Let c_{ij} be the cost per unit flow on the arc $(i,j) \in \mathcal{A}$. Moreover, the cost varies linearly with the amount of flow. Besides that, an integer upper and lower bound u_{ij} and l_{ij} on the amount of flow are defined for each arc $(i,j) \in \mathcal{A}$. For each node $i \in \mathcal{N}$ an integer b_i is referred to as the property of this node; if b_i is positive or negative, then node i is a supply or demand node, respectively. Otherwise, node i is a transshipment node. Let a decision variable x_{ij} be the amount of flow on arc $(i,j) \in \mathcal{A}$. The mathematical model is formulated as follows.

$$\min \sum_{(i,j) \in \mathcal{A}} c_{ij}x_{ij} \quad (2.5)$$

$$s.t. \quad \sum_{j:(i,j) \in \mathcal{A}} x_{ij} - \sum_{j:(j,i) \in \mathcal{A}} x_{ji} = b_i \quad \forall i \in \mathcal{N} \quad (2.6)$$

$$l_{ij} \leq x_{ij} \leq u_{ij} \quad \forall (i,j) \in \mathcal{A} \quad (2.7)$$

The objective function (2.5) minimizes the total costs such that the total outflow minus the total inflow in each node is equal to its demand/supply (see constraints (2.6)). Furthermore, the constraint set (2.7) indicates that the flow on each arc is between the lower and upper bounds of that arc. The constraint set (2.6) is also called the *flow conservation/required flow* constraint while the constraint set (2.7) is a *flow capacity* constraint.

Decision Support for Crew Rostering in Public Transit
Web-Based Optimization System for Cyclic and
Non-Cyclic Rostering

Xie, L.

2015, XIX, 172 p. 79 illus., 5 illus. in color., Softcover

ISBN: 978-3-658-08166-9