TACTICAL CREW PLANNING AT TURKISH STATE RAILWAYS

by

BİROL YÜCEOĞLU

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University

August 2009

TACTICAL CREW PLANNING AT TURKISH STATE RAILWAYS

APPROVED BY:

Assist. Prof. Güvenç Şahin, (Thesis Supervisor)

. . . .

Assist. Prof. Fatma Tevhide Altekin

. . . .

Assoc. Prof. Ş. İlker Birbil

. . . .

Assist. Prof. Kerem Bülbül

. . . .

Assist. Prof. Dilek Tüzün Aksu

. . . .

DATE OF APPROVAL: . . . .

TACTICAL CREW PLANNING AT TURKISH STATE RAILWAYS

Birol Yüceoğlu

Industrial Engineering, Master of Science Thesis, 2009

Thesis Supervisor: Assist. Prof. Güvenç Şahin

Keywords: crew planning, tactical planning, railroad, network flow, space-time network

Abstract

Tactical crew planning problem at Turkish State Railways (TCDD) involves finding the minimum crew capacity in a crew region required to operate a predetermined set of train duties assigned to the region by the headquarters. The problem is to be solved for each crew region by satisfying various rules and constraints associated with the requirements of the company. One of the most important constraints is the day-off requirement which makes the problem computationally intractable. In this study, we use a space-time network flow representation to solve the tactical capacity planning problem with day-off requirement. To solve the problem, we develop two solution approaches: the sequential approach and the integrated approach. In the sequential approach we mimic the current practice at TCDD and solve the problem in two stages. In the first stage, we solve a minimum flow problem over a space-time network by relaxing the day-off requirement. After obtaining the tentative schedules of crew members, we solve an assignment problem to fill-in the days-off in the tentative schedules by using additional substitute crew members. In the integrated approach, we solve a minimum flow problem with side constraints using a layered space-time network representation of the problem. We present the computational study on a real-life data set acquired from TCDD. We, then, study a higher level crew capacity planning problem. In this tactical-to-strategic level capacity planning problem, we minimize the total crew capacity of all regions by simultaneously considering multiple regions. We do this by re-assigning duties to different regions or allowing two neighboring crew regions share the train duties in different settings. For tactical-to-strategic capacity planning problem, we present the mathematical formulation of single-region and two-region models with various crew exchange policies. Given the large scale of the search space, we choose to employ a neighborhood search heuristic in order to solve the problem. The neighborhood search heuristic uses the minimum flow problem in the sequential approach as a subprocedure. We present the computational study again for the TCDD data set.

DEMİRYOLLARINDA TAKTİK SEVİYEDE EKİP PLANLAMA PROBLEMİ

Birol Yüceoğlu

Endüstri Mühendisliği, Yüksek Lisans Tezi, 2009

Tez Danışmanı: Yard. Doç. Dr. Güvenç Şahin

Anahtar Kelimeler: ekip planlama, taktik seviyede planlama, demiryolu, çizge akış, uzay-zaman çizgesi

Özet

Türkiye Cumhuriyeti Devlet Demiryolları'nda (TCDD) taktik seviyede ekip planlama problemi, ekip bölgelerinin önceden genel merkez tarafından belirlenmiş çizelgelerindeki görevleri gerçekleştirecek enküçük ekip üyesi sayısının belirlenmesini kapsar. Problem, şirketin gereksinimleriyle ilgili kural ve kısıtları sağlayarak her bölge için ayrı ayrı çözülür. Bu kısıtlardan en önemlilerinden biri de problemin çözümünü zorlaştıran tatil günü kısıtıdır. Bu çalışmada, taktik seviyede kapasite planlama problemini tatil günü kısıtıyla çözmek için bir uzay-zaman çizgesi gösterimi kullanıyoruz. Problemi çözmek için iki çözüm yaklaşımı geliştiriyoruz: ardışık yaklaşım ve tümleşik yaklaşım. Ardışık yaklaşımda halihazırda TCDD'de kullanılan çözüm yaklaşımını taklit ederek problemi iki aşamada çözeriz. İlk aşamada bir uzay-zaman çizgesi üzerinde tatil günü kısıtını göz ardı ederek bir enküçük akış problemi çözeriz. İkinci aşamada ise, bu problemin çözümünden elde edilen kesin olmayan ekip çizelgeleri üzerine yedek ekip üyeleri kullanarak tatil günlerini doldurmak için bir atama problemi çözeriz. Tümleşik yaklaşımda ise katmanlı bir uzay-zaman çizgesinde yan kısıtları olan enküçük akış problemi çözeriz. Bilgisayısal çalışmamızı TCDD'den alınan gerçek veri kümeleri üzerinde sunuyoruz. Taktik seviyede ekip planlama probleminden sonra daha üst seviyede bir ekip kapasite planlama problemini ele alıyoruz. Bu taktik-stratejik arası seviyedeki kapasite planlama probleminde birden çok bölgeyi aynı anda düşünerek bölgelerde ihtiyaç duyulan toplam ekip üyesi sayısını enküçükleriz. Bunu gerçekleştirmek için, görevlerin farklı bölgelere atanmasını ve bölgeler arası görev paylaşımı seçeneklerini değerlendiririz. Tek bölgeli ve iki bölgeli taktik-stratejik arası seviyedeki kapasite planlama problemlerinin farklı görev paylaşım kurallarıyla matematiksel modellerini sunuyoruz. Problemin büyük ölçekli olması sebebiyle açgözlü bir yöre araması algoritmasıyla problemi çözeriz. Açgözlü yöre araması algoritmasında ardışık çözümde geliştirdiğimiz enküçük akış problemini altyordam olarak kullanırız. Bilgisayısal çalışmayı yine TCDD veri kümesi üzerinde sunuyoruz.

Acknowledgements

I would like to express my gratitude to my thesis advisor, Dr. Güvenç Şahin, for his support, inspiration, and for tirelessly helping me. I would like to thank Dr. Ş. İlker Birbil for showing me the way and for his support. I would also like to thank Dr. Tevhide Fatma Altekin, Dr. Kerem Bülbül, and Dr. Dilek Tüzün Aksu for guiding me with their invaluable comments.

I would like to thank TÜBİTAK for the financial support I received during my studies.

Finally, I am grateful to everything that thrills me, my office mates for the pleasing atmosphere they created, Emre Dinçtürk for always being there when I turned my back, Cpt. Emre Köse for all those small rituals, and last but not least my family for just being there.

TABLE OF CONTENTS

List of Tables

List of Figures

1 INTRODUCTION

Transportation services planning is one of the most prominent and important research areas in operations research due to complexity of the problems arising in different modes of transportation such as airlines, railways and maritime. In the recent years, a vast variety of optimization problems relating to many aspects of operations of transportation companies are being studied extensively. For European railways, deregulation process in the 1990s pushed the companies to seek efficiency in their operations while the North American freight railway systems have started adapting operations research techniques in a need for effective decision making through their transition from tonnage-based services to scheduled services. The increasing level of competition with trucking companies as well as promotion of intermodal services have started elevating the level of such efforts.

Due to the complexity of the physical infrastructure and interrelated operations, there are several problems arising in railways. These problems are concerned with network configuration decisions (building new tracks, expanding line capacities, location of yards as well as their closing and opening), capital investment decisions (acquiring rolling stock, technological upgrades), allocation of resources, service scheduling, demand and revenue management to name a few. We refer the interested reader to Şahin [1] and Ahuja *et al.* [2] for an overview and a general framework of the railway planning problems. In this study, we are interested in crew-related planning problems at railways.

To understand the share of crew related costs in railways, it is sufficient to look at a few examples. According to the Dutch Railways example in Abbink *et al.* [3], there are more than 3000 drivers covering more than 14,000 timetabled daily trips put together in more than 1000 duties in 29 crew regions. A new planning system put into use in Dutch Railways is predicted to bring 6 million euros reduction in crew related costs, which corresponds to a 2% improvement. In Turkish State Railways (TCDD), crew-related costs (including managerial crew, which accounts for a small percentage of crew-related costs) constitute more than one third of general expenditures of the company during the time period from 1999 to 2003 even surpassing energy expenditures (Turkish State Railways [4]). Therefore, effective planning and efficient use of crew resources may lead to important savings.

The crew planning problem can be studied at all three levels of decision making processes in TCDD:

(i) The operational crew planning problem deals with managing the daily operations; the planning horizon is very short, usually a week. The final assignment of crew members to duties is determined at this level. The operational crew planning is concerned with different types of crew costs such as time-based compensations due to train duties, station duties, deadheading (transfer of crew members), and compensations due to disturbances (e.g. time spent away from the home station, assignment of inconvenient duties, overtime working hours, and rest periods exceeding a predetermined time). Assignment of duties by balancing both the workload and payments among the crew members is also considered at this level.

(ii) At the tactical level, the crew planning problem determines the number of crew members required in a region to operate the train schedule that is under the responsibility of the region, i.e. the minimum capacity requirement of the region. As the fixed crew salaries constitute a significant portion of crew-related costs in railway companies, the number of crew members under long-term contracts is an important decision to make. Yet, individual wages (including both the salaries and compensations) are usually ignored at this level according to the current practice.

(iii) At the strategic level, crew capacity planning problem is concerned with the system-wide capacity determination by integrating the regional problems with each other. Several aspects and environmental parameters of the problem that are considered as given at the regional level are questioned in order to see if any modifications in these parameters yield any substantial system-wide changes. These aspects include the number of regions, stations to be selected as regional bases, allocation of train duties among the regions, and locations of crew exchange stations. In addition, the parameters of the capacity planning problem, which are governed by the rules and policies of the railway company and by the regulations imposed by the labor unions, may also be evaluated. These parameters include restrictions on the duration of duty periods, overtime limits and deadheading rules. By modifying such parameters of the problem, railways may experience substantial reduction in crew-related costs.

In this study, we focus on the tactical and tactical-to-strategic level crew capacity planning problems.

Currently, TCDD is in the process of expanding its operations with the construction and planning of several high speed train lines. As a result of this expansion, the use of operations research techniques in crew planning becomes a necessity in the company as well as in other planning problems. Previously being ignored by the company, effective crew resources management may bring substantial improvements at all three levels of their decision making processes. In that respect, especially tactical and strategical crew capacity planning come into prominence as fixed crew costs are important cost measures in transportation systems. In addition, an optimal solution to the tactical level crew planning problem provides an input to the operational level problem as the number of available crew members in a region.

Our main contributions in this work are summarized as follows:

- We develop a space-time network representation for the crew capacity planning problem at TCDD.
- We develop two approaches to solve the crew capacity planning problem, particularly with the complicating day-off requirement constraint:
  - In our sequential approach, we solve the problem in two stages. At the first stage, we solve a minimum flow problem without considering the day-off requirement. At the second stage, we satisfy the day-off requirement by assigning some of the duties in the schedules obtained at the first stage to additional substitute crew members.
  - In our integrated approach, we formulate the problem on a layered version of the space-time network. Solving the corresponding integer programming formulation of the network flow problem, we obtain tentative crew schedules that comply with the day-off requirement.
- We formulate tactical-to-strategic level planning problems that consider the optimal allocation of train duties among the regions within a system-wide multiregional planning environment.
- We propose a solution method for the tactical-to-strategic crew capacity planning problem by using the minimum flow problem formulation of the tactical level problem as a subprocedure of a heuristic algorithm.

In Chapter 2, we define the tactical level crew capacity planning problem in detail and develop a space-time network representation of the problem along with the review of related studies in the literature. Chapter 3 follows by formulating the network flow problems of two different solution approaches to the tactical level planning problems. In Chapter 4, we present a computational study with real-life data in order to compare different solution approaches. Chapter 5 is dedicated to some tactical-to-strategic level planning problems; mathematical formulations of these problems as well as an efficient heuristic algorithm to solve them are proposed with a proof-of-concept computational study. Finally, Chapter 6 concludes with a summary of the thesis and some remarks on future research particularly on the strategic level planning problems.

2 PROBLEM DEFINITION AND NETWORK REPRESENTATION

Crew planning problems at railways have been studied several times for various environments considering particular railway companies, and nation-wide or region-wide systems. For instance Caprara *et al.* [5, 6, 7] focus on the Italian case whereas Vaidyanathan *et al.* [8] focus on the North American railways. There are, indeed, several commonalities among these problem environments due to universally accepted rules, regulations and labor union laws in addition to similar infra-structural properties. Therefore, problem definitions in different studies are found to be very close to each other. In the later sections, we mostly study the problem through the TCDD case and generalize it to the best of our ability. With this approach, we facilitate the further discussion in a clear and succinct way, but we also note that it is either straightforward or fairly easy to generalize the techniques and outputs of this study for other railway companies and nation-wide railways.

2.1 Problem Definition and Problem Environment

TCDD has eight crew-base regions. Each region has a central *home station* and is responsible for providing and managing the crew resources to operate a predetermined set of trains. The home station both plans and executes the crew-related operations. For each crew-base region, there is a predetermined set of stations called *away stations*. The away stations of a crew region can be either the home stations of other (usually neighboring, but not necessarily) crew-base regions or *intermediate stations* located between two home stations. In general at TCDD, a home station is connected with a railway line, also called as *rail-corridors* to home stations of neighboring regions. The so-called intermediate stations are the stations along these corridors (they can be considered to define *shadow-borders of the regions.*).

The predetermined list of trains for a crew-base region imply a set of train duties either starting or ending at the home station:

(i) For train duties starting at the home station, the crew-base is responsible for operating the train until it arrives at the away station. Yet, these trains may continue traveling to other stations. In this case, the train duty starts at the home station and ends at an away station.

(ii) For train duties ending at the home station, the crew-base is responsible for operating the train after it departs from the corresponding away station. Similarly, these trains may have started their journey at some station earlier than the away station and may continue traveling to other stations. In this case, the train duty starts at an away station and ends at the home station.

Therefore, at the away stations either the train journey starts (ends) or the crew replaces (is replaced with) the crew of other regions. Figure 2.1 shows an example of the region structure at TCDD with a home station H_0 at the center and four other home stations (H_1, H_2, H_3, and H_4) connected to H_0 with corridors ρ_1, ρ_2, ρ_3, and ρ_4, respectively. For some of these corridors, trains assigned to H_0 operate to and from an intermediate station, like in corridors ρ_1 and ρ_2 where intermediate stations I_1 and I_2 are used as crew exchange stations. The stations I_1, I_2, H_3, and H_4 constitute the shadow-borders of region H_0.

We focus on the planning problems related to a crew-base region with a single home station and multiple away stations. There are indeed multiple crew types such as the machinist, the conductor and the train attendant. Yet, the problem can be defined and solved for each crew type independently with respect to different rules and policies that applies to the particular type of crew.

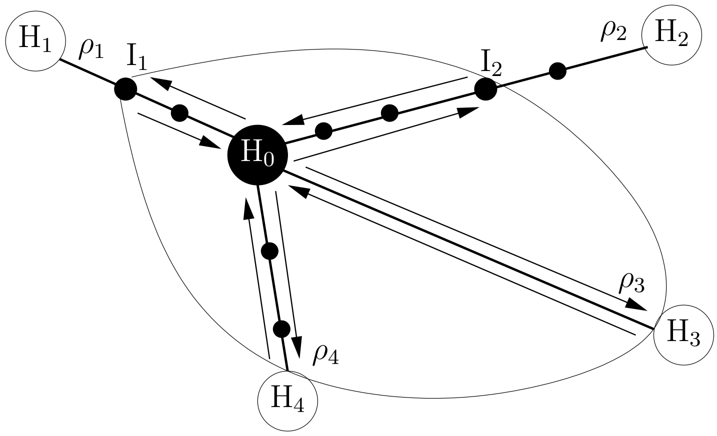

Figure 2.1: Region structure at TCDD.

The tactical level crew capacity planning problem of a particular crew type for a crew-base region determines the minimum number of crew members of that type required to operate the predetermined list of trains of the crew-base region. We consider a finite-length planning horizon that repeats itself periodically with respect to the schedules of trains. We assume that all crew members are at their home station at the beginning of the planning horizon, and each crew member has to end its duties at the home station at the end of the planning horizon. Accordingly, we assume that crew members also undertake duties for a repetitive periodic schedule although this assumption can be easily ignored at the operational level. Currently, TCDD uses a weekly schedule in which there are multiple copies (duplicates) of a train duty on different days of the week that are not necessarily every day of the week. Therefore for the TCDD case, the weekly train schedule is used in order to determine the minimum sufficient crew capacity at a base.

In addition to weekly train duties, there is another type of duty called *station duty*. A station duty is an 8-hour shift covered by a single crew member at the home station (in order to handle contingencies such as sickness of a crew member or a no

Parameter	Value at TCDD (hours)
On-duty time	1
Off-duty time	1/2
Double manning time	8
Excess duty time	12
Minimum home rest time	16
Maximum home rest time	48
Minimum away rest time	8
Maximum away rest time	24
Minimum deadhead start time	4
Maximum deadhead start time	24

Table 2.1: The environmental parameters of the problem at TCDD.

show-up). There are three shifts of station duty on a day: from 00:00 to 07:59, from 08:00 to 15:59, and from 16:00 to 23:59. In the subsequent parts of this thesis, we use the term *duty* to refer to both train duties and station duties unless we explicitly state one of them.

While solving the crew-related planning problems, there are several rules, regulations and policies we need to account for. Table 2.1 shows a list of parameters governed by the rules and policies of the company and the labor unions. According to the policies of TCDD, a crew member must report for a train duty an hour earlier than the departure of the train and can only finish his duty 30 minutes later than the arrival of the train. These time windows, respectively called *on-duty* and *off-duty* times, are used for filling paperwork and debriefs on the trip. If a train duty takes more than 8 hours, then there must be at least two crew members operating that train which is imposed by the double manning time. A crew member reaching the home station after completing a duty must take a *rest* of at least 16 hours and at most 48 hours. After 48 hours, the crew member must be assigned to another duty. A crew member reaching an away station by covering a duty has three options:

(i) The crew member can take an *away rest*. The away rest time must be at least

8 hours and must not exceed 24 hours.

(ii) If the crew member departing from the home station can return back to home in 12 hours by covering a second train duty, this is called an *excess duty*; the rules of away rest does not apply.

(iii) The crew member at an away station can be transfered to the home station without covering a duty. Transferring crew members to another location on trains covered by other crew members is called *deadheading*.

Usually in transportation companies, deadheading can be executed by other modes of transportation such as taxi, bus, or even airplane. In TCDD, the deadheading policy is slightly different; crew members are only allowed to deadhead by train. There are two types of deadheads:

(i) The first type of deadhead is *from away to home*. A crew member reaching an away station can take a short rest of 4 to 24 hours and ride on a train destined to his home station for deadheading. When there are multiple trains destined for the home station of the crew member, the crew member must take the one that will bring him back to his home station as early as possible after spending at least 4 and at most 24 hours at the away station.

(ii) The second type of deadhead is *from home to away*. In this case, the company sends multiple crew members from home station to an away station on a train in order to cover train duties from away to home that would be impossible to cover otherwise. For example, if there is a single train going from home to a particular away station and there are several trains from that away station to home after this train in the same period, the only option for covering such trains is by transferring more crew from home to away on the earlier trains.

The crew members deadheading from home to away do not necessarily have to be working through that duty, but they are subject to the same constraints as covering a duty when we evaluate their rest, direct connection, and deadhead options. In practice, this can be considered as a deadhead.

Deadheading of crew members has two benefits for the company. It avoids infeasibilities by sending multiple crews to away stations when coverage of multiple trains from that away station to the home station is not possible. Furthermore, by sending crew back to home station with the earliest opportunity, deadheading also minimizes the inefficient use of the workforce and helps the company deploy crew resources more efficiently.

In addition to the operational policies mentioned above, TCDD has another policy that makes the crew planning problem harder to solve. According to the company rules, each crew member has to take one day-off assuming a weekly planning horizon. This day-off should be spent at the home station and must be a whole day (from 00:00 to 23:59) and not any 24-hour period. This requirement is one of the major challenges that motivate our study and will be discussed later in detail.

2.2 Network Representation

In order to formulate the tactical crew scheduling problem for TCDD, we adapt the network representation in Vaidyanathan *et al.* [8] but modify it substantially in order to impose different policies and practical considerations applied in TCDD.

A space-time network should be defined with a set of nodes and a set of arcs. The nodes represent events in the space-time network and have two attributes: space and time, respectively representing the place (i.e. the station) and the time of the event. The beginning of a duty or the end of a duty are two examples for events that define a node. Arcs in the space-time network are directed arcs and connect subsequent events; arcs are used to represent activities whose beginning or end correspond to these subsequent events. For example, a *duty arc* emanates from the node corresponding to the beginning of a duty and enters the node corresponding to the end of that duty. Crew is the entity that flows on the arcs over the space-time network.

2.2.1 Elements of the Space-time Network

We first define the four types of essential nodes we use in the space-time network representation of the crew capacity planning problem:

- The *source node* represents the source of the crew resources; its space attribute is the home station and its time attribute is the beginning of the planning horizon. We suppose that all crew members are located at the source node, or at the home station, at the beginning of the planning horizon.

- The *sink node* represents the end of the duties during the planning horizon for a crew member. Its space attribute is the home station and its time attribute is the end of the planning horizon. We suppose that the crew members return to the home station at the end of the planning horizon.

- An *on-duty node* marks the beginning of a duty. For an on-duty node of a train-duty, the length of the time period between the time attribute of the on-duty node and the departure time of the train should be equal to the on-duty time. For an on-duty node of a station duty, the time attribute corresponds to the beginning of the station shifts (00:00, 08:00, and 16:00).

- Each on-duty node has a corresponding *tie-up node* which marks the end of the duty. For a tie-up node of a train-duty, the length of the time period between the time attribute of the tie-up node and the arrival time of the train should be equal to the off-duty time. For a tie-up node of a station duty, the time attribute corresponds to the end of the station shifts (07:59, 15:59, and 23:59).

The arcs in the space-time network represent the flow of crew as the crew is engaged with the activity represented by the arc. According to the type of events represented by the nodes and the corresponding activities, we use the following types of arcs:

- We create *source arcs* from the source node to every on-duty node at the home station (home on-duty node). These arcs represent the start of the first duty of the planning horizon for a crew. A crew leaving the source node over a source arc cannot cover a duty earlier than the time attribute of the head node of this source arc, which is an on-duty node.

- We create *sink arcs* from every tie-up node at the home station (home tie-up node) to the sink node. The flow of crew over a sink arc represent the end of the weekly duties for the crew. Time attribute of the tail node of a sink arc that has a flow on it marks the end of the weekly duties of a crew. A crew reaching the sink node over a sink arc cannot cover a duty that starts later than the time attribute of the tail node of this sink arc, which is a tie-up node.

- There exists a *duty arc* from an on-duty node to its corresponding tie-up node. Train duties and station duties are both represented by the same type of duty arcs. In general, a duty arc must have at least one unit of flow on it representing the fact that there is at least one crew member engaged with the activity (i.e. the duty) represented by the arc. If a duty takes more than double manning time then there must be at least two crew members covering the duty, corresponding to two units of flow on the arc.

- *Rest arcs* are created from a tie-up node to an on-duty node. The space attributes of these nodes must be the same. When these nodes are at the home station, the arc is a *home rest arc*; otherwise, it is an *away rest arc*. According to the rules at TCDD, a crew that ends a duty at the home station (including station duties) must take a rest regardless of his previous duty time. The duration of this rest must be greater than or equal to the minimum home rest time and less than or equal to the maximum home rest time. Therefore, we create rest arcs from a home tie-up node to home on-duty nodes whose time attributes are larger than that of the tie-up node by at least the minimum home rest time and at most the maximum home rest time. Similar rules apply for the away rest arcs considering the minimum away rest time and the maximum away rest time parameters.

- *Deadhead arcs*, used to represent away-to-home deadheads only, are created from an away tie-up node to the home tie-up node of the train that is used by the deadheaded crew member. According to the rules in TCDD, a crew that ends a duty at an away station must first take a rest before deadheading. The duration of this rest is at least the minimum deadhead start time and at most the maximum deadhead start time and we call this time period *feasible deadhead start window*. Therefore, creating deadhead arcs, we only consider trains departing from the same away station during the feasible deadhead start window. Furthermore, we only consider the train that brings the crew member to home in the earliest occasion and we create only one deadhead arc for every tie-up node that has a feasible deadhead opportunity. After deadheading, crews are subject to the same rest constraints. Home-to-away deadhead is represented by the duty arcs and we do not have to consider anything else for home-to-away deadhead. Figure 2.2 illustrates the away-to-home deadhead policy. Note that the train we choose for deadheading is not necessarily the one whose on-duty node has the smallest time attribute, as the speed of trains and their trip patterns can be different. We create a single deadhead arc, emanating from the away tie-up node and entering to the home tie-up node of the train duty used for deadheading.

- In order to represent the coverage of an excess duty by a crew member, we use *direct connection arcs* to represent this situation in the space-time network. A direct connection arc is created from an away tie-up node to an away on-duty node at the same away station. The direct connection arc is created if the difference between the time attributes of the on-duty node of the first duty and of the tie-up node of the second duty is smaller than the excess duty time. The excess duty of the crew member consists of the first duty, the waiting time between the two duties, and the second duty. As this description implies, the time attribute of the on-duty node of the second duty must be greater than or equal to the time attribute of the tie-up node of the first duty. In Figure 2.2 an example for the creation of direct connection arcs is given. To create a direct connection arc, the time between the time attribute of the home on-duty node of the first duty and the time attribute of the home tie-up node of the second duty should be less than or equal to the excess duty time. Furthermore, the time attribute of the tie-up node of the first duty should be smaller than or equal to the time attribute of the on-duty node of the second duty, allowing the crew member to feasibly cover the two duties.

Figure 2.3 is an illustration of the space-time network with different types of nodes and arcs. Note that the illustration only shows a small subset of the whole network.

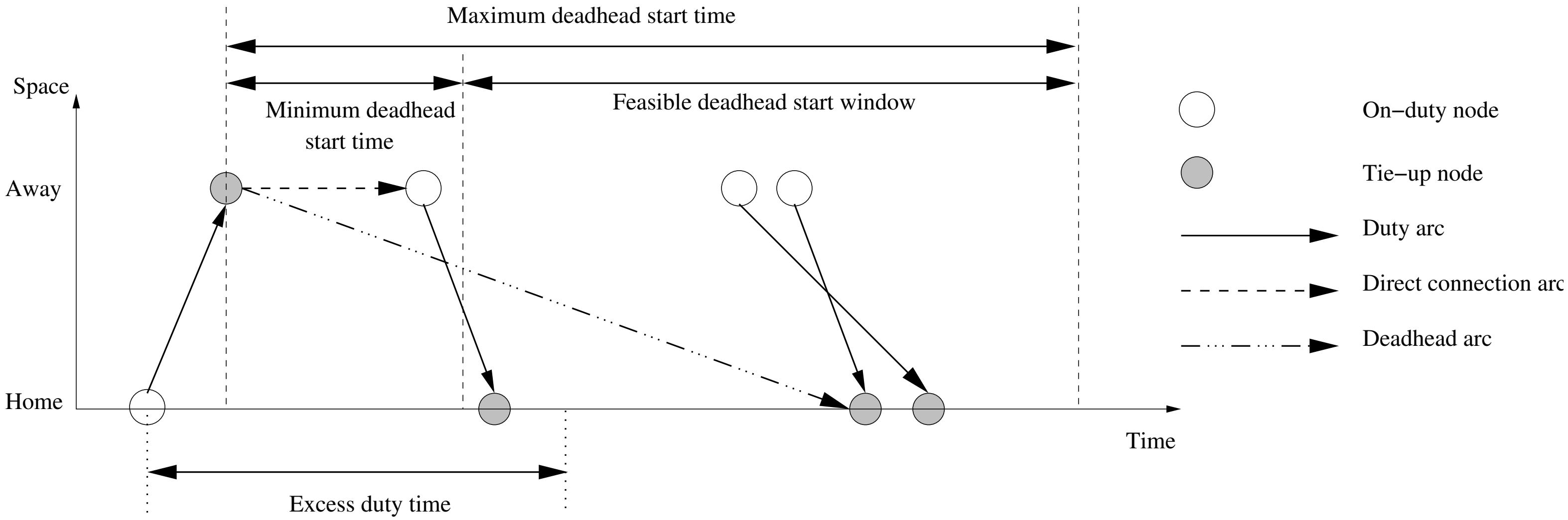

Figure 2.2: An illustration of direct connection and deadhead policies.

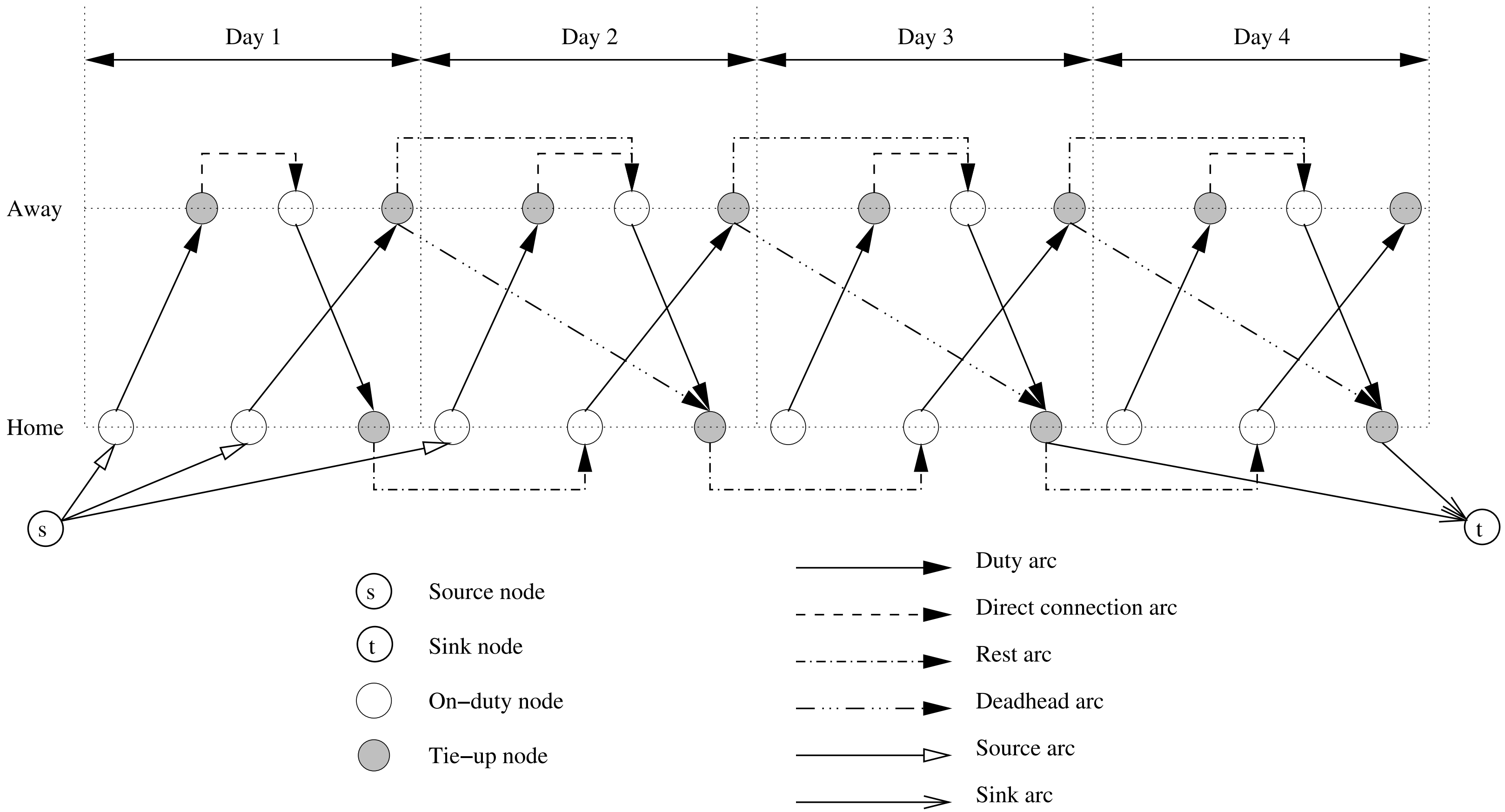

Figure 2.3: An illustration of the space-time network.

2.2.2 Handling Difficulties at the Beginning and the End of the Planning Horizon

We assume that each crew starts a week at the home station and has to return to the home station at the end of the week. However, this assumption cannot be easily represented in the space-time network due to the existence of following types of duties in the schedule:

(i) There may exist a train duty ending at an away station, where the crew member covering this train duty is unable to perform one of three options we discussed, namely away rest, deadhead, and direct connection. This occurs when the duty is close to the end of the planning horizon and the crew member has limited options of trains after reaching an away station.

    If a certain crew member cannot return to home station with one of the three options we discussed, we choose to connect the away tie-up node of this duty to the sink node as it is possible to send the crew back home through a deadhead or a duty from the beginning of the next week. This situation corresponds to creating an away tie-up node with only a sink arc emanating from the node.

(ii) There may exist a train duty starting at an away station close to the beginning of the planning horizon, and it is impossible to cover this train duty with crew members reaching the same away station and by satisfying their away rest requirement.

    If covering a duty from an away station to the home station is not possible because of a lack of crew members that can feasibly cover this duty, we move the on-duty node of the train to the end of the week by specifying its time attribute as the length of the planning horizon plus the original time attribute of the on-duty node (corresponding to pushing this duty to the beginning of the following week in a repetitive schedule). We call this type of node an *away early on-duty node*. We connect away early on-duty nodes to the sink node, and we do not create a tie-up node corresponding to an away early on-duty node. This is an admissible move as the tie-up node of an away early on-duty node is at the home station and can only be connected to the sink node. Away early on-duty nodes are subject to similar constraints as the regular on-duty nodes except that they are connected to the sink node. The arc connecting an away early on-duty node to the sink node represents a duty; therefore, it must have the necessary flow corresponding to the number of crew members required to execute the duty in a feasible solution of the problem.

2.2.3 The Network Flow Problem

Representing the problem with a space-time network, we intend to formulate the crew capacity planning problem as a network flow problem. The capacity planning problem aims at minimizing the number of crew members required to operate the predetermined list of trains in the schedule by covering the duties and honoring company policies and union regulations. The network flow problem on our space-time network corresponds to a minimum flow problem. The amount of flow emanating from the source node corresponds to the number of crew members required to operate the schedule. Hence, the out-flow from the source node is to be minimized. A path flow of one unit on the network represents the movements of a crew member and the activities she is engaged with. For each crew member, weekly duties start with a source arc, which is connected to an on-duty node. After covering some duties and potentially using different types of connections between them (rest, deadhead, and direct connection), the crew member reaches the sink node. The last tie-up node before reaching the sink node via a sink arc marks the end of his weekly duties.

We note that the operational crew scheduling problem involves costs associated with different types of activities and aims at finding a minimum cost assignment of crew members to duties. From the solution of the minimum flow problem, we may obtain a tentative schedule of duties. But, this feasible schedule does not consider any cost-related issues as they are usually associated with the operational level planning problem. The operational level planning problem, which assigns the duties to crew members by considering the costs of assignment, corresponds to a minimum cost flow problem. The solution of the minimum flow problem for capacity planning at the tactical level is feasible for the minimum cost flow problem at the operational level, but not necessarily optimal. Information on both types of problems and algorithms for solving them can be found in Ahuja *et al.* [9].

Solving the minimum flow problem to minimize the number of crew members corresponds to minimizing the fixed crew salaries, an important cost measure of the company. The minimum flow problem also provides an input to the minimum cost flow problem as the number of crew members required to operate the duties assigned to a region. The minimum cost flow problem minimizes the payments made to crew members with respect to their activities. However, the problem cannot be described without knowing available number of crew members since the minimum cost flow problem requires the amount of available demand and supply as input. In addition, the operational level problem is also concerned with the assignment of different types of duties in a balanced manner among the workforce. This last issue is indeed very critical in most crew planning problems in other transportation modes, too.

2.2.4 Day-off Requirement in TCDD

The network flow problem we describe in Section 2.2.3 can be used to solve the crew capacity planning problem with the given policies in TCDD. However, there is an important policy we did not mention up to this point, called the *day-off requirement.* According to the day-off requirement, every crew member must take a day-off each week. The day-off should be spent at the home station of the crew member, and it should include a complete day (from 00:00 to 23:59) and not any 24 hours.

The network representation of the problem and corresponding network flow problem do not consider the day-off requirement. Schedules of crew members obtained by the minimum flow problem are called *pseudo-feasible schedules* as they do not satisfy the day-off requirement. An additional effort is required to determine the number of additional crew members to fill-in for the duties on the day-off. We propose two solution approaches for the crew capacity planning problem with day-off requirement in Chapter 3.

In Mellouli [10] and Guo *et al.* [11], crew scheduling problem with day-off requirement is studied with a state-expanded aggregated space-time network approach. In this problem, consecutive duty days are defined as the accumulated state of the flow in the network, and crew members who work for five consecutive days (or who reach this state) take two days-off before being assigned to another duty. In our problem, the day-off requirement is not to be handled in this manner. In general, there is a certain number of days-off to be taken in a finite length planning horizon, and there is no consecutiveness restriction on the days-off and working days. Furthermore, we are studying a tactical level problem, and we are not interested in the previous state of the crew members, which bears more importance in an operational level problem.

2.3 Literature Review

The crew scheduling problem has been extensively studied in the operations research literature due its practical importance in different transportation industries such as airlines and railways. While the airline crew scheduling problem is one the most prominent research topics in operations research, the research on the crew scheduling problem on railways has been more modest. Airlines have adopted operations research techniques into their planning processes much earlier than the railways. In addition, railways have started using these techniques to solve their problems which are associated with larger capital investment such as network configuration, infrastructural planning, and rolling stock. For more information on scheduling and rostering problems in different industries, including crew scheduling at transportation services, we refer the interested reader to Ernst *et al.* [12] and references therein. In Möller [13], a review of different algorithms and models, as well as possible solution approaches for solving railway crew scheduling and crew rostering problems can be found.

The crew scheduling and associated problems are generally studied with two mainstream approaches: network flow formulations and set covering/partitioning type formulations. The network flow problem formulations usually depend on a space-time network representation of the problem, and require developing solution methods based on some relaxations of the problem, such as relaxation of a priority related constraint in Vaidyanathan *et al.* [8]. Set covering type formulations of the problem lead to developing decomposition-based methods and column generation.

In Vance *et al.* [14], an airline crew scheduling problem is studied. This study uses a similar approach to the conventional set partitioning and column generation approach. However, they present a duty-period based formulation, which unites flights under duties and duties under pairings, resulting in two similar subproblems for duty-period and pairing generation. The new formulation gives a better linear programming relaxation than the traditional set covering problem but it is more difficult to solve. In Anbil *et al.* [15], the airline crew pairing problem is solved using a column generation method and the Volume algorithm, which consists of finding near optimal solutions to the primal problem by generating feasible solutions to the dual problem.

Stojković *et al.* [16] and Medard and Sawhney [17] focus on operational airline crew scheduling problems. In the operational problem, the priority is finding a feasible schedule from the previously applied solution that is now disrupted because of a crew no-show or changes to flight schedules. The short term nature of the problem and the need to solve it frequently puts emphasis on the computation time. In their approach, Stojković *et al.* [16] first choose a smaller planning horizon and some crew members to reschedule. Then by using the same subprocedure they use for pairing generation (column generation and set partitioning), they obtain a solution for the new problem. Similarly, Medard and Sawhney [17], study integrating the crew pairing and rostering problems and develop an efficient method to solve the operational crew scheduling problem.

In our work, we follow the network flow approach and develop a space-time network representation of the capacity planning problem similar to the one developed by Vaidyanathan *et al.* [8]. In their work, Vaidyanathan *et al.* [8] study the crew scheduling problem at the operational level for North American railroads by using a multicommodity network flow approach to represent the assignment of crews with different qualifications to train duties. Their problem is computationally intractable due to a first-in-first-out (FIFO) requirement that is currently applied by Federal Railway Administration. According to this rule, crew members must be assigned to duties in the same order as they become available for covering a new duty. To solve the problem, they formulate an exact integer programming formulation of the problem based on a space-time network. As the FIFO constraint makes the problem much harder to solve, they also formulate a relaxed problem that is solved efficiently without the FIFO requirement. After obtaining the solution to the relaxed integer programming problem, they sequentially generate constraints that are violated in the previous solution, until all constraints are satisfied. This approach, however, leads to impractical computation times, especially for a scheduling problem on the operational level. To avoid this, they make use of the cost structures to perturb the costs of arcs in order to obtain near optimal solutions that comply with FIFO requirement. They also mention that the tools they develop can be used for problems on different levels.

Even though research on crew scheduling problem on a network flow formulation is limited, set partitioning and set covering formulations are more frequently used. Crew scheduling problem for Italian State Railways (Ferrovie dello Stato SpA) is studied by Caprara *et al.* [5, 6, 7]. In Caprara *et al.* [5], both the crew scheduling problem, where individual trips are united to form duties, and the crew rostering problem, where individual duties are united to form pairings are studied. For both crew scheduling and crew pairing problem, the authors propose a Lagrangian relaxation method and heuristic approaches based on Lagrangian costs, which yields good quality solutions even for large problem instances. In Caprara *et al.* [6], the crew scheduling problem is studied with three subproblems: pairing generation, pairing optimization, and rostering optimization. In the pairing generation phase, feasible pairings are generated using company policies. In the pairing optimization phase, a minimum cost subset of these pairings are selected by balancing different aspects like the number of pairings selected for each base. In the rostering optimization phase, pairings are put together to form larger blocks. In both pairing optimization and rostering optimization phases, Caprara *et al.* [6] use the Lagrangian relaxation and information obtained by Lagrangian costs to direct the optimization procedure, in a similar fashion to Caprara *et al.* [5]. Furthermore, Caprara *et al.* [7] develop a feedback mechanism between pairing optimization and rostering optimization to improve the quality of the solution. By updating pairing costs with respect to the solution of the linear programming relaxation of the rostering optimization problem, they improve on the current system used in Ferrovie dello Stato SpA.

Morgado and Martins [18] present a system that is used in Dutch Railways, called CREWS_NS, that significantly reduces the time required to generate the schedule of crews and the number of staff required to generate the schedules at the operational level. They mention that crew scheduling is done manually 6 to 12 months in advance relying on the expertise of schedulers in Dutch Railways. However, this approach causes problems as the train schedules may be altered in the meantime. To solve this problem, they present a white-box system, where it is possible for the scheduler to interact with the software. The software uses heuristics to improve the solution quality and makes use of constraint satisfaction techniques to reduce the search space for generating duties. The new system is put into use and predict a 4 million dollars decrease in crew related cost, which will be spent in placing crews to new duties. However, Kroon and Fischetti [19] claim that the CREWS_NS system has some disadvantages due to the greedy heuristic and limited backtracking possibilities to correct this problem. They present a new system that replaces CREWS_NS, called TURNI, relying on mathematical programming techniques. The system solves a set covering problem, where a new constraint for avoiding certain combinations of duties are added to the model. Moreover, they also add some suggested trips to the model without forcing crew members to cover them. To solve the model, authors use column generation, Lagrangian relaxation and heuristics, used for fixing some promising duties based on the information obtained from Lagrangian relaxation and the efficiency of the duties. The crew scheduling problem for Dutch Railways is also studied in Abbink *et al.* [3]. According to this study, there are more than 3,000 drivers in Dutch railways operating more than 14,000 timetabled daily trips, that constitute more than 1,000 duties in 29 regions. Considering that the problem is solved for a weekly schedule, the problem dimension is huge. They address the weakness of the TURNI system, which can be observed by rescheduling smaller partitions of schedules. They try to improve the current system by using different partitioning schemes to solve smaller problems simultaneously to reach more effective solutions. These partitionings are done with respect to the home station locations (solving 3 to 7 region problems together), with respect to train lines, and with respect to compatibilities and frequencies of being together for independent tasks. They claim to have reduced the crew-related costs at Dutch Railways by 2%, which corresponds to 6 million euros annually.

Freling *et al.* [20] present a heuristic branch-and-price algorithm to solve a large scale crew scheduling problem. The algorithm fixes a number of columns at each iteration to decrease the problem size and solves a smaller problem using the same procedure after generating new columns. Together with other improvements in the pricing problem, the algorithm is faster than the conventional approach but produces slightly worse results in terms of the objective function value. This is also used in Kroon and Fischetti [19], but the fixing procedure is relaxed when the solution is not improved.

Yunes *et al.* [21] present a hybrid branch-and-price method on a data set belonging to an urban mass transit company. In this approach, pricing stage is done with the help of constraint programming, whereas it is usually done by solving a problem (depending on the structure of constraints) on a network. By using constraint programming in the pricing stage it is possible to search the whole space of the feasible solutions efficiently, which makes solving the problems to optimality possible.

In addition to the current practices in the transportation systems, integrated approaches are gaining more and more importance. In Mellouli [10] and Guo *et al.* [11], partially integrated approaches to crew scheduling problem is studied. In

Mellouli [10], a state-expanded aggregated space-time network is defined and it is used for both aircraft/train maintenance routing problem and crew scheduling problem. In this approach, consecutive duty days are defined as the accumulated state of the flow in the network, where duties are set of feasible trips. For example, a vehicle on state 1 is in the first day of its operation after undergoing a maintenance. A vehicle operating for a certain time (three to four days in the example presented), or reaching that state, must receive maintenance, which resets its state to the initial value 1 after the required service (or day-off) time. Days-off for crew members are modeled similarly by using an aggregated space-time network. By aggregating vehicles or crew members that are at a certain location and decomposing the flows by disaggregating those parts further flexibility is obtained. By extending the network to contain maintenance and days-off, an integrated approach is developed. Maintenance and days-off are modeled as dummy duties carried on at maintenance bases and base respectively. The problem is solved for multiple home bases. Guo *et al.* [11] study only the crew scheduling problem by using the ideas mentioned by Mellouli [10].

In this section, we have discussed only a subset of relevant studies in crew and personnel scheduling. Yet, studies on crew scheduling in railways have all been covered to the best of our knowledge. The studies that are more relevant to ours are Vaidyanathan *et al.* [8] due to space-time network representation and Mellouli [10] and Guo *et al.* [11] due to the layered network idea. We believe that our study fills a gap in the literature by studying a tactical level capacity planning problem first time in railway-crew related literature. Since most of the literature focuses on operational level planning problems, the network flow problems are either multi-commodity problems or have hard-to-express constraints in a network flow problem. On the contrary, our major problem is a single-commodity problem and the challenging day-off requirement constraint is integrated in a fairly easy but sophisticated manner with the existing formulation characteristics. Adding to this, we also develop a framework for the higher level capacity planning problems based on the tactical level planning problems.

3 MATHEMATICAL FORMULATION AND SOLUTION APPROACHES

Currently at TCDD, the crew-related planning problems are solved with a two-stage manual approach. In particular for the capacity planning problem, they intend to find the minimum number of crew members without considering the day-off requirement at the first stage. Determining the capacity requirement and obtaining pseudo-feasible schedules for (original) crew members, substitute crew members are added in order to fill-in for the day(s)-off of the original crew member schedules. At the second stage, some of the duties of original crew members are assigned to substitute crew members. Then, the new assignments are checked for violations of regulations, and further changes are made to the schedules if necessary. Historically, the planning process has been executed with expert knowledge of the administrators at the crew regions and the headquarters.

We develop two approaches to solve the capacity planning problem with the day-off requirement:

1. In our first approach, called the sequential approach, mimicking the current two-stage approach at TCDD, we develop analytical models in order to make decisions at both stages as follows:

    - We formulate a minimum flow problem on the space-time network discussed in Section 2.2, in order to find the minimum capacity level without the day-off requirement.

- We formulate a selective assignment problem in order to find the minimum number of substitute crew members required to satisfy the day-off requirement of both original and substitute crew members.

2. In our second approach, called the integrated approach, we formulate a minimum flow problem on a layered network that integrates the day-off requirement into the space-time network. This approach gives the optimal solution to the capacity planning problem with the day-off requirement.

The solution methods used to solve the capacity planning problem with day-off requirement with respect to different approaches is summarized in Figure 3.1.

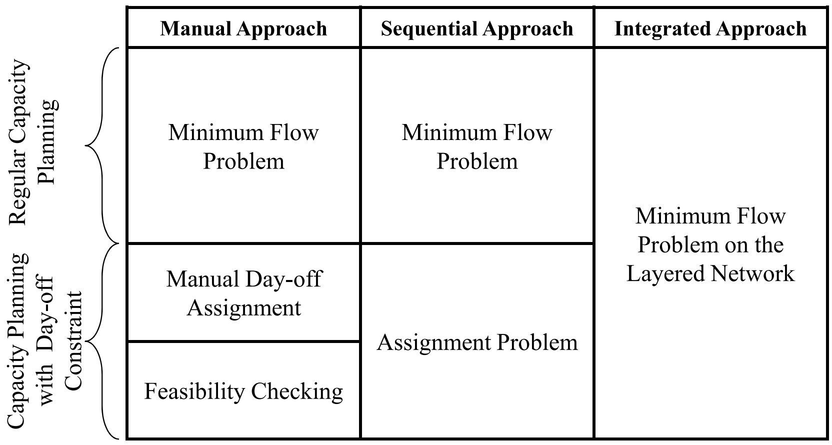

Figure 3.1: General framework for the capacity planning problem with day-off constraint.

3.1 Sequential Approach

In our sequential approach, capacity planning problem with day-off constraint is solved in two stages by mimicking the current practice at TCDD:

- Regular capacity determination, where the crew capacity of the region is determined without the day-off constraint

N	set of nodes in the network
s	source node of the network
t	sink node of the network
A	set of all arcs in the network
A_d	set of duty arcs in the network
A_s	set of crew source arcs
A_t	set of crew sink arcs
A_{n^+}	set of outgoing arcs at node n
A_{n^-}	set of incoming arcs at node n
c_a	number of crews that must be used to cover the duty represented by arc a

Table 3.1: Notation for the space-time network and minimum flow problem.

- Capacity determination with the day-off requirement, where we satisfy the day-off constraint by assigning some duties to additional substitute crew members by solving an assignment problem

In the first stage, pseudo-feasible schedules for (original) crew members are obtained by tracing the flow of the crew members in the optimal solution of a minimum flow problem. Then, some duties of the original crew member schedules are assigned to substitute crew members by solving an assignment problem to find the optimal solution according to the sequential approach.

3.1.1 Regular Capacity Determination: Minimum Flow Network Problem

Based on the network representation approach described in Section 2.2, we first depict the space-time network mathematically. Then, we present the mathematical model of the minimum flow problem we solve at the first stage of the sequential approach.

In Table 3.1, we present the notation for the space-time network and mathematical formulation of the minimum flow problem. We define the decision variable

$$x_a\text{: the amount of flow on arc } a.$$

The integer programming formulation of the minimum flow problem is as follows:

$$\text{Minimize} \sum_{a \in A_s} x_a \tag{3.1}$$

$$\text{subject to} \sum_{a \in A_s} x_a = \sum_{d \in A_t} x_d, \tag{3.2}$$

$$\sum_{a \in A_{n+}} x_a = \sum_{a \in A_{n-}} x_a, \qquad \forall n \in N \setminus \{s, t\}, \tag{3.3}$$

$$x_a \geq c_a, \qquad \forall a \in A_d, \tag{3.4}$$

$$x_a \in \mathbb{Z}_+, \qquad \forall a \in A. \tag{3.5}$$

The objective function (3.1) minimizes the flow emanating from the source node which corresponds to the number of necessary crew members for operating the weekly schedule. The problem we solve corresponds to a minimum flow problem as explained by Ahuja *et al.* [9]. Constraint (3.2) is the flow balance constraint between the source and the sink nodes, which ensures that the flow emanating from the source node is equal to the flow entering the sink node. We have the flow balance constraint of other nodes in (3.3). Constraint (3.4) is duty coverage constraint, which ensures for a duty arc the flow amount is at least as much as the number of required crew members, c_a. The integrality constraints on the variables are given in (3.5). The linear programming relaxation of an integer minimum flow problem yields an integer solution. Therefore, we can relax (3.5) as $x_a \in \mathbb{R}_+, \forall a \in A$.

An important property of the minimum flow formulation is unimodularity of the constraint matrix. The linear programming relaxation of an integer programming formulation with a unimodular constraint matrix yields an integer optimal solution. The constraint matrix of the maximum flow problem is unimodular [9]. It is possible to transform a maximum flow problem to a minimum flow problem by modifying the objective function and the right hand side of the constraint matrix (considering we replace variables having positive lower bounds). These changes do not affect the unimodularity of the constraint matrix and as a result, the optimal solution of a minimum flow problem is still integer. In addition to that, Ahuja *et al.* [9] show that the minimum flow problem can be solved in three steps, each requiring polynomial computation time. In the first step, we construct a feasible flow on the network by solving a maximum flow problem with lower bounds. After obtaining a feasible flow, we update the residual capacities of the arcs by considering the lower bounds on the flow values. Updating the network, we convert the feasible flow to a minimum flow by solving another maximum flow problem. This is achieved by pushing the maximum amount of flow from the sink node to the source node on the updated network. All of these steps are done in polynomial time, resulting in a polynomial time algorithm for the minimum flow problem.

3.1.2 Capacity Determination with Day-off Requirement

Considering the planning horizon as one week, a crew member at TCDD has to take one day-off in each week; this day-off starts at 00:00 and ends at 23:59, and it must be spent at the home station. TCDD uses a manual expertise-based approach for integrating the day-off requirement to find the additional number of crew members to fill-in for the day-off duties of the original crew. Inspecting the pseudo-feasible schedules from the first stage, for each original crew member that does not have a day-off, they manually assign some of its duties to additional crew members, which we call *substitute crew*. Substitute crews are subject to the same operational constraints and they also have to take a day-off each week. In order to avoid the current manual approach for integrating days-off, we redescribe the problem and formulate it as an assignment problem. As we are dealing with a capacity planning problem, our objective is (again) to minimize the number of substitute crews we have to add to the region for a solution that is feasible for both substitute and original crew.

To formulate the assignment problem, we use the pseudo-feasible schedules of original crews obtained by solving (3.1)-(3.5). As an input to this problem, we first determine for each pseudo-feasible schedule, the possible day-off time windows honoring the 00:00 to 23:59 rule. Given the possible day-off windows, the assignment problem assigns a substitute crew to only one day-off window in each pseudo-feasible crew schedule. We assume that a substitute crew covering duties for an original crew has to start and end its own duty at the home station. Due to this restriction, the assignment problem also considers avoiding the assignment of a set of incompatible (with respect to regulations) duties to crew members.

For a problem with a single day-off requirement, we determine the possible day-off windows as follows. Inspecting the pseudo-feasible schedules in an optimal solution to (3.1)-(3.5), we first find the number of days-off each crew schedule has. If a crew schedule does not have any days-off, we need to determine the possible day-off windows in this schedule. Next, we identify the duty-periods in each schedule; a *duty-period* corresponds to a period that starts with the beginning of a duty at the home station and ends with the arrival of the crew member to the home station (by covering another duty or with a deadhead). We start from the first duty-period and remove it to see whether removing it creates a day-off in the schedule. If removing this duty-period creates a day-off, we store this duty-period as a possible day-off window for the original crew and continue to look for other day-off windows starting from the next duty-period. If removing the first duty-period does not create a day-off in the schedule, we continue to remove subsequent duty-periods until we obtain a day-off window. As soon as we obtain a schedule satisfying the day-off requirement, we proceed by searching the next duty-period and repeat these steps.

Table 3.2 shows the notation we use in our algorithm to determine the day-off

C_c	the schedule of the crew member c
g_c	number of days-off the crew schedule C_c has
g	number of days-off each crew needs to have
I_c	set of day-off windows that allow crew member c to have more than g_c days-off
$i.start$	the start time of the day-off window i representing a day-off window
$i.end$	the end time of the day-off window i representing a day-off window
$i.crew$	the original crew schedule of the day-off window i representing a day-off window
$i.days$	the number of days-off that is obtained by removing duty i from the schedule

Table 3.2: List of parameters and attributes used in determining day-off windows.

Algorithm 1: Algorithm for finding day-off windows for a single crew member.

Input: C_c, g_c, g

$I_c = \{\}$

forall $i \in C_c$ **do**

  $C_{temp} = C_c$

  $W = \{\}$ */Set of removed duty-periods.*

  forall $j \in C_{temp} : j.start \geq i.start$ **do**

    C_{temp}.`remove`(j) */Removes duty-period j.*

    $W = W \cup \{j\}$

    if $\texttt{offdays}(C_{temp}) \geq g$ **then**

      $I_c = I_c \cup \{W\}$

      `break` */Day-off requirement is satisfied.*

    else if $\texttt{offdays}(C_{temp}) > g_c$ **then**

      $I_c = I_c \cup \{W\}$ */Day-off requirement is not satisfied but the crew has more days-off.*

Output: I_c

Original Crew Duties		Substitute Crew Duties		
Duty-Period Start Time (Day)	Duty-Period End Time (Day)	Duty-Period Start Time (Day)	Duty-Period End Time (Day)	Day-off (Day)
12:00 (M)	02:00 (T)	12:00 (M)	02:00 (T)	Monday
21:00 (T)	05:00 (W)	21:00 (T)	04:00 (R)	Wednesday
22:00 (W)	04:00 (R)	22:00 (W)	07:00 (F)	Thursday
23:00 (R)	07:00 (F)	23:00 (R)	07:00 (F)	Friday
03:30 (Sa)	09:00 (Su)	03:30 (Sa)	09:00 (Su)	Saturday

Table 3.3: An illustration of duty start and end times for an original and substitute crew.

windows for a single crew schedule. A day-off window consists of one or several duty-periods that are sequential in the original crew schedule. The day-off windows for substitute crew schedules has four attributes: day-off window start time ($i.start$), day-off end time ($i.end$), the original crew schedule that these duties in the period belong to ($i.crew$), and the number of days-off the original crew schedule has by not covering these duties ($i.days$). A generalized version, for g days-off, of the algorithm we use to determine the day-off windows is presented in Algorithm 1. The only difference in the generalized version is that we store day-off windows that let the crew member have more days-off than her current schedule, g_c, even if that does not satisfy the g day-off requirement. However, we keep removing duties until we reach g days-off. For each duty-period of the crew schedule (line 3), we begin removing duty-periods (line 7) until we reach at least g days-off for the schedule (lines 9-11). If we cannot reach g days-off by removing a set of duty-periods but create additional days-off for the crew schedule, we still keep the solution, but we do not terminate the search (lines 12-13). The computational complexity of the algorithm (for a single crew member) is $O(|C_c|^2)$.

An example is illustrated in Table 3.3 for $g = 1$. In this example, removing the first duty-period provides a day-off on Monday and is an acceptable window. However, removing only the second duty-period does not create a feasible day-off. Note that the crew schedule contains duties on both Tuesday and Wednesday; we

C	set of original pseudo-feasible schedules
C_r	set of original feasible schedules belonging to crew members with at least g days-off
C_p	set of original pseudo-feasible schedules belonging to crew members without g days-off
I_p	set of possible day-off windows belonging to schedules for $c \in C_p$
I_d	set of dummy day-off windows representing day-off for every day in the planning horizon
I	set of all day-off windows, $I = I_p \cup I_d$
T	set of days in the planning horizon
K	set of substitute crew members

Table 3.4: List of sets used in the assignment problem.

have to remove the next duty-period, too. By removing both duty-periods, we obtain a day-off on Wednesday. The crew member substituting for the original crew member has to cover both duties from 21:00 (T) to 05:00 (W) and from 22:00 (W) to 04:00 (R), allowing the original crew schedule having Wednesday as a day-off. After completion of the search, we end up with the possible day-off windows for a substitute crew in Table 3.3 on the right-hand side columns.

In the assignment problem formulation, we also have to consider the feasibility of the schedules of substitute crews. It means that any two consecutive duties assigned to a substitute crew schedule must have sufficient rest time between them (minimum home rest time). This approach provides a pseudo-feasible schedule for substitute crew members but does not guarantee that they take g days-off. In order to let the substitute crews take days-off, we create day-off windows, starting at 00:00 and ending at 23:59 for each day of the planning horizon with no crew attribute. We force the substitute crews to cover at least g of these duties. To avoid infeasible schedules for substitute crews, we need to represent the compatibilities among the day-off windows we obtain. We create a matrix $\mathbf{B} = [b_{ij}]$, which is an $|I| \times |I|$ matrix where

$$b_{ij} = \begin{cases} 1, & \text{if same substitute crew can cover both duty } i \text{ and duty } j; \\ 0, & \text{otherwise.} \end{cases}$$

If i and j are both day-off windows for a pseudo-feasible schedule, then the substitute crew has to be able to take a home rest between these duties. If, without loss of generality, i is a day-off then the duty j must not intersect with the time interval of i. Otherwise, we set $b_{ij} = 0$ and duties i and j are not compatible.

For the substitute crew assignment problem formulation, we define the following decision variables:

$$y_k = \begin{cases} 1, & \text{if substitute crew } k \text{ is used;} \\ 0, & \text{otherwise.} \end{cases}$$

$$w_{ik} = \begin{cases} 1, & \text{if substitute crew } k \text{ is assigned to duty } i; \\ 0, & \text{otherwise.} \end{cases}$$

The integer programming formulation of the substitute crew assignment problem is as follows:

$$\text{Minimize} \sum_{k \in K} y_k \tag{3.6}$$

$$\text{subject to} \sum_{i \in I_c} \sum_{k \in K} w_{ik}(i.days - g_c) \geq g - g_c, \qquad \forall c \in C_p, \tag{3.7}$$

$$\sum_{i \in I_d} w_{ik} \geq g y_k, \qquad \forall k \in K, \tag{3.8}$$

$$y_k \geq w_{ik}, \qquad \forall i \in I, \forall k \in K, \tag{3.9}$$

$$\sum_{\forall j \in I : b_{ij} = 0} w_{jk} \leq M(1 - w_{ik}), \qquad \forall i \in I, \forall k \in K, \tag{3.10}$$

$$w_{ik}, y_k \in \{0, 1\}, \qquad \forall i \in I, \forall k \in K. \tag{3.11}$$

The objective function (3.6) minimizes the number of substitute crews to be added to the capacity. Constraint (3.7) guarantees that each pseudo-feasible crew schedule is given at least g days-off. Constraint (3.8) leaves g days-off in a week for each substitute crew schedule. We, then, ensure in (3.9) that a substitute crew can be assigned to a duty only when the substitute crew is included in the schedule. Constraint (3.10) ensures that if a certain duty-period assignment is made to a substitute crew only compatible duty-period assignments are allowed for the same substitute crew. Constraint (3.11) indicates that the variables must be binary.

The substitute crew assignment problem, with a large number of binary variables and compatibility constraints, is computationally intractable. This observation was empirically confirmed in our experiments. Therefore, next, we present a heuristic algorithm to solve this problem.

Greedy Heuristic

In Algorithm 2, we present the greedy heuristic we developed for the assignment problem. In the algorithm, C_s denotes the set of schedules of substitute crew members and k denotes the size of the set K. In this algorithm, we only consider duties that let the crew schedules have at least g days-off, and we do not consider cases where having g days-off is possible by removing nonsequential duty-periods from the pseudo-feasible crew schedules. We, then, sort the possible day-off periods with respect to their duration, i.e. the time between the beginning of the first duty and the end of the last duty (or the end of the deadhead for deadheading crew members). Sorting the duty-periods in nondecreasing order of their duration, we repeat the following steps until all crew schedules have g days-off. We add a new substitute crew with an empty schedule (lines 9 and 10). Then, starting from the shortest duty-period, we check whether adding this duty causes infeasibility (a violation of the minimum home rest time, or of the day-off requirement of the substitute crew). If there is infeasibility, we skip this duty-period (line 12). If the new schedule is feasible, we add this duty-period to the schedule of the substitute crew (line 13). We modify the sets C_p and C_r accordingly, and delete duties with the same crew attribute from the set I (lines 14 and 15). After each pass over the duty-periods in the set I, we assign days-off to substitute crew members (line 16) and we fix the substitute crew schedule (line 17). At the end of the algorithm, we obtain the number of substitute crew members required to obtain feasible schedules for both substitute and original crew. The computational complexity of Algorithm 2 is $O(|C_p|\,|I_p|)$.

Algorithm 2: Algorithm for finding an upper bound on the assignment problem.

Input: C_p, I_p

$k = 0$ */Number of used substitute crews.*

$C_s = \{\}$ */Set of substitute crew schedules.*

forall $i \in I_p$ **do**

    if $i.days < g$ **then**

        I_p.`remove`(i)

I_p =`sort(`I_p`)` */Sort the remaining duties in nondecreasing order of their duration.*

while $C_p! = \{\}$ **do**

    $k = k + 1$

    $c_k = \{\}$ */Initialize the substitute crew schedule.*

    forall $i \in I_p$ **do**

        if *isFeasible*($c_k \cup \{i\}$) **then**

            $c_k = c_k \cup \{i\}$

            $I_p = I_p \setminus \{j \in I_p : j.crew = i.crew\}$

            $C_p = C_p \setminus \{i.crew\}$

    `assigndayoff`(c_k)

    $C_s = C_s \cup \{c_k\}$

Output: k, C_s

To conclude our discussion, in the sequential approach, we solve the tactical crew planning problem in two stages:

- In the first stage, we solve the regular capacity planning problem without considering the day-off requirement by formulating a minimum flow problem over a space-time network.

- In the second stage, we assign days-off to the crew schedules obtained in the first stage by using substitute crew members. In the second stage of the sequential approach, it is possible to use either an optimal solution of the assignment problem or a feasible, but not necessarily optimal, solution of the greedy heuristic.

3.2 Integrated Approach

In our integrated approach, we solve the capacity planning problem with the day-off requirement by extending the space-time network representation we discussed in Section 2.2. Contrary to the sequential approach, which yields a suboptimal solution, we obtain an optimal solution for the capacity planning problem with the day-off requirement. Furthermore, we do so by solving a single minimum flow problem on a layered network.

3.2.1 Layered Network Representation

In order to solve the problem with the day-off requirement, we enhance our network representation. The previous formulation must be modified to satisfy the day-off requirement; this leads to a different problem where there is no guarantee for the integrality of the solution. In this new formulation, we use the layer idea explored as state-expanded network representation by Guo *et al.* [11] and Mellouli [10]. For a generalized problem with g days-off, the network consists of $g+1$ layers from $Layer_0$ to $Layer_g$, with identical nodes, where the layer number marks the number of days-off a crew schedule includes in the partial schedule corresponding to the partial path from the source node on $Layer_0$ to the corresponding layer. Duty, deadhead, rest, and direct connection arcs are also identical for all layers. However, we have to slightly modify source and sink arcs:

- $Layer_0$ contains the source node of the network and the flow on this layer represents the movement of crews that have not yet taken any days-off.

- $Layer_g$ of the network contains the sink node of the network and the flow on this layer represents the movements of crews that have at least g days-off.

In essence, a unit flow emanating from the source node on $Layer_0$ has to reach, eventually, the sink node on $Layer_g$ after passing through $g+1$ layers of the network.

The source arcs and sink arcs in the network can be used to create days-off for crew members by using the following properties. As we already know that the head node of a source arc represents the beginning of the weekly duties for a crew member, we can conclude that Monday is a day-off for a crew member starting her weekly duties on Tuesday. Similarly, if a crew member ends her weekly duties on Saturday, we can consider Sunday as her day-off. To include this property to the network, we alter the creation of source and sink arcs as follows:

- For a home on-duty node on day γ we add a source arc emanating from the source node to the $Layer_{\gamma-1}$ version of the on-duty node by considering first $\gamma - 1$ days as day-off. For example, for a home on-duty node on the second day of the planning horizon, the source arc uses $Layer_1$ version of the on-duty node. As the supply arcs indicate the beginning of the weekly duties of a crew, we can consider the first day of these crews as a day-off. For the other copies of these nodes, we do not have to create additional supply arcs.

- For a home tie-up node on day $g - \gamma$, if the crew member can take the last γ days as day-off, then we can connect the node on $Layer_{g-\gamma}$ to the sink node with a sink arc. For example, a crew member reaching the home station on day $g - 1$ with a home tie-up node on $Layer_{g-1}$ can be connected to the sink node with a sink arc denoting the last day of the planning horizon as a day-off. However, in this case, we do not omit the sink arcs emanating from the higher layer copies of these nodes and create sink arc for copies $g - \gamma, g - \gamma + 1, \cdots, g$.

In order to emulate day-off situation and provide a connection between the layers of the network, we create additional home rest arcs by considering days-off, called *day-off arcs*. A day-off arc emanating from a home tie-up node enters a home on-duty node on the subsequent layers. Day-off arcs may contain up to g days-off. As a result of this, the crew members can take their days-off sequentially or non-sequentially. If the number of days-off is γ, we must guarantee that there are γ days (from 00:00 to 23:59) between the time attribute of the home tie-up node and the time attribute of home on-duty node. For crew members in $Layer_0$ of the network, we create day-off arcs between the home tie-up nodes in $Layer_0$ and home on-duty nodes in $Layer_\gamma$ of the network, to create γ sequential days-off for crew members. For a single day-off, if a crew terminates its duty with a home tie-up node at $Layer_0$ with a time attribute 21:00 of Tuesday, then its next duty can only begin at the beginning of Thursday on $Layer_1$. Considering the crew member has two days-off, her subsequent duty may start at the beginning of Friday on the $Layer_2$, and so on. In our approach, we did not have any constraints in the exercise of days-off. Crew members can take sequential or non-sequential days-off. In case there is a sequential day-off requirement, we can construct only two layers, independent of the g value, with day-off arcs containing at least g days-off. Furthermore, it is trivial to add different kinds of constraints on day-off arcs, like balancing the number of crew members having day-off on different days of the planning horizon, which adds to the value of the integrated approach. An illustration of the source, sink, and day-off arcs on a layered network is presented in Figure 3.2

3.2.2 Mathematical Formulation

After creating the layered network discussed in Section 3.2.1, we define the mathematical programming formulation associated with the layered space-time network. L

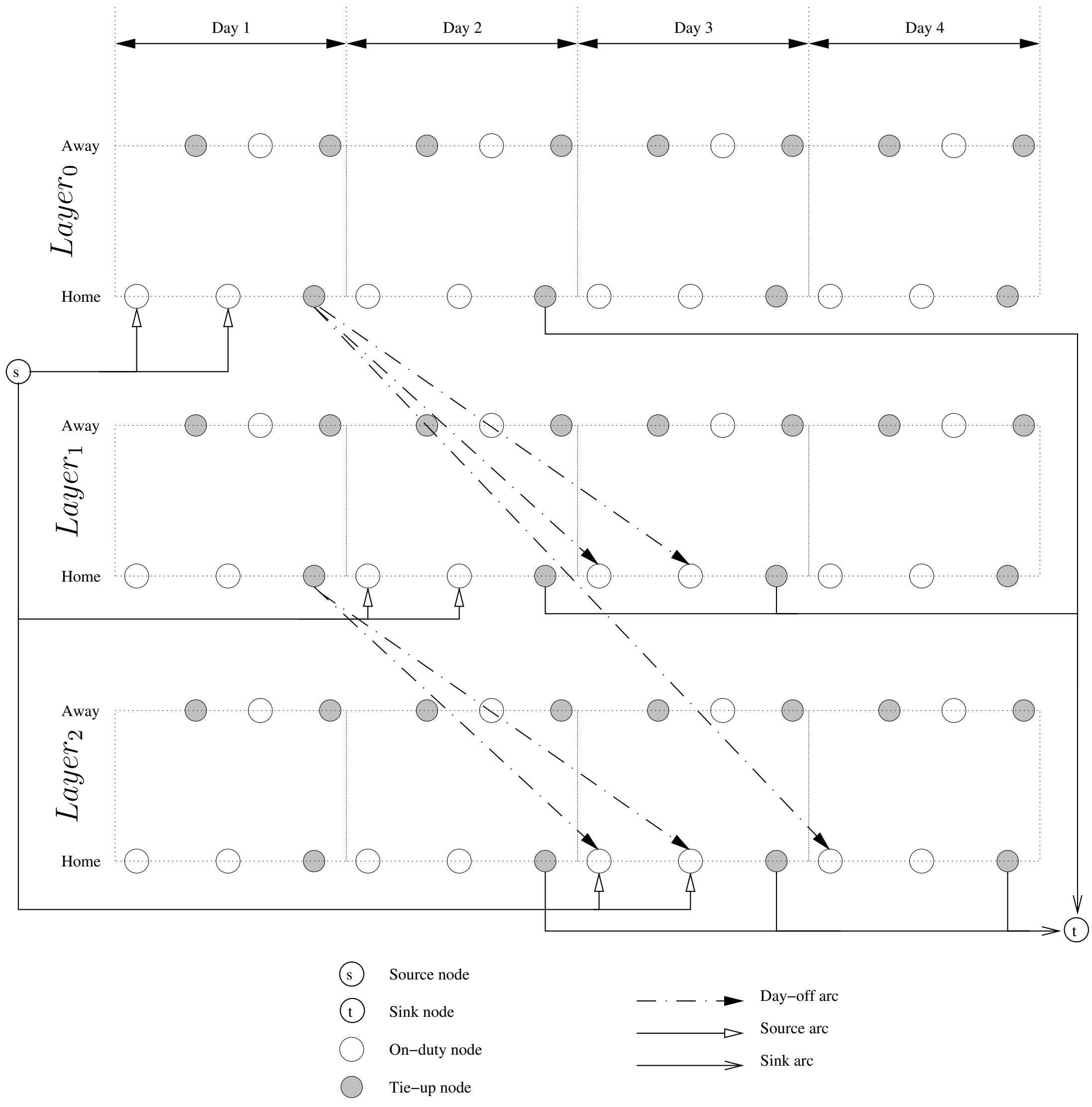

Figure 3.2: An illustration of source, sink, and day-off arcs on a layered space-time network.

represents the set of layers in the network. We define the decision variable x_a as the amount of flow on arc $a \in A$ and x_a^l as the amount of flow on copy of duty arc $a \in A_d$ on layer $l \in L$.

The integer programming formulation of the network flow problem is as follows:

$$\text{Minimize} \sum_{a \in A_s} x_a \tag{3.12}$$

$$\text{subject to} \sum_{a \in A_s} x_a = \sum_{a \in A_t} x_a, \tag{3.13}$$

$$\sum_{a \in A_{n^+}} x_a = \sum_{a \in A_{n^-}} x_a, \qquad \forall n \in N \setminus \{s, t\}, \tag{3.14}$$

$$\sum_{l \in L} x_a^l \geq c_a, \qquad \forall a \in A_d, \tag{3.15}$$

$$x_a \in \mathbb{Z}_+, \qquad \forall a \in A. \tag{3.16}$$

The objective function (3.12) again minimizes the amount of flow leaving the source node, which corresponds to minimizing the number of crew members required to operate the given schedule. Constraints (3.13) and (3.14) are flow-balance constraints for the source, the sink, and other nodes. The coverage constraint (3.15) guarantees that the total amount of flow on all copies of a duty arc is at least as much as the required amount, c_a, to ensure that duties are covered by the required number of crew members. The final constraint (3.16) is the integrality constraint on the decision variables.

Unlike the formulation (3.1)-(3.5), this is not a standard minimum flow formulation and cannot be solved in polynomial time. Constraint (3.15) puts a lower bound on the total amount of flow on a set of arcs, representing the versions of a duty on different layers. The lower bound is not on individual arcs, eradicating the unimodularity of the constraint matrix. Therefore, linear programming relaxation of the problem does not yield an integer solution, and there is not any known polynomial time algorithm to solve the problem. We observe that linear programming relaxation produces bounds that are close to optimal. However, the solution has fractional flow values, and converting a fractional solution to an integer feasible solution to the problem may not be easy.

4 COMPUTATIONAL RESULTS

In this section, we present the results of our computational study on a set of real-life data sets from TCDD. We have collected data on three crew regions: Haydarpaşa, Ankara, and Eskişehir. Among these districts, Eskişehir lies between the other two and is usually, but not necessarily, used as a crew exchange station for trains operating between Haydarpaşa and Ankara. For each crew district, we test four scenarios by modifying the length of the planning horizon and the day-off requirement. We solve the problem for one week and two weeks. For each planning horizon length, we create two problems with day-off requirements of one and two days. Crews belonging to Ankara district operates 35 train duties, while Haydarpaşa and Eskişehir operate 44 train duties each.

The results are given in Table 4.1. The first three columns show the crew district and the attributes of each scenario for the districts. OFV stands for the objective function value which corresponds to the number of crew members. For the sequential approach, MFP refers to the results of the minimum flow problem at the first stage, whereas GH-AP and OPT-AP refer to the greedy heuristic and optimal integer programming formulation results of the assignment problem at the second stage, respectively. For the integrated approach, L-MFP refers to the optimal integer programming formulation results of the layered network flow problem and R-L-MFP refers to the linear programming relaxation results of the layered network flow problem. The computational experiments are performed on a Intel Core 2 Duo 2.2 T7500 PC with 2 GB RAM; we use CPLEX 11.0 on OPL Studio 5.5 to solve the optimization problems.

Scenario Attributes			Sequential Approach								Integrated Approach			
			OFV				Time (ms)				OFV		Time (ms)	
District	Planning Horizon	Days-off	MFP	GH-AP	OPT-AP	Total	MFP	GH-AP	OPT-AP	Total	L-MFP	R-L-MFP	L-MFP	Total
Ankara	1	1	52	9	7	59	187	31	7312	7953	57	56.2	813	1281
	1	2	52	22	19	71	171	266	864891	865734	67	66.8	1234	1921
	2	1	52	4	4	56	469	125	5000	6203	54	53.2	5390	6328
	2	2	52	9	8	60	500	1906	995343	998374	58	57.5	92766	94484
Haydarpaşa	1	1	46	7	6	52	312	47	20281	21093	46	46.0	2531	3171
	1	2	46	17	14	60	328	172	498984	499953	53	52.4	4484	5515
	2	1	46	2	2	48	844	0	172	1796	46	46.0	11235	12672
	2	2	46	5	4	50	844	172	205797	207546	46	46.0	76797	79593
Eskişehir	1	1	76	16	16	92	281	296	937797	938843	79	78.6	1391	1968
	1	2	76	35	30	106	266	1593	(69.47%)	–	94	94.0	2282	3265
	2	1	76	7	5	81	735	437	326500	328546	76	76.0	11079	12453
	2	2	76	13	11	87	750	6250	(70.73%)	–	81	80.4	100406	102968

Table 4.1: Results for different approaches for solving the crew scheduling problem.

In all of the instances, integrated approach produces better results than the sequential approach in terms of the objective function value as expected. For instance, in Ankara district, with one week planning horizon and with a day-off requirement of one day, the sequential approach finds 59 crew members to operate the weekly schedule. In the first stage of the problem, the minimum flow problem, 52 crew members are used to cover the weekly schedule. In the second stage, 7 substitute crew members are added to fill-in the days-off for the original crew members. The optimal solution of the assignment problem improves on the solution found by the greedy heuristic, which requires 9 crew members. Yet, the integrated approach requires 57 crew members instead of the sub-optimal 59 of the sequential approach. To exemplify with Haydarpaşa district, for a planning horizon of one week and a requirement of two days-off, the sequential approach requires 60 crew members (46 at the first stage and 14 at the second stage). However, with the integrated approach, 53 crew members are sufficient to operate the given schedule while satisfying the day-off requirement.

In some instances, such as Haydarpaşa with one week planning horizon and with a requirement of one day-off, the objective function value of the integrated approach is equal to the objective function value of the minimum flow problem at the first stage of the sequential approach. This clearly implies that it is possible to operate the given schedule with day-off requirement by using the same number of crews found at the first stage of the sequential approach. We can easily conclude that the integrated approach is not only technically sophisticated, but it also helps the company save significantly in allocation of resources.

We can also see that greedy heuristic produces reasonably good initial solutions for the assignment problem. An observation we made was that providing an upper bound to the size of the set K helped us solve the problem in less computation time as the number of binary decision variables decreases sharply by decreasing the size of the set K. To help us reduce the computation time, we use the greedy heuristic to provide both an upper bound on the value of the size of the set K (and hopefully an elaborated one in comparison to possible trivial approaches) and an initial feasible solution. Even with the help of the greedy heuristic, there were two instances where the assignment problem failed to reach an optimal solution in 1000 seconds: Eskişehir district with a day-off requirement of two days (for both planning horizon choices). This is shown with "–" under the "Time" header. For these two instances, we show the gap between the best bound and the best feasible solution. Considering that the gaps are quite large, we can safely assume that solving these problems to optimality even with a significant increase in time limit does not seem possible.

In addition to providing better solutions, the integrated approach also required less computation time than the sequential approach in all but two instances: Ankara with a two weeks planning horizon and a requirement of one day-off and Haydarpaşa with two weeks planning horizon and a requirement of one day-off. We make this comparison by looking at the columns "Total" under the "Time" header for both sequential and integrated approaches. The column "Total" shows the time required for the construction of the space-time network and the time required by the corresponding solution approaches. This result shows that, in these instances, most crew schedules obtained at the first stage of the sequential approach honor the day-off requirement so that the assignment problem is easier to solve. The worst computation time for the integrated problem is slightly more than 100 seconds, still showing that the capacity planning problem of a region can be solved efficiently in practice.

From a managerial point of view, we make the following observations:

- With a given planning horizon length, the necessary number of crew members for operating the train schedule increases when we increase the number of days in the day-off requirement. For example, in Eskişehir region with one week planning horizon, number of crew members in the optimal solution increases from 79 to 94 when we increase the number of days in the day-off requirement from one to two.

- For a fixed number of days in the day-off requirement, decreasing the length of the planning horizon deteriorates the quality of the solutions. For example, in Eskişehir district with one day in the day-off requirement, the number of crew members in the crew regions decreases from 79 to 76 when the length of the planning horizon decreases from two weeks to one week.

In both cases the objective function value becomes worse when we increase the number of days in the day-off requirement.

From a computational point of view, changes in the length of the planning horizon and number of days in the day-off requirement reflect on the computational times of the sequential approach:

- For a fixed planning horizon, increasing the number of days in the day-off requirement makes the substitute crew assignment problem harder to solve, such as in Eskişehir region, when we reach the time limit by increasing the days in the day-off requirement to two.

- For a given number of days in the day-off requirement, decreasing the length of the planning horizon makes the problem harder to solve. For instance, in Ankara district with a planning horizon of two weeks, the computation time decreases from 998 seconds to 6 seconds, when we decrease the number of days in the day-off requirement from two to one.

In both cases the objective function value of the substitute crew assignment problem becomes worse when we increase the number of days in the day-off requirement. As a consequence, the number of substitute crew members in the assignment problem is expected to be larger. This, indeed, increases the number of decision variables and constraints in the assignment problem.

For the integrated approach increasing the length of the planning horizon or the number of days in the day-off requirement makes the problem harder to solve. However increasing the length of the planning horizon affects the computation time more than increasing the number of days in the day-off requirement.

5 SYSTEM-WIDE TACTICAL-TO-STRATEGIC PLANNING ON CREW REGIONS

Determining the minimum sufficient crew capacity levels for different crew-base regions is an important problem for TCDD, especially considering that they are beginning to use operations research tools for their operations, as it was stated by Havelsan in our personal communications [22]. Even though tactical capacity planning is important for the company, it does not provide any flexibility in terms of the allocation of train duties to other regions or sharing of duties among crew members located at different regions. The predetermined list of trains for each crew region and the crew exchange stations for the train duties have been used for years unless there was a substantial change in the train schedules. Studying the capacity planning problem with a more system-wide approach can bring substantial improvements in crew-related costs. A more system-wide approach could consider re-allocation of train duties among neighboring regions and evaluation of train duty sharing policies with respect to locations of exchange stations. In order to reduce the number of crew members working in all of the crew districts in TCDD, we formulate tactical-to-strategic level planning problems. In this setting, the borders of crew districts are assumed to be flexible (to different degrees for different problem formulations) as there are several candidate stations along the corridors between regions to execute crew exchanges.

Let us suppose that there are two neighboring crew districts, H_0 and H_1, and an intermediate crew exchange station between them at I_1. We consider two trains in

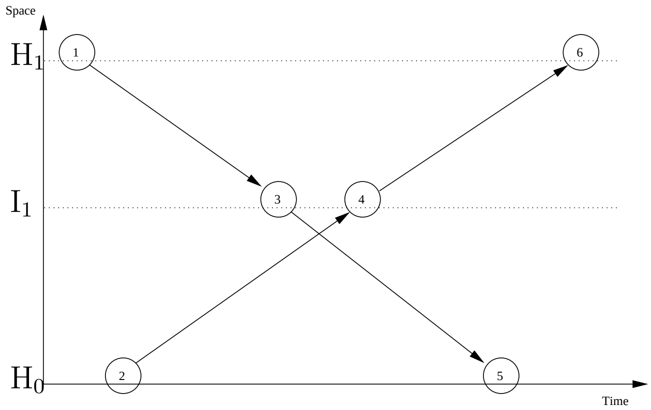

Figure 5.1: An illustration of the original network structure for a fixed crew exchange station.

the opposite direction one from H_0 to H_1 and another one from H_1 to H_0. Both trains use I_1 as their crew exchange station. The situation is illustrated in Figure 5.1. In this example, let us suppose that the time between nodes 1 and 6 is greater than the excess duty time and the time between nodes 2 and 5 is less than the excess duty time. We need four crew members to cover the train duties represented by the duty arcs $(1,3)$, $(2,4)$, $(3,5)$, and $(4,6)$.

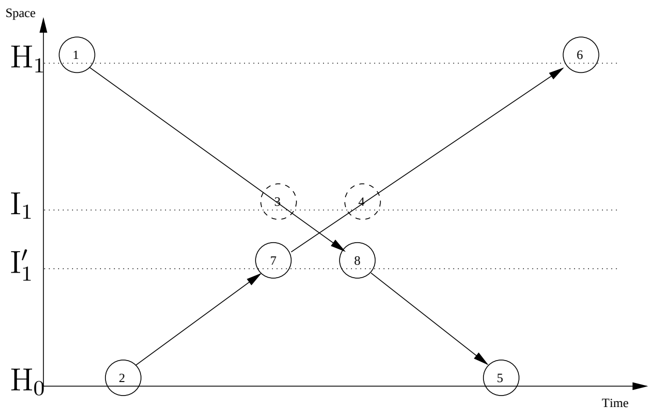

Figure 5.2: An illustration of the effects of flexible station change on single region.

A potential benefit of changing crew district borders (i.e. exchange stations) is exemplified in Figure 5.2. In this example, we consider the same trains; but, we consider the station I'_1 as the new exchange station. If crew exchange is executed at I_1', considering that the time between nodes 2 and 5 is smaller than the excess duty time, the crew covering arc $(2, 7)$ can also cover the duty on arc $(8, 5)$ via a direct connection arc $(7, 8)$. This modification on exchange stations decreases the number of crew members required at region H_0 from two to one. For region H_1, the number of crews required for covering these two duties is still two. These two longer duties does not worsen the crew requirement of region H_1 when we change the crew exchange station. As seen, in this small example, considering re-allocation of train duties even among two regions by changing the exchange station may bring easily observable savings to the company.

To exemplify further, we can also allow crew members belonging to one region cover the train duty for the entire trip, such as the crew members belonging to region H_0 or H_1 covering duties $(1, 5)$ and/or $(2, 6)$. This option can also decrease the overall number of crew members working in the two regions. In addition to that, it is possible to avoid double manning by dividing train duties between crew members operating in different regions. If a train duty taking more than 8 hours is divided in two parts, which results in two smaller train duties, there exists an opportunity to reduce the number of crew members over the two regions.

In the rest of this chapter, we discuss the mathematical models in two main categories: single region and two-region models. The new models help us determine the crew exchange stations optimally and re-allocate the train duties accordingly instead of the current approach where they are considered as given. Single region models are used only to illustrate the ideas on the tactical-to-strategic level capacity planning problem; their practical purposes are limited as we do not consider in these models the simultaneous effects of the new allocations and the new crew exchange stations on the other regions. In other words, the mathematical formulation would lead to results where the coverage of duties are minimum, by choosing the crew exchange stations as close to the home station as possible. Later on, we present two-region models, that take into account the effects of re-allocation on the two regions that share some train duties. However, even studying two region models is not sufficient as the schedules in TCDD are designed by considering firm decisions on the crew exchange stations. With multiple trains using the same station as the crew exchange station, we have limited room for improvement by only considering re-allocation of duties according to new crew exchange stations. For this purpose, we develop new deadhead policies that involve train duties shared between two regions in order to provide managerial insight to the planners about changes in the current deadhead policies.

5.1 Single-region Problem with Flexible Exchange Stations

In this part, we define different types of single-region problems. In single-region models, we study the effects of optimally determining crew exchange stations on the capacity of a single region, which brings the crew capacity planning problem to a more tactical-to-strategic level. In our previous definition of a crew region, we consider the away stations as defining the border of the crew region. In this higher-level planning problem, we consider the borders to be more flexible by formulating different types of problems.

In TCDD, a train service may be repeated on several days of the week at the same time of the day. As a result, there are copies of the same train duty on different days; in the schedule these trains are attributed with the same *train ID*. By allowing different levels of flexibilities in the crew exchange stations, we formulate three policies that are represented by three types of problems, described as follows:

1. In the *flexible crew exchange policy*, we allow train duties with the same ID use

different crew exchange stations on different days of the planning horizon.

2. In the *semi-fixed crew exchange policy*, trains with the same ID must use the same crew exchange station on each day of the planning horizon. However, trains with different IDs may use different crew exchange stations.

3. In the *fixed crew exchange policy*, all trains operating between the same two regions (i.e. along the same rail-corridor) must use the same crew exchange station.

If these flexibilities are integrated into the crew capacity planning problem under the current settings, existing crew exchange stations or away stations will be favored because of the number of trains using these stations. Furthermore, it is possible that crew members become stuck at an away station due to lack of appropriate train duties or a train duty is not covered due to lack of crew members at the station. As a result of this, when we are dealing with the tactical-to-strategic capacity planning problem by considering different possibilities for exchange stations, we must consider new types of deadheads. When we are working with a single train ID in a single region, altering the crew exchange station can cause the crews to become stuck at the new crew exchange station, where there is no other train to transfer them back home. Similarly, it is possible for a new crew exchange station not to have any incoming trains that provide crew members for train duties originating from this station. By exploring detailed schedules of crews and developing new rules for deadheading, we can avoid these situations. In order to further facilitate the discussion on the network representation, we first explain how the new deadhead policies are represented.

5.1.1 Integrating New Deadhead Policies into the Network Representation

In developing new deadhead rules, we consider two cases as deadheading from away-to-home and deadheading from away-to-away as follows:

1. To deadhead crew members from *away-to-home*, we create deadhead arcs from an away tie-up node to the earliest home tie-up node a crew can reach. In order to return home following an away tie-up, a crew must take a train after waiting for the minimum deadhead time. In addition, we have to make sure that the crew reaches home as soon as possible. This deadhead policy is the same as the deadhead policy discussed in Section 2.2.1. However, in this new approach, we consider the detailed schedules of trains; we also use trains operated by other regions to deadhead crew members to the home station. Considering that a crew member is located at an away station I_0 at time τ, we must find the fastest way to reach the home station from station I_0 by riding on a single train and by respecting the minimum deadhead start time and the maximum deadhead start time requirements. Specifically, we have to find the fastest way to reach the home station by considering the trains that are passing through the station I_0 no earlier than τ plus the minimum deadhead start time and no later than τ plus the maximum deadhead start time as required by the company policies.

2. We consider away-to-away deadheads in two ways by using either a direct connection or an away rest.

   (i) In the first case, where we consider the deadhead opportunity with a direct connection, we make use of the excess duty opportunity, discussed in Section 2.1. Therefore, we consider two train duties: the first one from home to away and the second one from away to home starting at a later time than the end of the first duty. The direct connection arc is created from the away tie-up node of the first duty and the away on-duty node of the second duty starting at a different away station. For creating the arc, we check two conditions:

- First, we have to guarantee that the time between the on-duty node of the first duty and the tie-up node of the second duty is less than the maximum duty time from home to home.
- Secondly, we have to find a feasible deadheading opportunity (from the end of the first duty to the beginning of the second one) between the two trips. A feasible deadhead must occur between the space attribute of the tie-up node of the first duty and the space attribute of the on-duty node of the second duty. Furthermore, the deadhead trip must start and end between the tie-up time of the first duty and on-duty time of the second duty.

(ii) The second way of modeling an away-to-away deadhead is considering the possibility of an away rest accompanied with a deadhead. In this case, a deadhead arc includes both an away rest and a deadhead trip in two ways:

- A crew reaching an away station, say I_0, at time τ first takes an away rest at least as long as the minimum away rest time, then travels deadheading to another station in order to start a new duty from that station to home. For constructing deadhead arcs considering an away rest, we have to find out how we can reach other stations from station I_0 by using trains that are passing through I_0 no earlier than τ plus the minimum away rest time allowed by the company.
- A crew reaching an away station, say I_0, at time τ first travels dead-

heading to another away station, then takes a rest at that station in order to cover another train duty to home station. In this case, we find out the earliest arrival time to other away stations from station I_0 no earlier than τ, and we let the crew member take an away rest after reaching the new station.

In both cases, the deadhead arc we create spans through a deadhead period and an away rest that is not shorter than the minimum away rest time and not longer than the maximum away rest time.

For the new deadhead policies, we consider that crew members ride only a single train for deadheading. Even though a more intricate approach is possible by allowing crew members use multiple trains for a faster transfer, it brings additional computational burden for possibly a small gain. Furthermore, using multiple trains may be disturbing for transfered crew members and is prone to more disruptions as the delay on one train can have effects on deadheading the crew member. For home-to-away and away-to-away with direct connection policies, we create every possible arc in the network. However, for away-to-away deadhead policies with an away rest, we only create the deadhead arc for the first duty that a crew member can cover after being transferred to another station. Furthermore, we only create deadhead arcs with an away rest for the duties shared by two regions.

5.1.2 Mathematical Formulation

We next formulate the crew capacity planning problem by considering different policies (flexible, semi-fixed, and fixed) to determine the crew exchange stations. As we noted earlier, the single-region problem formulation is depicted for only illustrative purposes in order to simplify the discussion on problem formulations for two regions and multiple regions. An optimal decision on crew exchange stations for the single

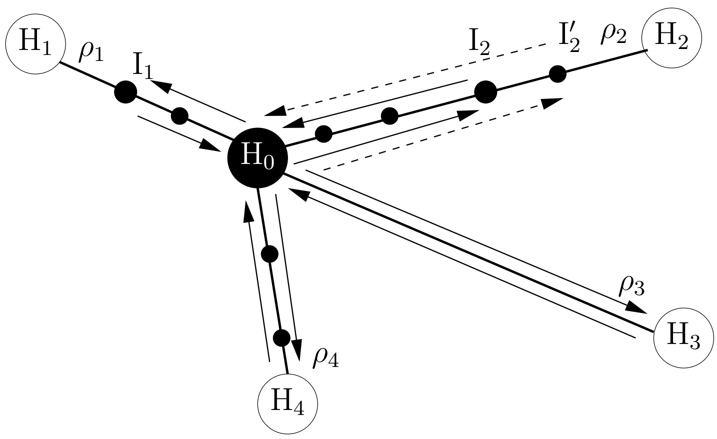

Figure 5.3: Corridors and crew exchange stations of a home station H_0.

region would favor closest stations to the home station. Therefore, the problem reduces to crew capacity planning problem discussed in Section 3.1.1. The new decision problem integrates a higher-level decision by determining the crew exchange stations into the original crew capacity planning problem. A similar approach to integrate a higher-level decision into a known problem formulation in the context of railway planning has been used by Liu *et al.* [23] for the yard location problem. The yard location problem determines the location of yards in a railroad network, which are used for grouping the cars into blocks by considering their routes. Liu *et al.* [23] solve the yard location problem by using a simplified version of the blocking problem, which can be solved efficiently, as a subprocedure.

Figure 5.3 illustrates the corridors between the home stations of neighboring crew-base regions. The home stations of the crew-base regions are denoted with H, H_0 denoting the region of interest. Corridors between two crew-base regions, denoted by ρ, are depicted using solid lines corresponding to a rail line between two home stations. Small circles on these lines show the possible crew exchange stations; the arrows represent train duties. The trains on corridor ρ_2 use station I_2 as crew exchange station, leaving the rest of the duty to be covered by crew members operating at crew-base region H_2. However, station I_2' is another option for a crew exchange

P	set of corridors shared with neighboring regions, indexed by p
V_p	set of stations belonging to corridor p, indexed by v
H	set of train duty IDs
G	set of days in the planning horizon
ρ_a	corridor attribute of arc a
μ_a	station attribute of arc a
ϕ_a	duty ID attribute of arc a
ω_a	duty day attribute of arc a
$\bar{A}_d$	set of selected train duties for change in exchange stations, $\bar{A}_d \subseteq A_d$
$\bar{\bar{A}}_d$	expanded set of train duties, $\bar{\bar{A}}_d \subseteq \bar{A}_d$, such that: $\forall \bar{a} \in \bar{A}_d$, $\exists a \in \bar{\bar{A}}_d$: $\mu_a = v$, $\forall v \in V_{p_a}$, $\phi_a = \phi_{\bar{a}}$, $\omega_a = \omega_{\bar{a}}$

Table 5.1: Notation for integer programming formulation.

station along the same corridor; selecting I_2' as a crew exchange station instead of I_2 can result in some savings with respect to the total number of crew members required in regions H_0 and H_2. Considering different policies and the possible changes in all corridors, the number of decisions we can make is substantial.

We formulate three problems based on the original crew capacity planning problem formulation. Additional notation for the integer programming formulation is given in Table 5.1. We define $\bar{A}_d$ as the set of selected train duties for change in the crew exchange stations. We, then expand the set $\bar{A}_d$ by creating a duty arc for each candidate crew exchange station. The new set is called $\bar{\bar{A}}_d$. Duties in the set $\bar{\bar{A}}_d$ share the same duty attributes except for the time attributes which are set according to the arrival time of trains to different crew exchange stations. In other words, for a single train duty in the set $\bar{A}_d$, we create a set of arcs (as much as the number of possible crew exchange stations) with the same train ID, duty day and corridor attributes. However, each arc has a different station attribute and different time windows associated with the time of arrival (departure) of the train to (from) the crew exchange station denoted by the arc. We define the common decision variable x_a as the amount of flow on arc a.

Case 1: Multiple Train IDs and Flexible Crew Exchange Stations

In this case, we consider multiple trains covered by a region in order to determine their crew exchange stations; we do not have any constraints on the crew change stations of trains with the same ID. In other words, the same train can exchange crews at one station on Monday and at another station on Tuesday. For the formulation, we define:

$$y^v_{phg} = \begin{cases} 1, & \text{if } v \text{ is used as crew exchange station for train ID } h, \text{ duty day } g \\ & \text{and corridor } p; \\ 0, & \text{otherwise.} \end{cases}$$

The integer programming formulation is written as follows:

$$\text{Minimize} \sum_{a \in A_s} x_a \tag{5.1}$$

$$\text{subject to} \sum_{a \in A_s} x_a = \sum_{a \in A_t} x_a, \tag{5.2}$$

$$\sum_{a \in A_{n^+}} x_a = \sum_{a \in A_{n^-}} x_a, \qquad \forall n \in N \setminus \{s, t\}, \tag{5.3}$$

$$x_a \geq c_a, \qquad \forall a \in A \setminus \bar{A}_d, \tag{5.4}$$

$$x_a \geq c_a y^{\mu_a}_{\rho_a \phi_a \omega_a}, \qquad \forall a \in \bar{\bar{A}}_d, \tag{5.5}$$

$$\sum_{v \in V_{\rho_a}} y^v_{\rho_a \phi_a \omega_a} = 1, \qquad \forall a \in \bar{A}_d, \tag{5.6}$$

$$x_a \in \mathbb{Z}_+, \qquad \forall a \in A, \tag{5.7}$$

$$y^v_{\rho_a \phi_a \omega_a} \in \{0, 1\}, \qquad \forall \rho_a \in P, \forall \phi_a \in H, \forall \omega_a \in G, \forall v \in V_{\rho_a}. \tag{5.8}$$

Compared with the minimum flow formulation in (3.1)-(3.5), the coverage constraint for duties in the set $\bar{\bar{A}}_d$ (5.5) is now conditioned on the selection of crew exchange stations. Constraint (5.6) selects only one crew exchange station for a train duty. Therefore, constraints (5.5) and (5.6) together guarantee that at least one arc belonging to each duty must be covered by the required number of crew members for that duty. Note that the formulation (5.1)-(5.8) does not imply an upper bound on the amount of flow on the other arcs representing a duty and home-to-away deadheading is still possible.

Case 2: Multiple Train IDs and Semi-Fixed Crew Exchange Stations

In this case, we consider a model, where train duties with the same IDs must use the same crew exchange stations. Yet, train duties with different IDs can use different crew exchange stations as long as the replicates of these trains on different days use the same station. As a result of this, we drop the day index g from the binary variable y^v_{phg}, and define:

$$y^v_{ph} = \begin{cases} 1, & \text{if train with train ID } h \text{ and corridor } p \text{ uses station } v \text{ as crew exchange} \\ & \text{station for each day of the week;} \\ 0, & \text{otherwise.} \end{cases}$$

The integer programming formulation is as follows:

$$\text{Minimize} \sum_{a \in A_s} x_a \tag{5.9}$$

$$\text{subject to}(5.2) - (5.4), (5.7), \tag{5.10}$$

$$x_a \geq c_a y^{\mu_a}_{\rho_a \phi_a}, \qquad \forall a \in \bar{\bar{A}}_d, \tag{5.11}$$

$$\sum_{v \in V_{\rho_a}} y^v_{\rho_a \phi_a} = 1, \qquad \forall a \in \bar{A}_d, \tag{5.12}$$

$$y^v_{\rho_a \phi_a} \in \{0, 1\}, \qquad \forall \rho_a \in P, \forall \phi_a \in H, \forall v \in V_{\rho_a}. \tag{5.13}$$

Compared to the previous formulation, (5.1)-(5.8), the only change in the formulation is the lack of day index in the binary variable y^v_{ph}. This change implies that the crew exchange stations are chosen common for the daily replicates of a train duty.

Case 3: Multiple Train IDs and Fixed Crew Exchange Stations

In this case, trains operating over a corridor between two regions must all change crews at the same station. As a result of this, we also drop the index h from the decision variable and define:

$$y_p^v = \begin{cases} 1, & \text{if all trains in corridor } p \text{ use station } v \text{ as crew exchange station;} \\ 0, & \text{otherwise.} \end{cases}$$

The integer programming formulation is written as follows:

$$\text{Minimize} \sum_{a \in A_s} x_a \tag{5.14}$$

$$\text{subject to}(5.2) - (5.4), (5.7), \tag{5.15}$$

$$x_a \geq c_a y_{\rho_a}^{\mu_a}, \qquad \forall a \in \bar{\bar{A}}_d, \tag{5.16}$$

$$\sum_{v \in V_{\rho_a}} y_{\rho_a}^v = 1, \qquad \forall a \in \bar{A}_d, \tag{5.17}$$

$$y_{\rho_a}^v \in \{0, 1\}, \qquad \forall \rho_a \in P, \forall v \in V_{\rho_a}. \tag{5.18}$$

Compared to the formulation (5.1)-(5.8), the only change in the formulation is the lack of day and train ID index in the binary variable y_p^v. This change implies that only a single crew exchange station is chosen for all daily replicates of all trains operating on corridor p.

Even though constructing these models help us understand the potential shortcomings of the tactical-to-strategic capacity planning problem, such as the infeasibility problems that may arise, studying the capacity planning problem on a single region does not have practical benefits. As we are trying to minimize the crew capacity of a region, the objective can be achieved by selecting the stations on a corridor that are close to the home station, without considering the burdens it may bring to the neighboring regions. For this purpose, we develop two-region problems, which deal with that problem.

5.2 Two-region Problem with Allocation of Trains

In this section, we study models that consider two neighboring regions simultaneously. Contrary to the single region models, by studying two-region models we consider the system-wide effects of the modifications carried out in the train schedules. As the train duties are divided between two regions, we have to make sure that crews belonging to different regions cover the entire train duty. For example, if crew exchange is executed at I_1 for a train duty from H_0 to H_1, then the first part of the duty (from H_0 to I_1) and the second part of the duty (from I_1 to H_1) must be covered by the crew members from H_0 and H_1, respectively.

5.2.1 Mathematical Formulation

For the two-region problem, we consider two policies for crew exchange stations: flexible and fixed policies. Similarly to the policies discussed in Section 5.1.2, in the flexible case, each daily replicate of a train can use any potential crew exchange stations whereas in the fixed case, all trains operating on the corridor must use the same crew exchange station. For illustrative purposes, we consider only a single train that runs from one home station to the other of two neighboring regions. Yet, extending the formulations to consider multiple trains is straightforward.

Table 5.2 shows the notation for the two-region models. Duties e and f have following properties for their attributes: $\rho_e = \rho_f = p$ and $\phi_e = \phi_f = h$ and $\omega_e = \omega_f = g$. The only difference in the two train duties is their origins and destinations and the time attributes associated with them.

Case 1: Single Train ID and Flexible Crew Exchange Station

In this case, we consider a single train whose duty can be shared by the two neighboring regions. If crew members from one region cover the trip from the origin to

E, F	two neighboring regions
p	the corridor between regions E and F
e, f	duties belonging to the sets $\bar{\bar{A}}_d^E$ and $\bar{\bar{A}}_d^F$ of regions E and F respectively

Table 5.2: Additional notation for integer programming formulation.

the crew exchange station I_1 (train duty e in Table 5.2), then crew members from the other region should cover the train duty from the crew exchange station I_1 to the destination (train duty f in Table 5.2). To formulate this problem, we have to create two networks belonging to two regions. In these two networks, there is a set of arcs whose coverage is dependent on each other in a pairwise fashion. Using the same example as above, the duty arc corresponding to the train duty e should be covered or not covered simultaneously with the arc corresponding to the train duty f. The two duty arcs are to be covered only when the crew exchange station is selected as I_1. They are both not necessarily covered when I_1 is not selected as the crew exchange station. Solving the two-region problem means solving two single-region problems simultaneously with a set of common decision variables selecting the exchange stations. The integer programming formulation is given as:

$$\text{Minimize} \sum_{h\in\{E,F\}} \sum_{a\in A_s^h} x_a \tag{5.19}$$

$$\text{subject to} \sum_{a\in A_s^h} x_a = \sum_{a\in A_t^h} x_a, \qquad \forall h \in \{E,F\}, \tag{5.20}$$

$$\sum_{a\in A_{n^+}^h} x_a = \sum_{a\in A_{n^-}^h} x_a, \qquad \forall n \in N^h \setminus \{s,t\}, \forall h \in \{E,F\}, \tag{5.21}$$

$$x_a \geq c_a, \qquad \forall a \in \cup_{h\in\{E,F\}}(A^h \setminus \bar{A}_d^h), \tag{5.22}$$

$$x_e \geq c_e y_{\rho_e \phi_e \omega_e}^{\mu_e}, \qquad \forall e \in \bar{\bar{A}}_d, \forall g \in G, \tag{5.23}$$

$$x_f \geq c_f y_{\rho_f \phi_f \omega_f}^{\mu_f}, \qquad \forall f \in \bar{\bar{A}}_d, \forall g \in G, \tag{5.24}$$

$$\sum_{v \in V_p} y^v_{phg} = 1, \qquad \forall g \in G, \qquad (5.25)$$

$$x_a \in \mathbb{Z}_+, \qquad \forall a \in (A^E \cup A^F), \qquad (5.26)$$

$$y^v_{\rho_a} \in \{0, 1\}, \qquad \forall \rho_a \in P, \forall v \in V_{\rho_a}. \qquad (5.27)$$

Constraint (5.23) guarantees the coverage of duty e by crew members from crew region E from the origin of duty until station v. Constraint (5.24) is the coverage constraint of duty f by crew members from region F from the crew exchange station v until the destination of duty. Constraint (5.25) is used to choose a crew exchange station for the duty. As we are dealing with a flexible crew change station policy, the constraints (5.23)-(5.25) must be honored for each replicate of the train during the week; together, they ensure that the train duty is covered as required for the entire length of the trip.

Case 2: Single Train ID and Fixed Crew Exchange Station

In this case, we again have to cover the whole trip for the train operating between two regions. Duties e and f have following properties for their attributes: $\rho_e = \rho_f = p$. However, the space and time attributes of the two train duties are different as it was the case in the previous models. We can write the following integer programming formulation by using the notation in Table 5.2:

$$\text{Minimize} \sum_{h \in \{E,F\}} \sum_{a \in A^h_s} x_a \qquad (5.28)$$

$$\text{subject to} (5.20) - (5.22), (5.26), \qquad (5.29)$$

$$x_e \geq c_e y^{\mu_e}_{\rho_e}, \qquad \forall e \in \bar{\bar{A}}_d, \qquad (5.30)$$

$$x_f \geq c_f y^{\mu_f}_{\rho_f}, \qquad \forall f \in \bar{\bar{A}}_d, \qquad (5.31)$$

$$\sum_{v \in V_p} y^v_p = 1, \qquad (5.32)$$

$$y^v_{\rho_a} \in \{0,1\}, \qquad\qquad \forall \rho_a \in P, \forall v \in V_{\rho_a}. \tag{5.33}$$

When compared to constraints (5.23)-(5.25), the only difference in constraints (5.30)-(5.32) is the lack of day and train ID index on decision variable y^v_p.

5.2.2 Solution Method for Two-Region Problem: A Neighborhood Search Algorithm

Solving the problems for two-region case requires adding many binary decision variables and simultaneously solving the capacity planning problem of two regions. Hence, it is exponentially more difficult to solve when compared to a single-region problem. We develop a solution method to efficiently solve these problems. We know that the capacity planning problem, when the crew exchange stations are given, can be solved with the minimum flow problem formulation described in Section 3.1.1. It is easy to observe that when the crew exchange stations are given, the two-region problems reduce to two separate capacity planning problems. Based on this observation, we devise a solution procedure for the two-region problems that contains the solution of the minimum flow problem formulation of the capacity planning problem. Let us assume that we are considering the re-allocation of a train duty between two regions, H_0 and H_1. Let us further assume that this train is assigned to region H_0 according to the current assignments. There are indeed several options for re-allocating the corresponding train duty. We can either assign the train duty fully to region H_1 or share the train duty between the two regions by selecting one of the stations along the corridor from H_0 to H_1 as the crew exchange station. To understand if re-allocation of this train would bring any savings in the total number of crew members required in the two regions, we first consider the extreme opposite solution to the current state: full assignment of the train duty to region H_1. Assigning the train duty to H_1 instead of H_0, we may observe the following changes in the number of crew members

Case	H_1	H_2	Overall
Case 1	Same	Same	Same
Case 2	Same	Increase	Increase
Case 3	Decrease	Increase	Same
Case 4	Decrease	Increase	Increase
Case 5	Decrease	Increase	Decrease
Case 6	Decrease	Same	Decrease

Table 5.3: Changes in the number of crew members required when a train duty is currently assigned to H_0 is assigned to H_1.

required:

- Case 1: The number of crew members in H_0 and H_1 stay the same, resulting in the same minimum crew capacity for the company.
- Case 2: The number of crew members in H_0 stays the same, whereas it increases in H_1, resulting in an increase in the minimum crew capacity.
- Case 3: The number of crew members in H_0 decreases, whereas it increases in H_1, resulting in the same minimum crew capacity.
- Case 4: The number of crew members in H_0 decreases, whereas it increases in H_1, resulting in an increase in the minimum crew capacity.
- Case 5: The number of crew members in H_0 decreases, whereas it increases in H_1, resulting in a decrease in the minimum crew capacity.
- Case 6: The number of crew members in H_0 decreases, whereas it stays the same in H_1, resulting in a decrease in the minimum crew capacity.

Therefore, six different cases may be observed. The changes according to these cases are also summarized in Table 5.3.

In Case 2 and Case 4, the total number of crew members required in the two region increases when the train duty is shifted from H_0 to H_1. This means that the current assignment of the train duty (to H_0) is better than the new allocation. In Case 1 and Case 3 , the number of crew members required in two regions stays the same when we shift the duty to H_1. In Case 5 and Case 6, the number of crew members required in two regions decreases, meaning that the new allocation of the train duty (to H_1) is a better option than the current assignment.

The algorithm we propose is based on a neighborhood search idea. We note that re-allocating a train duty to another region or splitting a train duty between two regions is a neighboring solution to the current solution. The neighborhood search algorithm is described in Algorithm 3. In the algorithm, TS denotes a feasible allocation of train duties among the regions of interest. TC denotes the train tuples whose allocations we consider modifying. The tuples may be of any size. Starting from an initial solution, which is an input to the algorithm, we initialize a greedy search procedure. At each iteration of the search procedure, by re-assigning a tuple of trains to the other region (i.e. to the region the duty is not assigned in the initial TS), we find the tuples that bring the most improvement in the number of required crew number members (lines 9-18). If there is a tie among the tuples we store all of them as candidates for re-allocating the corresponding duties (lines 17-18). If we fail to improve the current solution, we include tuples that does not deteriorate the solution as a candidate solution for re-allocation. For every tuple in the candidate set, we determine the common stations for the train duties in the tuple we are studying (line 20). We start dividing the train duties by beginning from the closest station to the best allocation, which is the closest station to H_1 in this specific example. After searching for all possible stations, we select the best crew exchange station (or the home station of H_1 if there is no improvement by splitting the train duties) and update the assignment of duties accordingly (lines 21-27). In re-allocation of a duty, even if there is a tie among different tuples or stations, we only store one solution (line 23). The next iteration will continue from this new solution. We repeat this procedure until we cannot improve the current incumbent solution by re-allocating or splitting duties. At the end of each iteration, we fix the train duties we altered (line 28) and we do not consider them in the subsequent iterations as candidates (line 9).

5.2.3 Computational Study

For the capacity planning problem where we consider the allocation of trains and new exchange stations, we perform a computational study with the same data set that we use for the tactical capacity planning problem in Chapter 4. Before presenting the results, explaining the characteristics of the physical infrastructure and trains operated by different regions can be insightful. The three crew-base regions of interest, namely Ankara, Eskişehir, and Haydarpaşa, lie on a single corridor between Ankara and Haydarpaşa, with Eskişehir situated between the other two regions. The train line between Ankara and Haydarpaşa is the most important part of TCDD network with several big cities located around the train line. Therefore, optimizing the crew operations of these three regions is very crucial for the business model of the company. There are several trains in the weekly schedule running between Ankara and Haydarpaşa. With respect to the train duties, some of them are completely assigned to Ankara and some of them are assigned to Eskişehir region with the duty already divided in two parts, between Ankara and Eskişehir, between Eskişehir and Haydarpaşa.

In the data set we used, Ankara region is responsible for operating 35 train duties, whereas Haydarpaşa and Eskişehir regions operate 44 train duties each. Most of these trains operate between the home stations of these regions. However, most of the train duties assigned to Haydarpaşa region are relatively short duties between Haydarpaşa and important cities close by. Among these three regions Eskişehir carries the largest

Algorithm 3: Algorithm for the neighborhood search.

Input: TS, TC
$best = \mathtt{OFV}(TS)$
$continue =$ `true`
forall $tr \in TC$ **do**
  $tabu[tr] =$ `false`
while $continue$ **do**
  $continue =$ `false`
  $TC_{best} = \{\}$ *Re-initialize the set of candidate tuples*
  forall $tr \in TC : tabu[tr] =$ `false` **do**
    $TS_{temp} = \mathtt{reassign}(tr)$ */Re-assign the train duties in tr to another region.*
    if $\mathtt{OFV}(TS_{temp}) < best$ **then**
      $TC_{best} = \{tr\}$ */Store the most promising train tuple.*
      $best = \mathtt{OFV}(TS_{temp})$
      $TS = TS_{temp}$
      $tr_{best} = tr$
      $continue =$ `true`
    else if $\mathtt{OFV}(TS_{temp}) = best$ **then**
      $TC_{best} = TC_{best} \cup \{tr\}$ */Store the most promising train tuples.*
  forall $tr \in TC_{best}$ **do**
    $ST = commonstations(tr)$ */Find the common stations for train duties in tr.*
    forall $st \in ST$ **do**
      $TS_{temp} = \mathtt{share}(tr, st)$*/Split the duty using the crew exchange station st.*
      if $\mathtt{OFV}(TS_{temp}) < best$ **then**
        $best = \mathtt{OFV}(TS_{temp})$*/Store the improvement obtained by duty sharing.*
        $TS = TS_{temp}$
        $tr_{best} = tr$
        $continue =$ `true`
  $tabu[tr_{best}] =$ `true`*/Fix the train duty and remove it from further consideration.*
Output: $TS, best$

workload with several trains operating to both Haydarpaşa and Ankara. Most of the trains assigned to Ankara region belong to the same corridor, with Eskişehir and Haydarpaşa frequently appearing as away stations. Ankara region also operates some duties to other cities lying on a different corridor.

In our study, we only consider trains operating between any two of the three crew-base regions as potential candidates for re-allocation and modifications in crew exchange stations. In other words, duties belonging to different corridors or operating between the home station (of one of the three regions) and an intermediate station (along the corridor) are considered as fixed in the train duty assignments of the three regions. In our experiments, we observed that considering reallocation of a single train usually leads to infeasible solutions when we try to split the train duty between two regions. For this purpose, we create train couples by observing the train schedules of regions. These couples have the same end points but the trains in the couple operate in opposite directions. However, this is not a restriction for our algorithm and it is possible to perform the search with a single train ID or with more than two train IDs. The problem we solve corresponds to a two-region problem with multiple train IDs. We employ a semi-fixed policy, which is more convenient to the crew members.

Table 5.4 shows the results. The first row of the table shows the number of crew members required with the current assignment of duties. We show a single iteration in two consecutive rows and consecutive iterations are separated by a horizontal line. If the train duties of the corresponding iteration are not to be shared by the two regions, this iteration is shown with "N/A" under the "Exchange Station" column. To exemplify, in the first iteration, we re-assign a train duty couple from Eskişehir to Haydarpaşa and reduce the number of required crew members by two, from 173 to 171 in total. This can be seen in column "Total" under the "Crews" header. We, then, try to split the duties by considering potential crew exchange stations, but we cannot improve the solution by splitting. In the second iteration, we assign another train duty couple from Eskişehir to Haydarpaşa and decrease the total number of crew members from 171 to 170. In this iteration, selecting Karaköy as a crew exchange station, we are able to reduce the total number of crew members to 167 (see columns under the “Splitting” header). We continue to search until no improvement is made at an iteration. In iterations 7 and 11, the re-allocation step fails to improve the current solution. However, after splitting duties the total number of crew members decreases and we are able to continue to search.

Iteration Number	Re-allocation: Current Region	Re-allocation: Duty Duration	Re-allocation: New Region	Re-allocation Crews: Esk.	Re-allocation Crews: Hay.	Re-allocation Crews: Ank.	Re-allocation Crews: Total	Splitting: Exchange Station	Splitting Duty Duration: Part I	Splitting Duty Duration: Part II	Splitting Crews: Esk.	Splitting Crews: Hay.	Splitting Crews: Ank.	Splitting Crews: Total
0				76	46	51	173							
1	Esk.	642 558	Hay.	70	50	51	171	N/A						
2	Esk.	584 582	Hay.	66	53	51	170	Karaköy	155 125	429 457	66	50	51	167
3	Esk.	481 570	Ank.	62	50	54	166	Yalınlı	181 178	300 392	63	50	51	164
4	Esk.	511 520	Ank.	59	50	54	163	Alpu	109 108	402 412	60	50	52	162
5	Esk.	629 586	Ank.	57	50	55	162	Alpu	87 113	542 473	57	50	54	161
6	Esk.	457 477	Ank.	55	50	55	160	N/A						
7	Esk.	1052 973	Hay.	51	54	55	160	Bozüyük	148 152	904 821	50	54	55	159
8	Esk.	688 603	Hay.	45	58	55	158	Arifiye	249 261	439 342	47	55	55	157
9	Hay.	432 406	Esk.	49	52	55	156	N/A						
10	Esk.	397 378	Hay.	47	53	55	155	N/A						
11	Esk.	446 487	Ank.	44	53	58	155	Polatlı	194 192	252 295	45	53	56	154

Table 5.4: Results of tactical-to-strategic capacity planning problem.

The results in Table 5.4 lead to an important observation. In seven of 11 iterations, 13 train duties constituting seven train couples last more then the double-manning time, 480 minutes. By splitting these duties, we obtain 16 duties and 13 of them can be covered by a single crew member. As we predict, avoiding duties longer than the double-manning time accounts for a significant part of the overall improvement.

With this example, we show that it is possible to reduce the total number of crew members in the three regions from 173 to 154 by assigning duties to other regions or splitting them considering different crew exchange stations. This result corresponds to a 11% decrease in the total number of crew members and can bring substantial benefits to the company.

Note that the result showed here is different from the one in Table 4.1 as Ankara region requires one less crew member. This change is due to the new away-to-home deadhead and away-to-away deadhead with direct connection policies we discussed in Section 5.1.1. This shows that the new deadhead policies are an improvement over the current policy and it should be considered by managers of the company. Note that in the initial assignment away-to-away deadhead policies with an away rest are not active as we only use them for train duties shared by two regions.

In order to see the effect of the new deadhead policies, we also perform our computational study by using the network representation discussed in Section 2.2, without considering new deadhead policies. The results are presented in Table 5.5. Without the new deadhead policies, we are able to execute seven iterations only in contrast to the 11 iterations in the previous case. There are only two train couples that was split with a new crew exchange station, whereas new deadhead policies have allowed seven train couples to be split in different crew exchange stations. However, even in this case, we succeed to reduce the number of required crew members from 174 to 165, resulting in a 5% improvement.

As a managerial insight, we can say that considering new deadhead policies can bring substantial improvements to the crew-related costs in the company. Not only the new deadhead policies produced better results in terms of the objective function value, they also served their purpose of avoiding infeasibilities as we were able to find a feasible solution in nearly all instances we solved. However, when we omit the new deadhead policies from the space-time network, there were several instances, where a candidate crew exchange station failed to produce a feasible solution. This observation can also be supported by the number of duty sharings between two methods (seven to two).

	Re-allocation							Splitting						
				Crews					Duty Duration		Crews			
Iteration Number	Current Region	Duty Duration	New Region	Esk.	Hay.	Ank.	Total	Exchange Station	Part I	Part II	Esk.	Hay.	Ank.	Total
0				76	46	52	174							
1	Ank.	304	Esk.	77	46	49	172	N/A						
		292												
2	Esk.	642	Hay.	71	50	49	170	N/A						
		558												
3	Hay.	403	Esk.	72	48	49	169	N/A						
		396												
4	Hay.	432	Esk.	74	45	49	168	N/A						
		406												
5	Esk.	511	Ank.	70	45	52	167	N/A						
		520												
6	Esk.	481	Ank.	67	45	55	167	Çardakbaşı	148	333	68	45	53	166
		570							136	434				
7	Esk.	688	Hay.	63	50	53	166	Arifiye	249	439	65	47	53	165
		603							261	342				

Table 5.5: Results of tactical-to-strategic capacity planning problem without extended deadhead policies.

6 CONCLUSION

In our work, we study the tactical crew capacity planning problem with day-off requirement at TCDD. The capacity planning problem is used to determine the necessary number of crew members in a region to operate the predetermined list of train duties assigned to the region by complying business rules and labor union policies.

In our study, we follow a network flow approach to solve the crew capacity planning problem represented by a space-time network constructed with respect to the rules and policies at TCDD. In order to represent the capacity planning problem with day-off requirement, we develop two solution approaches: the sequential approach and the integrated approach. In the sequential approach, we solve the problem in two stages, by solving a regular capacity planning problem without considering the day-off requirement in the first stage and filling-in day-off windows of the crew schedules obtained in the regular capacity planning problem in the second stage. In the integrated approach, we use a layered version of the space-time network in order to integrate the day-off requirement into the space-time network representation. We conclude from our computational study that the integrated approach is superior to the sequential approach in both solution quality and computation times.

After studying the tactical level problem, we formulate some tactical-to-strategic level planning problems focusing on system-wide improvements (or savings) in crew capacity by optimally re-allocating duties to different regions or allowing two regions optimally share a train duty by changing crew members at new crew exchange stations. We propose a greedy neighborhood search algorithm for the tactical-to-strategic crew capacity planning problem by using the minimum flow problem formulation of the tactical level problem as a subprocedure. The experiments we conduct result in substantial improvements in the system-wide crew capacity of the company highlighting the future research potential of strategic level problems in order to reduce crew-related costs at TCDD.

With respect to the state of current railroad planning and crew planning literature, this study is the first to focus on tactical and tactical-to-strategic level crew planning problems. In railroad crew planning literature, several variants of the operational level problem have been studied. Yet, we believe that our study is late enough for the fact that operational level problems require several inputs to be optimally determined at higher levels of the planning hierarchy starting with the number of crew members in each region. Therefore, this study opens a new research area in railroad planning that will also lead to other studies along the same path.

As future work, we intend to study other solution approaches for the tactical capacity planning problem, such as using set covering or set partitioning formulations combined with column generation. Using the integrated approach to solve the tactical-to-strategic capacity planning problem is another future research topic. However, considering the efficiency of the minimum flow problem in the sequential approach, the integrated approach would probably require additional work to decrease the computation time for the integrated approach before replacing the minimum flow problem formulation as a subprocedure. Devising a heuristic to provide an initial feasible solution to the model or exploring different solutions methods can be an option to solve the problem more efficiently. Improving the greedy neighborhood search in the tactical-to-strategic level problem by employing a more thorough search procedure can lead to better numerical results for this problem. It might also be interesting to focus on the exact integer programming formulations of the tactical-to-strategic capacity planning problem.

Bibliography

[1] G. Şahin, "New combinatorial approaches for solving railroad planning and scheduling problems," PhD Thesis, The University of Florida, Gainesville, FL, August 2006.

[2] R. K. Ahuja, C. B. Cunha, and G. Şahin, "Network flow approaches in railroad planning and scheduling," in *TutORials in Operations Research*, J. C. Smith, Ed. INFORMS, 2005, pp. 54–101.

[3] E. Abbink, J. V. Wout, and D. Huisman, "Solving large scale crew scheduling problems by using iterative partitioning," in *ATMOS 2007 - 7th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, C. Liebchen, R. K. Ahuja, and J. A. Mesa, Eds. Dagstuhl, Germany: Internationales Begegnungs und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2007.

[4] Turkish State Railways (TCDD), "General expenditures of TCDD," http://www.tcdd.gov.tr/genel/yillik/086.pdf, retrieved on August 1, 2009.

[5] A. Caprara, M. Fischetti, P. Toth, D. Vigo, and P. L. Guida, "Algorithms for railway crew management," *Math. Program.*, vol. 79, no. 1-3, pp. 125–141, 1997.

[6] A. Caprara, M. Fischetti, P. L. Guida, P. Toth, and D. Vigo, "Solution of large-scale railway crew planning problems: the italian experience," in *In: N.H.M. Wilson (Ed.), Computer-Aided Transit Scheduling, Lecture Notes in Economics and Mathematical Systems.* Springer, 1997, pp. 1–18.

[7] A. Caprara, M. Monaci, and P. Toth, "A global method for crew planning in railway applications," in *In: S. Voß and J. R. Daduna (Eds.), Computer-Aided*

Scheduling of Public Transport, Lecture Notes in Economics and Mathematical Systems, 505. Springer, 2001, pp. 17–36.

[8] B. Vaidyanathan, K. C. Jha, and R. K. Ahuja, "Multicommodity network flow approach to the railroad crew-scheduling problem," *IBM Journal of Research and Development*, vol. 51, no. 3, pp. 325–344, 2007.

[9] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, February 1993.

[10] T. Mellouli, "A network flow based approach to crew scheduling based on an analogy to an aircraft/train maintenance routing problem," in *In: S. Voß and J. R. Daduna (Eds.), Computer-Aided Scheduling of Public Transport, Lecture Notes in Economics and Mathematical Systems, 505.* Springer, 2001, pp. 91–120.

[11] Y. Guo, T. Mellouli, L. Suhl, and M. P. Thiel, "A partially integrated airline crew scheduling approach with time-dependent crew capacities and multiple home bases," *European Journal of Operational Research*, vol. 171, no. 3, pp. 1169–1181, June 2006.

[12] A. T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier, "Staff scheduling and rostering: A review of applications, methods and models," *European Journal of Operational Research*, vol. 153, no. 1, pp. 3–27, February 2004.

[13] J. Möller, "Seminar on algorithms and models for railway optimization crew scheduling," 28 August 2002. [Online]. Available: http://www.inf.uni-konstanz.de/algo/lehre/ss02/rails/moeller.pdf

[14] P. H. Vance, C. Barnhart, E. L. Johnson, and G. L. Nemhauser, "Airline crew scheduling: A new formulation and decomposition algorithm," *Operations Research*, vol. 45, pp. 188–200, 1995.

[15] R. Anbil, J. J. Forrest, and W. R. Pulleyblank, "Column generation and the airline crew pairing problem," *Documenta Mathematica*, vol. Extra Volume ICM, III, pp. 677–686, 1998.

[16] M. Stojković, F. Soumis, and J. Desrosiers, "The operational airline crew scheduling problem," *Transportation Science*, vol. 32, no. 3, pp. 232–245, 1998.

[17] C. P. Medard and N. Sawhney, "Airline crew scheduling from planning to operations," *European Journal of Operational Research*, vol. 183, no. 3, pp. 1013–1027, December 2007.

[18] E. M. Morgado and J. P. Martins, "CREWS_NS: Scheduling train crews in the netherlands," *AI Magazine*, vol. 19, no. 1, 1998.

[19] L. Kroon and M. Fischetti, "Crew scheduling for netherlands railways - destination: Customer," 2000.

[20] R. Freling, R. M. Lentink, and M. A. Odijk, "Scheduling train crews: a case study for the dutch railways," in *In Proceedings of CASPT 2000*, 2000.

[21] T. H. Yunes, A. V. Moura, and C. C. de Souza, "Solving very large crew scheduling problems to optimality," in *SAC '00: Proceedings of the 2000 ACM symposium on Applied computing*. New York, NY, USA: ACM, 2000, pp. 446–451.

[22] Havelsan, personal communication, 2009.

[23] J. Liu, R. K. Ahuja, and G. Şahin, "Optimal network configuration and capacity expansion of railroads," *Journal of the Operational Research Society*, vol. 59, pp. 911–920, 2008.