Decision Support for Disruption Management On High Frequency Transit Lines

by

Michel David Babany

Bachelor’s Degree in Engineering
Ecole Centrale Paris, 2012

Submitted to the Department of Civil and Environmental Engineering in partial fulfillment of the requirements for the degree of

Master of Science in Transportation
at the

Massachusetts Institute of Technology

June 2015

© 2015 Massachusetts Institute of Technology. All rights reserved.
Decision Support for Disruption Management On High Frequency Transit Lines

by

Michel David Babany

Submitted to the Department of Civil and Environmental Engineering on May 21, 2015, in partial fulfillment of the requirements for the degree of Master of Science in Transportation

Abstract

Incidents (due to equipment failures, passenger emergencies, infrastructure problems, human errors, etc.) routinely occur in metro systems. Such incidents can cause significant disruptions in service (from slowdown to full closure of the line), with serious impacts on passengers, especially in the core of high frequency lines operating near capacity. Disruption consists of two distinct phases. The incident phase is the period from the start of the incident to the moment when its cause has been resolved. The second phase of the disruption is the recovery, which starts at the end of the incident and lasts until normal service is restored. Dealing efficiently with disruptions is crucial and agencies use real-time control strategies to mitigate those impacts and improve performance.

This thesis proposes an approach for supporting controllers decision-making in the recovery phase of disruption management. While the method is applied to the Piccadilly Line on the London Underground, it is applicable to other high frequency transit rail lines.

After reviewing the main challenges controllers face during incident management and the main strategies they use, the thesis formulates the recovery phase problem as an optimization problem that integrates timetable revision and crew rescheduling (train reformation problem, TRP).

The approach focuses on modeling common control strategies such as short-turning and train renumbering. It explicitly incorporates the scarcity of resources and associated constraints, especially with respect to crews. The method consists of two phases: the generation of a large number of candidate journeys; and the selection of the journeys (recovery timetable) that optimize some measure of performance, involving the effectiveness of the recovery and the passenger service.
The model is first applied to an incident that happened on January 2014 on the Piccadilly Line. The actual controllers response is compared with the output of the train reformation problem, and a sensitivity analysis of the model parameters is performed. The results suggest that using more complex reformations and less short-turns may lead to better passenger service during the recovery phase. The train reformation problem is then applied to a hypothetical incident. The results support current practices that canceling trains during the incident phase enables a shorter and more efficient recovery.

The results of the case studies also suggest that a decision support system for controllers has the potential to improve service recovery. The thesis concludes with recommendations for how a decision support system could be implemented in practice, and how the method could be extended to the incident phase of service disruptions.

Thesis Supervisor: Nigel H.M. Wilson
Title: Professor of Civil and Environmental Engineering

Thesis Supervisor: Haris N. Koutsopoulos
Title: Professor of Civil and Environmental Engineering
Northeastern University
Acknowledgments

Before all, I would like to thank my research advisors, professors Nigel Wilson and Haris Koutsopoulos. Without your guidance, suggestions, knowledge, and even productive arguments, this thesis would have never been possible. I would also like to thank others at MIT, faculty and staff, in particular Ginny and Shannon.

I am very grateful to Transport for London who supported this research. I would like to thank especially Chris Baker, Dale Campbell, Nigel Kelt, Jemima Poole, Lauren Sager Weinstein, Chris Taggart, Andrew Toplis, and Sandra Wedell, for their time and advice. I would like to add special thanks to everyone in the Earl’s Court centre and in the crew depots, for having let me disturb them, and who help me discover so many fascinating facts about the Piccadilly Line.

I cannot forget to thank my friends in the Transit Lab: Steve Tuttle, Lauren Tarte and Gabriel Goulet-Langlois for the time we spent in London together; I must also mention the Transit Lab students of year ’14, who failed to sing my name correctly; every transit lab mate, who saw my progress in yoyo and juggling over the past two years; my MST classmates, for the coffee hours we spent together. Finally, thanks to all my friends I met in Boston, who have made these two years at MIT an incredible and extraordinary experience.
# Contents

1 Introduction ................................................. 17
   1.1 Context and Motivation ................................. 17
   1.2 Objectives ........................................... 20
   1.3 Research Approach .................................... 21
   1.4 Organization ......................................... 22

2 Disruption Management ...................................... 25
   2.1 High-Frequency Service ................................. 25
   2.2 Service Delivery Process ............................... 26
   2.3 Disruption ............................................ 27
   2.4 Service Control ....................................... 28
      2.4.1 Control Components ............................... 29
      2.4.2 Control Practices ................................. 33
      2.4.3 Emergency Timetable ............................. 37

3 Introduction to the Piccadilly Line and Literature Review .......... 39
   3.1 Background ........................................... 39
      3.1.1 Line Geometry ................................... 40
      3.1.2 Demand and Service ............................... 43
      3.1.3 Service Disruptions ............................... 46
      3.1.4 Service Control Organization .................... 48
   3.2 Previous Research on Piccadilly Line ................... 51
   3.3 Literature Review .................................... 52
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Decision Support Model: the Train Reformation Problem</td>
<td>57</td>
</tr>
<tr>
<td>4.1</td>
<td>The Train Reformation Problem</td>
<td>58</td>
</tr>
<tr>
<td>4.2</td>
<td>Journey Generation</td>
<td>62</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Timetable Expansion</td>
<td>62</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Group Reformation</td>
<td>64</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Station Arrival and Departure Times</td>
<td>66</td>
</tr>
<tr>
<td>4.3</td>
<td>Journey Selection</td>
<td>69</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Objective Function</td>
<td>70</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Constraints</td>
<td>74</td>
</tr>
<tr>
<td>4.4</td>
<td>Extension to the Incident Phase</td>
<td>77</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Incident Phase Performance Model</td>
<td>77</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Uncertainty Modeling</td>
<td>79</td>
</tr>
<tr>
<td>5</td>
<td>Train Reformation Problem: Implementation</td>
<td>85</td>
</tr>
<tr>
<td>5.1</td>
<td>Journey Generation</td>
<td>86</td>
</tr>
<tr>
<td>5.1.1</td>
<td>Timetable Expansion Stage</td>
<td>86</td>
</tr>
<tr>
<td>5.1.2</td>
<td>Reformation Stage</td>
<td>92</td>
</tr>
<tr>
<td>5.1.3</td>
<td>Station Arrival and Departure Times</td>
<td>98</td>
</tr>
<tr>
<td>5.2</td>
<td>Journey Selection</td>
<td>100</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Objective Function</td>
<td>100</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Constraints</td>
<td>108</td>
</tr>
<tr>
<td>5.3</td>
<td>Post-processing</td>
<td>110</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Fine-Tuning Module</td>
<td>110</td>
</tr>
<tr>
<td>5.3.2</td>
<td>TSM</td>
<td>112</td>
</tr>
<tr>
<td>6</td>
<td>Application</td>
<td>117</td>
</tr>
<tr>
<td>6.1</td>
<td>01/14/14 Incident</td>
<td>117</td>
</tr>
<tr>
<td>6.1.1</td>
<td>Controllers Response</td>
<td>118</td>
</tr>
<tr>
<td>6.1.2</td>
<td>Results and Comparisons</td>
<td>124</td>
</tr>
<tr>
<td>6.2</td>
<td>Hypothetical Incident</td>
<td>137</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Incident Description</td>
<td>138</td>
</tr>
</tbody>
</table>
6.2.2 Results .......................................................... 139

7 Conclusion ......................................................... 147

7.1 Research Summary ............................................. 147
7.2 Recommendations .............................................. 150
7.3 Limitations ....................................................... 151
7.4 Future Research ................................................ 152
List of Figures

1-1 Disruption Process ................................................. 19
1-2 Train Reformation Problem Structure ............................... 22

2-1 Disruption Process ................................................. 29
2-2 Control System and its application to disruption management .... 30
2-3 Short-turn Example ................................................. 34
2-4 Renumbering Example ................................................. 36

3-1 Piccadilly Line ....................................................... 40
3-2 Heathrow Branch Tracks .............................................. 41
3-3 Uxbridge Branch Tracks ............................................. 42
3-4 Tracks Shared with the District Line ............................... 42
3-5 East End Tracks ...................................................... 42
3-6 Passenger Flow during 30-min Peak of the Peak, Eastbound .... 44
3-7 Passenger Flow during 30-min Peak of the Peak, Westbound .... 45
3-8 Incident Cause Distribution ........................................... 47
3-9 Delay caused by Signal Failure (Data from 2013-2014) .......... 47
3-10 Incidents by Time of Day (Data from 2013-2014) ............... 48
3-11 Number of Incidents per Location (Data from 2013-2014) .... 50
3-12 Control Environment of the Piccadilly Line ..................... 51

4-1 Train Reformation Problem Inputs and Output ................. 59
4-2 Structure of the Train Reformation Problem .................... 59
4-3 Timetable Extension: Basic Operations ......................... 63
6-15 Waiting times at various locations ................. 144
6-16 TRP solutions recovery plans ..................... 146
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Top 20 Incident Locations in 2013 and 2014</td>
<td>49</td>
</tr>
<tr>
<td>5.1</td>
<td>Stop sequence of journey $C_{01}$</td>
<td>90</td>
</tr>
<tr>
<td>5.2</td>
<td>Group Formation</td>
<td>96</td>
</tr>
<tr>
<td>5.3</td>
<td>Group (A,15:45)</td>
<td>96</td>
</tr>
<tr>
<td>5.4</td>
<td>Group (B,15:45)</td>
<td>96</td>
</tr>
<tr>
<td>5.5</td>
<td>Station Arrival/Departure Calculation Example</td>
<td>100</td>
</tr>
<tr>
<td>5.6</td>
<td>Recovery Timetable before the Fine-tuning algorithm</td>
<td>112</td>
</tr>
<tr>
<td>5.7</td>
<td>Recovery Timetable after one iteration</td>
<td>113</td>
</tr>
<tr>
<td>6.1</td>
<td>Scenario parameters</td>
<td>125</td>
</tr>
<tr>
<td>6.2</td>
<td>Scenario results</td>
<td>127</td>
</tr>
<tr>
<td>6.3</td>
<td>Waiting Time at key stations during the Recovery Phase</td>
<td>135</td>
</tr>
<tr>
<td>6.4</td>
<td>Scenario parameters</td>
<td>140</td>
</tr>
<tr>
<td>6.5</td>
<td>Scenario results</td>
<td>141</td>
</tr>
<tr>
<td>6.6</td>
<td>Waiting Time at key stations during the Recovery Phase</td>
<td>143</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

This thesis presents an optimization model aimed at improving the disruption management process on high-frequency transit rail lines. Disruptions impact passengers dramatically and improving how they are handled is challenging. As part of the larger objective of building a decision support system for rail service controllers, the first part of the thesis focuses on the challenges that controllers face during service control. It then describes a method aimed at supporting real-time decision-making, especially by targeting the strategies required in the second phase of a disruption, referred to as the recovery phase, which is the period between the end of an incident and the time when the service returns to the normal schedule. The model is implemented and tested for the recovery phase of incidents on the Piccadilly Line.

1.1 Context and Motivation

Operations in public transit are planned in advance. The timetable, that contains information about the trips operated, their timing, and the vehicle and crew schedule, is developed typically on a 3 months to multi-year cycle. On a daily basis, the service experienced by passengers differs from the service planned, because of unexpected events called disruptions. Transport for London defines a disruption by: "disruption occurs where an incident causes the train service or the station service to be interrupted" (London, 2014a).
On high-frequency lines, the impact of disruptions on passengers and operations can be huge and this is even more true when the line operates at, or near, capacity. Because of the high-frequency, disruptions can propagate much faster than on low frequency services. It is harder to recover from a delay on a congested line, because there is less flexibility to move trains and change their routes. As a result, disruptions can have a dramatic impact on service reliability and become a major source of inconvenience for passengers. For instance, Transport for London estimates that during the 1-year period from April 1 2014 to March 31 2015, customers lost more than 18 million hours in the underground network due to incidents (London, 2014a).

Disruption management is the process of dealing with incidents in order to mitigate their effects, and consists of two distinct phases, as illustrated in Figure 1-1. The incident phase is the period from the start of the incident to the time when its cause has been resolved. During this phase, the first consideration is safety, then canceling trains to prepare for the recovery. The second phase of the disruption is the recovery, that starts at the end of the incident and lasts until normal service is restored. In this phase, the priority is to return to scheduled service as fast as possible.

On high-frequency transit lines, disruptions happen almost every day, and their importance varies widely: from a localized delay which will recover naturally to a total shutdown of the system. For instance, a single signal failure was serious enough to delay passengers for more than 84000 passenger-hours in the Piccadilly Line on February 2013 (London, 2015).

If an incident disrupts the service, following the scheduled timetable without any modification becomes difficult. Service controllers are in charge of adapting the service to these new constraints. Based on how much the service has deviated from the schedule, controllers decide to change the train itineraries in order to provide passen-
gers with good service and return to scheduled service quickly. The time horizon for control is much shorter than for scheduling: in the case of a disruption, decisions must be taken in real time. Controllers must constantly decide what service interventions (that consists of re-timing or rerouting trips) are needed in the next few minutes to safely mitigate the impact of the disruption.

Figure 1-1: Disruption Process

Despite the importance of service control, there is no systematic way to handle disruptions. Controllers usually work in small teams, their decisions are often based on experience and rules of thumb. They operate with other teams (e.g. signaling) as they collect information to make decisions. In managing disruptions, controllers must process a constant and complex flow of information. They take decisions to relieve the emergency, given the information available. It is difficult to take into account the impact of their decisions over the next few hours until the end of the disruption. Moreover, the emergency nature of incident management prevents controllers from testing new techniques. In this conservative environment, it is not possible to test two different strategies for similar incidents, due to potentially severe impacts on passengers. As a result, it is also difficult to identify (and document) alternative strategies for similar incidents that could be effectively compared and assessed.

Many system performance measures are based on disruptions and their impact. These measures are communicated to the public or are used as business targets (e.g. % of scheduled trips completed, hours lost by customers because of incidents). However,
these metrics are measured at an aggregate level based on the service provided, and
do not account for the strategies used in the disruption. The gap between operational
targets, passenger service, and controller objectives sometimes leads to decisions that
are not consistent. Having a tool that balances passenger and controller objectives is
an important goal of this research.

Although many models for timetable development, rolling stock scheduling, and
crew scheduling exist, few address the issue of rescheduling in case of incidents on
a subway line. Most often, existing rescheduling models have been applied in the
context of bus routes or low frequency lines, or do not integrate vehicle and crew
rescheduling. The characteristics of those environments differ from high-frequency
rail lines.

1.2 Objectives

This research aims at improving the disruption management process. To do so,
the research has targeted two main objectives:

To develop a framework for modeling service control in rail transit lines.
The lack of models for service control for high-frequency lines must be addressed.
This framework should be general enough to be applicable across systems. It
should also focus on what controllers do. Understanding their strategies is
crucial and few disruption tools are designed to be used by and to support,
controllers.

To develop a tool to support controllers for the recovery phase, either in
real time or offline. An offline tool could be used to assess alternative control
strategies. A real-time tool that would assist controllers is a longer term goal.
The main objective of this thesis therefore, is to develop and evaluate the main
principles on which to build such a tool. Targeting the recovery phase initially,
as opposed the full problem of incident and recovery management, provides
a good starting point. It largely avoids the problem of uncertainty. At the same time, a recovery tool is a good way to test the general approach, and, if successful, could be valuable to controllers.

1.3 Research Approach

Decision Support for Disruption Management is a long term research project that has been conducted for Transport for London (TfL). The research project is composed of three main phases. The first phase aims at advancing the understanding of disruptions and their management. The second phase focuses on designing a more robust crew schedule. This work is part of the third phase of the project, the development of real-time control strategies. More specifically, this study aims at providing controllers with real-time control strategy guidelines.

To do so, an optimization framework is developed in order to model the service and its control during recovery from service disruptions. The main element of the framework is the solution to what is called in this thesis, the Train Reformation Problem (TRP). The framework models service in a way that remains fairly close to what controllers experience every day. It puts an emphasis on the main strategies they apply, using similar information that controllers would have: the recovery is based on the schedule and is constrained by drivers and infrastructure. The value of information is crucial, as the complexity of the control decisions depends on the accuracy of the information available. The train reformation problem uses the same stream of information that controllers have, or could have, but over a longer period of time. When controllers can process the impact of their decision for the next hour based on the current situation, the TRP includes the impacts of interventions on the full recovery.

The structure of the problem is illustrated by Figure 1.2. It focuses on the two main recovery strategies (short-turns and reformations) and consists of two phases.
In the first phase of the TRP, the journey generation phase, journeys that represent a large number of possible short-turns and reformationss are generated. In the second phase, the journey assignment, a recovery plan accounting for the operational constraints (e.g. infrastructure and driver related constraints) and targets (e.g. providing a good passenger service and coming back to schedule) is obtained. The problem in this phase is formulated as an integer linear program and incorporates the above considerations, both in the objective function and the constraints. The solution to the TRP provides a recovery plan given the situation of the end of the incident phase. The impact of the decisions are evaluated for the full recovery period. The approach is implemented and tested on the Piccadilly Line in London.

![Figure 1-2: Train Reformation Problem Structure](image)

### 1.4 Organization

Chapter 2 introduces disruption management. It highlights the challenges specific to high-frequency transit lines, and describes the classic service delivery cycle. It then focuses on disruptions, their causes and impacts, and explains the different components and factors involved in real-time service control. Finally, it discusses the different control practices and strategies.

Chapter 3 briefly describes the Piccadilly line, its main characteristics, its geometry, and the operation of the line. It then conducts a literature review about scheduling and disruption rescheduling approaches.
Chapter 4 describes the general approach of the Train Reformation Problem. It explains how the components of service control are modeled and describes the two phases of the problem. It also suggests ways to deal with the uncertainty of an incident in terms of its duration.

Chapter 5 explores the model implementation in the context of the Piccadilly Line. It details additional assumptions required in order to apply the general framework accounting for the characteristics of the Piccadilly Line. It presents the constraints that are in place, and how the Train Reformation Problem has been adapted to the recovery phase.

Chapter 6 applies the method in two cases: an actual incident and a hypothetical one. The first case study deals with an incident that occurred in January 2014 and examines the application of the methodology to the recovery phase. The second case study is a hypothetical incident to examine the interconnections between decisions in the incident phase and the recovery phase. It reports and analyses the results, and compares, where applicable, these results with how controllers actually managed the incident.

Chapter 7 summarizes the approach of the Train Reformation Problem and highlights its strengths and weaknesses. It also makes recommendations about how this research can be used in London and in other systems. Finally, it suggests ways for researchers to build upon the model and to improve disruption management.
Chapter 2

Disruption Management

This chapter describes the general framework for disruption management. Section 2.1 explains the specificity of high-frequency rail lines. Section 2.2 describes the process that leads to service delivery. Section 2.3 shows how disruptions impact the service, and section 2.4 details the process of managing the service in case of disruptions.

2.1 High-Frequency Service

Service frequency is a very important attribute of a line, because it impacts both the operations side (service constraints) and the passenger side (passenger behavior). This research focuses on high-frequency lines, which come with specific challenges. In low-frequency services, the timetable is very important, as passengers coordinate their arrival with the trip departure time. As a result, the key metric for low-frequency services is on-time performance.

In high-frequency services, customer behavior is different, with passengers arriving at the stations more or less randomly. They only know the frequency and the timetable is often not even communicated. The important performance metric in this case is waiting time, which is a function of headway and service reliability.
Another attribute of high-frequency lines is that they serve more passengers. They operate closer to capacity to accommodate the demand: higher frequency means more trains, which implies more congestion on the line, which in turn limits the possibilities of delaying trains, and rerouting trains. Canceling a train in high-frequency services is only a minor inconvenience. Though it impacts the crowding and reduces the capacity, it does not prevent passengers from completing their trips, more or less as planned.

Therefore, the operating environment of rail high-frequency transit lines is different from lower frequency services. This difference has a direct impact on the disruption management.

2.2 Service Delivery Process

To understand how the service is managed in case of disruptions, it is important to understand how the service is planned. The first step of service planning is to estimate the demand to decide the span and the frequency of the service. The frequency of the line, expressed in trains per hour, depends on the passenger demand and historical data but is also constrained by the capacity of the infrastructure, and the resources (drivers and trains) available. Establishing a timetable that determines the route and the timing of trips which achieve the service targets is an important next step. The recovery time, i.e. the amount of time between two consecutive trips used to recover operational delays, the capacity of the depots, and the numbers of trains available are accounted for in the timetable development phase. The next step is rolling stock scheduling, which determines what trains will cover the trips of the timetable. During this phase, the number and locations of spare trains are also decided. The last step of the scheduling process is the crew scheduling. Shifts that cover the trips of the timetable are created, and operators are assigned to these shifts. Drivers may drive several trains during their shift. Each of these driving periods is called a driving task. The crew shifts must account for the labor rules, the number of drivers available,
and the crew facilities. These steps are conducted essentially sequentially, and the scheduling process takes several months and is typically repeated two to three times a year.

The operations are managed daily, based on the service plan. Even if the timetable is not communicated to the public, it remains crucial from an operational point of view, as it specifies what the drivers and trains must do. The actual service delivered, which is experienced by passengers, is different from the service plan, because the operations are by nature uncertain (because of passengers, drivers, potential failures, etc.). The role of service controllers is to adapt the service in case of unexpected events on the line.

2.3 Disruption

A high-frequency transit line is a complex system, and deviations from normal service cannot be avoided. The line is subject to natural variability and perturbations that prevent the exact deployment of the timetable on any given day. Small perturbations are called disturbances. Disturbances are expected and caused when normal events (such as driving time between two stations, or dwell time at stations) last longer than expected. Disturbances impact the general reliability of the line and may degrade the operational performance. However, the service can recover from disturbances, due to the slack time between consecutive trips.

On the other hand, disruptions are unexpected events that lead to larger perturbations. In case of disruptions, the timetable is not able to absorb the delays. It is necessary to adapt the operating plan to recover from the delay and return to normal service. In general, disruptions can lead to large delays, poor service and cancellations.

As will be discussed in chapter 3, causes of incidents vary. Incidents may be
caused by passengers (passenger sick on train, suicide) or supply related events (driver not available, signal failure, train failure, etc.). However, the impact of the incident on the operating side can manifest itself in only a few different ways. A number of incident types result in a blockage of a section of the tracks. As high-frequency lines generally have only a single track in each direction, the local blockage propagates and leads to a blockage of the line. A disruption can also lead to reduced capacity of the line: in some cases, disruptions prevent trains from moving at full speed, which reduces the capacity. For instance, a train failure or a signal passed at danger does not necessarily block the line, but it reduces the speed of the train. Disruptions can also impact a station. For instance, a terrorist alert at a station can lead to a station closure, without blocking the line. Disruptions can also impact a single train, without affecting the rest of the line. This may happen, for example, if the driver is late and fails to start their shift on time, or when a problem disables a train in depot.

A disruption can be divided into two phases, as illustrated in Figure [2-1]. The incident phase is the period from the start of the incident to the time when its cause has been resolved. The environment during the incident phase is very uncertain, especially at the start of the incident. Before the cause of the incident is identified, since controllers have no information about the cause, they are also unsure about its severity and likely duration. Later in the incident phase, the duration of the incident can sometimes be estimated, but the uncertainty still remains. The primary delay caused by the incident is propagated to other trains on the line (known as the knock-on effect). The second phase of the disruption is the recovery, that starts at the end of the incident and lasts until normal service is restored.

2.4 Service Control

In control theory, a controller is a system component that receives a command output and a feedback signal, and uses them to provide the input to the "plant" (see Figure [2-2]), which is the component that provides the final output. Figure [2-2]
Figure 2-1: Disruption Process

illustrates the similarity in the structure of transit service control and a theoretical control system. The first diagram (adapted from the book *Feedback control theory* Doyle et al., 1992), illustrates the general structure of a control-based system. In control theory, a disturbance signal is added to the command inputs, which makes the plant output different than expected from just the command. Adding a feedback loop and a controller addresses the error issue: a controller acts on both the command input and a measure of the output in order to provide a corrected input to the plant. Disruption management in public transit uses the same principles, where the timetable is the command input. Controllers’ role is to make sure that the service follows the timetable, even if the conditions change continuously due to unexpected events.

2.4.1 Control Components

The difference in the environment of the incident phase and the recovery phase leads to two different control behaviors. During the incident phase, the uncertainty and the congestion on the line lead controllers to reduce the service: they cancel trains, which give them more flexibility to recover in case the incident is longer than expected. The recovery phase is less uncertain. The controllers reintroduce trains and try to recover from the delay. During this process controllers make decisions guided by several objectives and constraints. The first priority of disruption management is to ensure the safety of passengers and staff. This sometimes requires canceling or holding trains, to avoid putting passengers and staff at risk. Even if this safety protocol often exacerbates the delay caused by the incident, safety must remain the
first priority at all times.

After safety, controllers aim at ensuring good passenger service in spite of the disruption. For this reason, controllers adapt the service in order to:

**Carry as many passengers as possible:** High-frequency subway lines often operate close to capacity. The delays and other constraints resulting from disruptions lead to a reduction in the number of passengers carried. Hence, providing enough capacity so that passengers can reach their destinations is a priority.

**Reduce the travel time:** Blockages and delays lead to longer travel times. Managing disruptions requires reducing both waiting times and in-vehicle travel time.

**Reduce crowding:** Disruptions lead to longer headways. Longer headways mean that more passengers are waiting on the platform for the next train, which
leads to crowding on the platforms, denied boardings, and in-vehicle crowding. Crowding is very uncomfortable and is a major source of inconvenience.

When the line operates near capacity, the delay caused by the incident leads to a gap in service downstream. Trains are concentrated upstream from the incident location, which challenges the capacity even more. The throughput is affected by the following parameters:

**Signaling capacity:** The signal system is one of the main bottlenecks of a train system. To prevent collisions, the signal control system prevents train from entering a track section if the next section is not free. Signal spacing is the main determinant of the minimum headway between two trains. Therefore, controllers must ensure that the number of trains in a section can be accommodated.

**Track capacity:** Most sections of high-frequency lines are single track. That means that the service must be suspended if a train blocks the track. Location of sections with more than one track, or with reversing capacity is important, since many control interventions happen there.

**Location of the trains:** Controllers need to consider where the trains are, and which ones are available for service. They may also have the possibility to use spare trains to replace a disabled train, or to support sections of the line operating with reduced traffic.

**Location of the train depots:** When not in service, trains must be in sidings or in depots. Moreover, the number of trains at each depot at the end of day, the stock balance, must respect the timetable.

In addition to capacity, controllers have to deal with driver-related constraints. If the timetable becomes infeasible during disruptions, controllers can change it. The crew timetable does not have such flexibility. It restricts control actions during recovery, because it is subject to more constraints. The crew timetable is composed of driver shifts, that are each composed of several tasks, which consist of driving a
train for part of the day. The shifts must respect driver work rules. For example, the work time over a day, a half day, and even for each task is limited. Unlike trains, drivers are not available before the beginning of their shift. After the scheduled end of their shift, even if their last task is not completed, on some systems, drivers cannot be required to work overtime. Similarly to the trains, train operators must start and finish their shift at their home depots, but unlike trains, actual shifts must happen within the scheduled period. Controllers must also be aware that spare drivers may be available to be used on short notice.

Accounting for all these constraints and objectives is very challenging. There is too much information to digest in real time. In some cases, such as the Piccadilly Line, the tasks of managing the crew and the schedule during a disruption are not handled by the same person. To be able to manage the service efficiently, it is important to set manageable, intermediate goals. For instance, maintaining even headways at key stations is a simple intermediate goal that helps to provide good passenger service. Balancing the sequence of the trip destinations, especially for lines with branches, is also a simple way to ensure that passengers will not wait too long before a train arrives that serves their destinations.

But the most important intermediate goal for controllers is to bring the service back to schedule as soon as possible after an incident. Controllers know that the schedule operates an even headway, under which passenger travel times are low. Controllers also know that the drivers work rules are satisfied by the schedule. Therefore, it is very attractive to try to return to schedule. The problem lies in the transition from the disrupted state to the normal schedule. To be able to recover quickly, it is easier to cancel more trains, that will be later reintroduced. Therefore, there is a trade-off between “cut deep” and recovering quickly, and maintaining a good service during a (longer) recovery.
2.4.2 Control Practices

Controllers use a number of strategies to modify the schedule to recover from delays.

Running Time changes: Slowing down a train can be used to make headways more even, or to change the train order at junctions.

Holding: Holding a train is the action of stopping a train and delaying its departure. Most holding takes place at stations. Holding is required in the case of a blockage during the incident phase. Holding can also be used to make headways more even and avoid trains bunching, to reorder trains at junctions, or to wait for a late driver. Holding increases in-vehicle time.

Early dispatching: Is the opposite of holding. It consists of the train departing early. It can be used to even headways, or to deal with expected longer dwell times during the trip.

Withdrawal: A train is withdrawn when it is prematurely diverted to a siding or to a depot, before it completes its trip. Passengers must vacate the train before its withdrawal. Withdrawal relieves the train congestion on the line, and enables more flexibility in the recovery, by having one more driver and train available later on.

Short-turn: A train is short-turned if it is reversed before reaching its scheduled destination. Short-turning facilitates recovering delay, and filling a gap in service in the other direction. Figure 2-3 illustrates an example of service recovery using a short-turn. A 20 minute incident happens in front of Train F, which causes a gap in service and congestion on the line. Short-turning Train H allows filling the service gap and returning to a normal schedule. Short-turning a train is inconvenient for passengers going to the end of the line, and for passengers originating at the end of the line, because the service is going to be less frequent.
**Extension:** Opposite of short-turn. The train goes past its scheduled destination. This allows serving more passengers, bringing back to schedule an early train, or filling a gap in the other direction.

**Diversion:** Consists of rerouting the train to a destination on another branch of the line. It is used to avoid the incident location, and to add service to a branch impacted by the disruption. If the running time on the other branch is different, it is also a way to recover the delay or the earliness of the train. It can be inconvenient to passengers who must alight the train, if the diversion is decided after the beginning of their trip.

**Expressing:** Consists of skipping scheduled stops. Can be used to shorten the duration of the trip, but is mainly used to avoid an incident or for security reasons.

**Cancellation:** Consists of canceling an entire trip, as opposed to withdrawal where the train is pulled out of service in the middle a trip. A withdrawal is often followed by the cancellation of the next scheduled trip. Like withdrawals, canceling trains is used to relieve congestion on the line, make resources available later, and facilitate the reintroduction of the trains on time.

**Adding an out-of-service trip:** When a train must be moved from one depot or
siding to another, this intervention consists of moving the train without passengers. It can be used to re-balance the stock, or make a train available where it is needed.

**Adding Service:** Extra trips that were not scheduled on the timetable. It can be used for the same reason as an out-of-service trip, but the train will stop at the stations between the two depots. It can also be a spare train used to provide passengers with more service, by filling a gap.

**Adding a shuttle service:** When the traffic is severely disrupted, short-turns can be used heavily to recover delay. In this case, the section at the end of the line will suffer from low-frequency service. Adding a shuttle service at the end of the line allows serving passengers by bringing them to the stations where more trains are stopping. As the route served by the shuttle is usually short, a single shuttle train can often serve a section with a decent frequency, while other trains recover delay with short-turns.

**Renumbering:** Renumbering (also called reforming) consists of changing the number of a train unit. When the number changes, the train must operate a new set of trips from the timetable. It is often used in conjunction with short-turns, extensions, and diversions. Figure 2-4 illustrates the utility of renumbering trains. The first line shows where trains A to D should be located, according to the service timetable. However, Train A is 5 minutes late, trains B and D 8 minutes, and train C 7 minutes. By changing the train numbers (A becomes D, B becomes A, C becomes B and D becomes C), the total delay is transferred to the new train D, which is now 29 minutes late, while the three other trains are now on time. A single short-turn of train D, combined with the renumbering, allows the recovery of 4 trains. Moreover, passengers of the renumbered trains are not impacted if the destinations are unchanged.

**Stock Renumbering:** Stock renumbering consists of renumbering one train in service with one spare train. If Train A is running late at the end of its first trip,
it is possible to dispatch a spare train on time for the second trip. The spare train assumes the train number A. The original train A will finish its trip late, then replace the spare train in the depot.

**Spare Driver Substitution:** If a driver is not available for a trip, it is possible to replace them with a spare driver.

**Spare Driver Early Relief:** If, because of the disruption, a driver cannot complete their task without violating a work rule or being late for their next task, it is possible to use a spare driver to drive the train to complete the task. The relieved driver will be able to finish their task on time, or early enough to begin their next task on time.

Spare drivers are a scare resource, since they are mainly designed to replace drivers absent because of sickness. Furthermore, it is more difficult for a controller to keep track of drivers than monitoring trains. For both reasons, the management of the crew is often decentralized in the depots. In this organization structure, controllers do not have perfect information about crew availability. Their interventions are taken
based on where the drivers would be, assuming that the crew timetable was followed, and with the hope that the service interventions will be compatible with the crew availability.

The combination of short-turn/renumbering is one of the most common strategies. It has the advantage of reducing the delay of several trains with a minimum number of train reroutings. However, renumbering does not reduce the driver delay. For this reason, renumbering is particularly attractive just before reliefs. If a train is renumbered well in advance their relief, its driver will have to drive trips that are likely not compatible with their own constraints: the relief time and location of the new train number might not be consistent. However, if the relief happens close to the end of the driver task, a new driver will soon take over the renumbered train. For the new driver, it does not matter that the train has been previously renumbered. Because they are both common and effective, this thesis focuses the short-turn and renumbering strategies.

2.4.3 Emergency Timetable

When a disruption is extremely severe, it may be difficult to recover enough delay to return to the timetable. In such cases, controllers may introduce an emergency timetable. An emergency timetable consists of operating a limited service until the end of the day. The emergency timetable is not strictly a timetable: the timing of the trips is not known exactly. The emergency timetable is based on the fact that if headways are increased, the available resources (trains and drivers) will be sufficient to operate without requiring a detailed working timetable. If a driver must finish their duty, due to limited service, it is very likely that another driver will be available to run the next trip. Emergency timetables are not included in this research.
Chapter 3

Introduction to the Piccadilly Line and Literature Review

The Piccadilly Line of the London Underground system is the subject of detailed analysis in this thesis. This chapter presents relevant information about the Piccadilly Line, focusing on the infrastructure, service characteristics, and demand. It also provides some background about the characteristics of the disruptions that take place as well as the structure and organization of the control system to deal with them. This chapter closes with a review of the relevant literature about disruption management and rescheduling.

3.1 Background

The Piccadilly Line is one of 11 lines of the London Underground (LU), the oldest subway system in the world. The ancestor of what would later become the London Underground started in 1863, when the Metropolitan Railway opened. Several other subway lines operated by different private companies were opened and all merged in 1933 into the ancestor of the London Underground, the London Passenger Transport Board. The central part of the Piccadilly Line was opened in 1906. The line was extended in 1932 and 1933 and then did not have further major changes until Heathrow was added to the line in 1977. The line was extended to terminals 4 and 5 in 1994.
3.1.1 Line Geometry

The Piccadilly Line is a very important part of the London transportation system (see Figure 3-1). It has three branches. The trunk starts at Acton Town and goes North-East to Cockfosters, passing through Central London. The stations from South Kensington to King’s Cross are major touristic destinations. The King’s Cross - St. Pancras station is a major hub with its transfers to 5 other underground lines, buses, regional and international trains.

Figure 3-1: Piccadilly Line

On the West side, the Piccadilly Line is the only underground line to serve Heathrow, offering a less expensive and more accessible alternative compared to other trains, such as the Heathrow Express and Heathrow Connect. Figure 3-2 shows the layout of the tracks on the Heathrow branch. Acton Town is a very important station,
as it is the last station before the line splits into two branches. It is also a crew depot, and a main reversing facility, with sidings to the east and west of the station. Northfields is the major train depot on the west part of the line, and is also a smaller crew depot. There are other reversing points, at South Ealing, Boston Manor, Hounslow Central and Hatton Cross, but they are used very rarely.

![Figure 3-2: Heathrow Branch Tracks](image)

The Uxbridge branch is the North-West part of the line. There are no reversing points and facilities between Acton Town and South Arrow. Figure 3-3 shows the track layout between Subdury Town and Uxbridge. South Arrow is a small depot where some trains are stored overnight. From Rayners Lane to Uxbridge, the line is also served by the Metropolitan Line. The Uxbridge Stabling and Ruislip sidings are used by the Metropolitan and Central Line, but not the Piccadilly Line.

Between Acton Town and Baron’s Court, the Piccadilly Line and the District Line (shown in green) share the same infrastructure, as shown in Figure 3-4. The Piccadilly trains use the two central tracks and do not stop between Hammersmith and Acton Town (except for Turnham Green in the early morning). Brompton Road and Down Street are disused stations that were closed before World War II. This part of the line has reversing facilities at Acton Town, between Hammersmith and Barons Court, at Hyde Park Corner, and Down Street.
With the exception of one reversing point at King’s Cross, the trunk of the line is simple. On the East end, shown in Figure 3-5 trains are able to reverse at Wood Green, Arnos Grove and Cockfosters. Arnos Grove is the crew depot in the east part of the line. Trains reverse there using the central tracks and the stabling sidings are used overnight and to store backup trains. A train depot with maintenance facilities is located between Oakwood and Cockfosters.
3.1.2 Demand and Service

In the financial year 2013/2014 (from April 1, 2013, to March 31, 2014), the London Underground network carried 1,265 million passenger-trips, which correspond to a 33% increase over a 10-year period (London, 2014a). The latest numbers that are available for the Piccadilly Line are from 2011/2012, with more than 210 million passenger-trips. Figures 3-6 and 3-7 (Ravichandran, 2013), show the passenger flows during the AM peak 30 minutes in 2012. Since then, the demand has grown, but the capacity has remained constant. These figures show that the line operates at, or close to, the capacity. Between Russell Square and King’s Cross, even a single train cancellation leads to the demand exceeding capacity.

41 trains are in service at 6:00am, and the number rises during the peak period, to 78 running a 150-second headway service. 68 trains are in service between the peaks, and this number increases again for the evening peak (London, 2014b). The service runs roughly from 5:00am to 1:00am. During peak hours, the service operates with a headway of 2.5 minutes. Because the Piccadilly serves Central London, the service on weekends is also important. On Saturdays, 76 trains run between 3:00pm and 6:00pm. The service is lower on Sunday, when only 68 trains are operating during the day.

The frequency is lower at the ends of the line. Eastbound trips end at Cockfosters or earlier in Arnos Grove. The destination of many trips on the Uxbridge branch finish at Rayners Lane, because the Metropolitan Line provides the main service on this section. On the Heathrow branch, some trips end at Northfields, especially during peak hours. The other reversing points are used only in the early morning, late night or during service disruptions.
Figure 3-6: Passenger Flow during 30-min Peak of the Peak, Eastbound
Figure 3-7: Passenger Flow during 30-min Peak of the Peak, Westbound
3.1.3 Service Disruptions

Disruptions are a major source of inconvenience on the London Underground network. Disruptions cause delays, overcrowding condition on trains and at stations and contribute to unreliable service. Transport for London uses a metric called Lost Customer Hours (LCH) that measures how much additional perceived time is wasted by passengers because of service disruptions (London, 2014a). Perceived journey time depends on the comfort of the trip: for instance, a minute spent standing in a crowded train is weighted more than a minute sitting in an empty train. In FY 2013/2014, LCH exceeded 20 million hours, which corresponds to an excess journey time of 5 minutes and 12 seconds per passenger-trip. On the Piccadilly Line, the estimated LCH value was 1.9 million hours, corresponding to an excess journey time of 4 minutes. In the past 10 years, the Piccadilly Line reduced LCH by almost 40%.

Figure 3-8 shows the most frequent causes of disruptions in the Piccadilly Line and their impact for CY 2013 and 2014. There are 5 main categories of incidents. The most important cause is signals: there are a large number of signals along the line. If just one fails, trains need to slow down and the traffic is disrupted. The second cause is customers & public, which corresponds to incidents due to passengers (doors blocked, passenger sick on train, suicide, etc.). The three other main causes are staff (driver late, signal passed at danger, etc.), track & civils (track switch failure, etc.) and fleet (train failure).

An important challenge in dealing with incidents is their unpredictability. Even if prevention programs, such as better driver training programs, or better maintenance processes, can prevent many incidents and decrease their numbers, when they happen, their location, cause, and duration remain unpredictable. Figure 3-9 illustrates the distribution of delays due to signal failure. The high variability highlights the unpredictability of incident duration. When a signal fails, trains cannot pass it faster than 5 miles per hour, which leads to a reduced line capacity. However, the problem
is often caused by a loose electrical contact. The connection may be reestablished when a train passes, and that is why most delays are short. However, in some cases, an incident may last more than an hour.

Figure 3-10 shows the distribution of the time when incidents occur. Incidents can happen at any time of the day. They are slightly more frequent during peak periods when there are more trains on the line. Their impact in LCH is also higher in these periods because they impact more passengers.
Finally, Table 3.1 and Figure 3-11 show that incidents can happen anywhere on the line. They are more frequent at more "active" stations, such as depots (Northfields, Acton Town, Cockfosters and Arnos Grove) or reversing stations (Hammersmith and Rayners Lane for instance). The higher number of incidents in the East end of the line can be attributed to the aging signaling system, which is different from the rest of the line (from Arnos Grove to Cockfosters).

3.1.4 Service Control Organization

The majority of the service control management takes place in the control center, which for the Piccadilly Line is located at Earl’s Court. The Service Control Manager (SCM) is in charge of the control organization. Service control is organized in three shifts of eight hours. During each shift, a Service Manager (SM) is in charge of the service control. The Service Manager manages two service controllers, one per direction. The service controllers make decisions about changing the route of trains.
Table 3.1: Top 20 Incident Locations in 2013 and 2014

<table>
<thead>
<tr>
<th>Location</th>
<th>No. Incidents</th>
<th>LCH</th>
<th>Average Delay in minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acton Town</td>
<td>657</td>
<td>729,789</td>
<td>7</td>
</tr>
<tr>
<td>Arnos Grove</td>
<td>537</td>
<td>299,394</td>
<td>6</td>
</tr>
<tr>
<td>Cockfosters</td>
<td>273</td>
<td>78,405</td>
<td>7</td>
</tr>
<tr>
<td>Hammersmith</td>
<td>247</td>
<td>313,267</td>
<td>6</td>
</tr>
<tr>
<td>Northfields</td>
<td>209</td>
<td>128,795</td>
<td>6</td>
</tr>
<tr>
<td>Rayners Lane</td>
<td>166</td>
<td>62,986</td>
<td>10</td>
</tr>
<tr>
<td>Northfield Depot</td>
<td>165</td>
<td>35,203</td>
<td>7</td>
</tr>
<tr>
<td>Oakwood</td>
<td>127</td>
<td>49,709</td>
<td>6</td>
</tr>
<tr>
<td>Heathrow T5</td>
<td>114</td>
<td>44,326</td>
<td>7</td>
</tr>
<tr>
<td>Wood Green</td>
<td>110</td>
<td>139,259</td>
<td>5</td>
</tr>
<tr>
<td>Kings Cross St. Pancras</td>
<td>107</td>
<td>160,014</td>
<td>7</td>
</tr>
<tr>
<td>South Arrow</td>
<td>86</td>
<td>32,618</td>
<td>11</td>
</tr>
<tr>
<td>Finsbury Park</td>
<td>82</td>
<td>49,665</td>
<td>5</td>
</tr>
<tr>
<td>Cockfosters Depot</td>
<td>79</td>
<td>1,205,457</td>
<td>4</td>
</tr>
<tr>
<td>Russel Square</td>
<td>75</td>
<td>75,919</td>
<td>5</td>
</tr>
<tr>
<td>Green Park</td>
<td>73</td>
<td>176,252</td>
<td>5</td>
</tr>
<tr>
<td>Barons Court</td>
<td>71</td>
<td>57,744</td>
<td>5</td>
</tr>
<tr>
<td>Ealing Common</td>
<td>71</td>
<td>53,207</td>
<td>14</td>
</tr>
<tr>
<td>Covent Garden</td>
<td>68</td>
<td>30,507</td>
<td>3</td>
</tr>
<tr>
<td>Earl’s Court</td>
<td>67</td>
<td>105,550</td>
<td>6</td>
</tr>
</tbody>
</table>

(where they go) and their timing (when they go). In addition, there are 7 signalers, in charge of controlling different sections of the line. They control the train signals at complex intersections (at junctions, sidings and reversing points). There is also a Line Information Specialist (LIS) who gives real-time information to all drivers on the line every 30 minutes. The LIS can also contact station supervisors.

The crew depots are managed by the Train Operations Managers (TOM). The Train Operations Service Managers (TOSM) are to crew depots what SM are to service control. However, in case of disruptions, the Duty Train Service Managers (DTSM) are in charge of the crew schedule. They make sure that every duty is covered, and use spare and available drivers in case a driver is late or absent. In case of a disruption, if the service is modified by controllers, the DTSM must consider the
changes impacts on the driver schedule and ensure that every modified journey is still covered. They communicate with controllers and can veto their control actions, if they are not feasible from a crew schedule point of view.

Stations are managed by Duty Station Managers (DSM) who are in contact with the Service Managers. Often they are the managers that are the closest to the incident location, and can provide relevant information to the controllers. Finally, there are two Duty Reliability Managers (DRM), based on the East and the West side of the Line. They go to incident locations, and investigate their causes. They are in contact with the Service Manager, help to resolve many problems and incidents, and confirm when the incident is over. Figure 3-12 shows a simplified diagram of the organization structure of the staff responsible for dealing with incidents in the Piccadilly Line.

Figure 3-11: Number of Incidents per Location (Data from 2013-2014)
3.2 Previous Research on Piccadilly Line

The MIT Transit Research Group has conducted several studies related to disruption management. Carrel (2009) described in detail the control environment of the Central Line in London. He used the signaling and automatic vehicle location data to reconstruct train operations during service disruptions. He then compared the timetable with the disrupted service to better understand control strategies and their impact on passenger service.

Freemark (2013) conducted a thorough analysis of disruptions in the Piccadilly Line. He developed a new metric, passenger accumulation, which uses the Automatic Fare Collection (AFC) data to measure the impact of disruptions on passengers. He also developed approaches to provide passengers with better information about disruptions using AFC data.
Ravichandran (2013) explored the robustness of the crew schedule. He studied two crew schedules of the Piccadilly Line and compared how their differences in allocation of slack time impacted service performance. He developed an event-based simulation tool to evaluate the robustness of crew schedules during service disruptions and evaluate the impact of changing labor rules on service performance.

### 3.3 Literature Review

Most railway services are based on a timetable, which determines the trips and the timing (departure and arrival times) of the trains that operate on the network. On passenger railroads, the service plan is created several months in advance, and the process is repeated a couple of times a year. The construction of the service plan consists of three main stages, most often done sequentially. The first step, the timetable scheduling problem, consists of deciding how many trips will operate, where they will run, and at what time. The second step is the rolling stock scheduling problem. During this step, train units are assigned to cover the trips in the timetable. The last step deals with the problem of assigning crews to trains in a way that satisfies work rules (crew scheduling). The timetable works fine during normal service, but during disruptions, it may be necessary to reschedule the railway system. Cacchiani et al. (2014) provide an exhaustive literature review of the research on rescheduling.

Train timetable rescheduling, which aims at modifying the route and the timing of the trips, has been the subject of several studies. Existing methods address the problem at various levels of detail. Corman et al. (2010) apply a rerouting and rescheduling algorithm in a case study of the Dutch rail network. The model adjusts the timing of the trips, and the tracks of the routes. However, the destinations are unchanged and there is no consideration for passengers. Pellegrini et al. (2014) develop a model which minimizes delays subject to detailed infrastructure constraints. Canca et al. (2014) expand the scope of rerouting operations, dealing more specifically with
the question of short-turns.

If the infrastructure constraints are included in the following works, they are no longer at the center of the model. For instance, Veelenturf et al. (2014) propose a model to minimize the delay and the number of cancellations. The model allows rerouting trains, but only in one specific case: the stops can be modified but only if rerouting can avoid the disrupted area. Many models also integrate the rolling stock rescheduling problem. Nielsen et al. (2012) aim at adapting the rolling stock assignment in response to a disruption, after the timetable has been modified. Even if the model does not tackle the timetable scheduling, it is interesting because it tries to balance the train stock at different parts of the line at the end of the day.

Passenger demand is considered in some approaches. For example, Fekete et al. (2011) develop a model which allows trips to be delayed or canceled, and trains to operate as shuttles or to be short-turned. This model is developed for high-frequency services, and the objective function maximizes the number of trips over the recovery period. The number of trips is used as a proxy for passenger service. Neither the delay of trains, nor the regularity of headways are included in the objective function. Adenso-Díaz et al. (1999) tackle the passenger level of service directly in the objective function, which maximizes the number of passengers carried. The model considers retiming, rerouting and train cancellations. The model is applied to an intercity railway network. Intercity rail operations differ from urban metro systems, since the goal is to carry passengers to their destination, while delays are secondary. Shen and Wilson (2001) and Puong and Wilson (2008) both develop models that include holding and short-turning strategies on high-frequency services. The models include denied boardings and longer dwell times due to overcrowding, while the objective functions account for waiting and in-vehicle times. O’Dell and Wilson (1999) develop a similar model which also includes the expressing strategy. In Song (1998), the various control strategies are applied to the problem of dispatching trains at terminal in
case of disruptions. Louwerse and Huisman (2014) develop a timetable rescheduling model accounting for passenger service, and the train stock balance (without including rolling stock rescheduling explicitly in the formulation). A number of papers (for example in Cadarso and Marín (2012), Cadarso et al. (2013), and Cadarso and Marín (2014)) propose models that simultaneously consider rolling stock and timetable rescheduling and apply them in the case of disruptions in the Madrid regional rail network. The objective function includes the number of passengers carried, a measure of comfort, and operating costs.

A limitation of the above models is that they do not address the issue of crew rescheduling. Cohn and Barnhart (2003) give a good example of the crew scheduling model, but in the airline industry. Crew schedules are one of the most limiting factors in service recovery. During disruptions, the crew schedule often becomes infeasible. Adjusting the crew duties to accommodate a disrupted service is an important but difficult task, due to the level of flexibility with respect to crew constraints. Using modified service and rolling stock timetables as input, Rezanova and Ryan (2010), and Huisman (2007) address the crew rescheduling problem, applied to disruption management, and planned disruptions respectively. The timetable and rolling stock rescheduling are handled separately. Veelenturf et al. (2012) introduce the possibility of modifying the timetable in the crew rescheduling phase by allowing the model to delay train trips.

Walker et al. (2005) attempt to solve both crew rescheduling and timetable rescheduling simultaneously. However, in their model, the timetable rescheduling is limited to changing the timing and order of the train departures and passenger service is not accounted for.

In general, a common way to model the timetable rescheduling problem is to adopt an event-based rescheduling approach, and was used in many of the above models (for
instance Veelenturf et al. (2014), Nielsen et al. (2012), and Louwerse and Huisman (2014). The original timetable is divided into events. These events are most commonly the stops at station and their departure times. The event-based approach consists of modifying these events, by changing their departures times, which train will cover the event, or canceling them. In these cases, the models emphasize the delay, which must be minimized.

From this literature review, several gaps can be noted. Models that aim at maintaining adherence to the timetable usually ignore major reroutings (such as short-turns). Crew and service timetable rescheduling are rarely integrated, with the exception of Walker et al. (2005) (but in this case, short-turns are not included). Issues specific to high-frequency passenger services, such as the importance of the delay and the limited capacity of the trains are rarely mentioned. Furthermore, most of the above works addresses the recovery problem at a theoretical level. Few papers take into account the way controllers operate and their daily practices. Decision support systems must be acceptable to controllers. To be acceptable, they need to be somewhat similar to the controllers approach, which means they need to adopt a different approach, closer to the controller practices. The large number of models that ignore the impacts of crew constraints on the timetable and on the rolling stock rescheduling is indicative of the preferred approach in the literature. In the control room, crew constraints are not just constraints that must be respected during disruption management, but they actually drive the controllers decisions and greatly influence, if not govern, the implemented timetable modifications.
Chapter 4

Decision Support Model: the Train Reformation Problem

This chapter describes the general approach to the decision support system. The decision support system for the recovery phase of service disruptions is built around the Train Reformation Problem (TRP). The train reformation problem aims at capturing strategies that controllers use every day in recovery from disruptions in a consistent and systematic way while mitigating the impact of the incidents. The train reformation problem assumes a deterministic environment. Under this assumption, given perfect information about the line, the location of the trains and the demand, the solution of the train reformation problem is a service plan that respects the crew constraints and provides good service to passengers. The approach consists of two parts: the first part is the generation of a large number of potential journeys (which is the service operated by a train from the end of the disruption to the end of the recovery) and is discussed in Section 4.2. These journeys are generated by applying a combination of reroutings and renumberings that modify the original journeys in the timetable. The second stage is an optimization problem that selects which of those potential journeys will be part of the service plan. The optimization accounts for the limited availability of resources, including the number of trains, the track capacity, and the crew schedule. The solution also takes into account the quality of service provided to passengers and the preferences of the control team. This stage is described
Section 4.4 proposes a way to expand the train reformation problem to the incident phase of service disruptions. The train reformation problem is not directly applicable to the incident phase. In the TRP approach, each journey is generated independently, without accounting for the interactions between the trains, which may result in a schedule where running times are inaccurate. This problem can be especially serious in the incident phase. Section 4.4.1 discusses how running time performance models and simulation tools can be used to correct the errors that may result. The second major difference between the incident phase and the recovery is in uncertainty. The uncertainty associated with the duration of the incident has a significant impact on the decisions that controllers must make during the incident phase. Section 4.4.2 discusses how the train reformation problem can be modified for use in the incident phase considering uncertainty.

4.1 The Train Reformation Problem

The train reformation problem is the central part of the proposed decision support approach. Its goal, illustrated in Figure 4-1, is to provide an optimal service plan for the recovery phase, given information about the location of the trains at the end of the incident and the timetable.

The train reformation problem aims at representing strategies that controllers use every day to recover from disruptions. Only if the model is closely related to what they do on a daily basis, is it likely to be useful to controllers. The approach to solve the problem is the following. Generally, during recovery, controllers use some combination of short-turning, reintroducing, and train renumbering. At the same time, they consider the constraints and the interactions with other trains. The result is the journey that each the train should follow during recovery.
The approach to the train reformation problem divides this process into two parts, as shown in Figure 4-2. The first part generates potential journeys for each train. These journeys correspond to a combination of rerouting and renumbering actions applied to the journeys from the timetable. Note that short-turnings and cancellations are just special cases of rerouting. To renumber trains, the notion of train groups, which is a set of trains that can be renumbered together, is introduced. At the end of this stage, a large number of possible journeys have been generated. These journeys differ with respect to their path, stop times, as well as their relief, determined according to the different train blocks that have been used to generate the trips.

Figure 4-2: Structure of the Train Reformation Problem
The second stage is an optimization problem that selects which of the potential journeys will be included in the solution. The optimization problem accounts for the limited availability of the resources, such as the number of trains in service or canceled, the track capacity, and the crew schedule constraints. The inclusion of a candidate journey in the solution is also based on the quality of service provided to passengers and the strategy used.

The following terminology is used in this document. These terms have been chosen to fit the formulation, and to make each concept in the model understandable. These do not necessarily mean the same as when used by transit agencies.

- A **station** is a location where trains can stop. It may or may not be used by passengers and may or may not be used to store trains: it refers to stations, sidings or depots.

- A **train stop** is the event of a train stopping at a station. A train stop is characterized by the station where the train stops, the direction, the arrival time and the departure time.

- An **itinerary** is a sequence of stations where the train stops. The stations are not necessarily passenger stops. Note that it is different from a sequence of train stops. A sequence of train stops is not only an itinerary, it also contains the arrival and departure times at each station.

- A **journey** is the sequence of stops that a train serves during the disruption. A journey is not completely defined until its list of **operator swaps** has been determined.

- An **operator swap** is the event of changing the driver of the train at a station at some time during the day. To make reading easier, the word *swap* may be used to designate operator swaps.

- A **relief** and a **pick-up** are the two events included in an operator swap. The relief is the event of a driver stepping off the train. A relief is characterized
by its location, its actual time, its scheduled time in the crew timetable, and the driver identity. As the operator swap always happens at a train stop, a swap is almost always linked to a single train stop. The location of the swap is the station of the corresponding stop, and the actual time the arrival time at that stop. An exception occurs when the relief has been canceled because the whole train block has been canceled. In this case, the relief is not linked to any train stop. Another relief attribute is its driver number, which identifies which driver is relieved, and its train number, which is the number of the train from which the operator is scheduled to be relieved. Similarly, a pick-up occurs when a driver boards a train and starts to drive it: it has the same attributes as a relief.

- A train block or a driver task is a sequence of stops between two successive operator swaps.

- A break is the period of time between the end of a driver task and the beginning of their following task.

- A journey is associated with one or more train numbers. Originally, a train number is what identifies a journey from the timetable. But after a reformation, a journey will be composed of several train blocks that originally belonged to different journeys in the timetable. As a result, a journey will have two numbers.

- A candidate is a journey generated at the end of the journey generation stage. The service plan is composed of journeys that are chosen among the candidates.

- A trip is a sequence of stops, over which the train does not change direction. A train journey can be divided into several trips. A trip is not only a one-way sequence of stops: a journey must also reverse at the first and last stops of each of its trips.

- A flag is a Boolean variable associated with a candidate, used to indicate journeys that share an attribute. For instance the flag “red locomotive” would be equal to one for candidates whose locomotives are red.
4.2 Journey Generation

The different operations used to generate potential journeys in the train reforma-
tion problem can be classified into three groups. The first group includes all actions
that modify the itinerary, i.e. the sequence of stations, but maintain the train num-
ber. Short-turning, expressing and re-routing are the basic operations in this group.
These operations are applied directly to the timetable: the sequence of stops of one
train from the timetable will be modified to produce several journeys which are vari-
ations on the original journey. Because a single journey of the timetable generates
many journeys, this stage is called timetable expansion. Operations in the second
group alter the trip number of a train and are referred to as train reformations. Fi-
ally, operations in the last group change the running times and hold trains. In the
approach, each category is modeled and applied consecutively. The whole process
can generate tens of thousand candidates from a hundred originally scheduled in the
timetable.

4.2.1 Timetable Expansion

The first category of operations expands the number of journeys by modifying
the itinerary of the original journeys. Short-turns and extensions are the two ba-
sic operations which can modify an itinerary. In this context, short-turning deletes
the sequence of stops scheduled between two consecutive stops at the same station.
The extension operation adds a new sequence of stations to an itinerary. The first
and the last station of the added sequence must be the same. Applying a succes-
sion of short-turns and extensions is sufficient to generate any type of rerouting. For
instance, rerouting can be done by applying one short-turn and one extension suc-
cessively. Figure 4-3 illustrates how short-turns and extensions change an itinerary.
In the short-turned journey (4-3a), stop C is skipped. The extended journey (4-3b)
adds stop E in the middle of the itinerary. The diversion (4-3c) happens when an
extension to D is added in place of the canceled stations of the short-turned itinerary.
Figure 4-3: Timetable Extension: Basic Operations
To make generation more flexible, more relevant, and computationally more efficient, it is possible to add constraints which guide the generation toward preferred strategies. Examples of such constraints include:

- Limit operations in time. No itinerary will be modified after the time chosen to end the recovery.

- Allow short-turns only if no stop where an operator swap occurs is deleted.

- Limit short-turns or extensions to a restricted set of reversing stations.

- Allow extensions only if they are combined with short-turns, to form a diversion, such as in (4-3c).

4.2.2 Group Reformation

The first group of operations was applied to each journey of the timetable independently: it left their numbers unchanged. The second group is applied to two journeys with different train numbers and combines the journeys. These operations are called renumbering or reformation. Figure 4-4 shows an example of reforming two journeys: Journey 400 and Journey 500. The renumbering decision is taken at time $t_{dec}$ (usually the end of the incident). The top left of the figure shows that before the reformation, journeys 400 and 500 are composed of several train blocks. To reform these journeys, they are divided into two parts. The first part of the journey contains the train blocks starting before the "decision time", which is the time when it is decided to renumber, (the "First Leg") and the second part contains those starting after that time ("Second Leg"). Reforming these journeys together concatenates the first legs with the second legs, to form 4 journeys. The right hand side of the figure shows the result of the reformation. Two journeys are the same as the original journeys 400 and 500. On the other hand, journeys 400-500 and 500-400 are composed of train blocks that were originally in 400 and 500.
When controllers reform trains, they need to ensure that there will be an operator available before and after the reformation to drive the train. In the case of Journey 400-500, the driver of block 500-3 must be present at the relief location of the operator of 400-2. Otherwise, there will be no driver to take over the train 400-500 after the driver of 400-2 has left it. This constraint leads naturally to the notion of a train group. A train group is defined as a set of journeys that can be renumbered together, i.e. they share a common swap location. Once a group has been formed, every journey of the group is split into two sub-groups. The train blocks occurring before the relief are assigned to the first subgroup and those after the relief are assigned to the second group. New journeys are generated by concatenating a first leg with a second leg. The swap station is at the junction of the two legs: the arrival at the swap station is in the first leg, but the train departure belongs to the second leg. The example of Figure 4-4 shows a group with two journeys before reformation. Groups are generally larger and contain tens of journeys.

From this general principle, there are three methods to perform the renumbering process, each one being broader and more complex than the previous one:

1. The simplest way to renumber is to fix a single decision time at $t_{\text{dec}}$. Journeys will belong to the group determined by the location of the first swap happening after $t_{\text{dec}}$.

2. It is possible to have several decision times ($t_{\text{dec}1}, t_{\text{dec}2}, \ldots$). That means that journeys do not necessarily have to be reformed simultaneously. Each journey will belong to one group for each decision time.

3. The last method allows modeling multiple renumbering operations. To do so, decision times can become vectors instead of points. A decision time vector with 3 decision times corresponds to 3 successive renumberings. Under this scheme, a group is not characterized by a single swap location, but by its sequence of swap locations for each decision time. Finally, a group is divided into several
subgroups: a journey will have \( n+1 \) legs, where \( n \) is the number of renumberings. Each leg corresponds to the train blocks occurring between two reformations.

### 4.2.3 Station Arrival and Departure Times

At this stage of the journey generation, only the itineraries of the journeys have been changed. However, to transform an itinerary into a valid candidate, it is crucial to set the arrival and departure times at stations.
Initial Conditions

The positions of the trains at the end of the incident phase are required. By looking at the location of trains from the automatic vehicle location system and comparing it with the timetable, it is possible to infer the delay of all trains. This delay becomes an input to the decision support system and the actual position of trains becomes a starting point for the recovery phase.

Using the delay at the end of the incident is also an easy way to simulate a typical incident. The delay at the end of the incident is the result of the strategy used for the incident phase. Figure 4-5 shows two kinds of delays, resulting from two different strategies in the incident phase. The top curve represents the delay if every train is held to maintain the scheduled headways. With this strategy, the delay curve is a step function. Notice a gap that is created when the trains upstream from the incident are not held. The bottom scenario shows what happens when trains are only separated by the minimum headway. The traffic after the incident is congested, and trains far from the incident are on time, which results in a triangle shape.

Modeling the impact of incidents at the beginning of the recovery with a delay function has several advantages. First, it is intuitive and easy to understand. It is easy to obtain the delay curve from a timetable or a waterfall diagram. It is a good way to provide a realistic starting point: it is able to either replicate easily what actually happened, or different delay functions can be used to generate starting points consistent with different control strategies typically deployed during the incident phase.

Holding variations

At this point, the question of holding trains becomes relevant. If a journey were short-turned or withdrawn, the main purpose was to make it on-time. However, after
these operations, the journey will be early. A withdrawn train will not be reintroduced immediately after its withdrawal: doing that would make the journey very early (it would run ahead of the schedule). Therefore, at one point during the itinerary, the train will be held or stalled long enough, so its journey runs on time when its service is resumed. That is why it is useful to generate these *holding variations*. They emphasize the reintroduction strategy that controllers use after they have canceled (or withdrawn) a train.

However, there are many places where the trains can be held. For instance, consider a journey running early after a short-turn. One possibility is to hold the journey as soon as it becomes early at the short-turn. Another possibility would
be to hold the train even before it has been short-turned, when the journey is still late. Of course, this is generally not a good solution because it could make the first operator swap even later. Finally, it is possible to divide the holding into several shorter holds at stations. For instance, if a journey is ten minutes early, it could be held ten times at one minute each or for ten minutes once. Section 5.1.3 in the next chapter details the holding methods that have been implemented for the Piccadilly Line application.

4.3 Journey Selection

At the end of the generation phase, a large number of candidates have been generated. For each journey, the stop locations and times are known, as well as the driver identity, and the original schedule of both the train and the driver. From these data, a constrained optimization problem is formulated, that aims at determining which candidate journeys will be selected in the recovery plan. The objective function captures passenger service costs (i.e. waiting times), and includes a schedule adherence component and a recovery schedule complexity component.

The journey selection problem is subject to three kinds of constraints, represented in Figure 4-6. Each constraint has a counterpart in the objective function that uses the same feature of the candidates. Both passenger service cost and infrastructure constraints are built upon the stop sequence of the train. The crew dimension is responsible for the crew constraints and the schedule adherence cost, which represent how close the service plan is to the original schedule. Finally, the different flags that are associated with candidates are used for the complexity cost and the constraints said basic (uniqueness of assignment, and strategy oriented constraint).
4.3.1 Objective Function

The objective function is given by:

\[ \min F(x) = W(x) + H(x) + \Theta(x) \quad (4.1) \]

Where:
- \( F \): objective function.
- \( x \): vector of binary decision variables. \( x_i = 1 \) if the journey \( i \) is selected in the service plan, 0 otherwise.
- \( W \): passenger service cost.
- \( H \): schedule adherence cost.
- \( \Theta \): recovery plan complexity cost.

Passenger Service Cost

The passenger service cost assesses the impact of the recovery plan on customer service. The service quality is represented by a function that approximates the passenger generalized cost. To do so, each candidate is assigned to several demand sources,
which are static entities with which a demand factor is associated. For instance, a journey can be associated with each origin-destination pair (OD) it serves. Considering all the trains that serve the OD pair in a given time period allows calculating the service frequency for this origin-destination pair, and in turn the waiting time.

To estimate the waiting times, the time horizon is divided into several time bins. Each combination OD pair/time bin (ODT) \( od, t \) (\( o \) stands for the origin, \( d \) for the destination and \( t \) is the center of the time period) is associated a variable \( n_{od,t} \) equal to the number of trains serving the ODT.

\[
W(x) = \sum_{o,d,t} d_{od,t} f_w(n_{od,t}) 
\]

Where:
- \( W(x) \): passenger service cost.
- \( x \): vector of binary decision variables. \( x_i = 1 \) if the journey \( i \) is selected in the service plan, 0 otherwise.
- \( d_{od,t} \): represents the demand of the ODT \( od, t \).
- \( f_w(n_{od,t}) \): function that approximates the waiting time of the passengers.

\( W \) is a function of \( x \). At the end of the journey generation, a binary matrix \( W = (w_{(od,ti)}) \) is generated, with \( w_{(od,ti)} \) equal to 1 if journey \( i \) serves the demand source \( od, t \), 0 otherwise. The matrix \( W \) is of dimension \( n_{od} \times n_t \times N \), where \( n_{od} \) is the number of origin-destination pairs, and \( n_t \) the number of time periods, and \( N \) the number of candidate journeys.

We define \( n \) as vector of size \( n_{od} \times n_t \), given by:

\[
n = Wx 
\]

\( n_{od,t} \), the number of trips serving ODT \( od, t \) is a member of vector \( n \).
In Section 5.2.1 where the implementation of the problem is discussed in detail, it is proposed to modify $W$ to mitigate the boundary effects of the time bins. According to this modification, a train stop can be assigned to more than one time bin with a degree of membership $w_{(od,t),i}$ between 0 and 1. For instance, journey $i$ serves the OD-pair A-B at 12:45. Assuming 30-minute intervals, $w_{(od,t),i}$ may assume values $w_{(A\rightarrow B,12:30),i} = w_{(A\rightarrow B,13:00),i} = 1/2$ rather than $w_{(A\rightarrow B,12:30),i} = 0, w_{(A\rightarrow B,13:00),i} = 1$.

The function $f_w(\cdot)$ represents the service quality provided to the users. Having $f_w$ proportional to the average waiting time is a simple way to model the passenger service. Under the assumption of even headways, it is approximated as half of the average headway (which is determined by $1/n_{od,t}$). Section 5.2.1 provides more detail about the calculation and the implementation of these terms in the Piccadilly Line application.

**Schedule Adherence Cost**

One of the main goals of disruption management is to recover the normal service. The adherence cost assesses how effective the recovery has been, which means how fast the service approaches the timetable. The best way to do that is to consider the timings of operator swaps. The service is on time if every operator swap is on time. Once an operator swap is on time, following the scheduled timetable until the end of the day is sufficient to keep the journey on time. Therefore, if every operator swap is on time, the service is following the timetable. As a result, the timing of operator swaps is a very good measure of how close to the schedule the recovery plan is.

Schedule adherence can be measured during the whole disruption, or at a predetermined time point. In the first case, the adherence depends on all operator swaps following one incident. In the second case, the adherence depends on all operator swaps occurring after a recovery time target. This second option is attractive if the journey generation is limited in time. If the model formulation requires that all con-
trol actions take place before time $t_{end}$ (input to the journey generation), it makes sense to calculate the cost of the delay for the first swaps happening after $t_{end}$. This is discussed in detail in Section 5.2.1

In both cases, the cost function is calculated as follows:

$$H_i = \sum_{r \in R_i} f_d(t(r) - t_0(r)) + \sum_{p \in P_i} f_d(t_0(p) - t(p)) \quad \forall i \in J \tag{4.5}$$

$$H(x) = H^T x \tag{4.6}$$

Where:

- $J$: set of candidates.
- $R_i$: set of reliefs of journey $i$ to consider.
- $P_i$: set of pickups of journey $i$ to consider.
- $t(r)$: time of relief $r$ in journey $i$.
- $t_0(r)$: scheduled time of relief $r$.
- $t(p)$: time of pickup $p$ in journey $i$.
- $t_0(p)$: scheduled time of pickup $p$.
- $f_d(\cdot)$: function equal to 0 when $x < 0$. The function increases non-linearly (e.g. $\exp$) so large delays are penalized more than small ones.

**Recovery Plan Complexity**

The idea of the recovery plan complexity component is to penalize service plans that are difficult to implement from a controller’s point of view. For instance, renumbering every journey 3 times is almost impossible, because there is too much information to communicate. Therefore, some journeys, which are considered simple to implement, could receive a bonus, whereas complex renumbered journeys would be more expensive. Another way to use the complexity component is to penalize plans that are very different from a desired strategy. For example, if the strategy is to skip
station A, journeys stopping at A would be penalized.

4.3.2 Constraints

The "optimal" recovery plan has to satisfy a numbers of constraints related to the capacity of the line, the crew and the train availability.

Capacity

The capacity constraints are of the form:

\[ T x \leq C \]  \hspace{1cm} (4.7)

Where:

- \( C \): capacity vector of the infrastructure, in seconds. \( C \) is of size \( k = m.n_t \), where \( m \) is the number of infrastructure elements with limited capacity and \( n_t \) is the number of time periods.
- \( T \): occupancy matrix of the journeys, of dimension \( N \times k \) where \( N \) is the number of journeys in \( J \).

The constraint is very similar to the passenger service component. A journey is associated with several combinations of infrastructure elements/time-period (IT) that are each limited in capacity. Examples of infrastructure elements are a track between two stations, a station, or a platform. The time each journey occupies the IT \( s,t \) (\( s \) stands for the infrastructure element, \( t \) is the center of the time period) is limited. For instance if the time period is 15 minutes, no more than 10 trains could stop at the station in each time period. It is also possible to include a safety buffer in the occupancy direction. If the minimum headway at a station is 150 seconds, the occupancy coefficient will be at least 150 seconds: no more that 6 trains can stop during the time period, which ensures that the minimum headway is respected. For the same reason as the passenger service component, it may be useful to allocate a stop to several ITs to mitigate the boundary effect of the time bin.
Crew Availability

The duty schedule imposes several constraints. The work time must not exceed a certain duration in a day, and a meal break must happen before the driving time reaches a threshold. And obviously, drivers must finish their previous task before starting their next one. This leads to two kinds of constraints: the first ensures that drivers are physically available, which means that they are present on time to complete each of their tasks in the recovery plan. The second type of constraint ensures that work rules are not violated (e.g. the total work time in a day or in a week does not exceed the limit).

The constraint that ensures the availability of the drivers in the service plan can be written as:

\[ Rx \geq K_1 \]  

(4.8)

Where:

- \( R \): relief matrix of dimension \( N \times P \), where \( P \) is the number of driver tasks, and \( N \) the number of candidates in \( x \). Its members \( r_{ji} \) are different of 0 if the journey \( i \) uses the swap \( j \) from the timetable. The value of \( r_{ji} \) is related to the swap time of relief \( j \) in journey \( i \).

- \( K_1 \): vector of the time that drivers need at their relief (i.e. the minimum time to walk from one task to another, the meal break minimum time, or the shift start time when the driver task is their first task of the day).

The second type of constraint, that ensures that the work rules are respected, is modeled by the following equation.

\[ Gx \leq K_2 \]  

(4.9)

Where:
• **G**: Driver constraint matrix. Each column corresponds to a candidate, and each row to a driver’s work rule. An element \( g_{ji} \) is the amount of time that the candidate \( i \) contributes to work rule \( j \) of a driver (e.g. driving time, time available in depot...).

• **K**: vector that constrains these work rules (e.g. driving time limit, work time limit...).

### Uniqueness Constraints

The uniqueness constraints ensure that a driver task is not assigned twice. Let \( A = (a_{p,i}) \) be a matrix of dimension \( P \times N \), where \( P \) is the number of driver tasks from the timetable, and \( n \) the number of candidates. \( a_{pi} \) is equal to 1 if candidate journey \( i \) uses the driver task \( p \). The uniqueness constraint is expressed as:

\[
Ax \leq 1 \quad (4.10)
\]

Where \( 1 \) is a vector \( \{1, 1, \ldots 1, 1\} \) of size \( P \).

### Strategy Oriented Constraints

It is possible to associate several attribute flags with each journey, such as "this journey is a holding variation" or "this journey has not been renumbered". Then it is possible to introduce constraints with respect to "preferred strategies". Let \( A^+ \) and \( A^- \) be two binary matrices with \( p_1 \) and \( p_2 \) rows and \( N \) columns (\( N \) is the number of candidates). Each flagged strategy corresponds to a row of \( A^+ \) or \( A^- \). \( A^+ \) corresponds to constraints of the type "no more than ... candidates in the solution that follow the strategy ..." and \( A^- \) corresponds to "exactly ... candidates in the solution that follow the strategy ...". Let \( b^+ \) and \( b^- \) be two vectors of integers of size \( p_1 \) and \( p_2 \).
\begin{align*}
\mathbf{A}^+ \mathbf{x} & \leq \mathbf{b}^+ \\
\mathbf{A}^= \mathbf{x} & = \mathbf{b}^= 
\end{align*}

(4.11)  \quad (4.12)

For instance, the constraint "no more than 8 reformed journeys in the solution" would add a row to the first equation, and the new member of $\mathbf{b}^+$ would be equal to 8, while a constraint "exactly 8 reformed journeys in the solution" would correspond to a row in the second equation.

### 4.4 Extension to the Incident Phase

The train reformation problem, described in the previous sections, deals exclusively with the recovery phase. The characteristics of the incident phase of a service disruption are different from the recovery phase, because of the uncertainty and the potential blockages of the line. This section discusses two approaches to adapt the train reformation problem so it can be applied to the incident phase.

#### 4.4.1 Incident Phase Performance Model

In the proposed approach to the train reformation problem, candidates are generated independently and running times are computed before the journey selection. The implication is that the running times associated with journeys may not be accurate because they do not account for interactions between trains. Interactions themselves depend on the kind of recovery plan used, while the recovery plan depends on the impact that the interactions have on the train reformation problem. This problem was addressed in the recovery phase by assuming that journeys use the running times from the timetable. However, such an assumption is not valid for the incident phase. Incidents impact running times, either directly (signal failure) or indirectly (congestion due to the signal failure). The indirect impact of incidents on running times is difficult to assess, because the propagation of the incident due to congestion de-
pends on the operated service itself. This subsection proposes two ways to improve the calculation of running times, which are necessary in order to formulate a control strategy during the incident phase.

**The speed based running time performance model**

The speed-based running time performance model assumes that different types of disruptions are characterized by a known speed profile, which gives the speed of the train as a function of time and location. The approach is similar to the shockwave theory in traffic flow theory. The incident is represented as an event that creates a wave. This wave propagates along the line and impacts the speed of trains, upstream from the incident location.

An incident is characterized by a vector \( \mathbf{p} \) of parameters, such as its location, duration, and spatial expansion, which is the extent of the line it affects. \( \mathbf{p} \) is used to derive the average speed \( S(s_1, s_2, t, \mathbf{p}) \) between two stations \( s_1 \) and \( s_2 \) at time \( t \). Given the speed \( S(s_1, s_2, t, \mathbf{p}) \), the running time \( RT(s_1, s_2, t, \mathbf{p}) \) between two stations \( s_1 \) and \( s_2 \) of a train departing \( s_1 \) at time \( t \), separated by distance \( d(s_1, s_2) \) is calculated by:

\[
RT(s_1, s_2, t, \mathbf{p}) = \arg_x \left\{ \int_t^{x} S(s_1, s_2, \tau, \mathbf{p}) d\tau = d(s_1, s_2) \right\} \quad (4.13)
\]

The main difficulty in using this model is the relationship between the incident characteristics \( \mathbf{p} \) and the speed values. For instance, Figure 4-7 shows an example of several speed curves at different locations during an incident. The orange curve, for the traffic located downstream on the line, is not affected by the incident and the speed curve remains flat. Upstream of the incident, the green curve shows that the traffic is suddenly interrupted and recovers sharply. Between these two points, the disruption does not impact immediately the service. Some time is needed before the speed starts to decrease. Similarly, the disruption needs some time to propagate after the incident, and the speed does not recover immediately. Under this model,
Figure 4-8 illustrates the effect of the speed reduction on a waterfall diagram. The small arrows represent the speed. If the arrow is horizontal, the speed is equal to 0. Red arrows show how the disruption propagates along the line. The figure also shows train trajectories, which are stream lines that follow the speed vectors. The parameters (i.e. slope of the speed profile, etc.) of the model can be calibrated so that the trajectories of the trains located close to the incident become realistic.

The model is static and does not capture the effect of a specific control plan on the speed profile. For instance, if there is no train on the line, the running time between two stations should not be changed by the incident, because there is nothing to impede a train to run at its maximum speed. On the contrary, it is the congestion that causes the propagation of the delays due to the incident and slows down trains.

**Role of a Simulation Model**

The train reformation problem does not require a simulation model, but it can be useful. It is possible to use a simulation model as an evaluation tool. Evaluating a recovery plan with simulation would be more accurate than using the output of the optimization problem. It is also possible to use a simulation to calibrate the model parameters. For example, it could be used to estimate the parameters of the speed model described in the previous section.

**4.4.2 Uncertainty Modeling**

The other main difference between the incident and the recovery phase is the uncertainty, especially with respect to incident characteristics (e.g. duration). The train reformation problem is based upon the assumption that the characteristics of the incident are known. However, this is not the case during the incident phase. Because of the uncertainty, controllers do not make decisions the same way during the incident phase based on the same principles as in the recovery phase. For instance, during the incident phase, they are likely to cancel more trains, not only to relieve
Far from incident
Upstream
Close to incident
Upstream
At the incident
Downstream of the incident

Figure 4-7: Speed curves

Figure 4-8: Trajectories and Speeds under the Speed Model
the congestion, but also to have more resources available and more flexibility in case the incident is more severe than expected. Therefore, it is important to consider the train reformation problem in a broader context that can handle uncertainty.

**Simple Approach**

The easiest way to handle uncertainty is to generate several scenarios of the incident, i.e. same cause, time and location, but with different durations. For each of these scenarios, the train reformation problem can be solved for the recovery phase. It is possible to evaluate how a service plan behaves under the other scenarios, and select the best overall solution (for example, the solution that remains feasible or provides good passenger service across scenarios), across all scenarios.

**Probabilistic Approach**

The previous approach fails to capture the caution that drives controller actions due to the uncertainty. It is important to include the uncertainty directly in the train reformation module.

Let us assume that there is a set of possible scenarios \( I^1, I^2, \ldots, I^N \) of different incident durations, with probabilities \( p^1, p^2, \ldots, p^N \). The objective function \( F(x) \) can be modified to represent the expected cost of a control plan during the incident phase (Equation 4.14), or to represent the worst possible cost (Equation 4.15).

\[
F_{\text{uncer}}(x) = \sum_{k=1}^{N} F(x|I^k) p^k \quad (4.14)
\]

\[
F_{\text{uncer}}(x) = \min_k (F(x|I^k)) \quad (4.15)
\]

The challenge with equations 4.14 and 4.15 is the definition of \( x \). In the recovery phase problem formulation, the journeys included in the solution \( x \) depend on the incident itself, since the set of candidates is generated specifically for each scenario. For the combined incident/recovery problem, the definition of \( x \) must be modified
so that the objective function can be estimated under every scenario. All scenarios share the same location and start time, which means that the scenario-specific sets of candidates share the same starting point. Let us denote by $t_{min}$ the duration of the shortest scenario. Then, the candidates for the first $t_{min}$ will be the same in any scenario. In addition, we use the following notation:

- $N$: number of scenarios.
- $x^k$: decision vector of size $N(x^k)$, built upon the set of candidates under scenario $I^k$.
- $x = (x^1, \cdots, x^N)$: decision vector of all sets of candidates. $N(x) = \sum_{k=1}^N N(x^k)$.
- $x^k_i$: binary decision variable, member of $x^k$. It is equal to 1 if candidate $i$ of scenario $I^k$ is included in the recovery plan of scenario $I^k$.
- $x_{t_{min}}(i,k)$: part of the journey $i$ that occurs during the first $t_{min}$ after the beginning of the incident.
- $J_{t_{min}} = \bigcup_{i,k} \{x_{t_{min}}(i,k)\}$: collection of the distinct $t_{min}$ first minutes of the candidates, of size $N(J_{t_{min}})$.
- $B$: binary matrix of dimension $N(x) \times N(J_{t_{min}})$. Its members $b_{ij}$ are equal to 1 if $t_{min}(i,k) = j$, where $j \in J_{t_{min}}$. It means that the first $t_{min}$ minutes of candidate $i$ of scenario $I^k$ is $j$.

Then the objective function is:

$$\minimize \quad F_{uncer}(x) = \sum_{k=1}^N F(x^k|I^k)p^k \quad \text{or} \quad F_{uncer}(x) = \min_k (F(x^k|I^k))$$

subject to

$$y_j = \frac{\sum_{k,i} b_{ij} x^k_i}{N}, \quad \forall j$$

$$y_j \in \{0, 1\}$$

(4.16)

The introduction of $y$ and the new constraint ensures that if a candidate is selected, the first $t_{min}$ minutes of its journey will be operated in every scenario. Figure 4-9 shows an example. When an incident happens at $t = 0$, three scenarios are consid-
Scenario 1 assumes that the incident will last 10 minutes, scenario 2, 20 minutes and scenario 3, 30 minutes. The right part shows that $x^1$, $x^2$, and $x^3$ are common in the first 10 minutes, while $x^2$, and $x^3$ are common from 0 to 20 minutes. However, the service plans can be totally different during their corresponding recovery phases.

Figure 4-9: Probabilistic TRP Example

The implementation of the joint incident/recovery TRP is very similar to the TRP for the recovery phase. The initial conditions correspond to the beginning of the incident phase. They are the locations of the trains and drivers at the beginning of the incident. For each scenario, the journey generation is done in parallel. The timetable expansion and the reformation phases are processed as described in Section 4.2. The objective function components $F(x^k|I^k)$ and the constraints are estimated as in Section 4.3, except for the new constraints and the modified objective function of equation 4.16. Only the stop time calculation changes, because the speed performance model is used to calculate arrival times.
Chapter 5

Train Reformation Problem: Implementation

The general approach to the train reformation problem (TRP) was discussed in chapter 4. The scope of the general problem was presented and a number of possible simplification were identified. This chapter presents the train reformation problem implementation for the Piccadilly Line.

The need for a good performance model to estimate running times and calculate stop times during the incident phase is a major challenge in the TRP solution and its implementation. Section 4.4 presented different ways to extend the TRP to the incident phase. However, the focus of this thesis is on the recovery phase, using the delay as a proxy for the initial positions of the trains. In this model, the initial positions of the trains at the beginning of the recovery phase are based on the service during the disrupted day. Their position is characterized by the difference between the expected arrival time of the train at the first station after the end of the incident and its scheduled arrival time. This difference can be positive if the train is late, or negative if the train is early (often the case if the train was short-turned by controllers during the incident phase). The trains that were withdrawn during the incident are also withdrawn in the model, but may be reintroduced in the recovery timetable.
5.1 Journey Generation

To simplify the problem, only the main reversing stations of the Piccadilly Line are included in the model, as shown in Figure 5-1. Trains can reverse at the ends of the line and at Acton Town, Arnos Grove, and Northfields. They can also reverse toward the trunk at Rayners Lane and Wood Green. These five stations account for the vast majority of reversals in the Piccadilly Line. It was decided not to include reversing points that are rarely used in practice during disruptions.

Figure 5-1: Line Geometry as Implemented

5.1.1 Timetable Expansion Stage

Method

A simplified version of the timetable expansion phase has been implemented. It is important to limit the number of possibilities to reduce the computation time. At the same time, there is a trade-off between the number of journeys generated and the quality of the solution provided. Therefore, the journey generation should not be over simplified. The following assumptions have been made:

- Extension operations are not considered. During operations, extensions are uncommon and are mainly used when a train is early because of a reformation or a previous short-turn. The other case where extensions are used is when the
train is sent to another branch because of congestion on the scheduled branch. The practical difficulty to implement extensions lies in the fact that the number of potential extensions is unlimited. In practice, because most trains are late rather than early, only few potential extensions are feasible. Therefore, short-turning is the main strategy considered.

- A time limit for control actions is implemented. Controllers think about what actions can be taken in the next few hours rather than for the rest of the day. Therefore, it is more realistic to set a limit to the control action time window. Moreover, section 5.2.1 will show that it is also a good way to influence the recovery duration.

- No short-turning is applied if it would require make an operator swap infeasible. This reduces the complexity of the problem considerably. Only short-to-medium duration cancellations (cancellations shorter than a train block) are generated. One justification for this assumption is that controllers seldom do longer short-turns that bypass swaps because they require moving the relief location, which adds more costs than it brings benefits. For trains that have been canceled by controllers, late reintroductions will be generated at a later stage of the TRP approach. The end of section 5.1.2 explains these reintroductions and introduces a way to study longer cancellations.

Algorithm (1), shown in Figure 5-2, is used to generate the potential journeys. The following notation is used:

- $C_0$: timetable journeys.
- $C_i$: set of generated journeys. The journeys in $C_i$ have been generated using $i$ short-turns. $C_{ij} \in C_i$ are the journeys.
- $t_s$ and $t_e$: start and end times of the period where control actions can be taken.
- $S(C_{ij})$: set of stops visited by journey $C_{ij} \in C_i$.
- $t_a(s^k)$: arrival time at stop $s^k \in S(C_{ij})$.
- $t_d(s^k)$: departure time at stop $s^k \in S(C_{ij})$.
- $n(C_{ij})$: train number of journey $C_{ij}$.
For each journey, the algorithm performs a double loop on the set of stations. It looks for potential short-turns (condition of line 6) that both satisfy the time limit conditions (lines 4 and 5) and that no swap station is bypassed (line 17). The journey is short-turned (line 7) and added (line 9) after we make sure that it is not a duplicate (line 8). The algorithm is run on every journey of the timetable, then recursively on the new journeys, until it becomes impossible to find new short-turns.

**Algorithm 1** Generate Short-turned variations

1: \( i \leftarrow 0 \)
2: while \( C_i \neq 0 \) do
3: \[ \text{for all } C_{ij} \in C_i \text{ do} \]
4: \[ \text{for all } s^1 \in S(C_{ij}) \text{ such as } t_s \leq t_d(s^1) \leq t_e \text{ do} \]
5: \[ \text{while } s^2 \in S(C_{ij}) \text{ such as } t_d(s^1) \leq t_d(s^2) \leq t_e \text{ do} \]
6: \[ \text{if } s^2 = s^1 \text{ then} \]
7: delete \([s^1 + 1, s^1 + 2 \ldots s^2 - 1, s^2] \)
8: \[ \text{if } \#C_{ij'} \in C_i \text{ such as } n(C_{ij}) = n(C_{ij'}) \text{ and } S(C_{ij}) = S(C_{ij'}) \text{ then} \]
9: \[ \text{add } j \text{ to } C_{i+1} \]
10: \( s^2 \leftarrow s^2 + 1 \)
11: \[ \text{else} \]
12: \( s^1 \leftarrow s^1 + 1; \text{goto 5} \)
13: \[ \text{end if} \]
14: \[ \text{else if } s^2 \text{ is a swap station then} \]
15: \( s^1 \leftarrow s^1 + 1; \text{goto 5} \)
16: \[ \text{else} \]
17: \( s^2 \leftarrow s^2 + 1 \)
18: \[ \text{end if} \]
19: \[ \text{end while} \]
20: \( s^1 \leftarrow s^1 + 1; \text{goto 5} \)
21: \[ \text{end for} \]
22: \( j \leftarrow j + 1; \text{goto 4} \)
23: \[ \text{end for} \]
24: \( i \leftarrow i + 1 \)
25: \[ \text{end while} \]
Initialize $C_{ij}$

$C_0 = $ Timetable

$i = 0$

Initialize $C_i$

$j = 0$

Initialize journey $C_{ij}$

$t_d(s_1) < t_s$

Try next station

$s_1++$

Try next journey

Is $j$ the last journey of $C_i$?

no

Is $s_1$ in the decision window?

not

$s_2 = s_1 + 1$

$s_2$ is in the decision window

Look at $s_2$

$s_2++$

Add new journey to $C_i$

Create new journey $NJ = C_{ij}$ then remove stations between $s_1$ and $s_2$

Is $NJ$ a duplicate?

no

Add new journey to $C_{i+1}$

$s_2 = s_1$

yes

$s_1 = 1$

Initialize Station

Figure 5-2: Algorithm \[1\] Flowchart
Example

Let $t_s = 15:30$, $t_e = 17:00$ and $C_0 = C_{01}$, where $C_{01}$ is a timetable journey composed of the sequence of stops $(s^i)$, where $i$ goes from $a$ to $k$. The sequence of stops is shown in the following table.

<table>
<thead>
<tr>
<th>Stop</th>
<th>Station</th>
<th>Scheduled departure time, $t_d(s)$</th>
<th>swap station</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s^a$</td>
<td>RLN</td>
<td>14:35</td>
<td>x</td>
</tr>
<tr>
<td>$s^b$</td>
<td>ACT</td>
<td>14:55</td>
<td>o</td>
</tr>
<tr>
<td>$s^c$</td>
<td>WGN</td>
<td>15:35</td>
<td>x</td>
</tr>
<tr>
<td>$s^d$</td>
<td>AGR</td>
<td>15:40</td>
<td>x</td>
</tr>
<tr>
<td>$s^e$</td>
<td>CFS</td>
<td>15:55</td>
<td>x</td>
</tr>
<tr>
<td>$s^f$</td>
<td>AGR</td>
<td>16:10</td>
<td>x</td>
</tr>
<tr>
<td>$s^g$</td>
<td>WGN</td>
<td>16:15</td>
<td>x</td>
</tr>
<tr>
<td>$s^h$</td>
<td>ACT</td>
<td>16:55</td>
<td>o</td>
</tr>
<tr>
<td>$s^i$</td>
<td>NFD</td>
<td>17:02</td>
<td>x</td>
</tr>
<tr>
<td>$s^j$</td>
<td>ACT</td>
<td>17:13</td>
<td>x</td>
</tr>
<tr>
<td>$s^k$</td>
<td>NFD</td>
<td>17:26</td>
<td>x</td>
</tr>
</tbody>
</table>

The following tree displays the different journeys that have been generated and the structure through which they have been created. The first level of the tree is $C_0$, the original set of journeys. The second level is $C_1$ and the generation stops after the third level $C_2$. Each line of each cell corresponds to a train block. To go down a level, a short-turn must be performed. This short-turn must be within a train block: to go from $C_{01}$ to $C_{11}$, only the second line, corresponding to the second train block is modified. From $C_{01}$ to $C_{11}$, the sequence WGN-AGR-CFS-AGR-WGR is short-turned. From $C_{01}$ to $C_{12}$, it is AGR-CFS-AGR. From $C_{01}$ to $C_{13}$, $C_{11}$ to $C_{21}$, and $C_{12}$ to $C_{22}$, the sequence ACT-NFD-ACT is deleted.

Algorithm 1 will perform the following steps:
Figure 5-3: Journey Generation Tree

1. $s^1 \leftarrow s^a$, $i \leftarrow 0$, $j \leftarrow 1 \Rightarrow C_{ij} = C_{01}$

2. As $t_d(s^1) = 14:35 < t_s$ at the beginning of the journey, $s$ is incremented until $s^1 \leftarrow s^c$.

3. $s^2$ is incremented until $s^2 \leftarrow s^g$ because $s^g = s^c = WGN$.

4. $(s^d \ldots s^f)$ are deleted.

5. $C_{11} = (s^a \ldots s^c, s^g \ldots s^k)$ is added to $C_1$

6. $s^2$ is incremented until $s^2 \leftarrow s^h$ and the loop breaks because $s^h$ is a swap station.

7. $s^1 \leftarrow s^d$

8. Following the same algorithm, $C_{12} = (s^a \ldots s^d, s^f \ldots s^k)$ is added to $C_1$

9. Because of $s^h$, the inner loop does not generate any trip and $s^1$ is incremented until $s^1 \leftarrow s^h$

10. Following the same algorithm, $C_{13} = (s^a \ldots s^h, s^j \ldots s^k)$ is added to $C_1$
11. $s^1 \leftarrow s^i$. As $t_a(s^i) > t_e$, the loop breaks.

12. $i \leftarrow 1, \Rightarrow C_i = C_1$

13. $j \leftarrow 1 \Rightarrow C_{ij} = C_{11}$

14. The algorithm goes on until $s^1 \leftarrow s^h$ and the journey $C_{21} = (s^a \ldots s^c, s^g \ldots s^h, s^j \ldots s^k)$ is added to $C_2$

15. $j \leftarrow 2 \Rightarrow C_{ij} = C_{12}$

16. The algorithm goes on until $s^1 \leftarrow s^c$ and the journey $(s^a \ldots s^c, s^g \ldots s^k)$ is generated. However, the journey is rejected because it is a duplicate of $C_{11}$

17. The algorithm goes on until $s^1 \leftarrow s^h$ and the journey $C_{22} = (s^a \ldots s^d, s^j \ldots s^h, s^j \ldots s^k)$ is added to $C_2$

18. $j \leftarrow 3 \Rightarrow C_{ij} = C_{13}$, but no new journey is added because only duplicates are created.

19. $i \leftarrow 2, \Rightarrow C_i = C_2$, but no journey is created

20. the algorithm terminates

5.1.2 Reformation Stage

The reformation stage generates new journeys from the journeys created by the timetable expansion phase. The last part of this section shows how canceled trains are handled.

Algorithm

The following two algorithms describe the method to generate a group, given a decision time $t_{\text{dec}}$, and the method to generate journeys after groups have been formed. To simplify the notation, a journey is noted $j_\bullet$ (instead of $C_\bullet$). The output of these two algorithms is a set of renumbered journeys: they will contain train blocks that originally belonged to two different journeys from the timetable. Figure 5-4 illustrates
the reformation process.

Let:

- \( t_{\text{dec}} \): time when the renumbering decision is taken
- \( \mathcal{C} \): set of journeys generated after the timetable expansion phase
- \( s^d(t_{\text{dec}}) \): station where the renumbering decision is made
- \( \mathcal{G}(t_{\text{dec}}, s) \): group of trains whose next swap after \( t_{\text{dec}} \) is at station \( s \)
- \( \mathcal{G} \): set of all train groups.
- \( \mathcal{G}_1 \): set of the first legs of \( \mathcal{G} \)
- \( \mathcal{G}_2 \): set of the second legs of \( \mathcal{G} \)

**Algorithm 2** Train Group Formation

1. for all \( j \in \mathcal{C} \) do
2. \( s \leftarrow s^d(t_{\text{dec}}) \)
3. while \( s \) is not a relief station do
4. \( s \leftarrow s + 1 \)
5. end while   //the while loop stops at the first swap after \( t_{\text{dec}} //
6. add \( j \) to \( \mathcal{G}(t_{\text{dec}}, s) \)
7. end for

**Algorithm 3** Journey Route Generation

1. for all \( \mathcal{G} \in \mathcal{G} \) do
2. for all \( j^1 \in \mathcal{G}_1 \) do
3. for all \( j^2 \in \mathcal{G}_2 \) do
4. concatenate \( j^1 \) and \( j^2 \)
5. end for
6. end for
7. end for

\( t_{\text{dec}} \) is the time at the end of the incident. The initial location are assumed known (from automatic vehicle location data). Setting the decision time later would imply a more uncertain position of the trains, and it would assume that controllers do not
manage trains at the beginning of the recovery phase. That is why a single decision, taken at the end of the incident phase has been implemented.

Referring to element 2 of the list in section 4.2.2, it is possible to implement multiple decision times, but this has not been implemented: the algorithm 2 would be applied several times with different $t_{dec}$, and each journey would belong to several groups. However, this was not implemented because the increase in computation time may be substantial.

Finally, multiple renumberings (element 3 of the list in section 4.2.2) per train should not be allowed. The first reason is that multiple renumberings are uncommon. When it happens, it is likely that controllers have lost track of which driver is in which train, or a spare driver has been used. Therefore, when they decide to do a second or third renumbering, controllers do not account for the various crew constraints and the decision is driven more by timetable adherence objectives. Considering that crew constraints are at the center of the model, it would be more consistent to relax driver constraints instead of using very complicated renumberings. The second problem with multiple renumberings is that the number of generated journeys increases exponentially with the number of renumberings. Given the fact that a large number of journeys are generated even with only one renumbering, adding multiple renumberings is not computationally feasible. Finally, the problem of the estimation of the location of the trains when they are reformed would become greater with multiple renumberings: having an accurate performance model becomes more important as the variability increases at each renumbering.

Example

This example illustrates the implemented method. Let us consider a reformation process taking place at 15:45, with 3 journeys to renumber. To simplify, only the
Journeys from Stage 1

Apply Incident delay to \( j \)

Identify \( s_d \), the first station after the end of the incident

Identify \( s \), the first swap station after \( s_d \)

Generate \( j^1 \), composed of stops occurring before \( s \)

Generate \( j^2 \), composed of stops occurring before \( s \)

Add \( j^1 \) to \( G_1(s) \)

Add \( j^2 \) to \( G_2(s) \)

For all groups \( G \)

Group \( G \)

Generate new journey \( J = (j^1, j^2) \)

For all pairs \( (j_1, j_2) \) of \( G \times G \)

Reformation

Figure 5-4: Reformation Flowchart
swap stations of the journeys are considered:

- \( J_1 = (C, 15:00), (A, 16:00), (B, 17:00), (A, 18:00) \)
- \( J_2 = (D, 15:15), (A, 16:15), (C, 16:30), (D, 17:15) \)
- \( J_3 = (A, 15:15), (B, 16:00), (B, 17:15), (C, 17:15) \)

Table 5.2: Group Formation

<table>
<thead>
<tr>
<th>Journey</th>
<th>( s^d(15:45) )</th>
<th>First Leg ( t_d = 15:45 )</th>
<th>Second Leg ( t_d = 15:45 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( J_1 )</td>
<td>(A, 16:00)</td>
<td>C-A</td>
<td>A-B-A</td>
</tr>
<tr>
<td>( J_2 )</td>
<td>(A, 16:15)</td>
<td>D-A</td>
<td>A-C-D</td>
</tr>
<tr>
<td>( J_3 )</td>
<td>(B, 16:00)</td>
<td>A-B</td>
<td>B-B-C</td>
</tr>
</tbody>
</table>

Table 5.2 shows how the groups are generated. \( J_1 \) and \( J_2 \) belong to \( G(A, 15:45) \), and \( J_3 \) belongs to \( G(B, 15:45) \). Tables 5.3 and Table 5.4 show how 8 journeys are generated from the 2 groups, including the original journeys. Also notice that the order of magnitude of the number of generated journeys is the square of the order of magnitude of journeys available at the beginning of the reformation stage.

Table 5.3: Group (A,15:45)  
Table 5.4: Group (B,15:45)

<table>
<thead>
<tr>
<th>First Leg</th>
<th>( J_1 ): A-B-A</th>
<th>( J_2 ): A-C-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>( J_1 ): C-A</td>
<td>C-A-B-A</td>
<td>C-A-C-D</td>
</tr>
<tr>
<td>( J_2 ): D-A</td>
<td>D-A-B-A</td>
<td>D-A-C-D</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>First Leg</th>
<th>( J_1 ): B-A</th>
<th>( J_3 ): B-C</th>
</tr>
</thead>
<tbody>
<tr>
<td>( J_1 ): C-A-B</td>
<td>C-A-B-A</td>
<td>C-A-B-C</td>
</tr>
<tr>
<td>( J_3 ): A-B-B</td>
<td>A-B-B-A</td>
<td>A-B-B-C</td>
</tr>
</tbody>
</table>

Reintroduction

In this section, we discuss how reformations are handled. For the trains that the controllers canceled during the incident phase, it is not possible to apply the group generation and the renumbering described above. The goal of reformation is to
renumber journeys so that they are on time. However, canceled trains are neither on time nor late. Moreover, they do not hinder the rest of the traffic. For this reason, it makes sense not to reform these trains, and simply reintroduce them as being on time. The following algorithm describes how trains are reintroduced. The train is stored in a depot near a station. In the timetable, its journey that would have run, had the train not been canceled, and would have stopped at this depot several times. If the train is reintroduced at a time that coincides with one of the stop times of the timetable, it would be reintroduced on time. Hence, each of these reintroduction possibilities is added to the list of candidates. Note that it was decided to allow reintroductions both during and after the end of the control period. Contrary to short-turns and reformations, resuming a train does not depend on the quality of the running time performance model. It is a simpler operation, and the number of possibilities increases roughly linearly with the time window (versus quadratic for short-turns): controllers are able to anticipate reintroductions earlier. Therefore, allowing late reintroductions has very few drawbacks. Let:

- \( C^c \subset C_0 \): the set of journeys canceled during the incident phase
- \( t_s \): start of the period where control actions can be taken
- \( S_j = (s^1 \ldots s^N) \): set of stations of the scheduled journey \( j \in C^c_i \)
- \( s^c \): station where the train is located after its cancellation.

**Algorithm 4** Reintroduction algorithm

1: for all \( j \in C^c \) do  
2:   for all \( s \in S_j \) do  
3:     if \( s = s^c \) and \( t(s) > t_s \) then  
4:       add \( (s \ldots s^N) \) to the list of candidates  
5:     end if  
6:   end for  
7: end for

It is possible to reform canceled trains to perform a stock renumbering (defined in section [2.4]): a canceled train will lose its number, while another train will inherit the
number of the originally canceled journey at relief, as in any other reformation. Then, when the canceled train is reintroduced, it would take the number of the reformed train. The only difference in the algorithm, compared to normal renumberings, is that canceled journeys are not split: they are placed directly in the second leg of train groups.

5.1.3 Station Arrival and Departure Times

At this stage of journey generation, only the itineraries of the journeys have been changed: arrival and departure times at stations must be calculated. Because the TRP is restricted to the recovery phase, it is easier to estimate the arrival/departure times.

During the recovery phase, if the capacity constraints (equation 4.7) are respected, nothing impedes trains as they run. For this reason, using the running times from the timetable is reasonable. Slightly longer dwell times can be expected, so slightly increasing running times at the start of the recovery phase should be considered. Yet, as the optimization will force the frequency to be similar to the schedule, after some time, the dwell time should not differ much from the schedule. The inaccuracy of the running times during the recovery phase should be limited. It is assumed that the running time is the same as the running time in the timetable.

Holding variations were introduced at the end of section 4.2.3. They are candidates whose journey is held in order to recover the earliness that may occur after short-turns. There is an infinite combination of holding patterns than can be generated from a single journey. It is important to find a balance in the number of holding candidates. On one hand, having these holding variations matters because they represent what can happen during operations. On the other hand, having too many candidates that differ only by the station where holding takes place is not very useful. Therefore, for each journey, at most three candidates are generated.
• The first candidate is simply the journey that comes from the reformation stage.

• The second candidate is a holding variation, called *simple holding variation*. If, from a certain swap happening at time $t$, every subsequent swap is early (the relief is early and the pick-up is late), the train will be held so it is on time. The holding location is the last reversing station before $t$.

• The third candidate is a holding variation called *cancellation holding variation*. If, from a certain swap at time $t$, every subsequent swap is more than 15 minutes early (the relief is early and the pick-up late), the train is withdrawn from service then reintroduced at the same location. The station where the train is removed is the last train depot before $t$.

The second variation represents classic holding that can happen if a train is running early. The third is similar to the second, with two differences: the train must be at least 15 minutes early, and the station where the train is withdrawn must be a train depot. This ensures that longer withdrawals happen at only a few stations where the capacity is high. Figure 5-5 represents the flowchart of the running time calculation stage.

![Stop Time Calculation Flowchart](image)

Figure 5-5: Stop Time Calculation Flowchart

**Example**

Let us consider the journey of the second line from Table 5.5. The circled stations are swap stations. The journey has been short-turned just after the relief at B. We
also have running times \(RT(A, B, t)\) and \(RT(B, A, t)\) equal to 15 minutes. At the end
of the reformation phase, only the first stop time is known. After applying the run-
ing time calculation, one journey is added to the candidates. Because the journey
is short-turned after the swap at B, the journey is 10 minutes early. Therefore, the
journey is held at B and a simple holding variation is added to the list of candidates.

<table>
<thead>
<tr>
<th>Stop</th>
<th>(\mathbb{A})</th>
<th>B</th>
<th>A</th>
<th>(\mathbb{B})</th>
<th>A</th>
<th>B</th>
<th>A</th>
<th>B</th>
<th>(\mathbb{A})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheduled Time</td>
<td>15:30</td>
<td>15:45</td>
<td>16:00</td>
<td>16:15</td>
<td>16:30</td>
<td>16:45</td>
<td>17:00</td>
<td>17:15</td>
<td>17:30</td>
</tr>
<tr>
<td>Candidate after RT calculation</td>
<td>15:40</td>
<td>15:55</td>
<td>16:10</td>
<td>16:25</td>
<td></td>
<td></td>
<td>16:50</td>
<td>17:05</td>
<td>17:20</td>
</tr>
<tr>
<td>Simple holding variation</td>
<td>15:40</td>
<td>15:55</td>
<td>16:10</td>
<td>16:45</td>
<td></td>
<td></td>
<td>17:00</td>
<td>17:15</td>
<td>17:30</td>
</tr>
</tbody>
</table>

### 5.2 Journey Selection

#### 5.2.1 Objective Function

The objective function of the train reformation problem was given in section \[4.3.1\]

\[
\min F(x) = W(x) + D(x) + \Theta(x)
\]  \[4.1\]

where \(W\) was the passenger service cost term, \(D\) the timetable adherence term, and \(\Theta\)
the complexity term. This section will discuss in detail how each term is calculated.
Passenger Service Cost

The service component was detailed in equation 4.2: \( W(x) = \sum_{o,d,t} d_{od,t} f_w(n_{od,t}) \), where \( d_{od,t} \) represents the demand of the origin-destination pair/time period (ODT) \( od, t \) and \( f_w(x) \) is a function that approximates the waiting time. This subsection will discuss these two terms.

Origin Destination Matrix  The duration of a time period is set to 15 minutes, which corresponds to the time range of the demand matrices provided by TfL. Yet, only a subset of stations has been integrated in the model (see Figure 5-1). To simplify the model, the 3 Heathrow stops were combined for the demand term. The demand matrix must be aggregated so that the demand for the ignored stations is taken into account.

Figure 5-6: Full Demand Matrix \( \Delta_t \)

Figure 5-6 illustrates how the demand matrix \( D_t = (d_{od,t}) \) is calculated from the full demand matrix \( \Delta_t = (\delta_{od,t}) \). A, D, and F are the major stations and B, C, and E are intermediate stations that are not included in the model. \( d_{AD,t}, d_{DA,t}, d_{AF,t}, d_{FA,t}, d_{DF,t} \) and \( d_{FD,t} \) will be calculated using the origin-destination matrix \( \Delta_t \) for
the full line. For instance,

\[
\begin{align*}
d_{AF,t} &= \delta_{AE,t} + \delta_{AF,t} + \delta_{BE,t} + \delta_{BF,t} + \delta_{CE,t} + \delta_{CF,t} \\
&= 4 + 0 + 8 + 0 + 12 + 2 \\
&= 26 \\
d_{AD,t} &= \delta_{AB,t} + \delta_{AC,t} + \delta_{AD,t} + \delta_{BC,t} + \delta_{BD,t} + \delta_{CD,t} \\
&= 5 + 2 + 1 + 5 + 1 + 3 \\
&= 17
\end{align*}
\]

Note that \(\delta_{od,t}\) is assigned to the ODT \(d_{o'd,t}\), where \(o'\) is the station before \(o\) included in the representation of the line, and \(d'\) is the major station after \(d\), hence \(\delta_{od,t}\) are assigned to a major ODT. For instance, a train serving \(AF, t\) serves necessarily \(CE, t\): its stop sequence is A-B-C-D-E-F, which includes C-D-E.

**Service Matrix**  This section explains how the term \(n_{od,t}\) of equation 4.2, the number of trains serving OD pair \(od\) in time period centered around time \(t\), is calculated. \(n_{od,t}\) is an element of the vector \(n = Wx\). \(W\) is called the *service matrix*, which associates all candidate journeys with the ODTs. \(n_{od,t} = 0\) means that no train is serving \(od, t\), and \(n_{od,t} = 10\) means that 10 trains are serving \(od, t\). \(w_{(od,t)i}\) is an element of \(W\), that specifies if the journey \(x_i\) contributes to the term \(n_{od,t}\). \(w_{(od,t)i} = 0\) means that journey \(i\) does not serve the OD pair \(od\) in the time period \(t\).

For the purposes of this application, the above strict definition of \(w_{(od,t)i}\) is relaxed, to let \(w_{(od,t)i}\) take any values between 0 and 1 to mitigate the boundary effects of the time intervals. Because of the variability of running times, a journey \(i\) scheduled to depart a station at 14:52 could leave at 14:53. Therefore, it is reasonable to define \(\{w_{(od,14:45)i}, w_{(od,15:00)i}\} = \{0.55, 0.45\}\) instead of having \(\{w_{(od,14:45)i}, w_{(od,15:00)i}\} = \{1, 0\}\). \(w_{(od,t)i}\) becomes the possibility of journey \(i\) serving passengers of the ODT \(od, t\).
To capture the above effect, the function $\lambda(\cdot)$, shown in Figure 5-7, is applied to the difference between $\tau$, the departure time of a journey, and $t$, the center of the time period time. $\lambda(\tau - t)$ is the probability that a stop departure planned at $\tau$ actually falls within the time period centered on $t$.

$\lambda(\tau - t) = \begin{cases} 
1 & : |\tau - t| \leq 4 \\
\frac{11 - |\tau - t|}{7} & : |\tau - t| \in [4, 11] \\
0 & : \text{otherwise}
\end{cases}$

The purpose of $\lambda$ is to avoid a threshold effect at the boundary of two time periods. Setting longer time periods would have mitigated the problem, as the probability of a train stop falling within a time period becomes closer to 0 or 1 when the time period length increases. However, it weakens the assumption of uniform headways, which is used to estimate the passengers cost. If train $i$ serves an OD $od$ at time $\tau$, the coefficient $\lambda(\tau - t)$ will be different from 0 if $t$ is within 11 minutes of the departure time. Notice that $\sum_{k \in \mathbb{N}} \lambda(x - 15k) = 1$, which ensures that the total probability of a train serving an OD over the time periods is always equal to one, even if the stop is assigned to several time periods.

A second function, $\sigma(od_1, od_2)$, is introduced to model transfers, where $od_1$ and $od_2$ are two ODs. $\sigma(od_1, od_2)$ is the probability for passengers of $od_2$ to take trains serving $od_1$. Let us consider passengers going from Uxbridge to Heathrow. There is no direct service, and passengers to Heathrow will take a train to Acton Town and transfer there. For these passengers, a train going from Uxbridge to Acton Town is useful, even if Acton Town is not their final destination. Therefore, $\sigma(od_1, od_2)$ specifies how serving the OD pair $od_1$ (like UXB-ACT) satisfies the demand of the OD pair $od_2$ (UXB-HRV).

- $\sigma(od_1, od_1)$ is always equal to 1.
- $\sigma(od_1, od_2)$ is equal to 1 if $od_2$ is an OD pair from the Heathrow branch to the Uxbridge branch (or opposite) and that the destination of $od_1$ is Acton Town,
as the situation in the example. This case captures the fact that, if there are 
no trains serving $od^2$ directly, passengers will board a train serving $od^1$ to reach 
their destination.

- $\sigma(od^1, od^2)$ can be between 0 and 1. For some OD pairs (for instance UXB to 
CFS), the frequency is low. Some customers will be willing to take a train to 
AGR, and transfer there. For instance, $\sigma(UXB - AGR, UXB - CFS)$ may 
take a value of 0.5.

Equation 5.1 is used to estimate $w_{(od,t)i}$.

$$w_{(od,t)i} = \max_{od^1 \in O, s \in S_i} \{ \lambda(\tau - t) \cdot \sigma(od^1, od^2) \cdot x_{i,s}^{od^1} \}$$  \hspace{1cm} (5.1)

Where:

- $S_i$: set of train stops for journey $i$.
- $\tau$: departure time of $s \in S_i$.
- $t$: time in the center of time period.
- $O$: set of OD pairs.
- $od^1$ and $od^2$: two OD pairs in $O$. 

Figure 5-7: Curves of $\lambda$ for different time periods
- $x_{i,s}^{od_i}$: binary variable equal to 1 if journey $i$ serves the OD pair $od^i$ when it leaves stop $s \in S$.

The max operation is needed to prevent multiple values in some cases. For example, a train going from Uxbridge to Cockfosters will first stop at Arnos Grove, where $\sigma(UXB - AGR, UXB - CFS) = 0.5$, and afterward at CFS, where $\sigma(UXB - CFS, UXB - CFS) = 1$. Without the max operation, $w_{i(UXB - CFS,t)}$ would be calculated twice, with two distinct values.

This equation calculates the service coefficient $w_{i(od^2,t)}$ of journey $i$ for the ODT $od^2, t$. It looks at every departure of the journey $i$ (term $\max_{s \in S}$) and at all possible OD pairs (term $\max_{od^1 \in O}$). If the journey $i$ does not serve the OD pair $od^1$ when it leaves stop $s \in S$, then $x_{i,s}^{od^1}$ is zero. Otherwise, $\lambda(\tau - t)$, the probability that journey $i$ serves the stop in period $t$, is multiplied by $\sigma(od^1, od^2)$, the probability that a passenger going to $d^1$ takes a train going to $d^2$.

**Passenger Service Cost Function**  The function $f_w$ is then applied to $w$. $f_w$ must represent the cost of waiting. Therefore, $f_w$ approximates the average waiting time which is inversely proportional to the number of trains. As the TRP is a linear integer problem, $f_w$ is approximated by a piecewise linear function. The number of pieces must be high enough to be a good approximation, but the complexity increases with the number of pieces. During the tests, 20 proved to be a good compromise. Moreover, $f_w$ was limited to an average waiting time of 22.5 minutes when the "number" of trains $n_{od,t}$ was lower than 0.3 trains per period. Figure 5-8 compares $f_w$ against the actual (nonlinear) function it approximates. The figure shows that $f_w$ follows the actual function very closely.

**Schedule Adherence Cost**

Getting back to schedule is one of the controllers’ most important objectives. Evaluating the delay of trains at the end of control interventions is a way to assess the effectiveness of the recovery. Because after the end of control interventions, jour-
neys follow the timetable, measuring the delay long after the recovery is equivalent
to measuring it immediately after the end of the control actions. Therefore, the com-
parison metrics between the recovery plan and the timetable is based on the delay of
the operator swaps two hours after the end of the control actions.

For the same reason, setting a limit to the action time is a way to play with the
duration of the recovery. It will either force the recovery to be shorter, or the high
delay cost will indicate that the recovery time is simply too short.

In addition to the delay 2 hours after the end of the control actions, the delay of
the relief for drivers that finish their shift and that happen during the recovery period
is also taken into account. Figure 5-9 illustrates how swaps in general contribute to
the calculation of the schedule adherence cost, or depending on the situation, intro-
duce hard constraints into the formulation.

In order to capture the increasingly higher cost of swap delay, an exponential
function is used:

\[ f_d(x) = e^{x/t_0} \quad \text{for} \quad x > 0 \]  \hspace{1cm} (5.2)

where:

- \( t_0 \): basic delay. The cost of the swap will be higher if the earliness (or delay) is higher than the basic delay. Because of the exponential form of \( f_d \), \( t_0 \) becomes a soft constraint for the tolerated delay. In the implementation, \( t_0 \) is equal to 10 minutes.

**Recovery Plan Complexity**

The complexity component promotes candidates that follow a preferred strategy. In the implementation, candidates that are not renumbered receive a cost bonus \( b \), if they are not reformed during the journey. This bonus is doubled if the journey is the same as the original journey of the timetable (i.e. without short-turn or holding).
5.2.2 Constraints

The optimization problem is subject to three constraints: capacity, crew constraints and uniqueness (see section 4.3.2). The implementation of those constraints is discussed in this section.

Capacity

The train reformation problem needs to consider capacity constraints to ensure that the selected candidates make a feasible recovery plan. Similar to the waiting time approach, the capacity constraint is applied not at the train level, but at the station-time interval level, IT. The occupancy matrix $T$ (of dimension $N \times k$ where $N$ is the number of journeys, and $k$ the number of IT) has elements $T_{i(st)}$ which specify how long Journey $i$ occupies IT $(st)$, where $s$ stands for the infrastructure and $t$ is the center of the time period.

To simplify, if a journey stops at station $s$, with $t_a(s)$ and $t_d(s)$ as its arrival and departure times, $T$ can be calculated using equation 5.3. The maximum in the formula ensures that any stop occupies the platform at least 90 seconds, which is an approximation of the minimum feasible headway.

$$T_{i(st)} = \max(90, t_d(s) - t_a(s)) \cdot \int_{t_a(s)}^{t_d(s)} \lambda(\tau - t) d\tau$$  \hspace{1cm} (5.3)

Equation 5.3 is a simplification of the actual implementation. Capacity at reversing stations is impacted by the train movements between the different platforms of a station. These movements depend on the train itineraries. For instance, a train reversing at Acton Town will occupy the westbound platform for the first 3 minutes, then will occupy the west siding, before using the eastbound platform for the last three minutes of the reversing. As a result, a single reversing stop at Acton Town impacts the occupancy of at least three ITs (eastbound, westbound and siding). TSM, the TfL simulation tool, discussed in detail in section 5.3.2, was used to choose the stations at which the occupancy constraint would be enforced. The constraints were
enforced at Acton Town (Eastbound, Westbound, West siding and East siding), Arnos Grove (Eastbound, Westbound, and the three platforms together), Cockfosters (the three platforms together), and Rayners Lane (Eastbound, Westbound and siding).

**Crew Availability**

Crew availability constraints were captured by equation 4.8.

\[ \text{Rx} \geq K_1 \]  

Matrix \( R = (r_{ji}) \) has \( N \) columns (one per candidate) and \( P \) rows, corresponding to the number of pickups between the end of the incident and two hours after the end of the decision time (after which they are involved in the delay calculation, see 5.2.1). Let \( p_j \) be a pickup, \( d_j \) its driver, and \( r_j \) -if it exists- the relief which immediately precedes \( p_j \) for driver \( d_j \).

- If journey \( i \) is not involved in the swap \( j \), \( r_{ij} = 0 \)
- If journey \( i \) contains \( p_j \), \( r_{ij} \) is the departure time of the task starting with \( p_j \).
- If journey \( i \) contains \( r_j \), \( r_{ij} \) is negative, and is the opposite of the arrival time of task preceding \( p_j \).
- If journey \( i \) contains \( p_j \) (respectively \( r_j \)), but the task involving it has been canceled, \( r_{ij} \) is set to an arbitrary large (small) number ensuring that the constraint will be respected. This can happen when a task has been canceled in the train generation phase because of short-turns, or because the journey is reintroduced after a cancellation in the incident phase.

- \( K_j \) is a slack time parameter depending on the type of relief. The value has been set to 35 minutes for a meal relief. For other reliefs, it has been set to 0 minutes. This setting implies that only meal breaks are enforced while other reliefs do not need margin.
• If column \( j \) corresponds to the first task of the day for the driver, \( K_j \) has been set to \(-5\) minutes. That assumes that drivers can be 5 minutes early for their first task of the day. This assumption relies on the observation that operators usually arrive before the beginning of their shift so as to be ready 5 minutes early on the platform.

**Uniqueness**

The uniqueness constraints were described by equation 4.10. When the TRP is restricted to the recovery phase, the inequality becomes into an equality, which ensures that all trains are present in the solution. In this case, the journeys canceled by controllers are modeled with the reintroduction variations.

It is also possible to use the TRP to study the number of longer cancellations (more than one train block). If equation 4.10 is used as an inequality, it is not necessary to assign as many trains as in the schedule. Some train numbers will not appear at all in the solution: they are canceled. This assumes that canceled trains do not impact other trains during their withdrawal from service.

### 5.3 Post-processing

#### 5.3.1 Fine-Tuning Module

The capacity constraints and the passenger service cost are evaluated in 15-minutes intervals, assuming that the train arrivals are evenly spaced within this period. However, there is no reason that this will be true, and it is likely that the output of the optimization will have some train bunchings. Therefore, applying an algorithm that spaces out train arrivals has the potential to improve the solution. A holding heuristic is applied to the solution to the train reformation problem to deal with this problem.
The fine-tuning routine holds trains according to the headways at Wood Green for westbound trips and Actor Town for eastbound trips. The method takes into account the crew constraints. As the algorithm is applied directly to the recovery timetable, it is not necessary to respect the occupancy constraints from the optimization phase. The occupancy is nonetheless taken into account. Trips are held in order to even the headways (and decrease the average waiting time). The optimal holding is $t_{n} - t_{p}$, where $t_{n}$ and $t_{p}$ are the departure times of the next and the previous train. The algorithm is run several times to resolve potential conflicts. These conflicts occur, since the algorithm is sequential while the optimal departure time of a train depends on the departure times of the previous and following trains, which are themselves modified by the holding module.

However, this time must be reduced if:

- The available time to reverse becomes too short. The constraint $t_{rev} \geq t_{min}^{rev}$, that makes reversing possible, must be respected at the origin and at the destination of the trip.

- The driver swap constraints are not respected (an additional 60-second margin has been allowed for the fine-tuning routine).

The fine-tuning module is illustrated through an example. To simplify, let us assume that the driver swap constraint must be respected, with no additional slack time. Table 5.6 gives the departure time of a trip before holding. $t_{max}$ is the maximum time this train can be held, and $t_{min}$ the maximum that this train can start early.

- At Cockfosters, because of the minimum reversing time, $t_{min} \leftarrow 3$min.

- At Arnos Grove, the new driver arrives no earlier than 16:54, but the train is scheduled to leave at 16:56. Therefore $t_{min} \leftarrow 2$min.
Table 5.6: Recovery Timetable before the Fine-tuning algorithm

<table>
<thead>
<tr>
<th>Station</th>
<th>Arrival Time</th>
<th>Departure Time</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFS</td>
<td>16:30</td>
<td>16:40</td>
<td>minimum reversing time of 7 minutes</td>
</tr>
<tr>
<td>AGR</td>
<td>16:55</td>
<td>16:56</td>
<td>operator swap:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- the relief driver arrives at 16:54</td>
</tr>
<tr>
<td>WGN</td>
<td>17:05</td>
<td>17:05</td>
<td>N/A</td>
</tr>
<tr>
<td>ACT</td>
<td>17:45</td>
<td>17:50</td>
<td>operator swap:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- the driver has his next task at 17:48</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- the relief driver arrives at 17:44</td>
</tr>
<tr>
<td>HRV</td>
<td>18:15</td>
<td>18:25</td>
<td>minimum reversing time of 6 minutes</td>
</tr>
</tbody>
</table>

- At Acton Town, the new driver arrives 6 minutes before the train must leave, which is more than the current value. Therefore, $t_{\text{min}}$ remains at 2 minutes. However, the relieved driver has his next task starting at 17:48, so $t_{\text{max}} \leftarrow 3\text{min}$.
- Finally, the reversing time at Heathrow does not limit $t_{\text{max}}$

Table 5.7 shows the new recovery timetable after one iteration under different scenarios. Under scenario A, $t_n = 17:09$ and $t_p = 17:03$. The train can be held for 1 minute and leave Wood Green at 17:07, because 1 minute is less than $t_{\text{max}}$. On the contrary, if $t_n = 17:07$ and $t_p = 16:57$, the ideal departure time at Wood Green would be 17:02. However, because of $t_{\text{min}}$, it cannot be earlier than 17:03, or the relief constraint at Acton Town would be violated. Note that in scenario B, the train would still be able to leave Arnos Grove one minute earlier, which would improve the headways. This minute difference would be fixed in the next iteration of the algorithm.

5.3.2 TSM

Transport For London uses a simulation tool called Train Service Model (TSM). TSM is used during the London Underground scheduling process, to test a new timetable. It assesses its feasibility and resiliency of the timetable in case of delays. It is also used to test modifications of the timetable due to planned work. Finally,
**Table 5.7: Recovery Timetable after one iteration**

<table>
<thead>
<tr>
<th>Station</th>
<th>Scenario A</th>
<th>Scenario B</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Arrival Time</td>
<td>Departure Time</td>
</tr>
<tr>
<td>CFS</td>
<td>16:30</td>
<td>16:41</td>
</tr>
<tr>
<td>AGR</td>
<td>16:56</td>
<td>16:57</td>
</tr>
<tr>
<td>WGN</td>
<td>17:05</td>
<td>17:06</td>
</tr>
<tr>
<td>ACT</td>
<td>17:46</td>
<td>17:51</td>
</tr>
<tr>
<td>HRV</td>
<td>18:16</td>
<td>18:25</td>
</tr>
<tr>
<td>Headway (after holding)</td>
<td>2 min</td>
<td>4 min</td>
</tr>
<tr>
<td>Headway (after holding)</td>
<td>3 min</td>
<td>3 min</td>
</tr>
</tbody>
</table>

it is part of the controllers training process, to help them become more familiar with the different constraints and strategies of service control. TSM is an event-based simulation model. It simulates the movement of trains from terminus to terminus. The simulated train service follows an input timetable. Passengers move through the system to reach their destination. They can transfer, be left-behind, or block the doors, which in turn impacts train operations. As a result, crowding may increase the dwell time and lead to further delays in the simulation.

TSM uses a number of input files. The demand file specifies the OD demand matrix in 15-minute intervals.

The infrastructure file is the most important one. It contains many parameters describing the line geometry. It lists the stations (each of which can contain several platforms) as well as the tracks linking the stations. The capacity of each track and platform, the inter-station runtimes, and the type of station (train depot, siding, etc.) are also specified. To model the line, the characteristics of the rolling stock must also be specified, and the different signal rules, such as “do not enter station A if link 1 is occupied” or the priority rules at junctions must be added. Finally, each station is translated to 3 or 4 letter codes that are used in the timetable file. One station can have several codes, but which platform is used under which circumstances would
change with the code. For instance, a train going westbound at Acton Town has to go through "ACTP". However, a train reversing at Acton Town east siding (ACTS) must stop at "ACT", because there are no links between the station ACTS and the platforms related to the code ACTP.

The last important file is the timetable file. Each physical train has several trips composed of several stops. For each stop, the arrival time and the name of the station are defined. How the trains behave vis-à-vis the timetable depends on the parameters entered in the infrastructure file, especially for the train priority at depots and reversing stations. It is not possible to specify directly in the timetable file that a train must be held 5 minutes at Acton Town. The holding strategy is a global parameter specified in the infrastructure file. To implement a specific holding action, the TSM user can adjust the running time to the next station to be longer than usual. Depending on the TSM parameters, this can be interpreted in two ways: the train may be considered early and be held at Acton Town, or TSM may allow the train to continue its journey despite its earliness.

TSM can be used to evaluate TRP recovery plans. However, the TRP recovery plan has to be translated to TSM input, with great detail and accuracy. For instance, the reversing times (time between to last stop of a trip and the first stop of the next one) must be higher than the station specific minimum reversing time. Finally, even if the timetable file is in the right format, it is likely that TSM ends up in gridlock: a small inaccuracy in the reintroduction or a wrong priority discipline at depots can result in a full line blockage.

For all these reasons, TSM was not used to evaluate the recovery plan. TSM is clearly not designed to study recovery plans, because it is too sensitive to the input. TSM is designed for operations planning, but cannot be used to assess recovery plans. In the context of this research, TSM was useful to identify the main infeasibilities
of the TRP solutions, and define the stations with capacity constraints that had to be enforced. However, the value of a simulation tool should not be underestimated. The integration of a simulation model (TSM or another one) with the TRP is a very promising future research direction.
Chapter 6

Application

In this chapter, the train reformation problem is applied under two different scenarios. The first example is an incident that happened on January 14, 2014. The second example demonstrates how, using a hypothetical incident, it is possible to use the train reformation problem to help decisions for both the incident and the recovery phase.

6.1 01/14/14 Incident

On January 14, 2014, at 5:35pm, the operator of train 340 heading Eastbound toward Cockfosters, reported smoke in the tunnel between Manor House and Turnpike Lane, at the North East end of the line (see Figure 6-1). The smoke was confirmed by the driver of the following train (252) at 5:37pm. The driver of the next train (train set #277) had to evacuate the train. The Manor House station supervisor boarded it and the train left Manor House without passengers at 5:40pm. The smoulder was confirmed by the station supervisor at 5:50pm. The electricity was turned off between Manor House and Wood Green at 5:52pm, for the fire to be extinguished. The power was reestablished at 6:03pm. The Piccadilly Line Duty Reliability Manager boarded the next train, which had also been evacuated at Manor House, to confirm that it was safe to resume service. He left Manor House at 6:05pm and confirmed that the tunnel was clear at 6:14pm. The management of the incident focused on safety: no
passengers were in the tunnel during the fire, and the power was switched off promptly. As a result, the fire incident was resolved as fast as possible. Yet, the incident lasted for 35 minutes and the control team reported a 39-minute delay. After investigation, it was determined that the cause of the fire was the accumulation of litter and paper which had come in contact with an electric cable.

Figure 6-1: Incident Location

6.1.1 Controllers Response

Figure 6-2 shows the time-space diagram of the train movements during the incident and the beginning of the recovery phase. h is the Heathrow branch of the Piccadilly Line. a shows the beginning of the incident, when train 340 first noticed the smoulder. b corresponds to train 277 with the station supervisor on it. d is train
341, from which the duty reliability manager allowed the service to resume. i corresponds to the end of the incident. During the incident phase, most trains are held at the stations. All trains shown with the arrows f were canceled either during the incident phase, or in the early stages of the recovery. In addition, two trains that were scheduled to begin their journey at Northfields at 6:00pm were also canceled. Train c was short-turned at King’s Cross and train e operated a shuttle service between Cockfosters and Arnos Grove, while most trains during the time period indicated by g were short-turned at Arnos Grove or Wood Green. Between the end of the incident and 9:30pm, 23 trains had been short-turned at Arnos Grove and 13 at Wood Green. In comparison, according to the timetable (shown in Figure 6-3), during this period, only 20 trains are reversed at Arnos Grove and none at Wood Green.

Figure 6-2: 14/01/14 Incident Phase

Figure 6-4 shows the controllers response during the incident and the recovery phases. Figure 6-5 compares the headway in the timetable and in the controllers re-
response for the trunk section, from trains in the Heathrow branch at Hatton Cross, and for trains leaving Cockfosters and Uxbridge. The disruption was updated to “minor” at 8:30pm and the service had mostly recovered by 10pm. Eastbound, the service had mainly recovered on the trunk section by 8:30pm. Headways were even, and most canceled train services were resumed between 7:30 and 9:00pm. However, a major deviation from normal service happened at the Uxbridge end of the line. To return to schedule on time faster and maintain the stock balance, controllers decided not to run most trains to Uxbridge, counting on passengers transferring to the Metropolitan Line at Rayners Lane. The Metropolitan and Piccadilly lines share the same tracks between these two stations. In the westbound direction, headways increased between 7 and 9pm (Figure 6-5a). However, this increase did not happen immediately after the end of the incident. The westbound direction did not experience any direct congestion because of the incident: rather, it suffered from knock-on effects. The reason lies within the trains which were canceled at Acton Town and Northfields at 6:00pm. Those trains would normally start their service in the Eastbound direction. They were canceled to relieve the congestion due to the delay caused by the incident. The
canceled journeys subsequently missed their westbound trips, and this led to longer headways.

It is possible to distinguish 4 phases in the controllers response:

1. 5:35-6:15pm: Incident phase and beginning of the recovery phase. Trains are held during the incident. Trains on the west part of the line are canceled to relieve congestion in the line.

2. 6:15-7:30pm: Trains are short-turned Eastbound at Arnos Grove and Wood Green and the service is reduced at Cockfosters (Figure 6-5b). The short-turnings are combined with reformations. Shuttle service is offered between Arnos Grove and Cockfosters.

3. 7:30-9:00pm: Most trains are reintroduced in the western portion of the line. Headways in the westbound direction are maintained even, and most trains are short-turned before reaching Uxbridge, with many cancellations on the Heathrow branch (Figure 6-5c). The service to Cockfosters is resumed.

4. 9:00-11:00pm: The demand is not as high. Controllers reintroduced the last trains to reestablish the train balance between the train depots. The service to Uxbridge is still mainly taken care of by the Metropolitan Line (the impact is visible on the right of Figure 6-5d).
Figure 6-4: Controllers Response Time-Space Diagram
Figure 6-5: Scheduled and Actual Headways
6.1.2 Results and Comparisons

The train reformation model was applied to this incident. The first step in applying the TRP is to establish the initial conditions at the beginning of the recovery phase. By 6:15pm, four trains were already canceled (two at Acton Town and two at Northfields). Train 274 was running the shuttle service between Arnos Grove and Cockfosters. These trains will be reintroduced during the recovery phase. During the incident, one train was short-turned at King Cross and became early. Other trains were running with delays ranging from 0 to 31 minutes. All eastbound trains before Manor House were late, but most westbound trains were still on time at the end of the incident.

With these initial conditions, the TRP model was run under several scenarios. For each scenario, we used a different set of parameters \((\alpha_c, \alpha_p, \alpha_d)\), which correspond to the weight coefficients associated with the complexity cost, the passenger service cost and the adherence delay cost respectively. An instance with \((\alpha_c = 0, \alpha_p = 1, \alpha_d = 0)\) means that the TRP optimizes the best service for passengers, regardless of the delay at the end of the recovery.

The optimization software used for the solution of the problem was Gurobi 6.0 (Gurobi Optimization, 2015). The TRP ran on a Windows 8.1 Pro 64-bit machine, with 8.00 GB of RAM, and an Intel Core i7-2630QM 2.00GHz processor. Gurobi estimates the lower bound of the objective function: if the relative gap between the lower bound and the current value of the objective function is less than 1%, the optimization terminates and the current solution is returned.

Table 6.1 lists the scenarios and their parameters. Most scenarios set the end time of control decisions \(t_{\text{end}}\) at 8:00pm, which implies a duration for the recovery phase of about 4 hours. They all enforce the occupancy constraints except for scenario K. The computation time necessary to generate the candidates \((t_{\text{gen}})\) grows expo-
nentially with the length of the control action period. The optimization time varied between scenarios, from 35 seconds to more than 8 minutes. In general, the CPU time decreases with fewer constraints (e.g. scenario K), fewer objective terms (C-D-E), and fewer candidates (I). The last column ($t_{opt}$) shows the CPU time needed to find a solution with a relative gap less than 10% (the CPU time for a solution within 1% is highly variable and not a reliable measure). In most scenarios, such solutions were found within 100 seconds. However, 8 minutes were needed in scenario J, where $t_{end} = 9:00$pm. For this scenario, no solution with a gap less than 4% was found even after 8 hours of computation.

Scenario A is the baseline. Scenario B uses a lower value for $\alpha_d$, which means the delay has less importance, while it is the opposite in scenario F. $\alpha_c$ is higher in scenario G, which means that renumbering and short-turning trains are more costly. The objective function of scenarios C, D and E includes only passenger cost, delay cost, and complexity cost respectively. Scenario I models a 3-hour recovery, while scenario J models a 5-hour recovery. Scenario K is useful to assess the influence of the occupancy constraints on the solution.

Table 6.1: Scenario parameters

<table>
<thead>
<tr>
<th>Scenario</th>
<th>$t_{end}$</th>
<th>$\alpha_p$</th>
<th>$\alpha_d$</th>
<th>$\alpha_c$</th>
<th>Occupancy Constraints</th>
<th>$t_{gen}$ (in s)</th>
<th>$t_{opt}$ (in s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>20:00</td>
<td>1</td>
<td>500</td>
<td>5</td>
<td>yes</td>
<td>141</td>
<td>104</td>
</tr>
<tr>
<td>B</td>
<td>20:00</td>
<td>1</td>
<td>500</td>
<td>5</td>
<td>yes</td>
<td>141</td>
<td>61</td>
</tr>
<tr>
<td>C</td>
<td>20:00</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>yes</td>
<td>141</td>
<td>71</td>
</tr>
<tr>
<td>D</td>
<td>20:00</td>
<td>0</td>
<td>5000</td>
<td>0</td>
<td>yes</td>
<td>141</td>
<td>45</td>
</tr>
<tr>
<td>E</td>
<td>20:00</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>yes</td>
<td>141</td>
<td>27</td>
</tr>
<tr>
<td>F</td>
<td>20:00</td>
<td>1</td>
<td>50000</td>
<td>5</td>
<td>yes</td>
<td>141</td>
<td>76</td>
</tr>
<tr>
<td>G</td>
<td>20:00</td>
<td>1</td>
<td>5000</td>
<td>50</td>
<td>yes</td>
<td>141</td>
<td>50</td>
</tr>
<tr>
<td>H</td>
<td>20:00</td>
<td>1</td>
<td>5000</td>
<td>0</td>
<td>yes</td>
<td>141</td>
<td>81</td>
</tr>
<tr>
<td>I</td>
<td>19:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>yes</td>
<td>37</td>
<td>18</td>
</tr>
<tr>
<td>J</td>
<td>21:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>yes</td>
<td>525</td>
<td>305</td>
</tr>
<tr>
<td>K</td>
<td>20:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>none</td>
<td>106</td>
<td>33</td>
</tr>
</tbody>
</table>

Table 6.2 shows the results from the train reformation problem. The number of
late journeys is estimated at the end of the recovery, which is two hours after \( t_{end} \).

AWT stands for Average Waiting Time. The AWT (OF) corresponds to the waiting time as estimated in the objective function. AWT (after fine-tuning) is the average waiting time after the fine-tuning module is applied, based on the departure times of the trains. For all scenarios, this waiting time accounts for passengers from 6:13pm to 9:30pm. Renumbered journeys correspond to the number of journeys that were reformed. Original Journeys are journeys that have been neither short-turned nor renumbered.

Note that the expected waiting time according to the timetable is slightly more than two minutes. The controllers response led to an average waiting time of 5 minutes. This estimate is based on a number of assumptions and may overestimate the actual waiting time. It is the waiting time that passengers would experience if they were to take only direct trains (without considering transfers). For instance, a shuttle service was operated between Cockfosters and Arnos Grove. In this case, passengers going to the rest of the line see their waiting time overestimated in the table, because their trip would involve a transfer. The estimated waiting time based on the TRP solutions in significantly lower than the controllers response and ranged from 200 to 230 seconds. There are a few exceptions, notably scenarios D and E, where the waiting time was 3.5 and 5.5 minutes respectively. However, it should be pointed out that waiting time was not part of the objective function in those scenarios.

Scenarios A to H were used to study the influence of the \((\alpha_c, \alpha_p, \alpha_d)\) parameters. We should mention that the adherence cost was almost constant for all scenarios, but was higher (10,000-fold increase) for scenarios C and E, which led to a significantly higher number of late trains. The reason is that the adherence cost is an exponential function of lateness. Therefore, the weight of the adherence cost does not significantly impact the solution, because its exponential term dominates \(\alpha_d\). For this reason, scenarios A,B, and F have similar results. However, setting this cost to 0 (scenarios
Table 6.2: Scenario results

<table>
<thead>
<tr>
<th>Scenario</th>
<th># of Late Journeys</th>
<th>AWT (OF, in s)</th>
<th>AWT (after fine-tuning, in s)</th>
<th>Original Journeys</th>
<th>Renumbered Journeys</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>7</td>
<td>232</td>
<td>228</td>
<td>20</td>
<td>48</td>
</tr>
<tr>
<td>B</td>
<td>9</td>
<td>214</td>
<td>206</td>
<td>21</td>
<td>43</td>
</tr>
<tr>
<td>C</td>
<td>26</td>
<td>194</td>
<td>207</td>
<td>1</td>
<td>65</td>
</tr>
<tr>
<td>D</td>
<td>8</td>
<td>436</td>
<td>338</td>
<td>2</td>
<td>62</td>
</tr>
<tr>
<td>E</td>
<td>25</td>
<td>308</td>
<td>226</td>
<td>52</td>
<td>72</td>
</tr>
<tr>
<td>F</td>
<td>6</td>
<td>237</td>
<td>214</td>
<td>19</td>
<td>49</td>
</tr>
<tr>
<td>G</td>
<td>7</td>
<td>236</td>
<td>219</td>
<td>25</td>
<td>39</td>
</tr>
<tr>
<td>H</td>
<td>7</td>
<td>232</td>
<td>233</td>
<td>5</td>
<td>61</td>
</tr>
<tr>
<td>I</td>
<td>17</td>
<td>244</td>
<td>297</td>
<td>22</td>
<td>45</td>
</tr>
<tr>
<td>J</td>
<td>3</td>
<td>230</td>
<td>225</td>
<td>13</td>
<td>53</td>
</tr>
<tr>
<td>K</td>
<td>3</td>
<td>211</td>
<td>190</td>
<td>25</td>
<td>43</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Controllers</th>
<th>Response</th>
<th>302</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timetable</td>
<td>0</td>
<td>210</td>
</tr>
</tbody>
</table>

C and E, where the delay component in the objective function is 0) has a dramatic effect on the delay at the end of the incident.

Scenarios H-A-G differ mainly in the importance of the complexity cost. The number of original journeys increases and renumbered journeys decreases with the ratio $\alpha_c/\alpha_p$ (Figure 6-6). The average waiting time remains stable.

The results from scenario I show that stopping control decisions at 7:00pm is premature: at the end of the recovery phase, 17 journeys are still late. The delay has not been recovered and the average waiting time, 297 seconds, is higher than in other scenarios. Using a longer recovery window (scenario J) is more realistic. The number of late journeys and the waiting times are reasonable. Moreover, for longer recovery scenarios, hard constraints (occupancy and crew) are applied until 11:00pm (compared with 10:00pm for scenarios with $t_{end} = 8:00pm$), which means that the recovery plan is likely to be feasible an hour longer. However, the number of candidates increases the complexity of the optimization problem significantly (which may compromise its use in real-time). Finally, disabling the occupancy constraints seems
to give reasonable results. However, the time-space diagram in Figure 6-7 shows that increased bunching takes place, compared to other solutions.

Figures 6-8 and 6-9 show for both directions the service recovery and the solution according to scenario G, which represents one of the best TRP solutions. Note that controllers reversed more trains at the beginning of the recovery at Wood Green, which led to better service westbound until 7:30pm. However, controllers tried to recover the delay more aggressively afterwards, which resulted in poorer service later in the evening. In the westbound direction (Figure 6-8), the main difference between the TRP and the controllers solutions is at the ends of the line. Controllers prefer to operate trains to Rayners Lane and let the Metropolitan line satisfy the demand beyond this point. Under the TRP solution, in scenario G, more trains go all the way to Uxbridge and Cockfosters. Finally, the average waiting time under scenario G is 27% lower than the controllers response. However, it should be pointed out that this difference is partly due to the decisions by the controllers to make passengers going to Cockfosters and Uxbridge transfer at Arnos Grove and Rayners Lane. Figure 6-10 compares train departures at key stations for the timetable, the controllers response and the scenario G solution. It shows clearly that controllers reduce the service at
the ends of the line (Arnos Grove and Uxbridge Eastbound), while the solution under scenario G uses more trains in these areas.
Figure 6-8: Time-space Diagrams, Westbound
Figure 6-9: Time-space Diagrams, Eastbound
Figure 6-10: Train departures at key stations
Table 6.3 shows the average waiting time per 30-minute interval during the recovery phase. Controllers manage to limit the waiting time until 7:30 pm with aggressive use of short-turning. However, these initial interventions introduce new constraints on the recovery: to get back on time they had to reduce the service provided at Uxbridge. Meanwhile, in scenario G, the TRP solution, which is generated by evaluating the impact of the interventions over the whole recovery period, the decisions taken at the beginning of the recovery do not compromise the possibility of providing a good service later. Figure 6-11 illustrates how scenario G provides better service at the ends of the line. In particular, 6-11b shows that the strategy of short-turning trains at Arnos Grove used by the controllers led to very good service on the trunk (ACT Eastbound and WGN Westbound lines) at the beginning of the recovery. However, this was not sustainable and the service deteriorated later. If needed, it is possible to increase the weight of the waiting time at the beginning of the recovery or in the trunk to obtain a solution with waiting in the trunk as low as in the controllers response.

Figure 6-12 shows the different train groups and the reformation cycles under scenario G. Journeys in bold correspond to the journeys that were not short-turned. The braces represent the train groups. Journeys in these braces are candidates to be combined in reformation. The figure displays the reformation patterns: the left number corresponds to the number of the train before reformation, and the arrow points to the new train number after reformation. Because of the uniqueness of the train numbers, reformation must form a cycle, so that every number appears once in the left and right columns. The reformation cycles are generally more complex in the TRP solutions than in controllers strategies. This also explains why more trains are able to go to Cockfosters. When complex reformation cycles are included in the solution, fewer short-turns were required to recover the service. As a result, more trains can go to Cockfosters, and more reversing points are used, which increases the capacity of the line and improves passenger service.
Figure 6-11: Waiting times at various locations as a function of time of day
Table 6.3: Waiting Time at key stations during the Recovery Phase

<table>
<thead>
<tr>
<th></th>
<th>18.5</th>
<th>19</th>
<th>19.5</th>
<th>20</th>
<th>20.5</th>
<th>21</th>
<th>21.5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Timetable</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACT Eastbound</td>
<td>85</td>
<td>100</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>104</td>
<td>106</td>
</tr>
<tr>
<td>CFS</td>
<td>119</td>
<td>109</td>
<td>124</td>
<td>161</td>
<td>173</td>
<td>155</td>
<td>136</td>
</tr>
<tr>
<td>RLN Eastbound</td>
<td>161</td>
<td>198</td>
<td>222</td>
<td>182</td>
<td>292</td>
<td>273</td>
<td>289</td>
</tr>
<tr>
<td>UXB</td>
<td>186</td>
<td>247</td>
<td>194</td>
<td>376</td>
<td>461</td>
<td>468</td>
<td>600</td>
</tr>
<tr>
<td>WGN Westbound</td>
<td>114</td>
<td>135</td>
<td>130</td>
<td>150</td>
<td>158</td>
<td>140</td>
<td>141</td>
</tr>
<tr>
<td><strong>Entire Line</strong></td>
<td><strong>112</strong></td>
<td><strong>130</strong></td>
<td><strong>120</strong></td>
<td><strong>131</strong></td>
<td><strong>133</strong></td>
<td><strong>134</strong></td>
<td><strong>135</strong></td>
</tr>
<tr>
<td><strong>Controllers Response</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACT Eastbound</td>
<td>277</td>
<td>136</td>
<td>83</td>
<td>148</td>
<td>158</td>
<td>139</td>
<td>189</td>
</tr>
<tr>
<td>CFS</td>
<td>677</td>
<td>3280</td>
<td>2855</td>
<td>3053</td>
<td>1028</td>
<td>264</td>
<td>399</td>
</tr>
<tr>
<td>RLN Eastbound</td>
<td>302</td>
<td>286</td>
<td>310</td>
<td>199</td>
<td>744</td>
<td>949</td>
<td>668</td>
</tr>
<tr>
<td>UXB</td>
<td>207</td>
<td>481</td>
<td>525</td>
<td>2460</td>
<td>2507</td>
<td>2503</td>
<td>1319</td>
</tr>
<tr>
<td>WGN Westbound</td>
<td>151</td>
<td>214</td>
<td>241</td>
<td>393</td>
<td>411</td>
<td>339</td>
<td>327</td>
</tr>
<tr>
<td><strong>Entire Line</strong></td>
<td><strong>216</strong></td>
<td><strong>299</strong></td>
<td><strong>290</strong></td>
<td><strong>425</strong></td>
<td><strong>335</strong></td>
<td><strong>277</strong></td>
<td><strong>255</strong></td>
</tr>
<tr>
<td><strong>Scenario G</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACT Eastbound</td>
<td>131</td>
<td>113</td>
<td>92</td>
<td>96</td>
<td>106</td>
<td>154</td>
<td>233</td>
</tr>
<tr>
<td>CFS</td>
<td>690</td>
<td>243</td>
<td>332</td>
<td>482</td>
<td>167</td>
<td>214</td>
<td>325</td>
</tr>
<tr>
<td>RLN Eastbound</td>
<td>260</td>
<td>431</td>
<td>453</td>
<td>992</td>
<td>1322</td>
<td>853</td>
<td>751</td>
</tr>
<tr>
<td>UXB</td>
<td>261</td>
<td>447</td>
<td>448</td>
<td>1189</td>
<td>1560</td>
<td>636</td>
<td>667</td>
</tr>
<tr>
<td>WGN Westbound</td>
<td>304</td>
<td>425</td>
<td>251</td>
<td>260</td>
<td>221</td>
<td>142</td>
<td>210</td>
</tr>
<tr>
<td><strong>Entire Line</strong></td>
<td><strong>238</strong></td>
<td><strong>267</strong></td>
<td><strong>184</strong></td>
<td><strong>222</strong></td>
<td><strong>191</strong></td>
<td><strong>177</strong></td>
<td><strong>236</strong></td>
</tr>
</tbody>
</table>
Figure 6-12: Scenario G Reformations
An interesting observation is that under scenario G, the Wood Green station is almost never used to reverse trains. Controllers used it to supplement the reversing capacity of Arnos Grove. As more trains go all the way to Cockfosters under scenario G, using Wood Green is no longer required. However, the limited use of Wood Green should be viewed positively. If controllers decide that the use of Arnos Grove is not feasible because of its aging signaling system, the solution can still be improved by reversing some trains at Wood Green instead. In general, all reversing points that are not modeled (e.g. Hammersmith, Hyde Park Corner or South Arrow) are possibilities for controllers to improve TRP recovery plans.

Finally, it is important to mention the different context of the two solutions. TRP makes all decisions at the end of the incident phase, by estimating what the position of the trains later in the recovery will be, taking into account the impact for the entire period. Controllers make decisions continuously, with information about current conditions. However, their decisions most likely have a limited ability to look ahead due to the complexity of considering all possible future scenarios. In this case study, unlike the controllers decision process, the TRP was solved only once, at the end of the incident. It is possible and recommended that a decision support system based on TRP be used in a rolling horizon model. The system is used several times during a disruption, using the updated positions of the trains and drivers. If the TRP is implemented so it can be used on a rolling horizon (for example, solved every 10 minutes), its potential may improve significantly.

### 6.2 Hypothetical Incident

The Piccadilly line incident application demonstrated that TRP can be useful to deal with the recovery phase of a disruption. This section discusses how the train reformation problem, designed in the first place for the recovery phase, can be used to support decisions during the incident phase, using a hypothetical incident.
As in the previous case study, we assume that the duration of the incident is known. We also assume that, at the end of the incident, the delay of every active (i.e. not canceled) train is known. The TRP model outputs a service plan for the recovery phase, and a list of trains that must be canceled during the incident phase, in order to be able to operate this recovery plan. In this application, the TRP does not provide a service plan in the incident phase. It only identifies trains to be canceled, without any details about when cancellations will take place. Other than the list of trains that must be canceled during the incident phase, there is no modification in the TRP model.

6.2.1 Incident Description

A hypothetical incident occurs during a weekday, just after the morning peak. At 10:00am, close to the Hammersmith station, a signal is malfunctioning. The west-bound direction is impacted first, while the eastbound signal fails 5 minutes later. The signals are repaired at 10:20am. As a result, the line suffers from a 20-minute delay westbound and a 15-minute delay eastbound.

During the incident, the controllers hold trains that are not canceled at stations. Figure 6-13 shows the delays of trains in each direction at the end of the incident, assuming no cancellations. The x-axis indicates the location of the trains. The labels above each point correspond to the train number. The delay is maximum at the incident location and decreases upstream.

This delay curve is realistic assuming a hold-at-stations incident phase strategy. A 20-minute delay is a relatively minor disruption, and controllers do not have time to short-turn trains in the trunk. However, the delay curves include all trains. The TRP has to be modified to include cancellations during the incident phase. A way to introduce cancellations is to keep the inequality (equation 4.10) in the problem formulation. This way, the solution does not require as many journeys in the solution.
as scheduled trains in the timetable. The optimization will select a set of journeys that will be on time and that provide a good service to passengers. The train journeys that are not included in the solution are considered as canceled in the incident phase. As it is easy to reintroduce a train, the recovery timetable may reintroduce some of them before the end of the recovery.

6.2.2 Results

The case study aims at showing that it is possible to use the TRP to support cancellations decisions during the incident phase. Table 6.4 shows the parameters of the different scenarios. The cancellation column specifies if cancellations are allowed. Occupancy constraints are critical in the beginning of the recovery. The incident causes congestion on the line, and the limited capacity causes the train cancellations. Assuming a minimum headway of 120 seconds, it is not possible to find a solution with no cancellations. Acton Town was the bottleneck on the line and could not accommodate all trains and reversing with this minimum headway.

To study how cancellations can improve the service, even when they are not strictly
necessary, we set the minimum headway to 90 seconds, and introduce the parameter \( \alpha_{\text{can}} \), which represents the cost of a cancellation in the objective function. In practice, setting a very high \( \alpha_{\text{can}} \) value is a way to find a solution with the minimal number of cancellations possible (scenario U). Scenarios S to N are used to study the impact of the length of the decision period on the solution.

Table 6.4: Scenario parameters

<table>
<thead>
<tr>
<th>Scenario</th>
<th>( t_{\text{end}} )</th>
<th>( \alpha_p )</th>
<th>( \alpha_d )</th>
<th>( \alpha_c )</th>
<th>Cancellations</th>
<th>( \alpha_{\text{can}} )</th>
<th>( t_{\text{gen}} ) (in s)</th>
<th>( t_{\text{opt}} ) (in s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>12:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>no</td>
<td>NA</td>
<td>179</td>
<td>28</td>
</tr>
<tr>
<td>Y</td>
<td>12:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>yes</td>
<td>0</td>
<td>170</td>
<td>19</td>
</tr>
<tr>
<td>X</td>
<td>12:00</td>
<td>1</td>
<td>5000</td>
<td>50</td>
<td>no</td>
<td>NA</td>
<td>180</td>
<td>26</td>
</tr>
<tr>
<td>W</td>
<td>12:00</td>
<td>1</td>
<td>5000</td>
<td>50</td>
<td>yes</td>
<td>0</td>
<td>173</td>
<td>15</td>
</tr>
<tr>
<td>V</td>
<td>12:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>yes</td>
<td>100</td>
<td>170</td>
<td>26</td>
</tr>
<tr>
<td>U</td>
<td>12:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>yes</td>
<td>1000</td>
<td>170</td>
<td>20</td>
</tr>
<tr>
<td>T</td>
<td>12:00</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>yes</td>
<td>0</td>
<td>170</td>
<td>44</td>
</tr>
<tr>
<td>S</td>
<td>11:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>no</td>
<td>NA</td>
<td>39</td>
<td>6</td>
</tr>
<tr>
<td>R</td>
<td>11:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>yes</td>
<td>100</td>
<td>39</td>
<td>6</td>
</tr>
<tr>
<td>Q</td>
<td>11:30</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>no</td>
<td>NA</td>
<td>80</td>
<td>13</td>
</tr>
<tr>
<td>P</td>
<td>11:30</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>yes</td>
<td>100</td>
<td>80</td>
<td>14</td>
</tr>
<tr>
<td>O</td>
<td>13:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>no</td>
<td>NA</td>
<td>830</td>
<td>76</td>
</tr>
<tr>
<td>N</td>
<td>13:00</td>
<td>1</td>
<td>5000</td>
<td>5</td>
<td>yes</td>
<td>100</td>
<td>823</td>
<td>80</td>
</tr>
</tbody>
</table>

Table 6.5 shows the results of the TRP. The last column is the number of trains that are canceled during the incident, which means that they are not included at all in the solution. The other columns are similar to Table 6.2. The TRP output is a list of train cancellations in the incident phase as well as a recovery plan. Scenario T finds the recovery plan that provides the best passenger service, subject to the crew and capacity constraints. It must be viewed as the lower bound on the waiting time. It is not necessarily the best recovery plan, because the delay is not included in the objective function: with only 4 trains canceled, but many remberbergings and a very high schedule adherence cost, the solution under scenario T operates a service leading to a 179-second average waiting time during the recovery phase. The difference of the AWT in the objective function between T and other scenarios is significant (at least 50 seconds). However, this difference disappears after the fine-tuning step.
Table 6.5: Scenario results

<table>
<thead>
<tr>
<th>Scenario</th>
<th># Late Journeys</th>
<th>AWT (OF)</th>
<th>AWT (after fine-tuning)</th>
<th>Original Journeys</th>
<th>Renumbered Journeys</th>
<th>Canceled Trains</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>4</td>
<td>278</td>
<td>192</td>
<td>20</td>
<td>51</td>
<td>0</td>
</tr>
<tr>
<td>Y</td>
<td>1</td>
<td>299</td>
<td>210</td>
<td>19</td>
<td>57</td>
<td>17</td>
</tr>
<tr>
<td>X</td>
<td>5</td>
<td>289</td>
<td>194</td>
<td>29</td>
<td>37</td>
<td>0</td>
</tr>
<tr>
<td>W</td>
<td>1</td>
<td>302</td>
<td>199</td>
<td>28</td>
<td>40</td>
<td>13</td>
</tr>
<tr>
<td>V</td>
<td>2</td>
<td>283</td>
<td>184</td>
<td>21</td>
<td>52</td>
<td>8</td>
</tr>
<tr>
<td>U</td>
<td>4</td>
<td>280</td>
<td>188</td>
<td>21</td>
<td>51</td>
<td>0</td>
</tr>
<tr>
<td>T</td>
<td>27</td>
<td>185</td>
<td>179</td>
<td>0</td>
<td>71</td>
<td>4</td>
</tr>
<tr>
<td>S</td>
<td>8</td>
<td>231</td>
<td>182</td>
<td>25</td>
<td>46</td>
<td>0</td>
</tr>
<tr>
<td>R</td>
<td>2</td>
<td>243</td>
<td>195</td>
<td>25</td>
<td>51</td>
<td>11</td>
</tr>
<tr>
<td>Q</td>
<td>5</td>
<td>300</td>
<td>194</td>
<td>21</td>
<td>51</td>
<td>0</td>
</tr>
<tr>
<td>P</td>
<td>0</td>
<td>304</td>
<td>188</td>
<td>21</td>
<td>53</td>
<td>10</td>
</tr>
<tr>
<td>O</td>
<td>4</td>
<td>228</td>
<td>183</td>
<td>24</td>
<td>47</td>
<td>0</td>
</tr>
<tr>
<td>N</td>
<td>2</td>
<td>238</td>
<td>181</td>
<td>24</td>
<td>47</td>
<td>8</td>
</tr>
<tr>
<td>Timetable</td>
<td>0</td>
<td>251</td>
<td><strong>135</strong></td>
<td>79</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

It is interesting to note that allowing cancellations does not necessarily lead to inferior passenger service compared to a scenario without cancellations. Allowing cancellations significantly reduces the value of the schedule adherence cost. The flexibility allowed by the cancellations enables shorter recoveries. When cancellations are allowed, the number of late journeys at the end of the recovery is less than 2. In addition, there are less reliefs that are late during the recovery phase.

Scenarios Y-V-U differ only in their value of $\alpha_{can}$. Scenario Y does not penalize cancellations, while the cost of a cancellation in scenario U is very high. In scenario Y, the high number of cancellations enables the recovery plan to have a very low delay cost. However, the number of running trains decreases: in scenario U, the solution cost is identical to that of solution Z. Figure 6-14 shows that there is no real trend in the passenger service component: when $\alpha_{can}$ increases the number of canceled trains drops, while the average waiting time stays fairly constant. However, canceling train allows less trains to be late at the end of the recovery. Canceling trains seems to result to shorter recovery durations.
Table 6.6 and Figure 6-15 show the waiting times under these three scenarios. At the end of the recovery, the average waiting time is longer when more trains are canceled, because when headways are evenly spaced, operating more trains decreases the waiting time (Figure 6-15a). Figure 6-15b shows that since the extra canceled were trains serving the Rayners Lane branch, their cancellations had a huge impact on this section of the line. This is also noticeable in the higher value of the waiting time at Uxbridge under scenario Y. Notice that a sudden bump in waiting time happens at 11:00am at Rayners Lane under scenario U, where no trains are canceled. Another sudden higher waiting time can be pointed out at 11:30am at Cockfosters. These increases in waiting times can be explained by the capacity constraints. The difficulty of respecting these constraints becomes the main driver in the optimization. The set of feasible solutions is reduced and peaks in waiting times may happen. On the other hand, when trains are canceled, the feasible space is larger, and these peaks do not appear.
Table 6.6: Waiting Time at key stations during the Recovery Phase

<table>
<thead>
<tr>
<th>Timetable</th>
<th>10.5</th>
<th>11</th>
<th>11.5</th>
<th>12</th>
<th>12.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT</td>
<td>86</td>
<td>85</td>
<td>88</td>
<td>91</td>
<td>87</td>
</tr>
<tr>
<td>CFS</td>
<td>127</td>
<td>125</td>
<td>125</td>
<td>102</td>
<td>109</td>
</tr>
<tr>
<td>RLN</td>
<td>247</td>
<td>292</td>
<td>344</td>
<td>387</td>
<td>287</td>
</tr>
<tr>
<td>UXB</td>
<td>690</td>
<td>570</td>
<td>564</td>
<td>601</td>
<td>453</td>
</tr>
<tr>
<td>WGN</td>
<td>96</td>
<td>110</td>
<td>124</td>
<td>138</td>
<td>134</td>
</tr>
<tr>
<td><strong>Entire Line</strong></td>
<td><strong>90</strong></td>
<td><strong>126</strong></td>
<td><strong>133</strong></td>
<td><strong>144</strong></td>
<td><strong>141</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario Y</th>
<th>10.5</th>
<th>11</th>
<th>11.5</th>
<th>12</th>
<th>12.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT</td>
<td>165</td>
<td>116</td>
<td>148</td>
<td>100</td>
<td>98</td>
</tr>
<tr>
<td>CFS</td>
<td>160</td>
<td>164</td>
<td>249</td>
<td>202</td>
<td>139</td>
</tr>
<tr>
<td>RLN</td>
<td>204</td>
<td>458</td>
<td>341</td>
<td>644</td>
<td>740</td>
</tr>
<tr>
<td>UXB</td>
<td>1736</td>
<td>1365</td>
<td>1083</td>
<td>1231</td>
<td>675</td>
</tr>
<tr>
<td>WGN</td>
<td>160</td>
<td>144</td>
<td>207</td>
<td>235</td>
<td>172</td>
</tr>
<tr>
<td><strong>Entire Line</strong></td>
<td><strong>126</strong></td>
<td><strong>192</strong></td>
<td><strong>215</strong></td>
<td><strong>220</strong></td>
<td><strong>217</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario U</th>
<th>10.5</th>
<th>11</th>
<th>11.5</th>
<th>12</th>
<th>12.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT</td>
<td>129</td>
<td>91</td>
<td>127</td>
<td>101</td>
<td>87</td>
</tr>
<tr>
<td>CFS</td>
<td>146</td>
<td>183</td>
<td>277</td>
<td>207</td>
<td>116</td>
</tr>
<tr>
<td>RLN</td>
<td>192</td>
<td>427</td>
<td>330</td>
<td>463</td>
<td>363</td>
</tr>
<tr>
<td>UXB</td>
<td>1134</td>
<td>1357</td>
<td>1245</td>
<td>909</td>
<td>389</td>
</tr>
<tr>
<td>WGN</td>
<td>170</td>
<td>143</td>
<td>154</td>
<td>209</td>
<td>150</td>
</tr>
<tr>
<td><strong>Entire Line</strong></td>
<td><strong>142</strong></td>
<td><strong>183</strong></td>
<td><strong>183</strong></td>
<td><strong>204</strong></td>
<td><strong>164</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario V</th>
<th>10.5</th>
<th>11</th>
<th>11.5</th>
<th>12</th>
<th>12.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT</td>
<td>121</td>
<td>101</td>
<td>89</td>
<td>87</td>
<td>75</td>
</tr>
<tr>
<td>CFS</td>
<td>137</td>
<td>160</td>
<td>339</td>
<td>181</td>
<td>166</td>
</tr>
<tr>
<td>RLN</td>
<td>194</td>
<td>740</td>
<td>361</td>
<td>454</td>
<td>274</td>
</tr>
<tr>
<td>UXB</td>
<td>827</td>
<td>458</td>
<td>1067</td>
<td>846</td>
<td>390</td>
</tr>
<tr>
<td>WGN</td>
<td>164</td>
<td>156</td>
<td>147</td>
<td>213</td>
<td>150</td>
</tr>
<tr>
<td><strong>Entire Line</strong></td>
<td><strong>145</strong></td>
<td><strong>208</strong></td>
<td><strong>181</strong></td>
<td><strong>198</strong></td>
<td><strong>164</strong></td>
</tr>
</tbody>
</table>

Scenarios S to P demonstrate that canceling trains becomes more valuable when a shorter recovery period is targeted. When \( t_{end} \) is at 11:00am, 11 cancellations are needed to reduce the delay cost. This number is lower in scenario N where \( t_{end} \) is at 13:00am. It is no longer necessary to cancel as many trains, because there are more opportunities to short-turn trains to get back on schedule. This result supports the conclusion that canceling during the incident phase is more effective than reforming.
Figure 6-15: Waiting times at various locations

and short-turning later to achieve a shorter recovery.

Figure 6-16 illustrates the time-space diagrams of scenario Z, which does not allow cancellations, while 8 trains are canceled in scenario V. Note that the AWT is similar in both cases (192 seconds versus 184 seconds), despite the lower number of trains in scenario V. This can be explained by less congestion in scenario V, leading to more even headways. On the contrary, recovery without cancellations leads to bunching,
which eliminates the advantage of running more trains.

This hypothetical scenario shows that it is possible to relax the number of trains to allow cancellations. These cancellations are important to get back on schedule faster. They relieve the congestion without too heavily impacting the average waiting time. Parameter $\alpha_{can}$ allows the identification of the cancellation strategy: with a high value, few trains are canceled, and these are the ones that should be canceled first. In this 20-minute delay incident, scenario V showed that canceling 8 trains (which is the number that controllers would choose, according to the 2-train cancellation per 5-minute delay rule of thumb they use) seems to be a good choice. Although these results are only indicative, they seem to confirm the idea that short recoveries (such as in scenarios S to P) need more cancellations that longer ones.

This example shows that TRP can be useful for the incident phase, using a simplified delay model. This implementation is consistent with the framework described in chapter 4 and 5 and assumes a known duration for the incident. The modified TRP provides not only the "optimal" recovery plan given a set of train positions, but help find an overall optimal strategy for the entire disruption.
Figure 6-16: TRP solutions recovery plans
Chapter 7

Conclusion

This first section of this chapter summarizes the research done in this thesis. The second section lists recommendations for how London Underground can use the findings from this work. Section 7.3 discusses the main limitations of the methodology that would prevent it from being fully implementable. Section 7.4 suggests topics for future research, in order to improve the solution of the train reformation problem, and develop and implement a decision support system.

7.1 Research Summary

Managing disruptions is a major challenge when operating high-frequency transit rail lines. In this environment, the high number of trains, the limited capacity of the tracks and the signaling system, and the constraints imposed by the crew schedule all make the recovery from service disruptions complex. Chapter 2 introduced the disruption management problem on high-frequency lines, emphasizing the importance of headways and the near capacity nature of the operating environment. The chapter introduced two main phases in disruption management: the incident phase, which lasts until the incident is resolved, and the recovery phase, that starts at the end of the incident until the service totally recovers. It detailed the different objectives and constraints that must be taken into account during disruption management, as well as the intermediate goals that make service control more manageable for controllers.
Finally, it described the main interventions that controllers use during a disruption to mitigate its impact.

Chapter 3 illustrated the major impacts disruptions have on the Piccadilly Line. The Piccadilly Line has a very high demand, even during week-ends. Most of the line operates on a single track in each direction. The combination of high demand and limited infrastructure with an aging signaling system restricts the capacity of the line even more and leads to limited resiliency in case of disruptions.

Rescheduling and disruption management are topics that have been addressed in the literature. However, few rescheduling methods deal with high-frequency transit rail lines. In particular, most of the work fails to capture the strong influence of crew constraints on disruption management. Furthermore, the specificity of a human control environment is not incorporated in most of the literature: most of the models do not take into account the controllers decision process.

Chapter 4 introduced the approach for the solution of the train reformation problem (TRP). The TRP is formulated as an optimization problem and is designed for the recovery phase of a service disruption. Given the demand, the characteristics of the line, and the location of the trains and drivers (as well as their schedules) at the end of the incident phase, the train reformation problem finds an “optimal” recovery plan that accounts for the passenger service, the effectiveness of the recovery, and the complexity of the plan. It explicitly incorporates the scarcity of resources and associated constraints, especially with respect to crews. The method consists of two phases: the first generates a large number of candidate journeys. The second phase adopts an optimization approach to select the journeys to be included in the recovery plan. While controllers make multiple decisions at any time during a disruption, typically based on short term considerations, the proposed solution to the TRP formulates a recovery plan, taking into account the impact of control interventions for the entire
recovery period.

Chapter 5 presented in more detail the different methods and algorithms used in the application of the train reformation problem to the Piccadilly Line.

In chapter 6, the train reformation problem was applied to an incident that happened in January 2014. The sensitivity of the solution with respect to the main parameters in the objective function (passenger service, schedule adherence, and complexity) was investigated. The solutions to the TRP under various scenarios showed the potential of the proposed approach to reduce the average waiting time during the recovery phase, compared with the controllers actual response (to the extent it was reproduced by the available data sources). The average waiting time of the most promising TRP solution was 27% lower. In particular, the solutions identified by the proposed method tend to provide more service at the end of the line, while controllers favored short-turning most trains at Arnos Grove and relying on a shuttle service beyond. The results also demonstrated that aiming at a very short recovery was impossible, as too many trains would remain late at the end of it.

The TRP methodology was also tested with a hypothetical incident. Assuming a known duration of the incident and the delay of every train at the end of it, the model was modified to allow train cancellations during the incident phase. As a result, the TRP provides a service plan for the recovery phase, as well as a list of trains that should be canceled in the incident phase in order to be able to achieve this plan. The results provided a better understanding of how cancellations can support a better recovery. In the example, allowing the cancellation of a few trains was sufficient to eliminate almost all delays at the end of the recovery phase (even for the shorter recoveries). Passenger waiting time did not show any significant improvement when no cancellations were allowed (but the train delays at the end of the recovery were higher).
7.2 Recommendations

The train reformation model has shown promising results. It demonstrated that it is possible to use a model to provide controllers with a recovery plan which mitigates impacts from disruptions in terms of passenger service and recovery duration. The results from the application of the model point to a number of initial recommendations related to the current practices for disruption management, although further detailed analysis is required to verify some of the findings.

**Frequency of short-turns at the end of the line.** The service provided at the ends of the line in the recovery phase is different in the controllers response from the TRP solutions. Controllers tend to short-turn many trains at Rayners Lane and Arnos Grove, while the TRP solutions provides more service at the end of the line. Controllers short-turn trains at these stations to recover delay, with a minimum impact on passengers, because the demand at the ends is low. The TRP case studies suggest that running trains to the end of the line is useful. This may be because TRP solutions manage to recover service more efficiently than controllers through more complicated reformation patterns with fewer short-turns. Having trains that go all the way to Cockfosters, as opposed to turning them short at Arnos Grove, means that more reversing points are used, which increases the line capacity, and may improve passenger service. However, it is important to mention again an important difference between the controllers actions and TRP. The controllers’ decisions were made in real time with only current information available. TRP solves the problem at the end of the incident phase only. Hence, some of the later decisions may not reflect what actually happened.

**Flexibility brought by canceling trains.** The case study with the hypothetical
incident suggested that canceling a few trains had no real impact on passenger service: the lower frequency is matched by the gain in performance of other trains in a less congested line, and by the shorter recovery enabled by more flexibility. Therefore, the number of trains in service should not be used as the main indicator to evaluate the efficiency control strategies.

**Crew schedule data and information system.** Crew constraints are at the center of the TRP formulation. They restrict control possibilities, and they are the main driver of the journey generation step. In the Piccadilly Line, crew and train controls are separated. Managing the drivers and modifying their schedule in case of disruptions is the responsibility of the DTSMs. Collaboration between the service controllers and the DTSMs is important and requires an effective information system. Currently, the location of drivers is not known in real-time. Drivers register their identification when they start driving a train, but this information is not communicated to the control side. Two types of information are critical for an effective decision support system. The location of drivers in real-time would be useful to DTSMs so they know the resources that are available. Currently, it is hard for a DTSM to know where drivers are after they leave the depot. Controllers should know the time left before a driver hits a work time parameter, as well as their location, so they can make more strategic decisions. This information is also needed by any decision support system.

### 7.3 Limitations

This thesis proposed a model to support controllers in decision making during disruptions. The proposed approach has a number of limitations.

- The TRP is currently restricted to the recovery phase. Chapter 3 proposed ways to extend the model to the incident phase, by including the uncertainty in the process, especially with respect of the duration of the incident. There
may also be other sources of uncertainty, such as dwell times at stations, late driver arrivals, etc., which were not explored. Furthermore, the model does not consider the possibility of new incidents happening during the recovery phase.

- The quality of the recovery plan is tied to the list of candidates generated during the first phase of the TRP approach. The approach mainly considers short-turns and reformations in this process, but one can argue that modeling spare drivers, extensions, etc. may be important as well, and further research is needed to identify the most important strategies to be included.

- The performance model used to generate the recovery timetable does not capture the effect of interactions between trains and journeys explicitly. As a result, the solution to the TRP is a simplification of an actual timetable: interactions between trains, even at the level of an intersection, if not modeled properly, can lead to a full blockage of the line.

- The above limitation resonates with the last recommendation of the previous section. A decision support system will not be adopted by controllers, unless the proposed solutions are feasible, implementable, and practical. Controllers already deal with a complex flow of information. The TRP aims to model the most common interventions used by controllers, but is nonetheless a simplified model of the line and of control interventions. Its simplified representation of the system may compromise its use by controllers who need to decide in real time on plan, with a low risk of infeasibility.

7.4 Future Research

The work presented in this thesis is an important step toward an efficient real-time decision support system. Several potential topics can be studied further to improve the model, and disruption management at large. Interesting future research can be grouped into three categories: model extensions, algorithm improvements, and practical aspects and implementation.
Model Extensions

Candidate generation. Short-turns and reformations are important and common interventions, but they do not necessarily represent the best possible decisions. Other types of interventions, such as extensions, multiple renumberings, or shuttle services, should be also considered, as they may improve the performance of the system. Moreover, generating only candidates that have serious potential to be included in the solution will improve the computational efficiency of the algorithm.

Integrated framework for the incident and recovery phases. Adapting the performance model to the incident phase is important. Including the specific characteristics of the incident phase, in terms of the variability and the impact of uncertainty on the service and the controllers decisions, requires a different formulation of the problem. In particular, looking at the propagation of an incident and of the knock-on effect is an interesting topic which has not previously been studied. The development of a performance model that includes the interactions between trains, which are more severe during the incident phase, is also necessary, and would be useful for both phases.

Algorithmic Improvements

Column generation formulation for the optimization. An interesting area of research is to change the sequential approach of the current solution of the train reformation problem. Instead of generating journeys prior to the selection, a column generation approach, in which candidates are generated as needed within the optimization algorithm, represents an interesting future research direction. Since a relatively small number of better candidates is expected to be generated, this approach can significantly reduce the computational time.
Practical aspects and implementation

Train reformation model implemented in a simulated environment. The TRP model has been tested in this research using various simplifications. The natural next step toward its implementation in a control room is to test such a tool in a simulated environment. In this context, controllers, with and without the assistance of a TRP-based tool, would try to mitigate the impact of an incident on a simulated high-frequency line.

Train reformation model as an evaluation tool. In addition to using the solution of the TRP for recommendations in real time, it is possible to use it as an evaluation tool. Comparing the performance metrics of the TRP solution with the actual response of controllers, based on the same initial conditions, can be used after the disruption to evaluate the response. In particular, it becomes possible to identify certain types of incidents that are generally difficult for controllers to manage. This information can help identify the conditions under which a solution based on the TRP would be more useful.

Model design based on human factors. Even if a decision support tool provides a good recovery plan and recommendation, there is no guarantee that such a plan would be adopted by controllers for various reasons. For instance, decisions included in the model must be close to what they do in order to increase the confidence that controllers have in the model. Therefore, human factors aspects of the design of the system are important and can influence the acceptability of the system by the controllers. Moreover, it is important to think about how a decision support tool could be used in an environment such as the Piccadilly line control centre, as controllers may be unable to use directly a recovery plan. Their workload must be taken into account: the question of the interface between the decision support system and controllers is important. For example, the decision support system may be designed to provide an ordered cancellation list, suggestions of promising reformation patterns, or warnings when delays threat to violate driver constraints.
Bibliography


