Model for Police Dispatching and Shift Scheduling

by

Vishal Vijayvargiya

B.Tech, National Institute of Technology Karnataka, 2013

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
School of Computing Science
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY
Fall 2017

Approval

Name:	**Vishal Vijayvargiya**
Degree:	**Master of Science (Computing Science)**
Title:	**Model for Police Dispatching and Shift Scheduling**
Examining Committee:	**Chair:** Faraz Hach Research Associate
	Binay Bhattacharya Senior Supervisor Professor
	Abraham Punnen Supervisor Professor
	Ramesh Krishnamurti Internal Examiner Professor School of Computing Science
Date Defended:	**December 8, 2017**

Abstract

Emergency services have been in existence since the beginning of recorded history, yet most efficient and effective use of resources in this field is still considered an open problem. In this thesis, we explore the challenges involved in police services, and present an approach for determining police force capacity for a given call for service data. We use mathematical programming for modeling police dispatching and shift scheduling, and test our method on real occurrence data.

Keywords: Police Dispatching; Shift Scheduling; Linear Programming

Acknowledgements

I am grateful to my supervisor, Dr. Binay Bhattacharya, for all the guidance and support he has provided throughout my studies at SFU. I would especially thank him for being patient with me, and motivating me at every step of my work. I owe great thanks to Vladyslav Sokol for all the advice and valuable insights regarding the problem formulation. I would also like to thank Patricia Brantingham and Caroline Brear for their time and help during the initial discussions. I have been lucky to have amazing friends and colleagues, who made this journey fun and enjoyable.

Contents

Approval ii

Abstract iii

Acknowledgements iv

Table of Contents v

List of Tables vii

List of Figures viii

1 Introduction 1
- 1.1 Background . . . 1
- 1.2 Existing Work . . . 1
- 1.3 Our Approach . . . 3
- 1.4 Thesis Overview . . . 4

2 Dataset 5
- 2.1 About . . . 5
- 2.2 Data Fields . . . 5
  - 2.2.1 Call Priority . . . 7
  - 2.2.2 Time Intervals . . . 8
- 2.3 Data Analysis . . . 9
- 2.4 Preparing Data for Experimentation . . . 15
  - 2.4.1 Data Filtering . . . 15

3 Police Dispatching 16
- 3.1 Preliminary . . . 16
- 3.2 Police Dispatching Problem . . . 16
- 3.3 Model Construction . . . 17
- 3.4 Big-M Method . . . 18
  - 3.4.1 Notations . . . 18

3.4.2 Variables . . . 19
3.4.3 Modeling Problem as Network Flow . . . 19
3.4.4 Mathematical Formulation . . . 19
3.4.5 Analysis . . . 21
3.5 Time Discretized Method . . . 21
3.5.1 Notations . . . 22
3.5.2 Variables . . . 22
3.5.3 Modeling Problem as Network Flow . . . 22
3.5.4 Mathematical Formulation . . . 24
3.6 Estimating Lower Bounds . . . 26
3.7 Implementation and Experimental Results . . . 26
3.8 Conclusion . . . 30

4 Shift Scheduling **31**
4.1 Background . . . 31
4.2 Demand Table for Shift Scheduling . . . 32
4.3 Model for Shift Scheduling . . . 32
4.3.1 Notations and Variables . . . 33
4.3.2 Mathematical Formulation . . . 33
4.4 Finding Number of Police Officers . . . 34
4.5 Implementation and Experimental Results . . . 34
4.6 Conclusion . . . 39

5 Shift Based Dispatching **40**
5.1 Objective Function for Model . . . 40
5.2 Mathematical Formulation . . . 40
5.3 Experiment and Results . . . 41
5.4 Conclusion . . . 45

6 Conclusion **46**

Bibliography **47**

List of Tables

Table 2.1 Fields in occurrence data 7

Table 3.1 Estimated lower bounds on officer count for each day of 1st week of September 2014 (based on 1 hour timeout limit) 27
Table 3.2 Estimated lower bounds on officer count for each day of 1st week of April 2014 (based on 1 hour timeout limit) 27
Table 3.3 Response time output for different count of police officers for Sept 6th 2014 data 27

Table 4.1 Estimated hourly demand of police officers during the 1st week of September 2014 35
Table 4.2 The number of police officers required in each 10-hour shift type for demand in table 4.1. First column is the type of shift, which is 10-hour shift. Second column is day of week, with 0 for Monday, 1 for Tuesday and so on. Third column gives start hour with value between 0-23. Fourth column is the output of the model, i.e. count of the police officers. 36
Table 4.3 The number of police officers required in each 8-hour shift type for demand in table 4.1. First column is the type of shift, which is 8-hour shift. Second column is day of week, with 0 for Monday, 1 for Tuesday and so on. Third column gives start hour with value between 0-23. Fourth column is the output of the model, i.e. count of the police officers. 37

Table 5.1 Results for 1st week of Sept 2014 with 10-hour shifts: showing total officer hours each day and total of overtime (in minutes) from all police officers 42
Table 5.2 Results for 1st week of Sept 2014 with 8hr shifts: showing total officer hours each day and total of overtime (in minutes) from all police officers 44
Table 5.3 Results on 1st week of Sept 2015 using schedule generated for Sept 2014 1st week. 45

List of Figures

Figure 2.1 Time mileposts and time intervals in police emergency response system 8

Figure 2.2 Plot showing count of Occurrence Number for each final priority for the year 2014. 10

Figure 2.3 Plot showing location details of priority 1 to 4 occurrences for the year 2014. 10

Figure 2.4 The trend of count of Occurrence Number for each month broken down by final priority for the year 2014. The marks are labeled by count of Occurrence Number. 11

Figure 2.5 The trend of count of Occurrence Number for first seven days of September broken down by final priority. 11

Figure 2.6 Plot showing percentage of priority 1 to 4 calls reported during each hour of a day for the year 2014. 12

Figure 2.7 The trends of average dispatch delay, travel time and response time (in minutes) for each month in the year 2014. The data is filtered on final priority, which ranges from 1 to 4. 13

Figure 2.8 The trend of count of Occurrence Number for each quarter of year 2014 and 2015. The data is filtered on final priority, which ranges from 1 to 4. 14

Figure 2.9 The trend of count of Occurrence Number for month of September in year 2014 and 2015. 14

Figure 3.1 A simple three job (in blue) network with two jobs having two copy nodes each (in green). Time horizon is of 24 hours, i.e., 1440 minutes 24

Figure 3.2 Response time statistics for September 6th 2014 with 30 police officers 28

Figure 3.3 Job count of police officers for September 6th 2014 with 30 police officers 28

Figure 3.4 Job schedule for each police officer for September 6th 2014 on time horizon of 1440 minutes (24 hours). Blue section marks the time duration when the police officer was attending a job. 29

Figure 4.1 Line graph showing officer demand and actual availability based on 10-hour shift with start times 12AM, 8AM, and 4PM 38

Figure 4.2 Line graph showing officer demand and actual availability based on 8-hour shift with start times 12AM, 8AM, and 4PM 38

Figure 5.1 Count of police officers on duty on each day based on 10-hour shift schedule 43

Figure 5.2 Response time analysis based on results of shift based dispatching for Sept 6th 2014 43

Figure 5.3 8hr and 10hr shift comparison based on total officer hours allocated for each day 44

Chapter 1

Introduction

1.1 Background

The police department in a city plays a crucial role in maintaining law and order by preventing crimes and responding to incidents. The way a policing agency operates in a city can have a profound effect on public safety and quality of life. Government and emergency services have a challenging task to devise techniques to utilize their limited resources effectively to live up to the public expectations. Doing so involves making decisions regarding various aspects of their operations. In policing, some of those elements include the number of patrol units, their geographic locations, dispatching policy, workforce scheduling, etc. Study of various police reports[13, 20] suggests that even today planners and administrators struggle to establish a reliable system to achieve policing efficiency and effectiveness in all situations. For example, the ever-changing number of calls for service with time and location affects the objective of attaining a desirable response time. This uncertainty then impacts police officer's availability for specific problem tasks or any other proactive work, as a lot of time is spent on responding to 9-1-1 calls. A significant amount of time and effort has been devoted in the last few decades to incorporate the advancement in technologies and research to enhance policing service model. In our work, we use the concepts of mathematics, operations research and computing science to address some of the issues in policing service.

1.2 Existing Work

Significant research in the study of police patrol allocations began in the early 1970s, when New York City Rand Institute analyzed emergency services and developed mathematical models for various aspects of operations. Larson[16] developed a hypercube queuing model which was a major advancement for police resource allocations across geographical regions at that time. The model aims to expand the state description of a queuing system with multiple servers to incorporate more complex dispatch policies by taking into account geographical and temporal complexities of the region. The hypercube in itself is an n-dimensional binary space for n servers with vertices representing a state space, which in this case would be the state of the cars. For example, if we use 1 for *busy* and 0 for *free*, a (100) state for three car system would mean car 1 and 2 being free and car 3 as busy. Once system configuration has been established, hypercube model could provide details about optimal units to dispatch at any particular state, and also calculate various performance measures, like, region-wide mean travel time, workload imbalance and inter-district dispatches. Application of this model in urban service systems have been further studied, for instance, Takeda et al.[4] used them for ambulance decentralization in urban areas and Chelst KR et al.[11] for police patrol deployments.

Traditional methods to determine the number of police officers on duty in a particular area at a specific time involved using quantitative methods of *hazard* or *workload* formulas. These techniques focus on weighing various factors which are significant for workforce allocation and generate a hazard score for each geographical area. Values of this hazard score determine the relative distribution of available police workforce among precincts in a city. For example, there are total N areas in a city, and the police department has developed a set of M relevant factors, then hazard score H_n for an area n is given by

$$H_n = w_1 \frac{f_{n1}}{F_1} + w_2 \frac{f_{n2}}{F_2} + \ldots + w_M \frac{f_{nM}}{F_M}$$
$$\text{where} \qquad F_m = f_{1m} + f_{2m} + \ldots + f_{Nm}$$

here f_{nm} is the amount of factor m in area n, and w_m is the weightage given to factor m. A value of say .10 for H_n would indicate that 10% of city police force be allocated to area n. If the factors or weights capture the police workload involved rather than the crime hazard itself, then they are regarded as workload formulas. Chaiken[10] in his report discusses how these formulas fail to incorporate other critical performance measures like delay in the call for service and total response times while balancing patrol allocation among areas. Another interesting scenario was mentioned by Larson[17], according to which using hazard formula could suggest the need for additional police personnel in the regions which were already overallocated, as more policing in an area could increase the number of arrests, and thus more crime being recorded. Hazard score for such an area would indicate assigning extra police officers in that area, and therefore stretching overworked precincts.

In modern times, much of the staffing and scheduling decisions are backed by software systems and simulators. Some of the applications which were developed include Staff Wizard, Police Resource Optimizing System (PROS), and Managing Patrol Performance (MPP). Such software often takes as input number of calls for service per hour, average response times, priority level among calls, unit deployed in an area, and other specific pieces of information. Vancouver Police Department's patrol deployment study of year 2007[13] talks about different approaches, and their advantages and disadvantages. They mention about Police Allocation Manual (PAM) approach which uses mathematical and logical relationships between workload data to approximate staffing needs. MPP, on the other hand, relies on queuing theory, probabilistic reasoning, and other mathematical models, to provide forecasts, predict staffing demands and run simulations. These tools offer a lot of benefits compared to manual planning. First, they enable planners to test different scenarios or any changes in allocation and calculate performance statistics in a reasonable amount of time. Second, because of their data-driven techniques, recommendations are in-line with realities and could easily be visualized. Though these applications have brought a definite upgrade in the industry, they are not self-sufficient to handle each aspect of policing operations. They have the disadvantage of not being flexible and transparent. Often human factor, regulations, and unforeseeable circumstances require manual intervention at every stage of process. Therefore carefully designed mathematical models combined with human reasoning could help to overcome the majority of complications in police agencies.

1.3 Our Approach

In our approach we concentrate on developing mathematical models for police patrol allocation strategies, specifically focusing on dispatching and shift scheduling. Our ultimate objective is to determine the size of the smallest police workforce along with their shift schedules to serve all the job requests in a region. Our work could be divided into three major stages:

First, we present a model for police dispatching for a known call for services with a fixed number of police officers on duty in a day. In this model we consider police officers have a 24-hour shift, i.e., they are available for any job during a day. We show how this model could help us in estimating lower bound on the number of police officers required to meet demand with a significant improvement in response time.

Second, using the estimates for the hourly requirement of police officers in the first stage, we design a shift schedule. We calculate the number of police officers required in each shift and thus obtain total number of employees needed.

Third, we show that the shifts generated from the second stage could fulfill the daily call for service requests with minimal overtime from fewer police officers.

1.4 Thesis Overview

This thesis is organized as follows:

In chapter 2, we discuss Waterloo Regional Police Service (WRPS) Occurrence Data, which is the dataset used for our experiments. We talk about various fields in the data and present statistics with the help of graphs.

In chapter 3, we focus on police dispatching and its intricacies. We elaborate on our model and show a mathematical programming formulation for the problem. Furthermore, we show how we can estimate police officer demand and analyze the output obtained from our model on various program parameters.

In chapter 4, we provide background about shift scheduling and give details about rod-schedule method[19].

Chapter 5 will contain specifics about the final stage of our whole approach where we use the shifts from the previous step for police dispatching.

Chapter 2

Dataset

2.1 About

For our studies, we are using publicly available Waterloo Regional Police Service (WRPS) Occurrence Data, which provides detailed statistics about police call information. WRPS aggregates this data from two systems, Computer Aided Dispatched (CAD) System and Records Management System (RMS).

Information gathered from data source description[1, 3] mentions: “WRPS uses the Intergraph CAD software to start an occurrence, dispatch officers, and maintain the status of all logged-in units. The creation of any new occurrence number in CAD is considered a ‘CAD event’ regardless of the source. CAD events may be generated from calls coming into the Communications Centre from a non-emergency line, from a 9-1-1 phone line, or initiated by an officer. A call may be canceled, duplicated from multiple people reporting the same incident, taken over the phone by resource desk, of a nature requiring a police report to be prepared or concluded with no report being necessary.”. Niche RMS is the records management system used by WRPS, which stores information regarding occurrences once they are marked closed.

WRPS releases data combined from these two systems in a CSV and XLS file which includes one full year of occurrence data. It should be noted that WRPS puts great effort in aggregating this data and making it available for the community to analyze.

2.2 Data Fields

Table 2.1 summarizes various fields present in occurrence data[2].

Name	Summary
Occurrence Number	Unique identifier for incident.
Geographic Location	The latitude and longitude detail for adjusted location of incident, generally nearest intersection.
Nearest Intersection Location	Street names for adjusted location information to nearest intersection.
Patrol Division	Patrol division based on address. The Region of Waterloo is divided into four patrol divisions for optimum police service delivery: WN=North; WC=Central; WS=South; WR=Rural.
Patrol Zone	Patrol zone based on address. Each patrol division is divided in six zones: : North zones=WN1-WN6; Central zones=WC1-WC6; South zones=WS1-WS6; Rural zones=WR1-WR6.
Municipality	The city or township based on the address of the occurrence
Reported Date and Time	Date and time when the event was received in the WRPS Communications Centre and an occurrence number was generated.
Initial Call Type	The 9000 code of the main call type assigned to the occurrence when it is initially created.
Initial Call Type Description	Description of initial call type.
Final Call Type	The 9000 code of the final call type assigned to the occurrence when it is closed.
Final Call Type Description	Description of final call type
Initial Priority	The first priority attached to the call
Final Priority	The final priority attached to the call
Disposition	Each occurrence is given a disposition representing the general outcome of the event. It may be canceled (CAN), duplicated (DUP), taken over the phone by resource desk (DPR), serious enough for a police report to be prepared (RTF), or unfounded with no report being necessary (NR).
Dispatch Date and Time	Captures the date and time of the first unit dispatched.

Name	Summary
Arrival Date and Time	The date and time that a dispatched unit first arrived on scene to the address of the occurrence.
Cleared Date and Time	The date and time that a dispatched unit last cleared from the address of the occurrence.
Call Dispatch Delay	Calculated time interval (in seconds) between Reported Date and Time, and Dispatched Date and Time.
Call Travel Time	Calculated time interval (in seconds) between Dispatched Date and Time, and Arrival Date and Time.
Call On-Scene Time	Calculated time interval (in seconds) between Arrival Date and Time, and Cleared Date and Time.
Call Response Time	Calculated time interval (in seconds) between Reported Date and Time, and Arrival Date and Time.
Call Service Time	Calculated time interval (in seconds) between Dispatched Date and Time, and Cleared Date and Time.
Total Call Time	Calculated time interval (in seconds) between Reported Date and Time, and Cleared Date and Time.
Total Unit Service Time	Calculated total service time (in seconds) of all units that were dispatched in CAD, excluding dispatching delay in the Communications Centre

Table 2.1: Fields in occurrence data

2.2.1 Call Priority

Each call received for service is assigned a priority based on its type. This priority may change over the time as more information is collected about the situation. Final Priority field in the dataset represents the last priority which was assigned to the call when the incident was marked closed in CAD. For our experiments we use this priority to decide if a vehicle should be dispatched or not.

Following list describes each priority code[2]:

0 - Officer Needs Assistance

1 - Immediate

2 - Urgent

3 - Routine

4 - Delay - When Zone Officer Becomes Available

5 - Differential Police Response (DPR) - officer not required to attend; taken over the phone at Police Reporting Centre (PRC)

6 - Collision Reporting Centre - motor vehicle collisions with property damage only; complainant attends PRC

7 - Officer Initiated - officer generated and/or present at that location

8 - Proactive - event specifically generated due to community project, directed patrol, strategic enforcement

9 - Administrative (Communications Alert); filed call not requiring police response

2.2.2 Time Intervals

Figure below depicts various time intervals involved in handling a call for service:

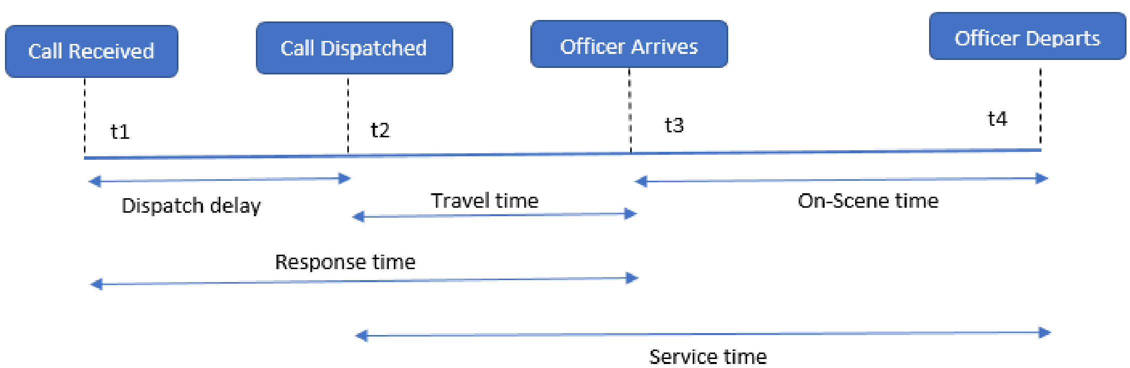


Figure 2.1: Time mileposts and time intervals in police emergency response system

Dispatch Delay (t2 - t1) - Duration until a dispatch is made for an incident. Call is usually placed in a dispatch queue and is responded when a unit becomes available.

Travel Time (t3 - t2) - Travel duration from unit's location to location of incident or call.

Response Time (t3 - t1) - Total delay in responding to a committed call for service request.

On-Scene Time (t4 - t3) - Time spend between a unit's arrival at the incident location to when it is cleared.

Service Time (t4 - t2) - Time spend for a particular call for service, i.e. travel time plus on-scene time.

2.3 Data Analysis

In this section, we perform data analysis on raw occurrence data for the year 2014. Study of this data gives valuable insights about the call for service patterns and helps us in making informed decisions for our studies. We should take into account that the statistics and graphs presented in this section are based on raw data of calls. Given the complexity involved in real-world policing operations, information released through dataset might not accurately represent exact details of how a request was really handled.

From figure 2.2 we notice that the number of emergency calls is least among all types, and communication alerts dominate overall occurrence data. Not all call for service requests require a police officer to travel to the incident location; thus we focus our study only on priority 1 to 4. Plot of location coordinates (figure 2.3) shows where the majority of calls are reported which could give information about where patrol units should be concentrated.

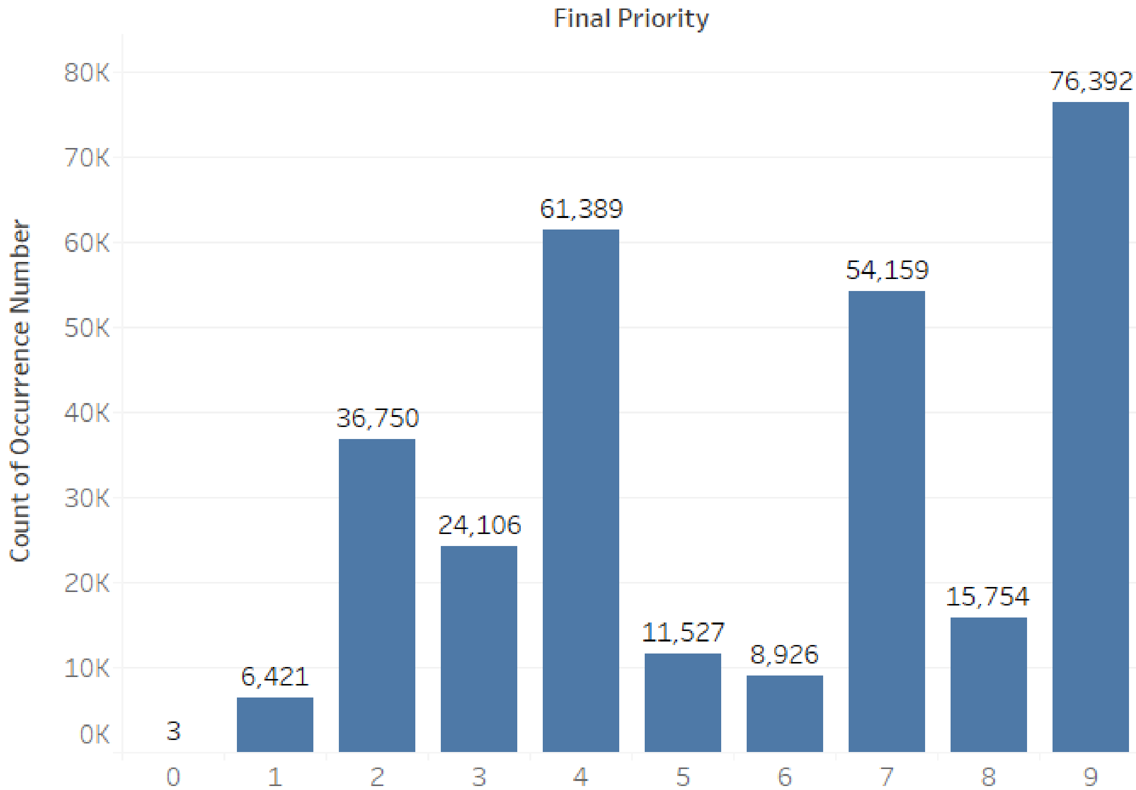


Figure 2.2: Plot showing count of Occurrence Number for each final priority for the year 2014.

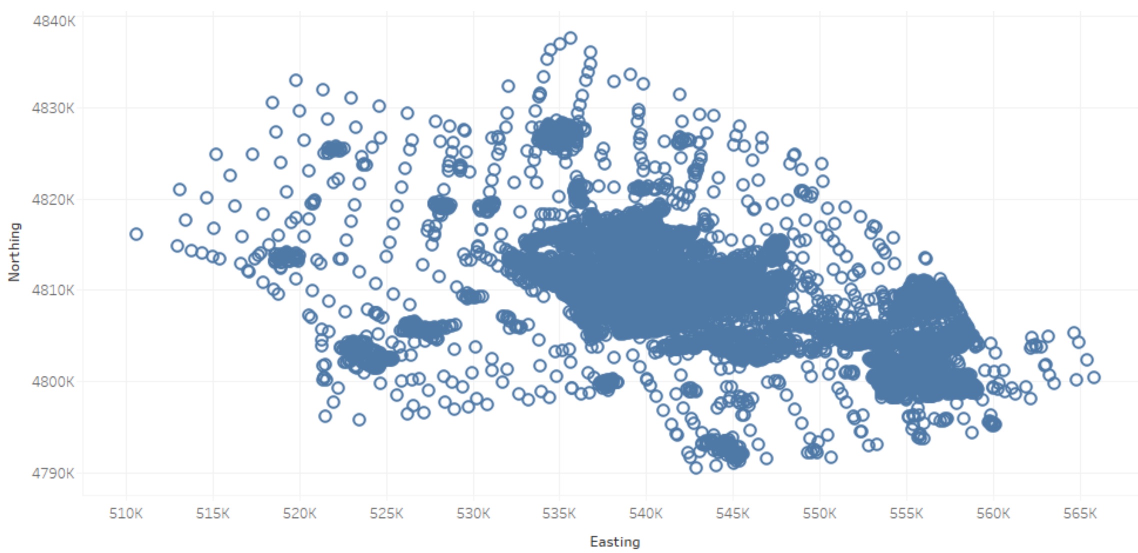


Figure 2.3: Plot showing location details of priority 1 to 4 occurrences for the year 2014.

In figure 2.4, the breakdown of calls in the year 2014 by month shows that the number of priority 1 calls during each month are more or less consistent, while there exists a lot of fluctuations for priority 4. With the help of figure 2.5 we can see that number of jobs increase during the weekend.

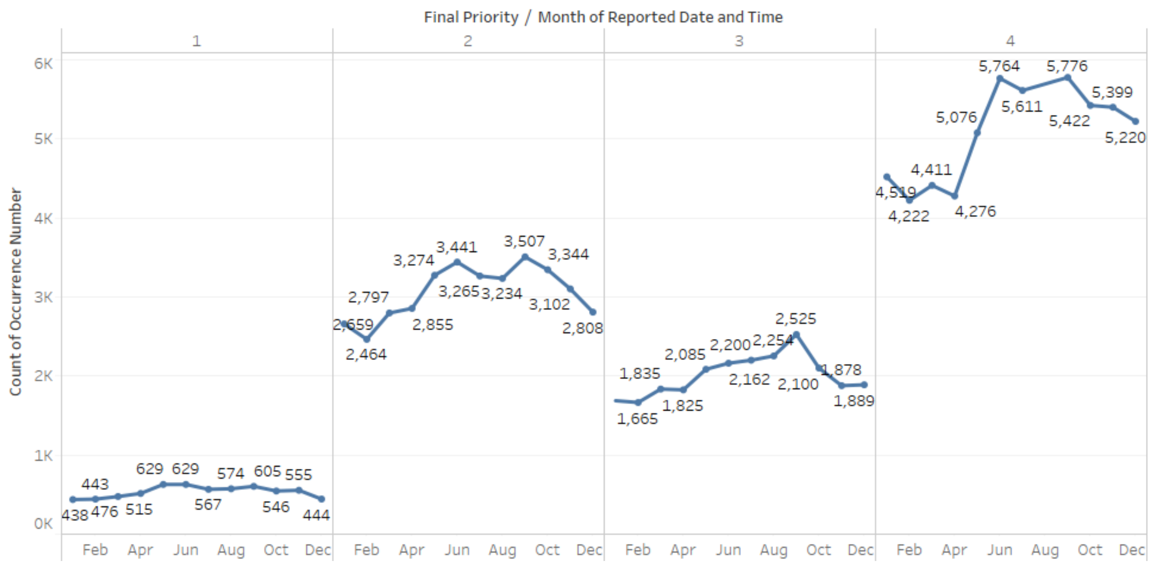


Figure 2.4: The trend of count of Occurrence Number for each month broken down by final priority for the year 2014. The marks are labeled by count of Occurrence Number.

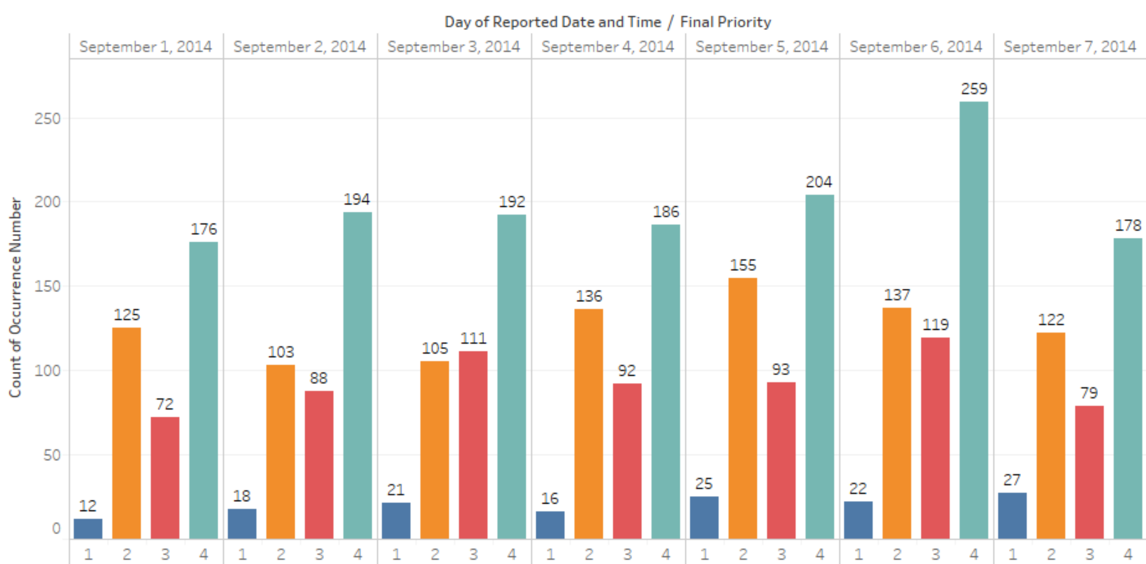


Figure 2.5: The trend of count of Occurrence Number for first seven days of September broken down by final priority.

Information about the distribution of calls (figure 2.6) during each hour of the day gives an insight about workload, as it can be seen that the evening hours are the peak hours when the majority of incidents are reported, with early morning hours being relatively moderate.

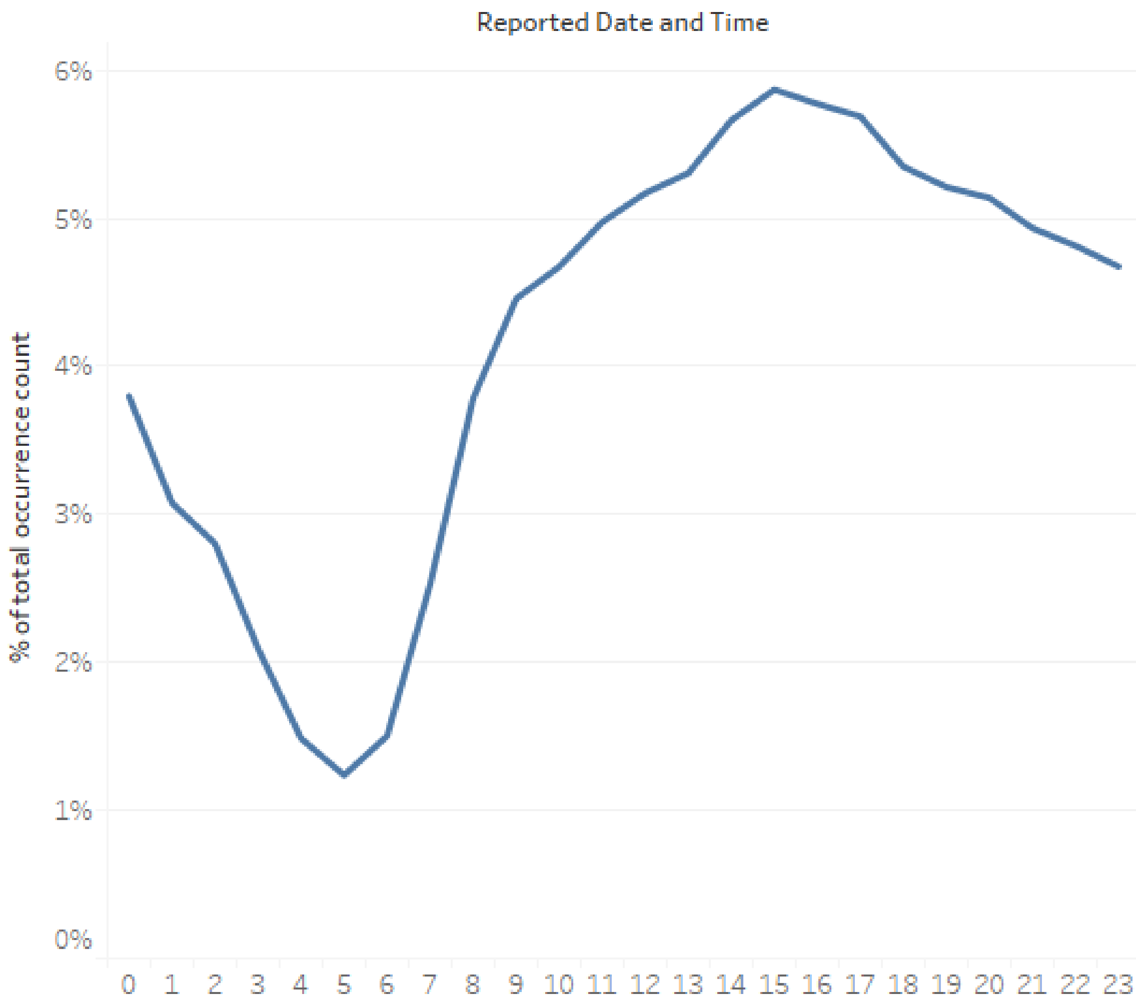


Figure 2.6: Plot showing percentage of priority 1 to 4 calls reported during each hour of a day for the year 2014.

Analysis of time fields on data suggests that on average approximately 10 minutes of travel time is required for jobs which is uniform throughout the year 2014. Higher value of average dispatch delay in figure 2.7 is due to priority 4 calls, as they are delayed because of unavailability of zone officers. There was a sharp swing for dispatch delay in February, but it is not clear if it is because of some erroneous data or some unique incidents.

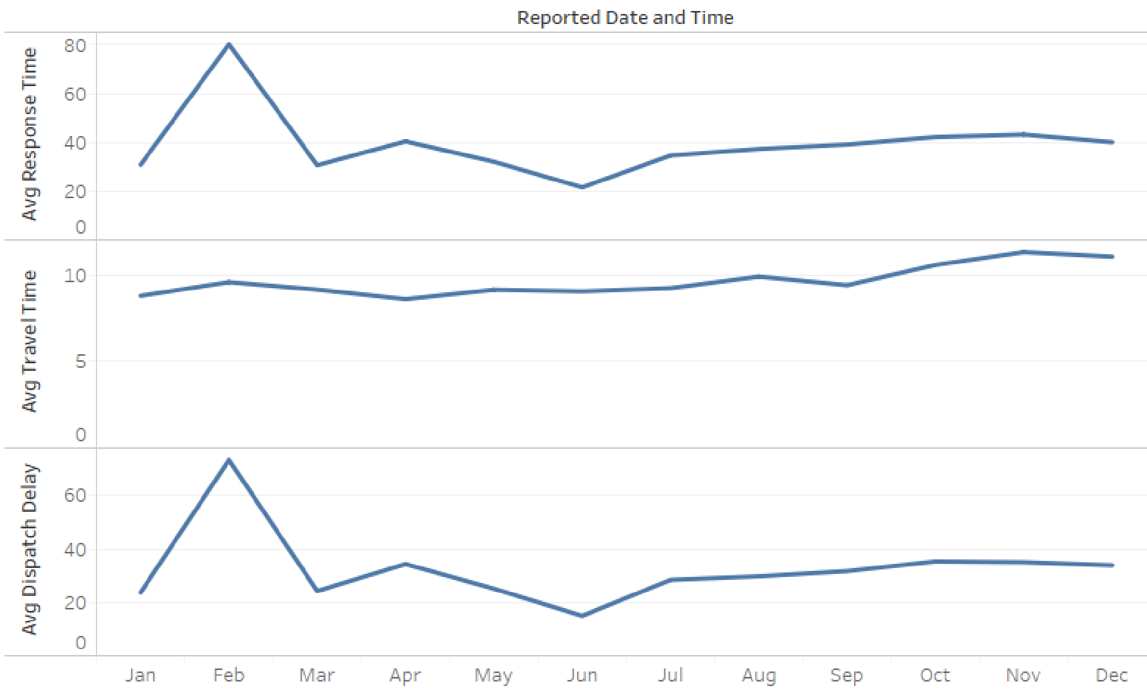


Figure 2.7: The trends of average dispatch delay, travel time and response time (in minutes) for each month in the year 2014. The data is filtered on final priority, which ranges from 1 to 4.

Another critical aspect is analyzing data from multiple years (figures 2.8, 2.9). Plotting occurrence entries for the year 2014 and 2015 reveals that the number of calls among various priorities has strikingly similar patterns in both the year. For priority 1-3, there isn't any major change in the number of occurrences in the year 2015 compared to 2014.

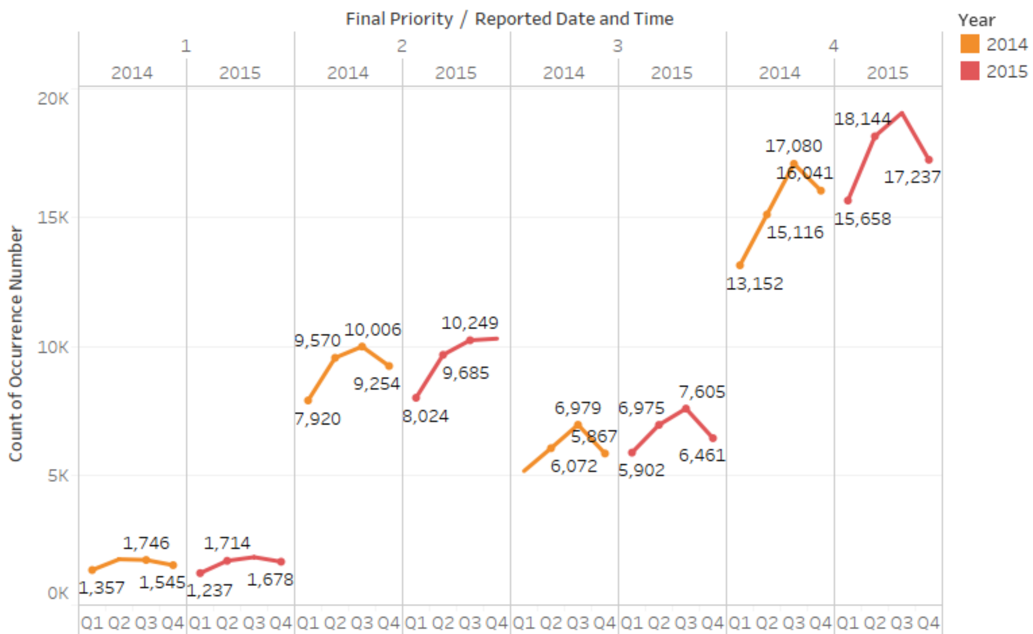


Figure 2.8: The trend of count of Occurrence Number for each quarter of year 2014 and 2015. The data is filtered on final priority, which ranges from 1 to 4.

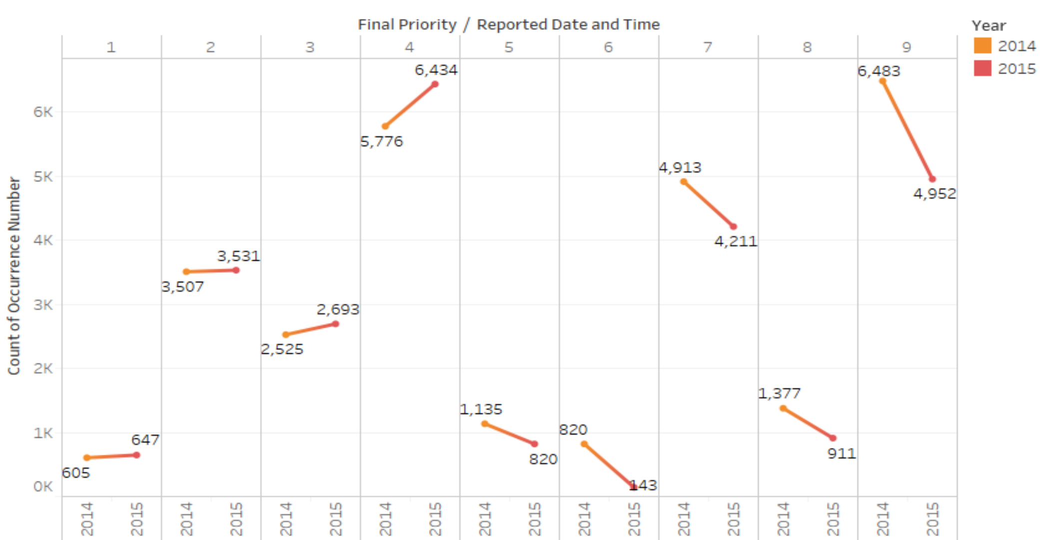


Figure 2.9: The trend of count of Occurrence Number for month of September in year 2014 and 2015.

2.4 Preparing Data for Experimentation

We choose occurrence data for the year 2014, which contains more than 2,90,000 records. Each record provides detailed information regarding a particular call for service and how it was handled. We will use information about location, reported date and time, cleared date and time, and priority for our dispatching model.

2.4.1 Data Filtering

To test our models we need sufficient and accurate data with all the required fields. We perform filtering based on various fields on original dataset.

Time fields: Since our work focuses only on police dispatching and shift scheduling, we eliminate records with missing dispatch date, reported date, cleared date. This operation removes significant data rows, which is apparent as most of the calls do not really require any police dispatch or are handled over the phone. We found some calls with very high values of on-scene time and response time. On the other hand, some data rows have time fields, like travel time, service time, dispatch delay, etc., with negative values. We were not able to find documentation explaining the significance of such values. Therefore we drop all the entries with on-scene and response time greater than 8 hours, and time fields with negative values.

Priority: WRPS occurrence dataset doesn't provide information regarding how many police officers respond to a call for service, nor if a police officer was dispatched for a particular incident. On the basis of description of call priorities, we consider only priority 1, 2, 3, 4, and 8 call types.

Location: Analysis of processed data shows that the location of some calls is outside the primary cluster of call requests. We pick jobs with Easting value between 532000 and 560000, and Northing between 4795000 and 4820000.

After this data cleaning operation, we still have a high number of call for service data for our studies. As we will see in next chapter, size of our model depends heavily on the number of incidents. Since urgent calls have precedence during peak hours, we decided to drop priority 4 calls between 6 PM to 4 AM for our experiments.

Chapter 3

Police Dispatching

3.1 Preliminary

In this chapter, we will discuss how we can estimate a lower bound on number of police officers for a known call for service data. To solve this problem, we decided to go with mathematical programming approach. Linear programming (LP) is the most fundamental tool in combinatorial optimization, where a problem in hand is represented with a linear objective function which needs to be minimized or maximized, and a set of linear constraints on nonnegative variables are used to represent the limitations. Mathematically, Linear Programming is an optimization technique which could be represented in the following standard form:

$$\text{Minimize} \quad c^{\mathrm{T}}x$$

$$\text{Subject to} \quad Ax \leq b$$

$$\text{and} \quad x \geq 0$$

where x represents variables whose values we want to determine, c and b are the known coefficients, $c^{\mathrm{T}}x$ collectively represents the objective function of our problem, inequalities $Ax \leq b$ and $x \geq 0$ are the constraints.

Linear Programming was invented by George Dantzig in 1947, and has been a growing area ever since. Industries in transportation, telecommunications, energy, and manufacturing have been using it for decades for optimizing their operations.

3.2 Police Dispatching Problem

Let's take an example of a call for service and understand the sequence of actions. Suppose an incident occurs where an offender poses a serious threat to people, and a police service is requested. This call for service is usually handled by a dispatcher, who collects the information and assigns a police unit to the incident. Police arrive on the scene and take necessary action, and if required may initiate a call for service themselves. A police officer may also need additional time to do any paperwork related to the incident. Once everything is in order, police may signal the call as cleared by passing this information to dispatcher, where it could be marked as completed or successful.

These service calls vary in terms of severity from regular administrative work to emergency situations. Based on the severity, each call for service needs to be handled separately. Generally, police departments assign a priority to each type of request on the basis of available information and predefined criteria. This priority decides how a particular call should be handled and what actions are required. Service calls with low priority often involve delay in dispatch so that high priority jobs do not get affected. Though delay in dispatch depends on a lot of factors, it mostly comes down to the availability of police units in that area. Since we cannot be sure about the nature of future service requests, it becomes really tricky to decide assignment of a police unit to a service request.

Analysis of occurrence data has also shown that the number of calls for service among different priorities varies based on season and day of the week. A constant number of police officers on duty for each day would then lead to under-manning on some day and over-manning on other. Thus decision on the number of police officers on duty on a particular day in a particular region affects the overall dispatching of police officers for incidents.

As it is evident that increase in the number of police personnel would be an easy fix to various objectives of efficient policing operation, this strategy leads to significant increase in costs to government and public. On the other hand, the insufficient number of personnel deployment would not only affect public but also police officers doing overtime to meet the demands. It couldn't be emphasized more that an efficient policing operation is of utmost importance which could bring improvement in the daily life of people in the city.

3.3 Model Construction

We will describe a model for police dispatching with an objective to minimize the response time. Response time is an essential metric for checking the efficiency of policing operations, and in our model, we will try to obtain best response time with given number of police officers.

We consider the static version of police dispatching, where we know all the details of the call for service for a day in advance. This variant is in contrast to the actual dynamic nature of dispatching where the next request is unknown to a dispatcher. We believe results from static version would give us valuable insights about domains which could be enhanced. We denote each call for service as a *job*, with a location and priority associated with it. A location would be the coordinates where unit needs to be present for handling the job. Priority would determine how many patrol units are required for that job and maximum possible dispatch delay. In our problem, we assign two police officers for priority 1 jobs and one for rest of the priorities. Also, each job has a start time and end time. Start time would be the time by which a police officer should arrive at the job location. End time would be the time when the police officer is cleared from the job. When it comes to using WRPS occurrence data, we will consider reported time as start time and call on-scene duration added to reported time as end time.

Each police officer has a shift during which he/she is available for accepting jobs. In our model we do not consider depot, and assume police officer is ready to take jobs immediately as soon as shift begins. A police officer can handle only one job at a time and would need to travel to the job location. Travel distance is calculated using euclidean distance between location coordinates. Once done with a job, he/she would devote some fixed time in proactive policing, during which no job would be assigned to that officer. Police officers would not take a job with start time outside their shift, but could undertake a job which started in their shift but ended beyond their shift. This would mean that police officer did overtime in order to satisfy the requirements.

3.4 Big-M Method

In this section, we will present a basic formulation which uses Big-M method.

3.4.1 Notations

We define a set of notations to model our problem:

$N = \{1, 2, \ldots, n\}$ - set of all job requests.
$P = \{1, 2, \ldots, p\}$ - set of available police officers.
t_j^{begin} - time at which job $j \in N$ is reported in the dataset.
t_j^{end} - time at which job $j \in N$ should finish if started at time t_j^{begin}.
$\tau_k^{\text{begin}}, \tau_k^{\text{end}}$ - time at which shift for police officer $k \in P$ starts and finishes.
c_j - number of police officers required to service job $j \in N$.
d_{ij} - time cost to move from location of job $i \in N$ to location of job $j \in N$.
proactive_time - a fixed proactive time after the end of each job.
E_j - maximum allowed delay in starting a job $j \in N$.

3.4.2 Variables

We use following variables in our formulation:

x_{ijk} - binary, determines if police officer k is going to do job j after job i.
s_{jk} - binary, determines if police officer k is going to do job j as first job.
f_{jk} - binary, determines if police officer k is going to do job j as last job.
e_j - non-negative continuous, determines delay in starting a job $j \in N$.

3.4.3 Modeling Problem as Network Flow

We model our problem as a network flow, where each job $j \in N$ is represented as a node, and a directed arc from job node a to job node b represents feasibility of doing job b after job a. We also add a start node s and a final node f to our graph to represent source and destination.

Handling priority 3 and 4 jobs

Jobs belonging to priority 3 and 4 are not urgent and therefore do not require immediate police dispatch. We will use e_j to measure the delay in starting a job j after its reported time. To control jobs getting postponed for an arbitrary long duration, we will set maximum allowed delay in responding to a job, denoted by E_j. For our model, we will consider E_j to be zero for priority 1 and 2 jobs, and 2 hours for priority 3 and 4 jobs.

Condition for arc between nodes

There is an arc from start node s to every other job node, and each job node has an outgoing arc to final node f. We will denote by $A^-(j)$ (respectively $A^+(j)$) a set of incoming (outgoing) arcs to (from) node $j \in N$. There exists an arc from node $i \in N$ to node $j \in N$, i.e. $(i, j) \in A^-(j)$ and $(i, j) \in A^+(i)$, if following constraints hold true:

$$t_i^{\text{end}} + d_{ij} + \mathit{proactive\_time} \leq t_j^{\text{begin}} + E_j \tag{3.1}$$

Constraint (3.1) ensures that police officer has enough time to perform proactive work and travel to his/her next job. We allocate a fixed proactive time after the end of the job, represented by *proactive_time*, to take care of any community work or administrative tasks. Also if job j could be postponed, we ensure that we don't eliminate an arc without considering the maximum delay.

3.4.4 Mathematical Formulation

Our IP objective function aims at minimizing response time, which is sum of delay (e_j) in starting a job and travel time (d_{ij}) between jobs. We use binary variables x_{ijk} specifying if police officer k traveled from job i to job j, s_{jk} to indicate if police officer k did job j as his/her first job, and f_{jk} if job j was done as a final job in his/her shift.

$$\min \sum_{j \in N} (e_j + \sum_{i \in N:(i,j) \in A^-(j)} d_{ij} (\sum_{k \in P} x_{ijk})) \tag{3.2}$$

subject to

police start - a police officer can do max one job as a starting job:

$$\sum_{j \in N} s_{jk} \leq 1, \qquad \forall k \in P \tag{3.3}$$

police finish - a police officer can do max one job as a final job. This constraint is redundant given condition 3.3, but it could help solver in finding bounds:

$$\sum_{j \in N} f_{jk} \leq 1, \qquad \forall k \in P \tag{3.4}$$

police officer transition through the node:

$$s_{jk} + \sum_{i \in N:(i,j) \in A^-(j)} x_{ijk} = \sum_{l \in N:(j,l) \in A^+(j)} x_{jlk} + f_{jk}, \qquad \forall k \in P, \forall j \in N \tag{3.5}$$

police officer shift constraint - first job should be after shift begins:

$$\sum_{j \in N} t_j^{\text{begin}} s_{jk} \geq \tau_k^{\text{begin}}, \qquad \forall k \in P \tag{3.6}$$

police officer shift constraint - final job start time should be within the shift. We choose a large enough value of M'_{jk} such that if f_{jk} is zero, start time of job j need not be within the shift of officer $k \in P$:

$$t_j^{\text{begin}} + e_j \leq \tau_k^{\text{end}} + M'_{jk}(1 - f_{jk}), \qquad \forall j \in N, \forall k \in P \tag{3.7}$$

police requirement for jobs:

$$\sum_{k \in P} s_{jk} + \sum_{k \in P} \sum_{i \in N:(i,j) \in A^-(j)} x_{ijk} = c_j, \qquad \forall j \in N \tag{3.8}$$

Big-M constraint - We choose a (big enough) value for M_{ij} s.t. if x_{ijk} is 0, constraint shouldn't add any extra restrictions on e_j and e_i:

$$M_{ij}(1 - x_{ijk}) + t_j^{\text{begin}} + e_j \geq t_i^{\text{end}} + e_i + proactive\_time + d_{ij}, \qquad \forall k \in P, \forall i, j \in N \tag{3.9}$$

e_j constraint - bound on maximum delay:

$$e_j \leq E_j, \qquad \forall j \in N \tag{3.10}$$

variables:

$$x_{ijk}, s_{jk}, f_{jk} \in \{0, 1\},\ e_j \geq 0, \qquad \forall k \in P, \forall i, j \in N \tag{3.11}$$

3.4.5 Analysis

Values of M play a significant role in Big-M formulations. If not selected carefully, they lead to numerical problems and weak linear relaxations[15, 9]. It is possible to choose appropriate values of M based on data inspections to avoid numerical issues, but still, the problem of weak relaxation could remain[12]. We tested this formulation on Gurobi Optimizer, a mathematical programming solver, with parameter value for MIP Gap at 0.10 and time limit at 7200 seconds (2 hours). We assume car speeds of 40 kph, and $proactive\_time$ after each job is considered as 20 minutes. Police officers have 24-hour shift, making them available for any job in a day. Value of c_j is two for priority 1 jobs, and one for other priorities. We use data of September 1st 2014 with 263 jobs from the filtered dataset discussed in chapter 2 for evaluating our model. We used different values of police count from 25 to 35, and couldn't obtain a feasible solution within time-limit.

3.5 Time Discretized Method

In order to obtain a solution in reasonable time and incorporate delay for jobs in our model, we need to modify our formulation approach. If we model lesser priority jobs as a single node with a fixed start time, it becomes difficult to bring flexibility in dispatching. Thus we create multiple copies of each priority 3 and 4 job at intervals of time δ from their original start time, with start time and end time of copies adjusted accordingly. This adjustment in start time for a job j would be mapped as delay, sh_j. For example, in our dataset, if there is a priority 4 job with start time 8 AM, δ is set to 20 minutes, and we are creating five copies of each priority 3 and 4 job, then five extra jobs would be created with start time 8:20 AM, 8:40 AM, 9:00 AM, 9:20 AM, and 9:40 AM with sh_j value 20, 40, 60, 80, and 100 respectively. These job copies would also be represented as a regular node in the network. We will use N' to denote the set of all the original jobs and their copies if any. We will describe by $J(z)$ the set of related jobs of job z, i.e., set of copies of z and job z itself. We call $J(z)$ as *job-set* of job z. For other priority jobs, $J(z)$ would only contain original job z. A police officer could be assigned to any of the jobs in the job-set to satisfy original service call request.

3.5.1 Notations

Let's define a new set of notations to represent our problem:

$N = \{1, 2, \ldots, n\}$ - set of all original job requests.
$P = \{1, 2, \ldots, p\}$ - set of available police officers.
$N' = \{1, 2, \ldots, n'\}$ - set of all job requests with copies.
$t_j^{\text{begin}}, t_j^{\text{end}}$ - time at which job $j \in N'$ starts and finishes.
$\tau_k^{\text{begin}}, \tau_k^{\text{end}}$ - time at which shift for police officer $k \in P$ starts and finishes.
c_j - number of police officers required to service job $j \in N'$.
d_{ij} - time cost to move from location of job $i \in N'$ to location of job $j \in N'$.
sh_j - delay in starting a job. For priority 1 and 2, it will be zero.
$proactive\_time$ - a fixed proactive time after the end of each job.
gap_time - maximum possible time interval between two consecutive jobs by a police officer.

3.5.2 Variables

We use following variables in our formulation:

x_{ijk} - binary, determines if police officer k is going to do job j after job i.
s_{jk} - binary, determines if police officer k is going to do job j as first job.
f_{jk} - binary, determines if police officer k is going to do job j as last job.

3.5.3 Modeling Problem as Network Flow

We again model our problem as a network flow, where each job $j \in N'$ is represented as a node, and a directed arc from job node a to job node b represents feasibility of doing job b after job a. We also add a start node s and a final node f to our graph to represent source and destination.

Condition for arc between nodes

There is an arc from start node s to every other job node, and each job node has an outgoing arc to final node f. We will denote by $A^-(j)$ (respectively $A^+(j)$) a set of incoming (outgoing) arcs to (from) node $j \in N'$. There exists an arc from node $i \in N'$ to node $j \in N'$, i.e. $(i, j) \in A^-(j)$ and $(i, j) \in A^+(i)$, if following constraints hold true:

$$t_i^{\text{end}} + d_{ij} + proactive\_time \leq t_j^{\text{begin}} \tag{3.12}$$

$$i, j \notin J(z) \qquad \forall z \in N \tag{3.13}$$

Constraint (3.12) ensures that police officer has enough time to perform proactive work and travel to his/her next job. We allocate a fixed proactive time after the end of the job, represented by *proactive_time*, to take care of any community work or administrative tasks. Next, with constraint (3.13) we restrict arcs between jobs belonging to the same job-set.

Edge elimination technique

One serious problem with constructing edges only based on previous criteria is that we add a lot of edges because of copy nodes in the network. In an optimal solution, only one incoming arc will be used out of all the arcs to nodes in a job-set of a job with c_j as 1, but our network size is unnecessarily inflated with all possible arcs between nodes. We propose a technique to eliminate a significant number of edges between nodes. Figure 3.1 is an elementary network of three original jobs, represented in blue. We created two copy nodes (in green) each for two of those jobs. Graph contains all the valid edges based on constraints (3.12) and (3.13). We have skipped start node and finish node for simplicity. Assume path indicated with arcs in orange is a feasible solution for the problem. The arc from J_a to J_{b2} is the first arc in the path. J_{b2} belongs to job-set $\{J_b, J_{b1}, J_{b2}\}$. We claim that if there is a job in the job-set of J_{b2} with start-time earlier than that of J_{b2}, with a valid arc from J_a (in this case J_a to J_{b1}), then we can obtain another feasible solution by eliminating J_a to J_{b2}, and considering J_a to J_{b1} instead. This is because any job node which is reachable from J_{b2} will also be reachable from a node in job-set of J_{b2} which has start-time before that of J_{b2}. Choosing a job earlier than J_{b2} would mean improvement in response time. Therefore we add the following two conditions for having an arc from node $i \in N'$ to node $j \in N'$.

There exists no job j' other than j in job-set $J(z)$, where $j \in J(z)$ and $z \in N$, s.t:

$$t_i^{\text{end}} + d_{ij'} + proactive\_time \leq t_{j'}^{\text{begin}} \tag{3.14}$$

with

$$t_{j'}^{\text{begin}} \leq t_j^{\text{begin}} \tag{3.15}$$

These two conditions limit arc from a job $i \in N'$ to only one copy of job $z \in N$. We select this by choosing the earliest copy job node in time which is reachable from job i.

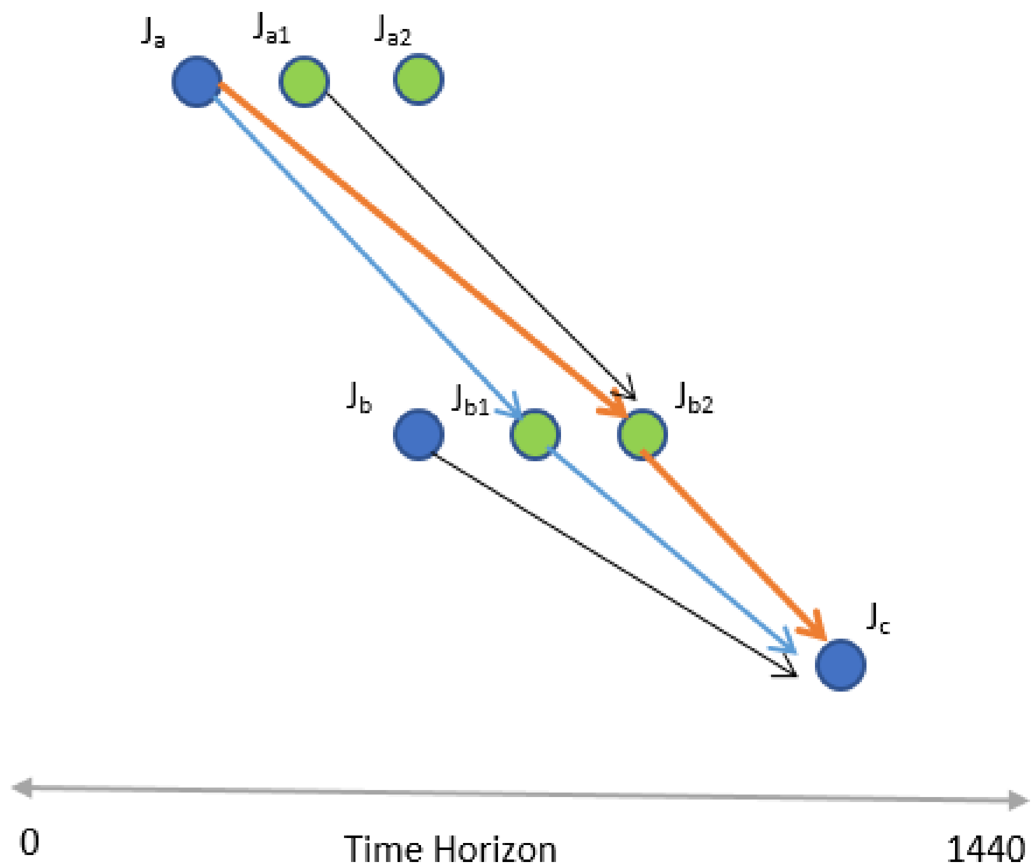


Figure 3.1: A simple three job (in blue) network with two jobs having two copy nodes each (in green). Time horizon is of 24 hours, i.e., 1440 minutes

To further restrict the size of the graph we introduce constraint (3.16), which removes arcs between job nodes which are far apart in time, precisely, the time difference between next job start time and current job end time should be less than *gap_time*. One additional benefit of this approach is that it helps in reducing the gap between consecutive jobs of a police officer. Thus there won't be a case where a police officer finished a job at 9 AM, and his next job was at 4 PM, if *gap_time* was set to 3hrs.

$$t_j^{\text{begin}} - t_i^{\text{end}} \leq gap\_time \tag{3.16}$$

3.5.4 Mathematical Formulation

Our IP objective function aims at minimizing response time, which is sum of delay (sh_j) in starting a job (because of picking a job which is a copy of original job) and travel time (d_{ij}) between jobs. We use binary variables x_{ijk} specifying if police officer k traveled from job i to job j, s_{jk} to indicate if police officer k did job j as his/her first job, and f_{jk} if job

j was done as a final job in his/her shift.

$$\min \sum_{j \in N'} ((\sum_{k \in P} sh_j(\sum_{i \in N':(i,j) \in A^-(j)} x_{ijk} + s_{jk})) + \sum_{i \in N':(i,j) \in A^-(j)} d_{ij}(\sum_{k \in P} x_{ijk})) \tag{3.17}$$

subject to

police start - a police officer can do max one job as a starting job:

$$\sum_{j \in N'} s_{jk} \leq 1, \qquad \forall k \in P \tag{3.18}$$

police finish - a police officer can do max one job as a final job. This constraint is redundant given condition 3.18, but it could help solver in finding bounds:

$$\sum_{j \in N'} f_{jk} \leq 1, \qquad \forall k \in P \tag{3.19}$$

police officer transition through the node:

$$s_{jk} + \sum_{i \in N':(i,j) \in A^-(j)} x_{ijk} = \sum_{l \in N':(j,l) \in A^+(j)} x_{jlk} + f_{jk}, \qquad \forall k \in P, \forall j \in N' \tag{3.20}$$

police officer shift constraint - first job should be after shift begins:

$$\sum_{j \in N'} t_j^{\text{begin}} s_{jk} \geq \tau_k^{\text{begin}}, \qquad \forall k \in P \tag{3.21}$$

police officer shift constraint - final job's start time should be within the shift:

$$\sum_{j \in N'} t_j^{\text{begin}} f_{jk} \leq \tau_k^{\text{end}}, \qquad \forall k \in P \tag{3.22}$$

police requirement for jobs - if copies of a job were created, then count of police officers over all job copies should be equal to original job demand. Only priority 3 and 4 jobs have copies, which have demand c_z as one:

$$\sum_{j' \in J(z)} (\sum_{k \in P} s_{j'k} + \sum_{k \in P} \sum_{i \in N':(i,j') \in A^-(j')} x_{ij'k}) = c_z, \qquad \forall z \in N \tag{3.23}$$

variables:

$$x_{ijk}, s_{jk}, f_{jk} \in \{0,1\}, \qquad \forall k \in P, \forall i,j \in N' \tag{3.24}$$

3.6 Estimating Lower Bounds

Values for various parameters govern the output of our model. One such important parameter is the number of police officers. Our focus in the first stage of our experiment is to estimate the lower bound on the number of police officers for satisfying demands in a day, which would then be fed into the second stage for shift scheduling. As this value would depend on shift length being used, we would consider 24-hour shift for all the police officers in this step. We run our model with different input values of police officer count and select the feasible solution obtained within the time limit with least number of police officers. By this approach, we estimate the minimum count of police officers for satisfying demand and then a dispatching strategy which reduces response time.

3.7 Implementation and Experimental Results

We capture the mathematical programming formulation mentioned in section 3.5.4 in an LP format, which could be passed to Gurobi Optimizer. LP format is an easy to read modeling format structured as a list of segments, where each segment represents a logical piece of the optimization model.

In this run, we consider a time horizon of 24 hrs, i.e., 1440 minutes. Police officers have 24-hour shift, therefore τ_k^{begin} is 0 and τ_k^{end} is 1440, making them available for any job in a day. Value of c_z is two for priority 1 jobs, and one for other priorities. We create five copies of priority 3 and 4 jobs, with δ as 20 minutes. We fix the value of *gap_time* at 180 minutes, which would give more compact scheduling of police officers. We assume car speeds of 40 kph, and *proactive_time* after each job is considered as 20 minutes. Changing values for these parameters would generate different solution, but considering a small set of combination would be sufficient to prove the core concept of dispatching and scheduling. For Gurobi Optimizer, we set parameter value for MIP Gap at 0.10 and time limit at 3600 seconds (1 hour).

We use the filtered dataset discussed in chapter 2 for evaluating our model. We will consider two weeks from the year 2014: September 1st to September 7th, and March 31st to April 6th. We will present results for both the weeks but would focus our attention on output from September 1st week for detailed analysis. Table 3.1 and 3.2 gives statistics obtained from execution of our model. We don't see any visible pattern in response time based on the number of jobs and police count. Results compiled on using different officer count on Sept 6th data (table 3.3) shows how response time decreases with increase in officer count.

Date	Number of Jobs	Police Estimate	Best Bound	Total Response Time	Gap
1-Sep	263	28	2712	2893	6.26%
2-Sep	264	31	3001	3076	2.44%
3-Sep	289	35	3891	3967	1.91%
4-Sep	306	33	3275	3386	3.28%
5-Sep	306	34	3882	3888	0.15%
6-Sep	311	30	3875	4150	6.62%
7-Sep	269	27	2739	2739	0.00%

Table 3.1: Estimated lower bounds on officer count for each day of 1st week of September 2014 (based on 1 hour timeout limit)

Date	Number of Jobs	Police Estimate	Best Bound	Total Response Time	Gap
31-Mar	281	41	3809	4152	8.26%
1-Apr	236	38	2040	2134	4.40%
2-Apr	258	33	3676	3992	7.92%
3-Apr	253	30	3184	3403	6.43%
4-Apr	262	33	2359	2379	0.84%
5-Apr	209	27	2668	2806	4.92%
6-Apr	229	27	2928	2938	0.34%

Table 3.2: Estimated lower bounds on officer count for each day of 1st week of April 2014 (based on 1 hour timeout limit)

Count of Police Officers	Best Bound	Total Response Time (in minutes)	Gap
30	3875	4150	6.62%
31	3012	3110	3.15%
32	2463	2473	0.40%
33	2069	2157	4.08%
34	1799	1932	6.88%

Table 3.3: Response time output for different count of police officers for Sept 6th 2014 data

Analysis of results for a particular day show some exciting aspects of our model. Figure 3.2 asserts that our model could generate very good response time for all the priorities compared to real-world response time targets. Report on General Duty Staffing Assessment for Surrey[6] mentions that 7 minutes of emergency response time is considered reasonable from the cost perspective, and provides satisfactory service to the public. With the current model, we were able to achieve an average of 3-5 minutes of response time for priority 1 and 2 calls.

Though response time is optimized, officer scheduling generated as a result has some issues. From figures 3.3 and 3.4 we notice that the job count for police officers is disproportional and some of the officers had no job assigned in the morning, but others had a busy schedule.

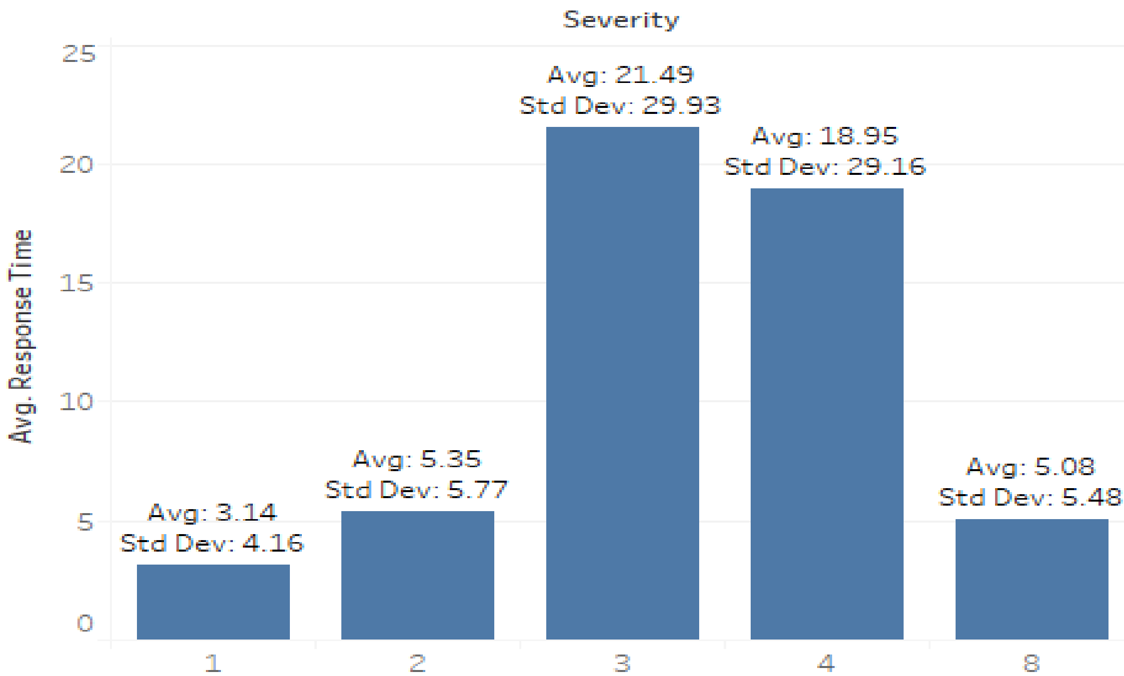


Figure 3.2: Response time statistics for September 6th 2014 with 30 police officers

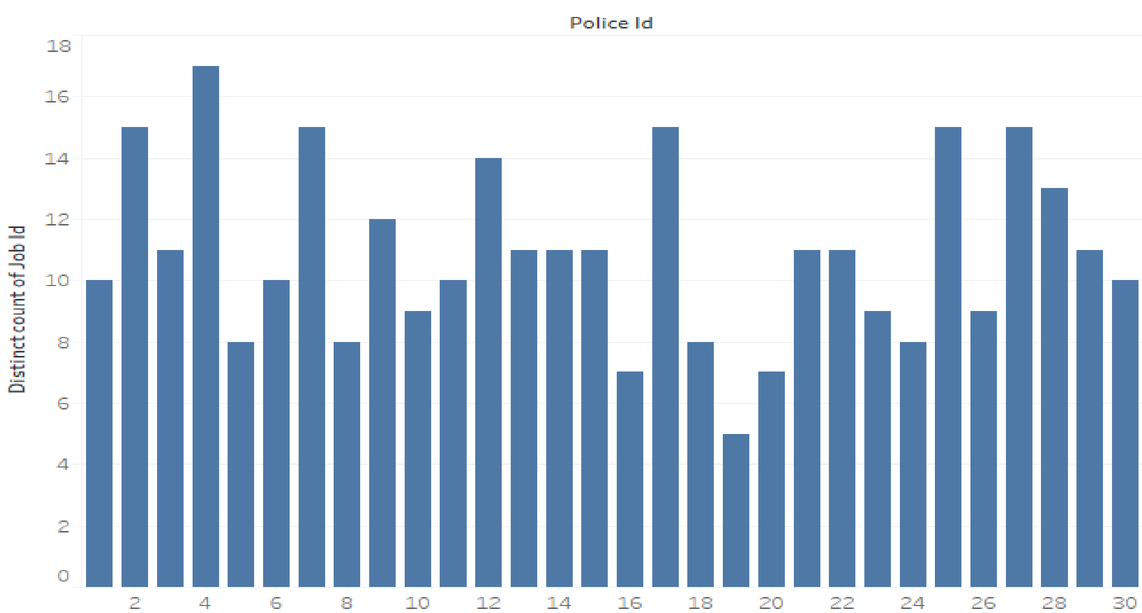


Figure 3.3: Job count of police officers for September 6th 2014 with 30 police officers

Figure 3.4: Job schedule for each police officer for September 6th 2014 on time horizon of 1440 minutes (24 hours). Blue section marks the time duration when the police officer was attending a job.

3.8 Conclusion

We explored the problem of police dispatching and presented a mathematical programming formulation with the aim to minimize response time. We evaluated our model on WRPS dataset and estimated officer demand for a week of September and April 2014. We also performed detailed analysis on the results for a single day, September 6th, and discussed statistics on response time and officer schedules.

Chapter 4

Shift Scheduling

4.1 Background

A shift schedule could be described as a record of employee availability on each day of a week. Industries or services which require 24/7 availability of personnel often implement their work schedule as a sequence of different types of shifts. A *shift* involves a collection of working hours on particular days followed by a rest period. In policing industry, the problem of generating shift schedules for police officers could be considered as an optimization problem in which demand for personnel must be satisfied with conditions on total working hours of police officers, work policies, and other preferences. Though manual approach of generating shifts is often a laborious task and limited to finding a feasible solution, it is still fairly common in many areas. Use of mathematical models, on the other hand, is not only easy and accurate but provides much better performance. Best example of this would be of San Francisco Police Department(SFPD), which moved away from hand designed schedules and implemented Police Patrol Scheduling System (1989)[14] and saw a significant upgrade in productivity with 20% improvement in response times.

Shift designs in other industries have also garnered major interest among researchers. One such problem is of *nurse scheduling*, where work schedule is developed for nurses in the health-care sector. Berrada et al.[7] and Miller[18] tackled this by developing mathematical programming models with the objective to optimize staffing needs and incorporating personal preferences of individual nurses. For shift scheduling in call centres, queuing models and heuristics were derived by Andrews et al.[5] and Buffa et al.[8]. Also, quality of shifts cannot be justified solely based on quantitative optimizations, in this regard work published by Vila et al.[21] discusses how various aspects of a shift affects police officer's health and performance. He mentions that a study conducted on officers with fewer workdays in a week showed lesser fatigue than those with 5-day on and 2-day off, 8-hour shifts.

Our aim concerning shift scheduling would be to consider a set of standard patrol shift types in the police departments and satisfy officer demands.

4.2 Demand Table for Shift Scheduling

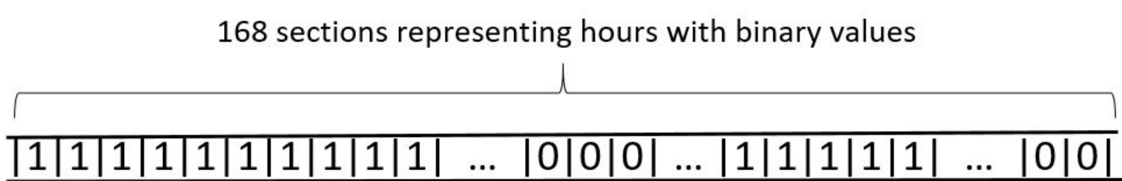
To put in simple terms, a demand table is a record of police officer requirements in each hour of a day or week. We generate this data for a day from the output of our dispatching model based on the jobs considered for that day. For each hour of a day, we count the number of police officers assigned to a job or doing some proactive duty. If we plan to create demand table for a week, we run our dispatching model for every day of that week.

It is essential to understand the significance of demand table with regard to shift scheduling. They provide patterns in requirement which could be used to adjust shift timings to optimize overall number of policing personnel on duty. It should also be noted that using demand tables alone doesn't suffice personnel requirements in the real world. This is because of nature of policing industry where work is carried out continuously, 24/7. Impact of leaves, vacations, training and other activities sharply increases the actual requirement of employees.

4.3 Model for Shift Scheduling

We aim to create an LP model for generating shifts based on demands. Since shifts are generally scheduled for a week, we use demand data for each hour of that week. Our LP model is inspired by the work done by Bruce Rout[19] at Simon Fraser University, where he proposes a rod-schedule method. In the rod-schedule method, we represent a length of time, in our case 168 (hours in a week), divided into sections on a rod. For instance, for a week, the first section would denote Monday 00:00 hour and last section Sunday 23:00. Each section holds a binary value, 1 if a police officer is working in that hour of the week, or 0 if not. A rod could then be used to model a week of shift for an employee, and a collection of such rods could be used to fulfill demands for a particular week.

Following is an example of a rod:



These rods could be used for describing any shift patterns, but only a set of shift schedules are practical. Policing industry, like any other industry is bound by union and government regulations which formulate working hours and timings for workers. Several studies are conducted to design shifts which try to accommodate workers safety, health, family care, and satisfaction, and effect of those shift schedules have shown better job performance and personal life for individuals. Therefore only specific shift patterns should be considered for scheduling. Details of 10-hour shift and 11-hour shift were found in various police reports[13, 20] with 4 or 5 different start times in a day.

In our model, we consider police officers work 40 hours a week. A shift schedule for a week is decided based on shift type, and start day and time of shift. For example, a 10-hour weekly-shift with four days on and three days off, having a start time of say Monday 8 AM, would have a schedule of 8 AM - 6 PM from Monday to Thursday, with rest days from Friday to Sunday. An 8-hour weekly-shift with five days on and two days off, having a start time of say Wednesday 4 PM, would have a schedule of 4 PM - 2 AM from Wednesday to Sunday, with rest days from Monday to Tuesday. A 40-hour work week on a rod representation would have forty ones and rest zeros. For instance, 10-hour shift with start time on Monday 12 AM would have ones at sections 0-9, 24-33, 48-57, and 72-83.

4.3.1 Notations and Variables

Let's define a set of notations to represent our problem:

$L = \{1, 2, \ldots, l\}$ - set of allowed shift types, i.e., 8-hour, 10-hour, 12-hour, etc.
$T = \{1, 2, \ldots, t\}$ - set of allowed start times, i.e., Monday 12 AM, Monday 8 AM, Tuesday 12 AM, Tuesday 8 AM, etc.
$\Re^*_{i,j}$ - a vector of ones and zeros, representing the rod with shift type $i \in L$ and start time $j \in T$, with length 168.

Variables:
$Q_{i,j}$ - number of officers following shift schedule based on rod vector $\Re^*_{i,j}$.

4.3.2 Mathematical Formulation

Considering each rod represents 40 hours of work in a week, our objective function would be to minimize total work hours:

$$\min \sum_{i \in L} \sum_{j \in T} 40 Q_{i,j} \tag{4.1}$$

subject to

demand satisfaction for each hour - $\Re^*$ is the binary constrained matrix with 168 rows and $(l * t)$ columns, where l and t is size of set L and T respectively. Columns in $\Re^*$ are rod vectors $\Re^*_{i,j}$, q is the vector with members $Q_{i,j}$, and d is the demand vector with members d_h, $h \in \{1,2,3...168\}$, where d_h is the demand in hour h of the week.:

$$\Re^* q \geq d \tag{4.2}$$

non negative values for $Q_{i,j}$:

$$q \geq 0 \tag{4.3}$$

4.4 Finding Number of Police Officers

Results from shift scheduling formulation would give us the number of police officers needed for each type of shift. Adding them would tell us the minimum number of police officers required to meet demands for that particular week. As mentioned in previous sections, this number is just based on police officer requirements for jobs. The actual required number would vary significantly based on policies and other work commitments. It is interesting to note that this number might still require police officers to work overtime if they take jobs which start in their shift but end outside their shift. Also, the count of police officers in each shift calculated based on a particular week's data would most likely fail to satisfy demands for other weeks. This is obvious as demands change across weeks, more so in different seasons.

4.5 Implementation and Experimental Results

We again use Gurobi Optimizer for our shift scheduling problem. To test our model we separately examine two shift types with total 40 working hours per week: 10-hour shift with 4 days on and 3 days off, and 8-hour shift with 5 days on and 2 days off. We also consider only fixed set of start times for weekly-shifts, precisely, 12 midnight, 8 AM and 4 PM, on each day of a week. Selecting this period will allow for 2-hour overlap between upcoming shift and current shift for 10-hour based schedule, while there won't be any such overlap for 8-hour based schedule. Overlaps could be significant in handling jobs reported around shift change. Demand data for September 1st week based on results from dispatching model is summarized in table 4.1.

MIP solver returns optimal results in few seconds, which are mentioned in table 4.2 and 4.3. Based on 10-hour shift, 139 police officers are required, while with 8-hour shift only 113 officers are needed. To better understand the meaning of this result we plot a graph between demand and number of police officers required based on the optimal solution. Figure 4.1 and 4.2 provide closeness of fit, where it could be seen that for 10-hour shift there are spikes during particular hours. This is caused by 2 hour overlap with our shift selection. In 8-hour shift we didn't consider overlap and thus obtain a much better fit to demand.

Date	9/1/2014	9/2/2014	9/3/2014	9/4/2014	9/5/2014	9/6/2014	9/7/2014
DayofWeek	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
0	9	9	6	10	11	13	16
1	12	10	14	15	13	17	20
2	14	9	12	15	13	19	21
3	18	7	6	12	11	19	22
4	18	7	7	15	11	17	19
5	18	9	10	7	11	12	15
6	17	10	8	4	15	9	18
7	16	15	15	15	18	14	14
8	22	23	33	27	32	19	23
9	23	30	33	33	34	27	26
10	25	31	33	33	33	28	27
11	27	31	33	28	34	30	27
12	28	30	33	32	34	30	27
13	28	31	33	33	32	30	27
14	28	31	33	33	33	30	27
15	28	31	33	33	34	30	27
16	28	31	33	33	34	30	27
17	28	31	33	33	34	30	27
18	28	31	33	33	34	30	26
19	28	31	33	33	34	30	27
20	28	31	29	27	34	30	26
21	28	28	29	30	30	30	27
22	27	23	29	30	30	29	24
23	23	24	25	28	26	26	20

Table 4.1: Estimated hourly demand of police officers during the 1st week of September 2014

ShiftType	DayOfWeek	StartHour	OfficerCount
10	0	0	3
10	0	8	7
10	0	16	7
10	1	0	2
10	1	8	11
10	1	16	11
10	2	0	4
10	2	8	9
10	2	16	9
10	3	0	6
10	3	8	6
10	3	16	6
10	4	0	6
10	4	8	8
10	4	16	8
10	5	0	3
10	5	8	7
10	5	16	7
10	6	0	7
10	6	8	6
10	6	16	6

Table 4.2: The number of police officers required in each 10-hour shift type for demand in table 4.1. First column is the type of shift, which is 10-hour shift. Second column is day of week, with 0 for Monday, 1 for Tuesday and so on. Third column gives start hour with value between 0-23. Fourth column is the output of the model, i.e. count of the police officers.

ShiftType	DayOfWeek	StartHour	OfficerCount
8	0	0	3
8	0	8	9
8	0	16	9
8	1	8	8
8	1	16	8
8	2	0	6
8	2	8	7
8	2	16	8
8	3	0	4
8	3	8	5
8	3	16	5
8	4	0	6
8	4	8	6
8	4	16	5
8	5	0	3
8	5	8	5
8	5	16	5
8	6	0	3
8	6	8	4
8	6	16	4

Table 4.3: The number of police officers required in each 8-hour shift type for demand in table 4.1. First column is the type of shift, which is 8-hour shift. Second column is day of week, with 0 for Monday, 1 for Tuesday and so on. Third column gives start hour with value between 0-23. Fourth column is the output of the model, i.e. count of the police officers.

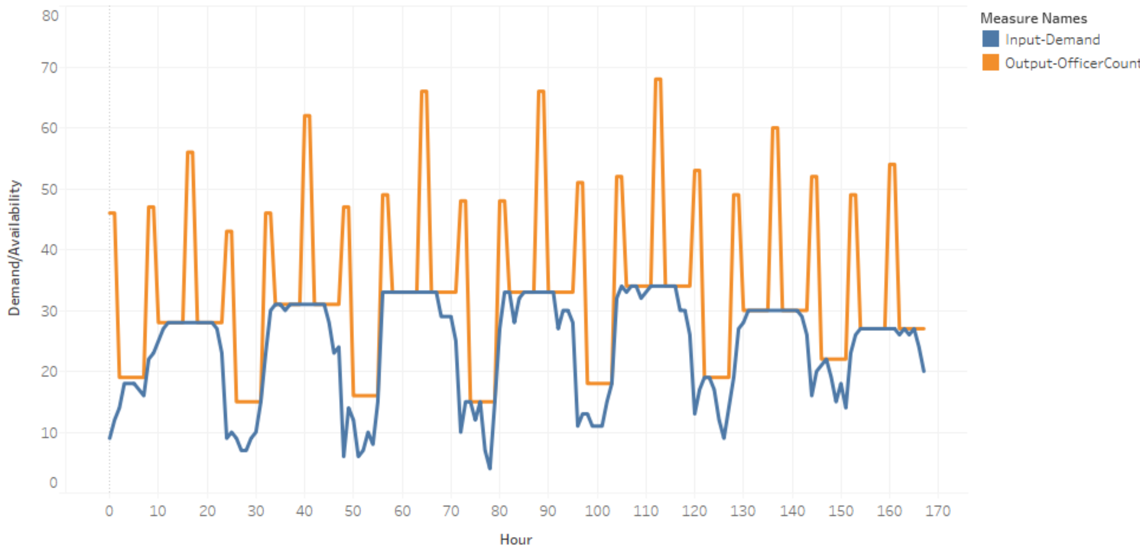


Figure 4.1: Line graph showing officer demand and actual availability based on 10-hour shift with start times 12AM, 8AM, and 4PM

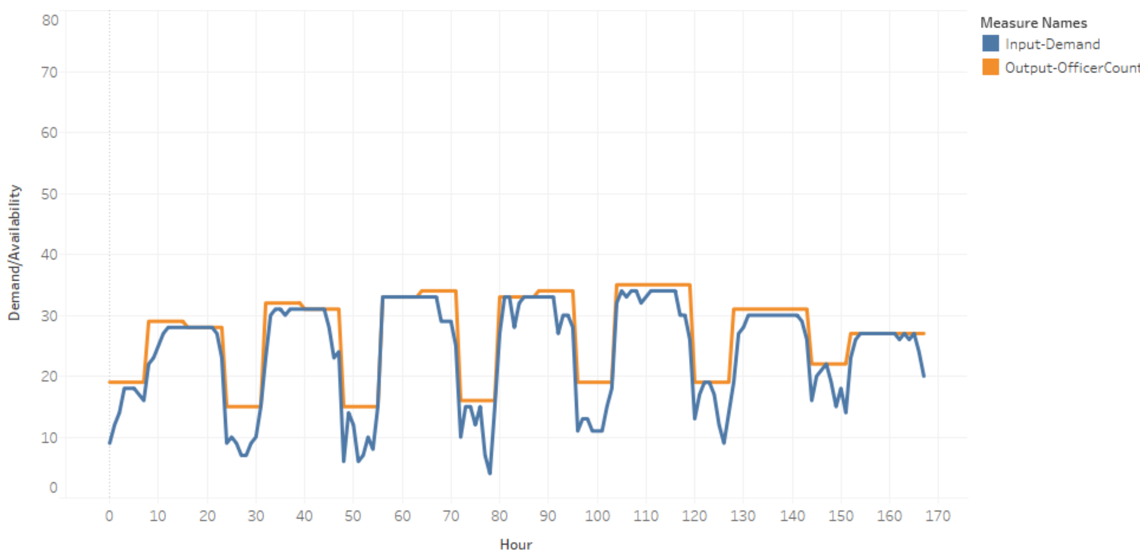


Figure 4.2: Line graph showing officer demand and actual availability based on 8-hour shift with start times 12AM, 8AM, and 4PM

4.6 Conclusion

We used the rod-schedule method to estimate personnel requirements to satisfy hourly demands of officers for a week. Results from 10-hour and 8-hour shift types with start times 12 AM, 8 AM and 4 PM were analyzed on WRPS occurrence data for the first week of September 2014. 10-hour based schedules with shift overlap require 139 police officers, while 8-hour based schedules without shift overlap require only 113 officers.

Chapter 5

Shift Based Dispatching

5.1 Objective Function for Model

In first stage of our approach we used officers with 24-hour shift, which is unrealistic in actual policing operations. In order to make our first stage model more practical, we need to make some modifications. To begin, we will use shift schedules generated using rod-schedule method for our dispatching decisions. A police officer is assigned a shift based on the requirements obtained from our shift scheduling run. This police officer would only take jobs belonging to his/her shift. Now input to our program would include the total number of police officers with their availability hours. In the first stage of our approach we optimized dispatching for better response times, but in this step, we would minimize officer overtime keeping other parameters same. In the real world, overtime plays a vital role in dispatching decisions, where an officer is often not allocated any new incident investigation if it is reported at the end of his/her shift, given other officers are available.

5.2 Mathematical Formulation

We add a constraint to calculate officer overtime and change our objective function to minimize it. Rest of the constraints and formulation remains same as in section 3.5.4. We use non-negative continuous variable y_k to represent total overtime done by police officer $k \in P$. Our modified formulation would look like:

$$min \sum_{k \in P} y_k \tag{5.1}$$

subject to

police start - a police officer can do max one job as a starting job:

$$\sum_{j \in N'} s_{jk} \leq 1, \qquad \forall k \in P \tag{5.2}$$

police finish - a police officer can do max one job as a final job. This constraint is redundant given condition 5.2, but it could help solver in finding bounds:

$$\sum_{j \in N'} f_{jk} \leq 1, \qquad \forall k \in P \tag{5.3}$$

police officer transition through the node:

$$s_{jk} + \sum_{i \in N' : (i,j) \in A^-(j)} x_{ijk} = \sum_{l \in N' : (j,l) \in A^+(j)} x_{jlk} + f_{jk}, \qquad \forall k \in P, \forall j \in N' \tag{5.4}$$

police officer shift constraint - first job should be after shift begins:

$$\sum_{j \in N'} t_j^{\text{begin}} s_{jk} \geq \tau_k^{\text{begin}}, \qquad \forall k \in P \tag{5.5}$$

police officer shift constraint - final job's start time should be within the shift:

$$\sum_{j \in N'} t_j^{\text{begin}} f_{jk} \leq \tau_k^{\text{end}}, \qquad \forall k \in P \tag{5.6}$$

police requirement for jobs - if copies of a job were created, then count of police officers over all job copies should be equal to original job demand. Only priority 3 and 4 jobs have copies, which have demand c_z as one:

$$\sum_{j' \in J(z)} \left(\sum_{k \in P} s_{j'k} + \sum_{k \in P} \sum_{i \in N' : (i,j') \in A^-(j')} x_{ij'k} \right) = c_z, \qquad \forall z \in N \tag{5.7}$$

police officer overtime constraint - calculates overtime done by a police officer:

$$\sum_{j \in N'} (t_j^{\text{end}} + proactive\_time) f_{jk} - \tau_k^{\text{end}} \leq y_k, \qquad \forall k \in P \tag{5.8}$$

variables:

$$x_{ijk}, s_{jk}, f_{jk} \in \{0, 1\}, \qquad \forall k \in P, \forall i, j \in N' \tag{5.9}$$

5.3 Experiment and Results

For experiments, we set the time limit of 2.5 hours on MIP solver. Again, we consider the same parameter values as used in 1st stage of approach. We consider a time horizon of 24 hrs, i.e., 1440 minutes. Values of τ_k^{begin} and τ_k^{end} is decided based on shift schedules. Value of c_z is two for priority 1 jobs, and one for other priorities. We create five copies of priority 3 and 4 jobs at repeated intervals of 20 minutes. We fix the value of *gap_time* at 180 minutes, which is the maximum possible time interval between two consecutive jobs by a police officer. We assume car speeds of 40 kph, and *proactive_time* after each job is considered as 20 minutes.

Table 5.1 is the digest of results for September 2014 1st week with 10-hour shift schedule. Third column of the table contains total officer hours based on shift based dispatching, which is calculated by adding all the shift hours of police officers available on a particular day. Fourth column is the total officer hours used, calculated based on the demand table 4.1. Then we report overtime best bound, and obtained solution by Gurobi Optimizer. We notice that officer shifts successfully handled all the jobs for the whole week. Total daily overtime values ranged from 341 minutes on Sept 4th to 1659 minutes on Sept 5th, which could be considered reasonable given the number of officers on duty each day(fig. 5.1). For example, on Sept 1st, average overtime is around 11.38 minutes per officer. Also, this overtime includes proactive time after their last job, which could be avoided if work is going beyond officer's shift. Response time values(fig. 5.2) see an increase compared to 1st stage results, which is expected as the focus in this stage was to minimize overtime.

Date	Number of Jobs	Total Officer Hours Available	Total Officer Hours Used (based on 1st stage 24hr shift)	Overtime (best bound)	Overtime (obtained)
1-Sep	263	748	549	922	922
2-Sep	264	764	544	160	462
3-Sep	289	816	586	560	1528
4-Sep	306	810	592	266	341
5-Sep	306	858	625	1246	1659
6-Sep	311	798	579	0	822
7-Sep	269	766	560	1278	1386

Table 5.1: Results for 1st week of Sept 2014 with 10-hour shifts: showing total officer hours each day and total of overtime (in minutes) from all police officers

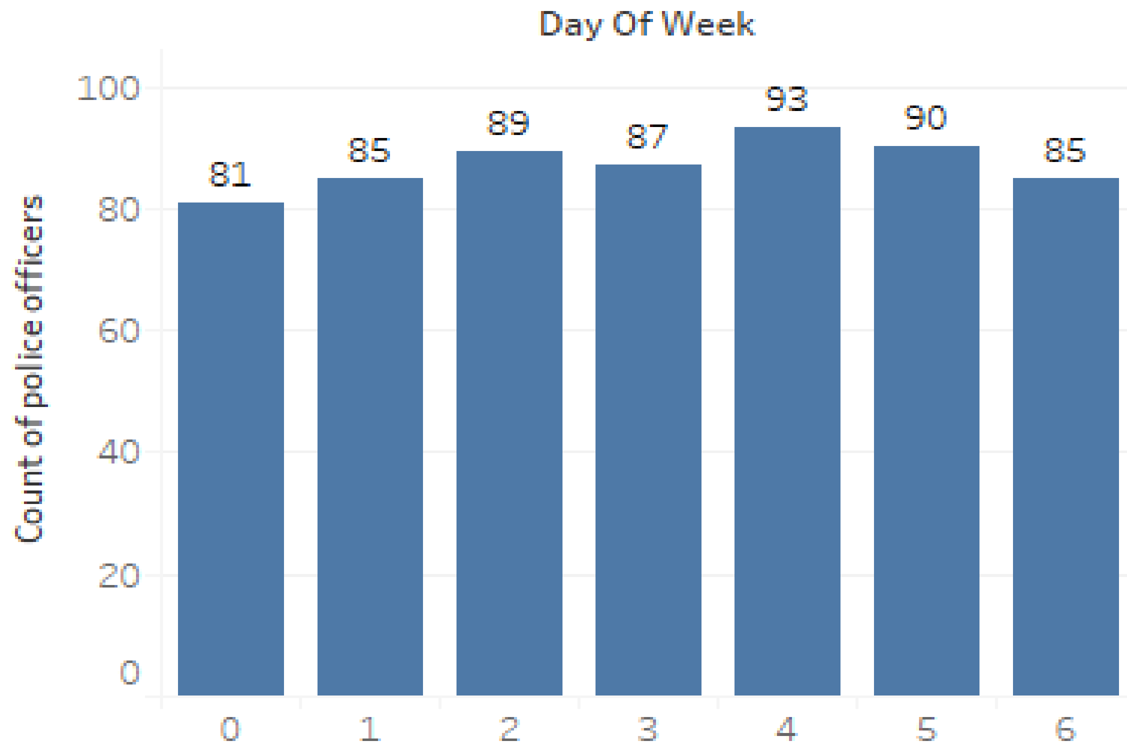


Figure 5.1: Count of police officers on duty on each day based on 10-hour shift schedule

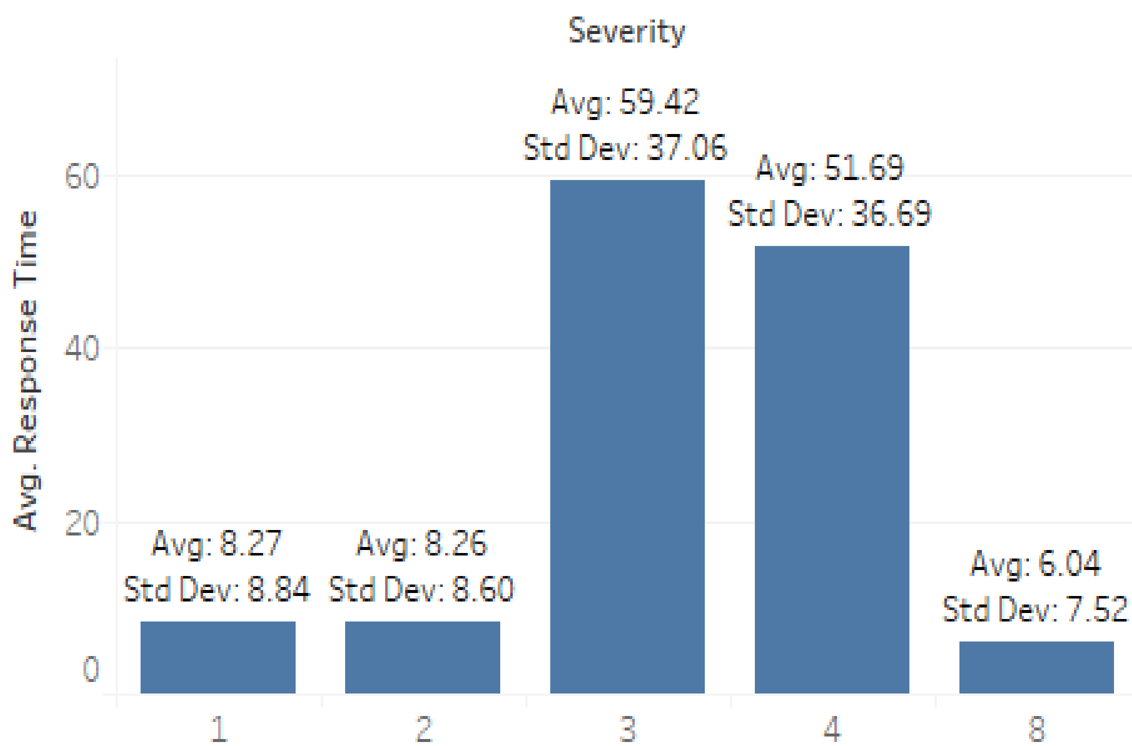


Figure 5.2: Response time analysis based on results of shift based dispatching for Sept 6th 2014

On 8-hour shift schedules (table 5.2), we get some interesting results, where total overtime varied from 1771 minutes to 3561 minutes. This is because of non-overlapping shift schedules in 8-hour shifts, as we considered shifts with start time 12-midnight, 8 AM, and 4 PM. Also, as a consequence total officer hours is reduced, as graphed in figure 5.3.

Date	Number of Jobs	Total Officer Hours Available	Total Officer Hours Used (based on 1st stage 24hr shift)	Overtime (best bound)	Overtime (obtained)
1-Sep	263	608	549	1600	2441
2-Sep	264	624	544	146	1771
3-Sep	289	656	586	1358	2823
4-Sep	306	664	592	44	2177
5-Sep	306	712	625	0	3561
6-Sep	311	648	579	148	2441
7-Sep	269	608	560	2322	2863

Table 5.2: Results for 1st week of Sept 2014 with 8hr shifts: showing total officer hours each day and total of overtime (in minutes) from all police officers

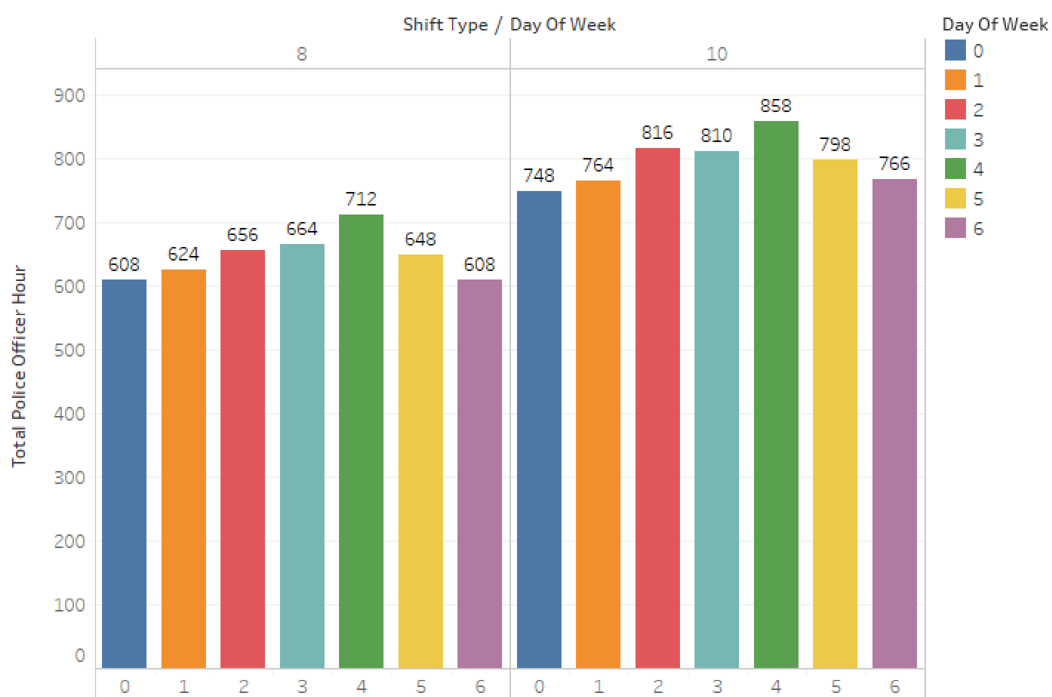


Figure 5.3: 8hr and 10hr shift comparison based on total officer hours allocated for each day

The most intriguing result was when 10-hour shifts generated based on the year 2014 data fulfilled the requirement of the year 2015. Outputs mentioned in table 5.3 show the successful assignment of officers for the jobs in 1st week of September 2015 with moderate overtime.

Date	Number of Jobs	Total Officer Hours Available	Overtime (best bound)	Overtime (obtained)
31-Aug-15	273	748	491	1888
1-Sep-15	265	764	92	1919
2-Sep-15	230	816	306	306
3-Sep-15	257	810	292	292
4-Sep-15	260	858	689	689
5-Sep-15	264	798	560	965
6-Sep-15	247	766	543	553

Table 5.3: Results on 1st week of Sept 2015 using schedule generated for Sept 2014 1st week.

5.4 Conclusion

We examined how our shift schedules affect police dispatching by evaluating our revised model on the first week of September 2014 data. First, we discussed results based on 10-hour shifts where allocated police officers could satisfy demand with moderate overtime. Next, we compared results for 8-hour shift schedules with 10-hour based schedules. Finally, we demonstrated how shift generated on the year 2014 data could meet demands for the year 2015 occurrences.

Chapter 6

Conclusion

In this work, we studied the various aspects of police patrol allocation strategies and presented a model for dispatching and shift scheduling. We first determine a dispatching plan by estimating smallest police force with 24-hour shift schedule. This schedule establishes the hourly police patrol requirements to successfully service the jobs minimizing the total response time. The second part involves selecting a smallest size police force with shift schedules to satisfy the hourly job load determined in the first part. The third part is to verify and determine the dispatching schedule of the job data using the police force with schedules established by the second part. We develop mathematical programming models for our problem and test them on Waterloo Regional Police Service Occurrence Data, evaluating the performance on various criterion, like response time, officer overtime, and total number of police officers. For the experiment data of the first week of September 2014, we estimated that 139 police officers with 10-hour based shift schedules with start times 12 AM, 8 AM, and 4 PM could satisfy the call for service requirements. Based on the proposed dispatching model, we obtained average response time of around 8 minutes for priority 1 and 2 jobs, while it was under an hour for priority 3 and 4.

We believe our approach gives an insight into the interactions between the resources needed and their assignment in policing agencies, which could then be used to analyze current operations and assist in any future policy changes.

Bibliography

[1] Annual Report 2016. `http://www.atyourservice2016.ca/occurrences.html`. Waterloo Regional Police Service. Accessed: 2017-11-16.

[2] Description of WRPS Occurrence Data Fields. Waterloo Regional Police Service.

[3] Waterloo Regional Police Service: About Our Data. Waterloo Regional Police Service.

[4] Renata Algisi Takeda, João Widmer, and Reinaldo Morabito. Analysis of ambulance decentralization in urban emergency medical service using the hypercube queueing model. *Computers and Operations Research*, 34:727–741, 03 2007.

[5] Bruce Andrews and Henry Parsons. Establishing telephone-agent staffing levels through economic optimization. *Interfaces*, 23(2):14–20, 1993.

[6] Peter Bellmio. General duty staffing assessment. Technical report, Surrey RCMP, 2014.

[7] Ilham Berrada, Jacques A. Ferland, and Philippe Michelon. A multi-objective approach to nurse scheduling with both hard and soft constraints. 30:183–193, 09 1996.

[8] Elwood S. Buffa, Michael J. Cosgrove, and Bill J. Luce. An integrated work shift scheduling system. *Decision Sciences*, 7(4):620–630, 1976.

[9] Jeffrey D. Camm, Amitabh S. Raturi, and Shigeru Tsubakitani. Cutting big m down to size. *Interfaces*, 20(5):61–66, 1990.

[10] Jan M Chaiken. Patrol allocation methodology for police departments. *The Rand Corporation*, 1975.

[11] Kenneth R. Chelst and Ziv Barlach. Multiple unit dispatches in emergency services: Models to estimate system performance. *Management Science*, 27(12):1390–1409, 1981.

[12] Alejandro Crema. Mathematical programming approach to tighten a big-m formulation. *Escuela de Computacion, Facultad de Ciencias, Universidad Central de Venezuela*, 08 2014.

[13] Simon Demers and Adam Palmer. Vancouver Police Department Patrol Deployment Study. 2007.

[14] Philip E. Taylor and Stephen Huxley. A break from tradition for the san francisco police: Patrol officer scheduling using an optimization-based decision support system. 19:4–24, 02 1989.

[15] Ed Klotz and Alexandra M. Newman. Practical guidelines for solving difficult mixed integer linear programs. *Surveys in Operations Research and Management Science*, 10 2013.

[16] Richard Larson. Hypercube queuing model: User's manual. *The Rand Corporation*, 1975.

[17] Richard C Larson. *Urban police patrol analysis.* Cambridge, Mass. MIT Press, 1972.

[18] Holmes Miller, William Pierskalla, and Gustave J. Rath. Nurse scheduling using mathematical programming. 24:857–870, 10 1976.

[19] Bruce Rout. An examination of resourcing and scheduling within the RCMP / by bruce rout.

[20] Edmonton Police Service. A New Patrol Delivery Service Model. 2007.

[21] Bryan Vila, Gregory B. Morrison, and Dennis J. Kenney. Improving shift schedule and work-hour policies and practices to increase police officer performance, health, and safety. *Police Quarterly*, 5(1):4–24, 2002.