TIFAR MODELING PACKAGE
FOR THE EVALUATION OF
EMERGENCY MEDICAL SERVICES
With EMS modeling results for the Amsterdam area

A thesis submitted to the
Delft Institute of Applied Mathematics
in partial fulfillment of the requirements

for the degree

MASTER OF SCIENCE
in
APPLIED MATHEMATICS

by

MARTIN VAN BUUREN
Delft, The Netherlands
December 2010

Copyright © 2010 by Martin van Buuren. All rights reserved.
MSc THESIS APPLIED MATHEMATICS

“TIFAR Modeling Package for the Evaluation of Emergency Medical Services”
With EMS modeling results for the Amsterdam area.

MARTIN VAN BUUREN

Delft University of Technology

Daily supervisors
Prof. dr. ir. K. I. Aardal
Prof. dr. ir. R. D. van der Mei

Responsible professor
Prof. dr. ir. K. I. Aardal

Other thesis committee members
Prof. dr. ir. C. Vuik
ir. H. N. Post

December 2010

Delft, The Netherlands
## Contents

1 Introduction ........................................... 11

2 Literature review .................................... 13

   2.1 Outline ............................................. 13

   2.2 General dispatch procedure ....................... 13

       2.2.1 Dispatch process ............................... 13

       2.2.2 Call events that gets logged in The Netherlands .... 15

       2.2.3 The response time in three parts ................. 16

       2.2.4 Road plan ...................................... 17

   2.3 Current operation in The Netherlands ............... 17

       2.3.1 Nationwide ..................................... 17

       2.3.2 RAV Agglomeratie Amsterdam .................... 20

   2.4 Shortest Path Problems ............................ 29

       2.4.1 Introduction ................................... 29

       2.4.2 Pre-processing algorithms ...................... 29

       2.4.3 Shortest-Path calculations ..................... 33

       2.4.4 Section conclusions ............................ 36

   2.5 Mathematical Models ............................... 37

       2.5.1 Introduction ................................... 37

       2.5.2 Classes of EMS models .......................... 37

       2.5.3 Similarities between all models ................ 37

   2.6 Deterministic Static Models ....................... 39

       2.6.1 Location Set Covering Model (LSCM, 1971) ......... 39
4 Results

4.1 Percentage in time .......................................................... 69
4.2 Coverage per base .......................................................... 71
4.3 Coverage of Agglomeratie Amsterdam .............................. 80
  4.3.1 Not covered potential A1 call origins with a response time of 15 minutes .... 80
  4.3.2 Single covered potential A1 call origins with a response time of 15 minutes .... 80
  4.3.3 Well covered areas in RAV Agglomeratie Amsterdam ............................ 81
  4.3.4 Coverage for A2 calls ................................................. 82
  4.3.5 Coverage for other RAV’s ............................................ 84
  4.3.6 Nearest base to each location in RAV Agglomeratie Amsterdam ............. 86
  4.3.7 Nearest hospital from each location in RAV Agglomeratie Amsterdam .......... 87
4.4 Adding additional bases .................................................. 88

5 Validity and Discussion .................................................. 91

5.1 Validity ................................................................. 91
  5.1.1 Regional borders .................................................. 91
  5.1.2 Agreement between GGD and VZA .................................. 91
  5.1.3 Assistance from other EMS teams .................................. 91
  5.1.4 Assistance from other RAV’s ..................................... 92
  5.1.5 No B calls ......................................................... 92
  5.1.6 Inaccurate speeds and road plan .................................. 92
  5.1.7 Distribution of accidents ........................................... 93
  5.1.8 Choice of EMS hospitals by an EMS vehicle ............................ 93
  5.1.9 No staffing issues ................................................ 93
  5.1.10 No refueling .................................................... 93
  5.1.11 Deterministic treating time and transfer time .............................. 93
  5.1.12 No cruising ..................................................... 94
  5.1.13 Time-independency ............................................... 94
  5.1.14 Dispatch and Relocation rules .................................... 94
  5.1.15 Initialization .................................................... 94
  5.1.16 Influence of randomizer and stop condition ............................ 94
CONTENTS

5.1.17 Concluding ......................................................... 95
5.2 Future research ...................................................... 95

6 Summary ................................................................. 97

A Abbreviations ........................................................... 99

B States within the TIFAR package .................................... 101

B.1 EMS vehicles ......................................................... 101
B.2 Call ................................................................ 102
B.3 Vertices ................................................................. 103
B.4 Standard instances ................................................... 103

C Used notations ........................................................... 105
Preface

The last nine months of the master program Applied Mathematics at Delft University of Technology consists of a project in which several skills learned are brought together. The results of this project are presented in a master thesis.

In this master thesis we will develop a tool that simulates emergency medical services (EMS). We will use this tool to model EMS dispatch in the region around Amsterdam, The Netherlands. The project is initiated by GGD, one of the two EMS providers in the city, who contacted Centrum Wiskunde en Informatica (CWI). Delft University of Technology and CWI have good relation, which explains how I got involved with this project.

To model EMS movements it is very important to have good geographical data and road information. Connexxion has a good relation with Delft University of Technology, and was kind enough to provide me with their GIS data and route planning interfaces.

I would like to start thanking you who I forget to thank in the next paragraph. There are so many people who helped and supported me in one way or the other, that it is impossible for me to include you all. This thesis taught me a lot and helped me to improve my capacities in scientific research, and would not be of this quality without your help.

I want to thank prof. dr. ir. Karen Aardal (TU Delft) and prof. dr. ir. Rob van der Mei (CWI) for giving me the opportunity to do my master thesis on such an interesting subject, and for their daily support. I also want to thank Simon Visser (GGD) for giving me the possibility of studying EMS care in practice by letting me spend two shifts on an EMS vehicle, and a few hours at the EMS dispatch center in Amsterdam. My thanks goes to EMS teams of the GGD and dispatchers of RAVAA for providing me with much valuable information and the unforgettable experience to get an insight in their daily work. I want to thank ir. Henk Post (Connexxion) of the department Future Technology (FT) for providing me with access to the GIS information and Connexxion route planners, and for the large amounts of feedback on my work. My thank goes to everyone of Future Technology at Connexxion for accompanying and supporting me during all the days I’ve spend with you. Furthermore I want to thank ir. Geert Jan Kommer (RIVM) for pointing me to some articles with highly valuable information. Last but not least, I want to thank prof. dr. ir. Kees Vuik (TU Delft) for taking place in my MSc committee.

M. van Buuren
Chapter 1

Introduction

Emergency medical services (EMS) are an important part of society, and provides an interesting field of research in Operation Research (OR). Because there are high costs involved in obtaining and maintaining EMS equipment, and highly qualified staff is needed, it is important to make sure that the resources are used efficiently.

Amongst the various amount of problems EMS managers face are:

- Which fixed base locations must be used, and do we use variable bases?
- How many vehicles must be available at a given time on a day at each base?
- How much EMS personnel do we need, and how should we schedule the crews?
- Which available EMS vehicle should we send to an incoming call?
- How and when do we redeploy resources under different system states?

These points originate from Goldberg (2004) and Restrepo (2008).

In this thesis we develop a simulation package to evaluate the effect of a dispatch strategy, that is called ‘Testing Interface For Ambulance Research’, abbreviated TIFAR. The TIFAR package provides a graphic user interface to see real time effects of decisions, and a mode to quickly compute final statistics. If one has the computing resources, detailed data and enough time to search for a solution, simulation is an excellent tool for obtaining high quality designs and operation rules, see Goldberg (2004). The concept is simple to communicate to decision-makers and the public, see Erkut, et al. (2007).

TIFAR can deal with a very large amount of vertices on which accidents can occur, which is uncommon for previous models. By using the best available EMS vehicle travel speeds in The Netherlands and the reuse of real data an outstanding result can be reached. In TIFAR it is possible to return to a base location without bringing a patient to a hospital, this so called EHGV feature is not seen (by us) in other models. These features are combined with real time visualization and an easy to maintain and extendable ansi C++ program code to achieve a good package for further use.
1 Introduction

Because our test case and the statistics we have used are from Dutch origin, we sometimes use Dutch abbreviations. Appendix A contains an overview of all abbreviations that are used in this thesis. Appendix B gives an overview with details of some classes from the TIFAR modeling package. When discussing mathematical models we use many variables. Appendix C contains a list with the variables that we have used, the meaning of each variable, and details were we use the variables.

Let us draw the outline of the thesis. This thesis consists of six chapters. The second chapter will contain our literature review, which consists of three parts: current operation of EMS in The Netherlands and the Amsterdam region, shortest path problems, and a review of existing EMS models. The third chapter describes the TIFAR model and the choices that are made while developing the evaluation tool. In Chapter 4 we analyze the Amsterdam area with TIFAR. Chapter 5 contains a discussion about the validity of the model, and provides insights for improvements and further research. The final chapter provides a summery of the thesis.

![Figure 1.1: Me at the Royal Palace on Dam Square, Amsterdam.](image)
Chapter 2

Literature review

2.1 Outline

Most of the research in the field of the planning and operation of Emergency Medical Services (EMS) has been done in the last forty years. Even though there are a few surveys focusing on the city of Amsterdam, the majority of the scientific articles are dedicated to other cities or describe general dispatch methods.

In the next section we will discuss the general procedure of EMS dispatch, and in Section 2.3 we describe EMS planning in The Netherlands and the city of Amsterdam in particular. In Section 2.4 we consider algorithms to determine shortest paths in a road network, that is needed as input for the overall algorithms to determine optimal EMS planning as discussed in Section 2.5.

2.2 General dispatch procedure

2.2.1 Dispatch process

Procedure in practice

When an accident occurs that involves EMS care, there will be a call generated by someone. Most of the calls are made to the national emergency number by a civilian (112 in most European countries, and 911 in American countries). The call operator - also referred to as the EMS dispatcher - will start a triage procedure to estimate the sincerity of the emergency call, and labels it with a priority. Also additional information will be asked, such as the victim’s name, age, current location and medical background (if available). The dispatcher will also ask for the name and telephone number of the person who makes the call, so that the dispatcher can reestablish the line if the phone connection is accidentally terminated. There are cases where people place a call, while the situation does not require EMS care, for example because the person can go to a first aid post by his or hers own means of transportation. If necessary the dispatcher gives instructions to the person who called in to tide over until an EMS vehicle arrives to take over the first aid.

EMS dispatch personnel know the current location of all EMS vehicles in their domain at all time using GPS and cellular communication networks that are regulated by computers. After the triage procedure,
the dispatcher contacts an available EMS vehicle according to certain guidelines, and sends it to the location where the call originates from. Without the loss of generality we assume this location to be the same as where the victim is located.

Then, the EMS vehicle will go to the call location, and provides help. When all injuries can be handled locally, the EMS vehicles will return to a predetermined location called a base. This type of call will be classified as an EHGV-call (Dutch: Eerste hulp, geen vervoer. English: First aid, no transporation). If the call is not EHGV, the EMS vehicle will provide transport to a nearby hospital, and returns to the base afterwards. This type of call is called declarable. As soon as an EMS vehicle returns to a base, it will become available for dispatch to a new call.

Schematic overviews of the dispatch procedure

In the literature, there are two approaches to make a schematic overview of the EMS dispatch procedure. We will discuss both of them.

EMS dispatch involves four major components: Accidents, the dispatch center (DC), EMS vehicles and geographical information about the street plan including bases for EMS vehicles and hospital locations.

Henderson and Mason (2005) suggested a schematic model that contains three of the four major components mentioned above. In this type of schemes, geographical information is omitted because it only has influence on the choice of the EMS vehicle, hospital and station; it is implicit included in the other three components.

![Figure 2.1: The model as in Henderson and Mason (2005), page 8.](image-url)
2.2 General dispatch procedure

When we consider the EMS dispatch from a (approximate) dynamic programming point of view (see Section 2.8), the state of each EMS vehicle at a certain moment in time, is given by a tuple \( \mathbf{a} = (\sigma_a, o_a, d_a, t_a) \) where \( \sigma_a \) denotes the status of the EMS vehicle, \( o_a \) and \( d_a \) are the origin and destination locations and \( t_a \) is the starting time of its current activity. When an EMS vehicle is not moving, \( o_a \) equals \( d_a \). Also, the state of a call is given in a tuple \( \mathbf{c} = (\sigma_c, o_c, t_c, p_c) \) where \( \sigma_c \) denotes the state of the call, \( o_c \) is the location of the call, \( t_c \) is the time when the call was generated and \( p_c \) is the priority of the call. Now we can put the ambulances and calls into finite-dimensional vectors \( \mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_M) \) and \( \mathbf{C} = (\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_N) \), where \( M \) and \( N \) denote the number of ambulances and calls, respectively. An event \( e \) is the ‘reason’ why a decision should be made, such as ‘a call arrives’ or ‘an ambulance must be chosen’. Using notation \( \tau \) for the current time, and \( e \) for the last event we can denote the state of the system by \( s = (\tau, e, \mathbf{A}, \mathbf{C}) \).

When we need to take a decision we discribe the systems trajectory by

\[(s_1, s_2, s_3, \ldots)\]

where \( s_1 \) denote the current states and \( s_2, s_3, \ldots \) denote predictions for the future. We will limit ourselves to relocations within the time span \([0, T] \) for some fixed value \( T \).

<table>
<thead>
<tr>
<th>Accidents</th>
<th>Emergency Medicle Service Vehicle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can have the following states:</td>
<td>Can have the following states:</td>
</tr>
<tr>
<td>Assigned To EMC Vehicle</td>
<td>Idle At Base</td>
</tr>
<tr>
<td>Queued For Service</td>
<td>Returning To Base</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Events on which we make decisions:</td>
</tr>
<tr>
<td>Call arrives and is placed in the [th position</td>
</tr>
<tr>
<td>EMS vehicle departs for scene of call</td>
</tr>
<tr>
<td>EMS vehicle arrives at scene of call</td>
</tr>
<tr>
<td>EMS vehicle leaves scene of call for hospital</td>
</tr>
<tr>
<td>EMS vehicle arrives at hospital</td>
</tr>
<tr>
<td>EMS vehicle finishes at hospital</td>
</tr>
<tr>
<td>EMS vehicle arrives at base.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Serving At Scene Of Call</th>
<th>Transferring Patient To Hospital</th>
</tr>
</thead>
<tbody>
<tr>
<td>Going To Hospital</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2.2: States of the model as in Maxwell et al. (2009)

2.2.2 Call events that gets logged in The Netherlands

For the past four years, all EMS movements are logged into a computer system. To have accurate data, it is important that EMS personnel uses their mobilophone correctly. This is a mobile device with ten buttons that is on board every EMS vehicle and logs the exact time the button is pushed,
and communicates this with a central server. The data is also used by the GIS system such that EMS dispatchers can see the current state of each vehicle in real time.

The times as displayed are logged into a central data are displayed in Figure 2.3. These states provide a manner to model the EMS process, too. More information about the mobilophone can be found in Appendix B.

![Figure 2.3: Call events that are logged into the central database, and their corresponding mobilophone number.](image)

### 2.2.3 The response time in three parts

Most governments measure the effectiveness of EMS care in the percentage of calls having a lower response time than a certain given time. This response time can be divided in three parts that can be optimized seperately.

#### Duration handling time call dispatcher, and time to contact an EMS team.

At the moment somebody calls the national emergency number, the call handling time (or rather: duration handling time at the call dispatcher and time to contact an EMS team) starts. As said before, the dispatcher starts a triage procedure and contacts an EMS team on their pager. As soon as the message appears on the pagers, the call handling time ends. Note that the dispatcher may still be on the telephone to give first aid intructions to tide over until the EMS team arrives.

![Figure 2.4: The response time devided into three parts, see Boers et al. (2010).](image)

#### Clear out time

As soon as the EMS team gets the message on their pager, the clear out time starts. The team goes to the vehicle and prepares to depart. When they left the base and start riding, they give a status 1 on the EMS vehicle’s mobilophone. At the moment this button is pressed, the clear out time ends.
Driving time

At the moment that the EMS driver gives status 1 on the mobilophone, the driving time starts. When the EMS vehicle arrives at the scene, status 2 will be given and the driving time ends. By some authors driving time is called service time.

Response time

Adding the call handling time, clear out time and driving time gives the response time.

2.2.4 Road plan

In all EMS models it is important to know the travel time between two locations. It would be very hard or even impossible to make good operational decisions if we do not know the travel times for EMS vehicles. Altough driving times are probabilistic, all models we encounter consider them as deterministic.

In all models, the road plan is represented by a graph $G = (V, A)$. Here $V$ is the set of vertices and $A$ is the set of arcs. In most models the arcs represent the roads and the vertices represent the road intersections. In (later versions of) the simulation package TIFAR which gets developed in this thesis we shall choose to take 6 position postal codes instead, like 1101AA, as the set of vertices, see Subsection 3.2.2. The arcs will represent the shortest paths over the roads between any two vertices. In Section 2.4 we will discuss ways to perform shortest path calculations.

2.3 Current operation in The Netherlands

2.3.1 Nationwide

EMS transportation in The Netherlands is discribed in ‘Wet Ambulancevervoer’ (WAV) (and in the near future in ‘Wet Ambulancezorg’ (WAZ)), the law on EMS transportation. The country is split into 25 safety regions, each called a ‘Regionale Ambulancevoorziening’ (RAV), see Figure 2.5.

Each RAV has one ‘Meldkamer Ambulancezorg’ (MKA), which is the dispatch center (DC) that also functions as the emergency call center. Regions Amsterdam-Amstelland (13) and Zaanstreek-Waterland (11) that are merged into one RAV called Agglomeratie Amsterdam (RAV AA) are the only exceptions since they are controlled from one MKA.

MKA personnel coordinate all movements of EMS vehicles, and are entrusted with the triage of calls. A call originates from someone calling the emergency number or from a direct line with the police, firefighters or a hospital. In the majority of regions there is a joined dispatch center where dispatchers of the police, firefighters and EMS sit side by side; a so called ‘gecolokeerde meldkamer’. After the dispatcher deemed the call legitimate enough to send an EMS vehicle, a priority is attached. In The Netherlands we make a distinction between three levels of priority: A1, A2 and B.
Figure 2.5: RAVs in The Netherlands.
Let us take a closer look to each of the priority levels:

**A1** An urgent call with an acute threat to the patient’s life. Vital functions of the patient are not or rarely present, or cannot be determined through the telephone. The EMS vehicle uses optical and visual signals and tries to get to the call location as soon as possible. Examples: Heart attack, reanimation, or serious traffic accidents.

**A2** The patient’s life is not under direct threat, but there might be serious injuries. The EMS vehicle may use optical and visual signals if the EMS personnel has discussed this with the MKA, but this only happens on rare occasions. Examples: A broken leg or a general practitioner asks for transportation.

**B** A call without A1 or A2 priority in which the patient must be transported within a given predetermined time interval. A typical B call exists of transferring a seriously ill person from one hospital to another, because this hospital is specialized in the patient’s condition. When a seriously ill person receives transport from an EMS vehicle to his or her home, it will be classified as a B call as well.

A major difference between A1 and A2 calls on one hand and B calls on the other, is that A1 and A2 calls are not known at forehand, whilst B calls can be planned in advance.

Sometimes, if there is a shortage of EMS vehicles in the RAV, it is possible to contact nearby regions and ask them to provide additional vehicles. At both planning and dispatch is not common to rely on other regions since they seldom occur; in the year 2009 RAV Agglomeratie Amsterdam assisted with 516 A1 calls and 273 A2 calls to other RAVs, and got assistance for 172 A1 calls and 53 A2 calls from other RAVs, see Boers (2010) on page 74.

EMS vehicles departing to their own RAV from a delivery from a call with priority B into another RAV will also check in with the RAV they are currently in, so that this region can dispatch them if necessary.

By serious accidents and disasters, there will be a so called GRIP-code (Gecoördineerde Regionale Incidentbetrokkene Procedure) applied that gives information about the sincerity of the disaster. Then an authority called GHOR (Geneeskundige Hulpverlening bij Ongevallen en Rampen) will coordinate EMS care. In this thesis, we only consider the daily operation and not situations where GRIP-codes and GHOR are included.

An EMS vehicle is occupied by two persons: a driver and a nurse. Both followed a special training for EMS operation. If an accident seems to be too serious for one EMS team, the EMS team can contact the MKA and request help from a second EMS team or even from a MMT (mobile medical team; a helicopter). Usually only one EMS vehicle is sent to a call. By reanimations or drownings the MKA sends two vehicles. In this thesis we only consider the case that one EMS vehicle is sent to each call.

In The Netherlands, EMS vehicles have exemptions from traffic signs, as stated in article 91 of ‘Regelement verkeersregels en verkeerstekens’. This means that EMS vehicles can make use of footpaths, cycle paths, wrong-way driving, special highway exits and more, as long as they do not risk causing traffic accidents and if it is required for their duties. When an EMS vehicle uses optical and visual signs, it may ignore traffic lights (except at railways and bridges) and the speed limit. Regionally, there may be additional exemptions. When using or developing route planners for EMS, it is important to keep these differences in relation to other vehicles in mind, and include shortcuts if possible.
2.3.2 RAV Agglomeratie Amsterdam

Providers and MKA

In RAV Agglomeratie Amsterdam there are two EMS providers: Gemeentelijke of Gemeenschappelijke Gezondheidsdienst (GGD) and Verenigd Ziekenvervoer Amsterdam (VZA). The two providers work together in a cooperation (Dutch: stichting) called Regionale Ambulance Voorziening Agglomeratie Amsterdam (RAVA, not to confuse with RAV AA by which we denote the geographical region). The MKA is part of RAVA and coordinate the EMS vehicles movements for both EMS providers. The reason that RAV Agglomeratie Amsterdam consists of two safety regions is that region Zaanstreek-Waterland is rather small, and therefore is not required to have its own MKA.

There is an agreement between GGD and VZA about who covers which parts of the RAV: GGD covers the northern part, city center and south-eastern part of Amsterdam and VZA covers the rest of the RAV. If it contributes to the improvement of a response time, EMS vehicles will enter each other’s territory.

Bases

RAV Agglomeratie Amsterdam has nine documented bases where EMS vehicles be located while not attending a call, see Table 2.1.

<table>
<thead>
<tr>
<th>Nr.</th>
<th>Name</th>
<th>EMS provider</th>
<th>Inhabitants Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Valckenierstraat</td>
<td>GGD</td>
<td>179265</td>
</tr>
<tr>
<td></td>
<td>Valckenierstraat 9-21</td>
<td></td>
<td>1018XB000022</td>
</tr>
<tr>
<td>2.</td>
<td>Karperweg</td>
<td>VZA</td>
<td>179265</td>
</tr>
<tr>
<td></td>
<td>Karperweg 19-25</td>
<td></td>
<td>1075LB00019</td>
</tr>
<tr>
<td>3.</td>
<td>Zaandam</td>
<td>VZA</td>
<td>196344</td>
</tr>
<tr>
<td></td>
<td>Heijermansstraat 76</td>
<td></td>
<td>1502DR00076</td>
</tr>
<tr>
<td>4.</td>
<td>Purmerend</td>
<td>VZA</td>
<td>141981</td>
</tr>
<tr>
<td></td>
<td>Cantekoogweg 17 - 19</td>
<td></td>
<td>1442LG00017</td>
</tr>
<tr>
<td>5.</td>
<td>Aalsmeer</td>
<td>VZA</td>
<td>135965</td>
</tr>
<tr>
<td></td>
<td>Zwarteweg 77A</td>
<td></td>
<td>1431VJ00077</td>
</tr>
<tr>
<td>6.</td>
<td>Amstelveen</td>
<td>VZA</td>
<td>188389</td>
</tr>
<tr>
<td></td>
<td>Fokkerlaan 50</td>
<td></td>
<td>1185JC00050</td>
</tr>
<tr>
<td>7.</td>
<td>Academisch Medisch Centrum (AMC)</td>
<td>GGD</td>
<td>146579</td>
</tr>
<tr>
<td></td>
<td>Meibergdreef 9</td>
<td></td>
<td>1105AZ00009</td>
</tr>
<tr>
<td>8.</td>
<td>Monnickendam</td>
<td>VZA</td>
<td>Unknown</td>
</tr>
<tr>
<td></td>
<td>De Werf 1</td>
<td></td>
<td>1141HL00001</td>
</tr>
</tbody>
</table>

Table 2.1: Base locations in RAV AA, October 2008. Source: RIVM & NZa.

The source of the base addresses is Mulder (2009) from RIVM. The number of inhabitants living closest to a base is obtained from Nederlandse Zorgautoriteit (NZa, 2008). This number does not respect RAV-boundaries. The location of more (less often used) bases are known to the researcher but omitted due to possible concurrence implications. Later on, we will add two extra bases to see their influence. For now, we will only consider these nine bases.
2.3 Current operation in The Netherlands

Figure 2.6: Hospitals and bases in RAV Agglomeratie Amsterdam.

<table>
<thead>
<tr>
<th>Nr.</th>
<th>&lt; 4:00 min</th>
<th>&lt; 11:00 min</th>
<th>&lt; 22:24 min</th>
<th>Code</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>575</td>
<td>10 666</td>
<td>27 079</td>
<td>1018XG00002</td>
<td>Valckenierstraat</td>
</tr>
<tr>
<td>2.</td>
<td>536</td>
<td>8 016</td>
<td>27 331</td>
<td>1075LB00019</td>
<td>Karperweg</td>
</tr>
<tr>
<td>3.</td>
<td>367</td>
<td>5 765</td>
<td>29 264</td>
<td>1502DR00076</td>
<td>Zaandam</td>
</tr>
<tr>
<td>4.</td>
<td>207</td>
<td>2 300</td>
<td>11 571</td>
<td>1442LG00017</td>
<td>Purmerend</td>
</tr>
<tr>
<td>5.</td>
<td>236</td>
<td>1 772</td>
<td>14 824</td>
<td>1431VJ00077</td>
<td>Aalsmeer</td>
</tr>
<tr>
<td>6.</td>
<td>585</td>
<td>5 061</td>
<td>26 759</td>
<td>1185JC00050</td>
<td>Amstelveen</td>
</tr>
<tr>
<td>7.</td>
<td>4</td>
<td>2 139</td>
<td>23 858</td>
<td>1105AZ00009</td>
<td>AMC</td>
</tr>
<tr>
<td>8.</td>
<td>324</td>
<td>1 621</td>
<td>24 942</td>
<td>1141HL00001</td>
<td>Monnickendam</td>
</tr>
</tbody>
</table>

Table 2.2: The number of postal codes within reach of certain times.
Using the same route plan software as used in the simulation package TIFAR we can count the amount of 6 position postal codes that can be reached from a base within a certain amount of time, see Table 2.2. Explanation of the chosen times follows later in this subsection.

Hospitals

In RAV Agglomeratie Amsterdam there are eight hospitals, see Table 2.3. Each hospital has its own entrance for EMS vehicles, but only AMC hospital can station EMS vehicles. Later on, we will add another hospital that can act as a base as well. When a patient needs transport to a vehicle, the choice of a hospital will be via a triage protocol that we will discuss later on in this section.

<table>
<thead>
<tr>
<th>Nr.</th>
<th>Name</th>
<th>Address</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td><em>Academisch Medisch Centrum (AMC)</em></td>
<td>Meibergdreef 9</td>
<td>105AZZ00009</td>
</tr>
<tr>
<td>2.</td>
<td><em>Sint Lucas Andreas</em></td>
<td>Jan Tooropstraat 164</td>
<td>1061AE00164</td>
</tr>
<tr>
<td>3.</td>
<td><em>VU Medisch Centrum</em></td>
<td>De Boelelaan 1117</td>
<td>1081HV01117</td>
</tr>
<tr>
<td>4.</td>
<td><em>BovenIJ</em></td>
<td>Statenjachtstraat 1</td>
<td>1034CS00001</td>
</tr>
<tr>
<td>5.</td>
<td><em>Onze Lieve Vrouwe Gasthuis (OLVG)</em></td>
<td>Oosterpark 9</td>
<td>1091AC00009</td>
</tr>
<tr>
<td>6.</td>
<td><em>Zaans Medisch Centrum</em></td>
<td>Koning Julianaplein 58</td>
<td>1052DV00058</td>
</tr>
<tr>
<td>7.</td>
<td><em>Waterlandziekenhuis</em></td>
<td>Waterlandlaan 250</td>
<td>1441RN0250</td>
</tr>
<tr>
<td>9.</td>
<td><em>Slotervaart Ziekenhuis</em></td>
<td>Louwesweg 6</td>
<td>1066EC00006</td>
</tr>
</tbody>
</table>

Table 2.3: Hospitals in RAV Agglomeratie Amsterdam, October 2010.

Road plan and additional exemptions

The city of Amsterdam differs in a couple of aspects from the rest of the country. Amsterdam is part of a large densely populated region called Ring City, or ‘Randstad’ in Dutch. This region has to cope a lot with traffic congestion. The A10 motorway is a ring road around the city of Amsterdam that has a hard shoulder over almost the entire length. With visual and auditory signals EMS vehicles may drive at a speed of 20kph above the speed of other vehicles, if the other vehicles drive between 30 kph and 80kph. However, if the speed of other vehicles is below the 30kph, EMS vehicles on a shoulder may not drive at speeds above 50kph. When not driving on a hard shoulder, the maximal speed of EMS vehicles is at most 40kph above the regular traffic. See ‘richtlijn 5.3-5.5’ in AZN and V&VN (2009).

The road plan of Amsterdam contains dedicated tracks for public transportation like trams and busses. EMS vehicles have an exemption to use them. Using these tracks provides a better flow of EMS traffic,
in particular when car tracks suffer from congestion.

The RAV is separated in two parts by the waters Het IJ and Noordzeekanaal. There are three tunnels (of which two are part of the A10 motorway) and one bridge to cross these waters. In case of emergency it is possible for EMS personnel to contact traffic control, which clears a tunnel lane from other vehicles by displaying red crosses on matrix signs above the road. It is also possible for EMS personnel to request bridges to be available for them.

Concluding, we can say that the RAV has to cope with traffic congestion, but emergency medical services have means to reach their destination by using hard shoulders, public transportation tracks and claiming lanes by contacting traffic control.

Demographical statistics per municipality

A satellite image of RAV AA with an overlay containing all municipalities is displayed in Figure 2.7. In Table 2.4 the demographic statistics of inhabitants of RAV Agglomeratie Amsterdam is displayed per municipality. Municipalities with only a few inhabitants per km² are rural (like Zeevang, Beemster, Waterland, Landsmeer and Ouder-Amstel) whilst municipalities with many inhabitants per km² are urban (like Amsterdam, Purmerend, Diemen, Zaanstad and Amstelveen). In 2009, The Netherlands had 16485787 inhabitants. RAV Agglomeratie Amsterdam provides EMS care for 7.56% of the Dutch population. All demographical data is obtained from CBS (2009).

<table>
<thead>
<tr>
<th>Municipality</th>
<th>Inhabitants</th>
<th>Inhabitants per km²</th>
<th>Total inhabitants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zeevang</td>
<td>6297</td>
<td>165</td>
<td></td>
</tr>
<tr>
<td>Beemster</td>
<td>8564</td>
<td>121</td>
<td></td>
</tr>
<tr>
<td>Wormerland</td>
<td>15900</td>
<td>410</td>
<td></td>
</tr>
<tr>
<td>Purmerend</td>
<td>78862</td>
<td>3364</td>
<td></td>
</tr>
<tr>
<td>Edam-Volendam</td>
<td>28483</td>
<td>1748</td>
<td>318455</td>
</tr>
<tr>
<td>Zaanstad</td>
<td>144055</td>
<td>1950</td>
<td></td>
</tr>
<tr>
<td>Oostzaan</td>
<td>9201</td>
<td>798</td>
<td></td>
</tr>
<tr>
<td>Landsmeer</td>
<td>10139</td>
<td>450</td>
<td></td>
</tr>
<tr>
<td>Waterland</td>
<td>16954</td>
<td>325</td>
<td></td>
</tr>
<tr>
<td><strong>Amsterdam-Amstelland</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Amsterdam</td>
<td>755605</td>
<td>4554</td>
<td></td>
</tr>
<tr>
<td>Diemen</td>
<td>24361</td>
<td>2033</td>
<td></td>
</tr>
<tr>
<td>Ouder-Amstel</td>
<td>13107</td>
<td>543</td>
<td></td>
</tr>
<tr>
<td>Amstelveen</td>
<td>79768</td>
<td>1924</td>
<td></td>
</tr>
<tr>
<td>Aalsmeer</td>
<td>28006</td>
<td>1367</td>
<td></td>
</tr>
<tr>
<td>Uithoorn</td>
<td>27660</td>
<td>1512</td>
<td></td>
</tr>
<tr>
<td><strong>Total:</strong></td>
<td><strong>1246872</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.4: Demographical Statistics per Municipality, 2009.

Choice of hospital by EMS personnel

If an EMS team has determined that a patient requires transportation to a hospital, a triage protocol is followed to determine the choice of the hospital. This triage protocol is described in Groen (2009).
In Figure 2.8 a summarized version is displayed: some medical terms are omitted or translated into non-medical jargon.

National protocols for EMS drivers and EMS nurses can be found in ‘Landelijk Protocol Ambulancezorg 7.1’ (2009), and protocols for EMS dispatchers in The Netherlands can be found in ‘Landelijke Standaard Meldkamer Ambulancezorg’ (2009).
2.3 Current operation in The Netherlands

EMS statistics in RAV Agglomeratie Amsterdam

EMS statistics over the year 2009 can be found in ‘Ambulances In Zicht’ by Boers et al. (2010). Additional EMS statistics for RAV Agglomeratie Amsterdam are obtained from Călinescu (2009) and Zuidhof (2010). We will discuss information that might help us in modeling EMS care for the region of Amsterdam.

Recall from Figure 2.4 that the response time can be divided into three parts. Most statistics are divided in these three parts.

Figure 2.8: Triage scheme for the choice of hospital.

---

EMS statistics in RAV Agglomeratie Amsterdam

EMS statistics over the year 2009 can be found in ‘Ambulances In Zicht’ by Boers et al. (2010). Additional EMS statistics for RAV Agglomeratie Amsterdam are obtained from Călinescu (2009) and Zuidhof (2010). We will discuss information that might help us in modeling EMS care for the region of Amsterdam.

Recall from Figure 2.4 that the response time can be divided into three parts. Most statistics are divided in these three parts.

---

EMS statistics in RAV Agglomeratie Amsterdam

EMS statistics over the year 2009 can be found in ‘Ambulances In Zicht’ by Boers et al. (2010). Additional EMS statistics for RAV Agglomeratie Amsterdam are obtained from Călinescu (2009) and Zuidhof (2010). We will discuss information that might help us in modeling EMS care for the region of Amsterdam.

Recall from Figure 2.4 that the response time can be divided into three parts. Most statistics are divided in these three parts.
A1 calls
In Table 2.5 we can see statistics how the mean response time for A1 calls is split for RAV Agglomeratie Amsterdam.

<table>
<thead>
<tr>
<th>Mean RAV AA</th>
<th>Mean Country</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duration handling time</td>
<td>2:44</td>
<td>1:52</td>
</tr>
<tr>
<td>Clear out time</td>
<td>1:16</td>
<td>1:09</td>
</tr>
<tr>
<td>Driving time</td>
<td>5:58</td>
<td>6:42</td>
</tr>
<tr>
<td>Response time</td>
<td>10:04</td>
<td>9:44</td>
</tr>
</tbody>
</table>

Table 2.5: A1 priority statistics, 2009. Time in minutes.

When adding the three times we can notice a difference of six seconds in the mean of RAV AA and one second difference in the country mean. The one second difference might be explained by a rounding error. For the six seconds difference I do not have an explanation.

We see that the mean response time of RAV Agglomeratie Amsterdam is 20 seconds higher than the mean response time of the country. The duration of the handling time of the call dispatcher in Amsterdam is highly above average. The reason EMS dispatchers in Amsterdam give is that people in urban areas call much sooner to the emergency number than the more sober-minded people in rural areas, which makes the triage procedure on the telephone much longer.

Boers et al. (2010) notice that urban areas and areas with a proactive dynamic EMS management have a lower clear out time, because vehicles that are already on the road (or rather: mobile readiness) do not have a clear out time.

On the other hand, the driving times in the city of Amsterdam are highly under the mean, because there are a lot of EMS vehicles per square kilometer. Because there is cancelation between long handling time of the call dispatcher and short driving time, the response time is close to the mean response time of the country.

<table>
<thead>
<tr>
<th>&lt;12:00</th>
<th>&lt;14:00</th>
<th>&lt;15:00</th>
<th>&lt;16:00</th>
<th>&lt;18:00</th>
<th>&lt;20:00</th>
<th>&lt;21:00</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAV AA</td>
<td>76%</td>
<td>88%</td>
<td>91.7%</td>
<td>94%</td>
<td>97%</td>
<td>98%</td>
</tr>
<tr>
<td>Country</td>
<td>76%</td>
<td>88%</td>
<td>92.0%</td>
<td>94%</td>
<td>97%</td>
<td>98%</td>
</tr>
</tbody>
</table>

Table 2.6: Percentage in time for A1 calls, 2009. Time in minutes.

In Table 2.6 we can see statistics how the mean response time for A1 calls is split for RAV Agglomeratie Amsterdam. The source of the data is Boers et al. (2010), pages 30 and 38.
A2 calls
In Table 2.7 we can see A2-statistics for RAV Agglomeratie Amsterdam.

<table>
<thead>
<tr>
<th></th>
<th>Mean RAV AA</th>
<th>Mean Country</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duration handling time</td>
<td>5:24</td>
<td>3:22</td>
<td>Boers et al. (2010, p. 42)</td>
</tr>
<tr>
<td>call dispatcher, and time to contact an EMS team.</td>
<td>2:12</td>
<td>1:39</td>
<td>Boers et al. (2010, p. 44)</td>
</tr>
<tr>
<td>Clear out time</td>
<td>11:22</td>
<td>10:51</td>
<td>Boers et al. (2010, p. 46)</td>
</tr>
<tr>
<td>Response time</td>
<td>19:53</td>
<td>16:15</td>
<td>Boers et al. (2010, p. 48)</td>
</tr>
</tbody>
</table>

Table 2.7: A2 priority statistics, 2009. Time in minutes.

We see similar trends as by the A1 calls. The handling time for the call dispatcher is much higher than the mean in The Netherlands, and we can say the same for the clear out time.

The driving time, on the other hand, is higher than the mean of the country. This was not the case by A1 calls. A reason that can be found is that the EMS vehicles may not use hard shoulders, have to wait for traffic lights and may not exceed the speed limit. With these constraints, EMS vehicles have the burden of traffic congestion. The cancelation effect we mentioned for A1 calls will not occur, and the response time is much higher than the country’s mean.

<table>
<thead>
<tr>
<th></th>
<th>&lt;20:00</th>
<th>&lt;25:00</th>
<th>&lt;28:00</th>
<th>&lt;30:00</th>
<th>&lt;32:00</th>
<th>&lt;35:00</th>
<th>&lt;41:00</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAV AA</td>
<td>59%</td>
<td>78%</td>
<td>85.7%</td>
<td>87.8%</td>
<td>90%</td>
<td>93%</td>
<td>97%</td>
</tr>
<tr>
<td>Country</td>
<td>76%</td>
<td>89%</td>
<td>93.0%</td>
<td>94.5%</td>
<td>96%</td>
<td>97%</td>
<td>99%</td>
</tr>
</tbody>
</table>

Table 2.8: Percentage in time for A2 calls, 2009. Time in minutes.

In Figure 2.8 we see that for all time constraints, RAV Agglomeratie Amsterdam scores under the countries mean. This data is gained from Boers et al. (2010), pages 50 and 52.

B calls
There are not much statistics known about the EMS statistics of B calls because there is no information available in Boers et al. (2010). For the GGD data we know from Zuidhof (2010) that the average amount of 34.77 B calls is served each day with a standard deviation of 8.53. The mean time an EMS vehicle is not available for dispatch because it is serving a B call, is 76.8 minutes. Since the median is 63.7 minutes, a normal distribution probably will not fit for simulation B calls, she states. This data is only from GGD vehicles that serve in the city of Amsterdam, and therefore is not likely to be representable for the entire RAV. The mean in the year 2008 for RAV Agglomeratie Amsterdam is 70.1 minutes, see Kommer (2008) on page 22. He does not state any deviations.

<table>
<thead>
<tr>
<th></th>
<th>A1 calls</th>
<th>A2 calls</th>
<th>B calls</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAV AA (absolute)</td>
<td>53 293</td>
<td>10 608</td>
<td>36 930</td>
<td>100 831</td>
</tr>
<tr>
<td>RAV AA (relative)</td>
<td>52.9%</td>
<td>10.5%</td>
<td>36.6%</td>
<td>100%</td>
</tr>
<tr>
<td>Country (absolute)</td>
<td>454 309</td>
<td>239 572</td>
<td>348 085</td>
<td>1 041 966</td>
</tr>
<tr>
<td>Country (relative)</td>
<td>43.6%</td>
<td>23.0%</td>
<td>33.4%</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 2.9: Amount of calls in RAV Agglomeratie Amsterdam and The Netherlands.
The amount of calls
The amount of calls for both RAV AA and the country’s average can be found in Table 2.9. It is not hard to see that 9.7% of all calls in the country are in RAV Agglomeratie Amsterdam. In this RAV, dispatchers give sooner a priority A1 label to a call. The higher percentage of B-calls can be explained by two academical hospitals (AMC and VUMC) and the AVL cancer hospital (excluded from the hospital list because it doesn’t handle regular patients, and the AVL building is next to the Slotervaart Ziekenhuis). If we omit all B-calls from the statistics of Table 2.9, we get Table 2.10.

<table>
<thead>
<tr>
<th></th>
<th>A1 calls</th>
<th>A2 calls</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAV AA (absolute)</td>
<td>53 293</td>
<td>10 608</td>
<td>63 901</td>
</tr>
<tr>
<td>RAV AA (relative)</td>
<td>83.4%</td>
<td>16.6%</td>
<td>100%</td>
</tr>
<tr>
<td>Country (absolute)</td>
<td>454 309</td>
<td>239 572</td>
<td>693 881</td>
</tr>
<tr>
<td>Country (relative)</td>
<td>65.5%</td>
<td>34.5%</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 2.10: Amount of calls in RAV Agglomeratie Amsterdam and The Netherlands without B-calls.

The data for Table 2.9 is obtained from Boers et al. (2010), page 58.

Declarable calls, EHGV-calls and Loss calls
Each call can be classified as a declarable call, EHGV call or loss call. An EHGV-call is a type of call where the EMS team can provide care locally, and the patient does not require transportation to a hospital. This status is determined by EMS personnel at the scene. Sometimes an EMS team arrives at the scene and misses the presence of a patient to take care of. This might be the result of a patient who left after the call was made, a patient who is not ready for transportation (B-calls), or a prank call to the emergency number. This kind of call can be classified as a loss call. Calls that are not EHGV or loss are classified as declarable. In the case a call is declarable a patient will be brought to the hospital. Statistics for declarable, EHGV and loss calls can be found in Table 2.11.

<table>
<thead>
<tr>
<th></th>
<th>Declarable</th>
<th>EHGV-calls</th>
<th>Loss calls</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAV AA (absolute)</td>
<td>74 470</td>
<td>17 084</td>
<td>8 827</td>
<td>100 831</td>
</tr>
<tr>
<td>RAV AA (relative)</td>
<td>74.3%</td>
<td>16.9%</td>
<td>8.8%</td>
<td>100%</td>
</tr>
<tr>
<td>Country (absolute)</td>
<td>810 015</td>
<td>183 571</td>
<td>48 380</td>
<td>1 041 966</td>
</tr>
<tr>
<td>Country (relative)</td>
<td>77.7%</td>
<td>17.6%</td>
<td>4.6%</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 2.11: Classes of calls in RAV Agglomeratie Amsterdam and The Netherlands.

Conclusions
In this section we have discussed the situation in The Netherlands and RAV Agglomeratie Amsterdam in general. The RAV has 1 246 872 inhabitants, nine hospitals and eight EMS bases. Of the 100 831 calls that were made, 52.9% were priority A1 and 10.5% were A2. We will use the data that is discussed as input for our modeling package that we develop.

The next section discusses way to perform route planning operations, and the section after that utilizes that information in EMS models.
2.4 Shortest Path Problems

2.4.1 Introduction

During an overall algorithm for EMS modeling, it is paramount to be able to acquire the distance between any two points in the RAV. We model the street plan as a weighted directed graph \( G = (\mathcal{V}, \mathcal{A}) \) where \( \mathcal{V} \) denotes the vertices and \( \mathcal{A} \) the arcs. An undirected edge (i.e. a road where we can travel in two directions) will be represented by two directed arcs. This distinction can come in handy when the speed in both directions differ. For the amount of vertices and arcs we assume \( 1 \leq |\mathcal{V}| < \infty \) and \( 0 \leq |\mathcal{A}| < \infty \). We will define \( n := |\mathcal{V}| \). Furthermore we assume \( G \) to be connected. For each arc \( \{i, j\} \in \mathcal{A} \) we denote the arc weight with \( d[i, j] \). For each pair of vertices \( i, j \in \mathcal{V} \) we will denote the length of the shortest path from \( i \) to \( j \) by \( t[i, j] \).

Since the performance of EMS care is measured in time latency, the distance between two vertices is measured in time instead of the physical distance between these two vertices. As a result, the arc lengths are positive, just like they would be when measured in physical distance. This implies that there are no negative cycles in the graph. We shall need this fact later on when discussing shortest path algorithms. The travel time \( t[i, j] \) is assumed to be time-independent. For solutions for time-dependent vehicular movements, see Horn (2000).

There are two possibilities during an overall algorithm where shortest distance calculations can be made: in the pre-processing or on demand during the processing. Pre-processing algorithms will calculate the shortest distance between every two vertices and store these travel times (and a way to reconstruct the path) in the computer’s memory. The advantage is that these calculations are only needed once, and shortest paths are quickly available when they are needed later on. A drawback is that the pre-processing is very time consuming and requires a huge amount of storage space (\( O(n^2) \) bytes, as can be easily seen). On the other hand, the use of on-demand algorithms will have a shorter pre-processing time and require less storage space, but are slower than a lookup of a value in a table produced in the pre-processing since a lot of shortest path calculations must be done during an overall algorithm.

In this section we will discuss two all-to-all shortest path algorithms for pre-processing (Floyd-Warshall and Johnson’s algorithm) and two algorithms that will do shortest-path calculations on demand during the processing (GSP and A*).

2.4.2 Pre-processing algorithms

As mentioned in Section 2.4.1, pre-processing algorithms will calculate the shortest path between every pair of vertices. This type of algorithm is also called all-to-all shortest path. First we will study the Floyd-Warshall algorithm, followed by a discussion about the Johnson algorithm.

Floyd-Warshall

The original version of the Floyd-Warshall algorithm can be found in Floyd (1962). For routing purposes we will adapt the algorithm slightly, since we do need to reconstruct the shortest path if we want to compute the location of an EMS vehicle at a certain moment in time. We introduce a new variable \( m[i, j] \) that stores the (temporary) best found length of a path from \( i \in \mathcal{V} \) to \( j \in \mathcal{V} \). During the iterations, \( m[i, j] \) will converge to the length of the shortest path from \( i \) to \( j \), and \( m[i, j] \) will finally equal this length.
Initialization
Initially we set \( m[i, j] = t[i, j] \) if there is a direct link from vertex \( i \) to vertex \( j \), and we set \( m[i, i] = 0 \) for each \( i \in V \). If arc \((i, j)\) ∈ \( A \) does not exist, we add arc \((i, j)\) and set \( m[i, j] \) initially to infinity. At completion, \( m[i, j] = t[i, j] \), the length of the shortest path \( t[i, j] \) from \( i \) to \( j \). The requirement that Floyd-Warshall does not allow negative cycles is fulfilled in the road plan as stated in Section 2.4.1.

The variable \( v[i, j] \) denotes the label of the predecessor of \( j \) on the path from \( i \) to \( j \). We need this information for path reconstruction. At initialization we will set \( v[i, j] = -1 \).

Algorithm
The algorithm runs as follow:

1. \textbf{For} \( i = 1 \) to \( n \)
2. \textbf{For} \( j = 1 \) to \( n \)
3. \textbf{If} \( m[j, i] < \infty \)
4. \textbf{For} \( k = 1 \) to \( n \)
5. \( u := m[j, i] + m[i, k] \)
6. \textbf{If} \( u \leq m[j, k] \)
7. \( m[j, k] := u \)
8. \( v[j, k] := i \)

\textbf{Algorithm 2.1:} The Floyd-Warshall algorithm.

Performance
It is easy to see that Floyd-Warshall has performance \( O(n^3) \) since it consists of three nested loops with \( n \) steps each.

Correctness & Necessity
Since the only proof that is found by us is for a slightly adapted version of Floyd-Warshall we will provide a proof of correctness. We will prove that the algorithm finds the shortest path from an arbitrary vertex \( s \in V \) to any other arbitrary vertex \( t \in V, s \neq t \). Since \( s \) and \( t \) are chosen arbitrary we will prove that the algorithm will give an all-to-all shortest path.

In Lines 2 – 6 of the algorithm we check for all choices of \( j, k \in V \) whether the triangle inequality holds while keeping \( i \) fixed, i.e.

\[
m[j, k] \leq m[j, i] + m[i, k].
\]

This is only necessary if \( m[j, i] \) is not infinity (Line 3). When the triangle inequality does not hold this means a detour from \( j \) via \( i \) to \( k \) is faster than taking the direct arc, hence we update the length between \( j \) and \( k \) and store that we take a detour via vertex \( i \). Even if the triangle inequality holds with equality, which means \( j, k \) and \( i \) are on a line, we store that the shortest path from \( j \) to \( k \) goes via \( i \), which we shall use for path reconstruction. The only way we do not store \( i \) as a predecessor is when the triangle inequality does not hold, and in this case the shortest path from \( j \) to \( k \) does not involve \( i \) at all. For an illustration, see Figure 2.9.

Now note that if \( i \neq s \) and \( i \neq t \) we have the following situation. Suppose we have done the iteration steps from Lines 2 – 8 for the fixed vertex \( i \). Since the connection between any two points \( j \) and \( k \), such that a path \( j \rightarrow i \rightarrow k \) exists, contains all shortest path information that vertex \( i \)
provides, vertex $i$ does not provide added value to the shortest path calculation any more, see Figure 2.9 for an illustration of this principle. Thus after this iteration we can consider the graph $G^1 = (V^1 := V \setminus \{i\}, A^1 := A \setminus \{\text{arcs incident to } i\})$, choose a vertex $i' \in V^1 \setminus \{s, t\}$, and run Lines 2 – 8 to ‘eliminate’ vertex $i'$ from the graph (with preservation of its information). We then get $G^2 = (V^2 := V \setminus \{i, i'\}, A^2 := A \setminus \{\text{arcs incident to } i \text{ or } i'\})$. Using induction, after $n - 2$ steps we get graph $G^{n-2} = (V^{n-2} = \{s, t\}, A^{n-2} := \{(s, t)\})$, which gives us the length of the shortest path $t[s, t] = m[s, t]$.

**Figure 2.9:** An example graph to demonstrate the elimination of a vertex.
In the case that \( i = s \) or \( i = t \) we see that running Lines 2 – 8 will not do any harm, since a shortest path from \( s \) to \( t \) will not visit \( s \) or \( t \), except as starting point or end point. (Since \( G \) represents a road network and hence have no negative cycles, both \( j \to s \to k \) and \( j \to t \to k \) will not occur.) Also, adding \( i, i', i'', \ldots \) back to the graphs \( G^1, G^2, G^3, \ldots \) will not affect the outcome of the iteration to find the shortest path between \( s \) and \( t \). For the shortest path between \( s \) and \( t \) they do not have any added value, but for any other choice of \( s \) and \( t \) they do, and we do include them, which results in the algorithm described before.

Since \( s \) and \( t \) are chosen arbitrarily, the algorithm finds the shortest path between any two vertices. The existence of a shortest path is guaranteed by the assumption that the graph is connected, which implies that \( [m[i,j]]_{i,j} \) becomes a complete graph. Notice that the completeness of the graph with arc weights \( [m[i,j]]_{i,j} = [t[i,j]]_{i,j} \) when the stop condition has been reached gives us the Necessity of the proof.

\[ \square \]

Path reconstruction

The path reconstruction goes via the recursive function which is described below in pseudocode. The path consists of an ordered array of adjacent vertices one has to visit to reach \( t \). The proof follows from the correctness of the algorithm that is proven earlier in this section.

1. path getShortestPath (s, t)
2. If \( v[s, t] = -1 \)
3. Return path \( t \).
4. Else
5. Vertex \( via := v[s, t] \).
6. Return path getShortestPath(s, via) + getShortestPath(via, t).

Algorithm 2.2: Path reconstruction for Floyd-Warshall.

Algorithm 2.4.2 can also be found under the name printpath in Cormen (1990), page 476.

Johnson’s Algorithm

Another well-known algorithm to compute the shortest path between every two vertices in a weighted directed graph without negative cycles is Johnson’s algorithm, as described in Johnson (1977). It works particularly well on sparse graphs. A sparse graph is a graph in which the number of arcs is much smaller than the number of arcs in a complete graph on the same vertex set. A realistic road map is an example of a sparse graph.

Johnson’s algorithm solves the problem in \( O(|V|^2 \log |V| + |V||A|) \) time, see Black (2004). This is slightly better than Floyd’s algorithm.
The algorithm works as follows, see Black (2004) as well:

- Add a new vertex with zero weight arcs to every other vertex.
- Run the Bellman-Ford algorithm to check for negative weight cycles and determine $g(v)$, the least weight of a path from the new vertex to vertex $v$.
- Reweigh the arcs using the vertex’ $g(v)$ values.
- Finally for each vertex, run Dijkstra’s algorithm and store the computed least weight to other vertices, reweighed using the vertices’ $g(v)$ values, as the final weight.

The algorithm of Bellman-Ford is described in R. Bellman (1958) and Section 2.4.1 of Bertsekas (1998). Bellman-Ford runs in $O(|V||A|)$ time complexity. Since a road network does not have negative cycles, we do not have to check for them.

Both Floyd-Warshall’s and Johnson’s algorithm are used to calculate the shortest path between every pair of vertices. The algorithms can be used in the pre-processing. Their performance in time is $O(|V|^3)$ and $O(|V|^2 \log |V| + |V||A|)$, respectively.

In Section 2.4.3 we will discuss two algorithms that calculate the shortest path between two given vertices.

### 2.4.3 Shortest-Path calculations

In Section 2.4.1 we considered two possibilities in the overall algorithm where we could do shortest path calculations: During pre-processing (see Section 2.4.2) or on demand while processing the overall algorithm. Now we will discuss the latter. This type of algorithms calculate the shortest path from one vertex to one other vertex; a so called one-to-one shortest path algorithm.

Two algorithms that can be used for this purpose are Generic Shortest Paths (GSP) and A*. There are other algorithms like the already mentioned Bellman-Ford, d’Esopo-Pape (Pape (1974)) and Pallottino, and it can even be proven that these are special cases of GSP including Dijkstra (Dijkstra (1959)), but we will limit ourself to GSP and A*.

### Generic Shortest Paths (GSP)

The Generic Shortest Paths algorithms, see Bertsekas (1998) and Klunder and Post (2006), is a way to calculate the shortest path between two vertices $s, t \in V$. This generic algorithm maintains a list of vertices $C$ that is called the candidate list, and a vector of labels $(c_1, c_2, \ldots, c_n)$, where each $c_j$ is either a real number or infinity. Furthermore, $v_j := v(j)$ denotes the predecessor of vertex $j$ on the path from $s$ to $j$. The algorithm only works if there are no negative cycles, which holds for real road networks as discussed in Section 2.4.1.

**Initialisation**

Initially we let the candidate list be the singleton containing the starting vertex, i.e. $C = \{s\}$. For this starting vertex we set travel distance 0 - from the starting point to the starting point has distance 0 - and as predecessor we take a dummy vertex with label $-1$. Mathematically, we write $d_s = 0$ and $v_s = -1$. For all other vertices $j \in V \setminus \{s\}$ we set distance $d_j = \infty$. 
Algorithm
The algorithm runs as follow:

1. While $C$ is not empty
2. Choose a pivot vertex $i$ from $C$.
3. Delete $i$ from $C$.
4. For each outgoing arc $(i, j) \in A$
5. If $d_j > d_i + d[i, j]$
6. $d_j := d_i + d[i, j]$
7. $v_j := i$
8. If $j \notin C$
9. Add $j$ to $C$.

Algorithm 2.3: The GSP algorithm.

Correctness & Necessity
The proof of both correctness and necessity can be found in Bertsekas (1998), Proposition 2.2b. This proposition states that at termination, the algorithm has found the shortest path from $s$ to every other vertex. In particular, the shortest path from $s$ to $t$ will be found. The GSP has an exponential run time complexity - even with no negative cycles, see Mehlhorn and Sanders (2008).

Path Reconstruction
Let $\ell$ denote the number of arcs on the shortest path from $s$ to $t$. If $t[i, j] < \infty$, which must hold since we have a connected graph by assumption, the shortest path is given by

$$s = v^\ell(t) \rightarrow v^{\ell-1}(t) \rightarrow \cdots \rightarrow v^2(t) \rightarrow v(t) \rightarrow t$$

where $v^r(t) := (v \circ v \circ \cdots \circ v)(t)$.

Proof: The algorithm finds the length of the shortest path from $s$ to each other vertex in the graph. Vertex $v(t)$ is the predecessor of $t$ on the shortest path from $s$ to $t$. (Please note: $v(t)$ and $t$ are adjacent vertices on this shortest path). Also, $v(t)$ has optimal length to $s$, so the shortest path from $s$ to $t$ has $v(t)$ and $(v \circ v)(t)$ as the two last vertices visit on the shortest path. Using induction, we see that the path reconstruction finds the shortest path.

GSP in relation to Dijkstra’s Algorithm
Dijkstra’s algorithm is a special case of the GSP. Dijkstra chooses an element from $\{i \in C | d_i \leq d_j, \forall j \in C\}$ as pivot vertex. Due to this choice, a vertex will not come back in $C$ once it is removed. Dijkstra runs in $O(|V|^2)$ time complexity, which can be reduced to $O(|A| + |V| \log |V|)$ using data structures like a binary heap, see Johnson (1972) and Barbehenn (1998).
2.4 Shortest Path Problems

A* Algorithm

The A* algorithm (pronounced as ‘A star’) is another special case of the GSP algorithm, that finds the shortest path from \( s \in V \) to \( t \in V \), see Klunder and Post (2006). It uses a heuristic \( h_i \) that estimates an lower bound on the distance in time from \( i \) to \( t \). One might want to choose the Euclidean distance from \( s \) to \( t \) divided by the maximal speed in the area between \( s \) and \( t \) for \( h_i \), typically 120 kph in The Netherlands but perhaps higher for EMS vehicles.

During the execution of the algorithm we choose an element from \( \{ i \in C \mid d_i + h_i \leq d_j + h_j, \forall j \in C \} \) as pivot vertex. One can think of A* as a special case of Dijkstra’s algorithm, see Theorem 1 from Ikeda et al. (1994).

Due to this heuristic, A* will search towards \( t \) instead of in a circular form around \( s \). The Generic Shortest Path algorithm will only terminate when the candidate list is empty. Therefore, a valid question rises whether using a so called dual feasible heuristic is really better than Dijkstra. Fortunately, one can prove something significant using this specific heuristic: once a vertex is removed from the candidate list, it will never enter it again. (See Hart (1972) for the proof). This means that we can abort the algorithm once \( t \) has left the candidate list \( C \), and in practice we see that A* gets terminated after less iterations than Dijkstra.

Though the complexity in time for both Dijkstra’s algorithm and the A* algorithm are the same, one can see in practice that the A* algorithm performs better.

Bidirectional Search

A bidirectional algorithm searches from both the start point \( s \in V \) to the end point \( t \in V \), and from the end point \( t \) to the start point \( s \). Therefore, it has two candidate lists \( C_s \) which holds the candidates from \( s \) to \( t \) and \( C_t \) which holds the candidates from \( t \) to \( s \). The two heuristics induced by the two directions are denoted by \( h_s \) and \( h_t \). Unfortunately we can not terminate the algorithm at the moment the two searched areas meet due to the fact that A* does not label the vertices permanently in order of their distance to the origin.

Ikeda et al. (1994) proved that A* becomes equivalent to Dijkstra if we modify arc lengths according to \( d'[i,j] := d[i,j] + h_s^i - h_s^j \). If we define \( d''[i,j] := d[i,j] + h_t^i - h_t^j \) for searches in the backwards direction, and take their average, we get arc lengths \( \bar{d}[i,j] = \frac{1}{2}(d'[i,j] + d''[i,j]) \). Because of this transformation we turn the label correcting A* into the label setting Dijkstra algorithm. In a label setting algorithm a label can be marked as permanent. Let us denote by \( R^s \) and \( R^t \) the sets of nodes which labels are marked as permanent.

For label setting algorithms we know the following: If \( u \in R^s \cap R^t \), the distance from \( s \) to \( t \) is given by

\[
\min_{i \in R^s, j \in R^t, (i,j) \in A} \{d_s^i + \bar{d}[i,j] + d_t^j, d_t^u + d_s^u \}
\]

See Klunder and Post (2006) for a proof. Thus if a node is permanently labeled from both directions, we only have to examine all possible paths that connects the sets \( R^s \) and \( R^t \) with at most one arc.
2.4.4 Section conclusions

During the planning of EMS movements it is important to know the distance in time between two locations. For this purpose we have modeled the road system by a graph, where the arc weights represent the travel times.

In an overall algorithm used to model the EMS planning, we can do shortest-path calculations at two points: we can either choose to calculate the shortest paths between every two vertices of the graph in the pre-processing, and store them in memory (this requires much storage, results in a (very) long pre-processing time, but shortest path information can be accessed very fast during the runtime of the overall algorithm), or we can choose to calculate shortest path on demand at processing (which gives us almost no pre-processing time and uses less storage, but it takes more time to acquire shortest path information while running the overall algorithm).

A schematic overview with the algorithms that we have discussed can be found in Table 2.12. Using a bidirectional search we can make the searched area smaller, and thereby making an algorithm faster in practice.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Processing</th>
<th>Type</th>
<th>Time complexity</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Floyd-Warshall</td>
<td>Pre-processing</td>
<td>All-to-all shortest path</td>
<td>$O(</td>
<td>V</td>
</tr>
<tr>
<td>Johnson’s</td>
<td>Pre-processing</td>
<td>All-to-all shortest path</td>
<td>$O(</td>
<td>V</td>
</tr>
<tr>
<td>GSP</td>
<td>Processing</td>
<td>One to all-shortest path</td>
<td>Exponential run time complexity in general.</td>
<td>Better for specific implementations.</td>
</tr>
<tr>
<td>A*</td>
<td>Processing</td>
<td>One to all-shortest path</td>
<td>$O(</td>
<td>A</td>
</tr>
</tbody>
</table>

Table 2.12: The algorithms that we have discussed in this section.
2.5 Mathematical Models

2.5.1 Introduction

The purpose of EMS care has two sides: first of all to save as many lives as possible, and secondly to provide a safe means of transportation for seriously ill people to be quickly available. An overall underlying assumption made in existing literature is that quicker response times result in better EMS care.

In Section 2.2.3 we saw that the response time can be discussed in three parts: call handling time, clear out time and driving time. In this section we will only consider the driving time by assuming that the handling time and clear out time are known and fixed.

In this section we will review existing literature on how to optimize EMS planning.

2.5.2 Classes of EMS models

The existing models to mathematically describe and optimize EMS care can roughly be divided in four classes. All these classes assume that there is a way to calculate the travel time between two points in the road network. How to perform these calculations is discussed in Section 2.4.

- **Deterministic Static Models**: These models try to find the best stationary distribution of the available EMS vehicles over all available bases.

- **Probabilistic Static Models**: These models take the probability into account that an EMS vehicle is busy, and try to find a stationary distribution of the available EMS vehicles over all available bases.

- **Dynamic Models**: This type of models tries to give insight into the best real time decisions during an EMS dispatch process.

- **Simulations**: Simulations try to evaluate one or more EMS dispatch strategies to get insight into good decision making.

Before we discuss each of these classes separately, we will talk about similarities between all models, and introduce an overall notation of the variables that we will use.

2.5.3 Similarities between all models

Although all models are different, they have some underlying assumptions in common. As said in the previous section, all models represent the road network by a graph $G = (V, A)$. $V$ is the disjunct union of the set of demand points $V$ and the set of bases $W$, i.e. $V = V \sqcup W$. Some articles call $W$ the set of potential base locations, but because we are going to make a programming package with fixed base location we prefer the term bases.

Furthermore, accidents only occur at demand points vertices $V$ of the graph $G$, and not on its arcs. Every inhabitant is assigned to the nearest vertex. In this way, every vertex $i \in V$ gets a density $d_i$ assigned to it. The choice of letting accidents happen only at vertices will not be a problem, other than losing some
precision by granularity: projections from the real starting point to the projected starting vertex have some distance, and the same holds for the end point, see Goldberg (2004). Also, all bases (and hospitals) are assumed to be at a vertex.

All models we discuss omit the possibility of emergency calls with priority B. Most of the algorithms do not even make a distinction between A1 and A2 calls.

Most governments measure the performance of EMS care by the percentage of calls that does not exceed a given time limit for the response time. All mathematical models consider the call handling time and clear out time as given. Subtracting these two times from the given time limit yields a maximum driving time that is allowed. Early models only consider one time limit $r$. When there are two time limits in place for different type of calls, we denote them with $r_1$ and $r_2$, ($r_1 < r_2$). Some areas have two types of EMS vehicles called ALS and BLS vehicles that each have a different maximum allowed driving time that we denote by $r_A$ and $r_B$, respectively. Later on we give more information about ALS and BLS vehicles.

Let us define some terminology we shall use. For each vertex $i \in V$ we define

$$W_i = \{ j \in W \mid t[i,j] \leq r \}.$$  

This set contains all bases that have a driving time of at most $r$ to vertex $i$. We say that vertex $i \in V$ is reachable from base $j \in W$, or equivalently that demand point $i \in V$ can be reached from base $j \in W$ within time $r$.

Similar, we define for each vertex $i \in V$ the sets

$$W^1_i = \{ j \in W \mid t[i,j] \leq r_1 \}$$

$$W^2_i = \{ j \in W \mid t[i,j] \leq r_2 \}$$

when the model contains two different maximum allowed driving times and

$$W^A_i = \{ j \in W \mid t[i,j] \leq r_A \}$$

$$W^B_i = \{ j \in W \mid t[i,j] \leq r_B \}$$

when we consider two types of EMS vehicles. If for some demand point $i \in V$ the inclusion $j \in W_i$ holds, we say that demand point $i$ is covered by base $j \in W$. Note that the terms ‘reachable’ and ‘being covered’ are equivalent.

The goal of the models is to find an optimal dispatch, hence we need to know how many vehicles should be put on each base. The amount of EMS vehicles that are assigned to base $j \in W$ is denoted by $y_j$, or $y^A_j$ and $y^B_j$ if we consider two types of EMS vehicles. We say that demand point $i \in V$ is covered by

$$\sum_{j \in W_i} y_j$$

EMS vehicles.

In many models we need to know by how many bases a demand point is covered. Only if demand point $i \in V$ is covered by at least one EMS vehicle, we define the binary variable $\lambda_i = 1$. Similar, only if demand point $i \in V$ is covered by at least two EMS vehicles we say that the binary variable $\mu_i = 1$. The generalization is defined by the binary variable $\nu^k_i$ which only equals one if demand point $i \in V$ is covered by at least $k \in \mathbb{N}$ EMS vehicles. In one model we need to have a binary variable equalling 1 only if there are exactly $k$ EMS vehicles covering demand point $i \in V$. Hence we define the variable $\xi^k_i$ to be 1 only if there are $k \in \mathbb{N}$ EMS covering demand point $i \in V$.

In this subsection we have mentioned the most important variables and assumptions of the existing models that describe EMS optimization. A complete overview of all variables used can be found in reference. In the following sections we sometime repeat the definition for the readers convenience.
2.6 Deterministic Static Models

The early models on EMS planning are not sophisticated. They are static IP-formulations that concentrate on finding a deterministic solution, and ignore stochastic considerations. See the survey paper by Brotcorne et al. (2010).

2.6.1 Location Set Covering Model (LSCM, 1971)

A (very) unsophisticated model for EMS planning is LSCM, see Toregas et al. (1971). This binary IP-model places at most one EMS vehicle at each base, in such a way that each demand point is covered at least once. The goal of this model is to minimize the number of EMS vehicles in the RAV such that each demand point can be reached within time $r$.

Minimize:  \[ \sum_{j \in W} y_j \]
Subject to:  \[ \sum_{j \in W_i} y_j \geq 1 \quad \forall i \in V, \]
\[ y_j \in \{0, 1\} \quad \forall j \in W. \]

Model 2.1: Location Set Covering Model

The LSCM gives a minimum amount of EMS vehicles needed to cover the entire RAV. A major disadvantage is that when an EMS vehicle gets dispatched, this will result in some demand points not being covered any more. A solution to counteract this will be discussed in the BACOP and DSM models.

Another disadvantage is that one may not have the number of EMS vehicles at ones disposal that is needed to cover all demand points. A way to find a good way to cover a RAV with exactly $M$ available EMS vehicles will be discussed in the MCLP model.

2.6.2 Maximal Covering Location Problem (MCLP, 1974)

The MCLP-model from Church and ReVelle (1974) will provide a maximal coverage over the population, with only $M$ available EMS vehicles at ones disposal. This model also assumes that one knows the population density in each demand point.

In the model, a new binary variable $\lambda_i$ is introduced for each demand point $i \in V$. A constraint then forces this variable to be equal to 1 only if its corresponding demand point can be reached within time $r$ by at least one EMS vehicle.

An underlying assumption of this model is that the number of potential occupied bases is not larger than the number of available EMS vehicles.
Maximize: \( \sum_{i \in V} d_i \lambda_i \)  
Maximize the population covered (at least) once.

Subject to: \( \sum_{j \in W_i} y_j = M \)  
Make sure that each EMS vehicle has a base assigned.

\( \sum_{j \in W_i} y_j \geq \lambda_i \)  
Only set \( \lambda_i = 1 \) if demand point \( i \) can be reached within time \( r_1 \).

\( y_j \in \{0, 1\} \)  
\( \forall j \in W \).

\( \lambda_i \in \{0, 1\} \)  
\( \forall i \in V \).

---

Model 2.2: Maximal Covering Location Problem

A disadvantage is that this model does not allow multiple EMS vehicles at each base, that results in uncovered demand points if an EMS vehicle gets dispatched. The BACOP and DSM models provide a backup strategy to prevent gaps in the RAV’s coverage when one EMS vehicle gets dispatched.

2.6.3 Backup Coverage Problems (BACOPs, 1986)

There are two BACOP-models, called BACOP1 and BACOP2. Both are described in Hogan and ReVelle (1986). The advantage of these models is that they encourage a double coverage of each demand point. This means that when one EMS vehicle is dispatched to an accident, every demand point is still covered by at least one other EMS vehicle.

Both BACOP models have the limitation that each base can only be occupied by at most one EMS vehicle. In total, we have to assign each of the \( M \) EMS vehicles to a base.

**BACOP1**

The BACOP1 model only rewards demand points that are covered twice. For this purpose, we define the binary variable \( \mu_i \) such that \( \mu_i = 1 \) if and only if demand point \( i \in V \) is covered by at least two EMS vehicles within time \( r \).

Recall that the amount of EMS vehicles that can reach a demand point \( i \in V \) within time \( r \) is given by the expression \( \sum_{j \in W_i} y_j \). Therefore, we include a constraint which states that \( \mu_i = 1 \) only if \( \sum_{j \in W_i} y_j > 1 \).

**BACOP2**

The disadvantage of the BACOP1 model is that the objective function rewards demand points that are covered only once equally much as demand points that are not covered at all. The BACOP2 model provides a way to include single covered demand points into the objective function with a weight factor
2.6 Deterministic Static Models

Maximize: \( \sum_{i \in V} d_i \mu_i \)  
Maximize the population covered (at least) twice.

Subject to: \( \sum_{j \in W} y_j = M \),  
Make sure that each EMS vehicle has a base assigned.

\( \left( \sum_{j \in W_i} y_j \right) - \mu_i \geq 1 \quad \forall \ i \in V \),  
Let \( \mu_i \) be 1 if and only if \( i \) is covered at least two times within time \( r_1 \).

\( y_j \in \{0, 1\} \quad \forall \ j \in W \),  
\( \mu_i \in \{0, 1\} \quad \forall \ i \in V \).

---

Model 2.3: Backup Coverage Problem 1

\( \theta \in [0, 1] \). Double covered demand points have weight factor 1. Note that we get BACOP1 if we choose \( \theta = 1 \), and we get MCLP if we choose \( \theta = 0 \).

We use the already defined binary variables \( \lambda_i \) for (at least) once covered demand points within time \( r \), and \( \mu_i \) for (at least) twice covered demand points within time \( r \).

Maximize: \( \theta \sum_{i \in V} d_i \lambda_i + (1-\theta) \sum_{i \in V} d_i \mu_i \)  
Maximize the population covered (at least) twice, and encourage single covered demand points (slightly).

Subject to: \( \sum_{j \in W} y_j = M \),  
Make sure that each EMS vehicle has a base assigned.

\( \mu_i - \lambda_i \leq 0 \quad \forall \ i \in V \),  
Make sure that if \( \mu_i = 1 \) then also \( \lambda_i = 1 \): What is covered twice is also covered once!

\( \left( \sum_{j \in W_i} y_j \right) - \lambda_i - \mu_i \geq 0 \quad \forall \ i \in V \),  
If \( i \) is covered once, let \( \lambda_i = 1 \). If there is a coverage by more than one EMS vehicle, let \( \mu_i = 1 \) as well.

\( y_j \in \{0, 1\} \quad \forall \ j \in W \),  
\( \lambda_i, \mu_i \in \{0, 1\} \quad \forall \ i \in V \).

---

Model 2.4: Backup Covering Problem 2
2.6.4 Double Standard Model (DSM, 1997)

A more recent model is the DSM from Gendreau, Laporte and Semet (1997). This is the first model that makes a distinction between two response times. This model, which can be seen as an extension of BACOP1, states that every call must be handled within time \(r_2\), but a fraction \(\alpha \in (0, 1)\) of the calls must be handled within the time standard \(r_1\) (with \(r_1 < r_2\)). To put it another way: The travel time that is needed to reach part \((1 - \alpha)\) of the population may take longer than \(r_1\), but no longer than \(r_2\).

Another advantage in DSM is that the bases are allowed to have multiple EMS vehicles assigned, with a maximum of \(M_j\) EMS vehicles on base \(j \in W\).

Maximize: \[\sum_{i \in V} d_i \mu_i\] Maximize the population covered (at least) twice.

Subject to: \[\sum_{j \in W} y_j = M,\] Make sure that each EMS vehicle has a base assigned.

\[y_j \leq M_j, \forall j \in W,\] The amount of EMS vehicles may not exceed the bases capacity.

\[\sum_{j \in W^2} y_j \geq 1, \forall j \in W,\] Every demand point must be reached within time standard \(r_2\).

\[\sum_{i \in V} d_i \lambda_i \geq \alpha \sum_{i \in V} d_i,\] Part \(\alpha\) of the population must be reached within time \(r_1\).

\[\mu_i - \lambda_i \leq 0, \forall i \in V,\] Make sure that if \(\mu_i = 1\) then also \(\lambda_i = 1\): What is covered twice is also covered once!

\[\left(\sum_{j \in W^1_i} y_j - \lambda_i - \mu_i\right) \geq 0 \forall i \in V,\] If \(i\) is covered once, let \(\lambda_i = 1\). If there is a coverage by more than one EMS vehicle, let \(\mu_i = 1\) as well.

\[y_j \in \mathbb{N} \forall j \in W,\]

\[\lambda_i, \mu_i \in \{0, 1\} \forall i \in V.\]

Model 2.5: Double Standard Model

Taking \(\alpha = 0.40\) says that 40% of the calls must be reached within time \(r_1\) and the other 60% must be reached within \(r_2\). It is not recommended to use this principle to model calls of A1 and A2 priority, because in general A1 calls are not near bases and A2 calls do not always occur at a greater distance from bases. A disadvantage of the DSM is that only one type of vehicles is taken into account. The TEAM model we will discuss next do consider two types of EMS vehicles.
The first model that allows multiple types of vehicles is TEAM, see Schilling et al. (1979). The model is an extension of the MCLP. The TEAM-model introduces two types of calls: Advanced Life Support (ALS) and Basic Life Support (BLS). It also introduces and two types of EMS vehicles (also referred to as ALS and BLS). ALS-calls must be handled by ALS-vehicles within time standard $r_2$, and BLS-calls must be handled by BLS-vehicles within time standard $r_1$ ($r_1 < r_2$). The model finds an assignment of the EMS vehicles to the bases such that a maximum will be obtained for the amount of people that can be reached by both types of vehicles, within their given time standards.

For this purpose we introduce $W^A_i := \{j \in V : t[i, j] \leq r_2\}$ and $W^B_i := \{j \in V : t[i, j] \leq r_1\} \forall i \in V$.

The amount of EMS vehicles that gets stationed within the RAV are denoted by $M^A$ and $M^B$.

Furthermore, we do not allow type BLS vehicles at a base that is not accompanied by a type ALS vehicle. This prevents the rent of too many bases. This restriction is not severe since BLS-vehicles are in practice high maneuverable and obtain higher speeds than the slower ALS-vehicles. Each base can house at most one vehicle of each type. The binary variables $y^A_j$ and $y^B_j$ state whether an EMS vehicle of type ALS or BLS is located at the base. The binary variable $\lambda_i$ equals 1 if and only if $i \in V$ is an element of both $W^A_i$ and $W^B_i$.

Maximize: $\sum_{i \in V} d_i \lambda_i$  
Maximize the population covered once by both type of vehicles.

Subject to:

\[ \sum_{j \in W^A_i} y^A_j \geq \lambda_i \quad \forall i \in V \]
If $\lambda_i=1$ this means it can be reached by an ALS-vehicle within time $r_2$.

\[ \sum_{j \in W^B_i} y^B_j \geq \lambda_i \quad \forall i \in V \]
If $\lambda_i=1$ this means it can be reached by a BLS-vehicle within time $r_1$.

\[ \sum_{j \in W} y^A_j = M^A, \]
Make sure that each ALS-vehicle has a base assigned.

\[ \sum_{j \in W} y^B_j = M^B, \]
Make sure that each BLS-vehicle has a base assigned.

\[ y^B_j \leq y^A_j \quad \forall j \in W, \]
An type B vehicle is only assigned to a base where a type A vehicle is assigned.

\[ y^A_j, y^B_j \in \{0,1\} \quad \forall j \in W, \]
\[ \lambda_i \in \{0,1\} \quad \forall i \in V. \]
2.6.6 Summary of static deterministic models

Deterministic static models were binary or integer linear programs, mainly developed during the early EMS planning. The LSCM model gives a minimal amount of EMS vehicles needed to cover a RAV. If the amount of EMS vehicles is known, then MCLP will ensure that most of the population can be reached within the given time standard. BACOP1 is a model that provokes double coverage, and BACOP2 also takes single coverage into account. DSM only looks to single coverage, but is more realistic in the sense that it considers two types of time constraints. TEAM allows two types of vehicles to work side by side.

The models that were discussed in this section are all deterministic. In the next section we will discuss probabilistic static models, which take the probability into account by which an EMS vehicle is dispatched and thus unavailable for a new dispatch.

2.7 Probabilistic Static Models

All probabilistic static models take into account that the probability that an EMS vehicle is available for dispatch is independent of the status of all other EMS vehicles.

2.7.1 Maximum Expected Covering Location Problem (MEXCLP, 1983)

Just like the LSCM-model, MEXCLP gives a lower bound for the amount of vehicles that is needed in the RAV, see Dashkin (1983). From statistics, we can estimate the busy fraction \( q \). This fraction states the amount of time that an ambulance is unavailable for dispatch, for example because it is already on its way to a call. Thus: If \( q = 0.2 \) this means that there is an 80% probability that the EMS vehicle is available for dispatch to an incoming call.

If we assume that a vertex \( i \in V \) is covered by exactly \( k \) EMS vehicles, we obtain the probability that none of the vehicles is available for dispatch to vertex \( i \) equals \((1-q^k)\). Thus the expected amount of people in \( i \) that is covered (i.e. the expected coverage demand) is given by \( E_k = d_i(1-q^k) \). The marginal expected covered demand of the \( k \)-th EMS vehicle is given by \( d_iE_k - d_iE_{k-1} = d_i(1-q^k) - d_i(1-q^{k-1}) = d_i(1-q)q^{k-1} \).

Taking the sum over the marginal expected covered demands yields

\[
E_\ell = \sum_{k=1}^{\ell} d_i(1-q)q^{k-1}. \quad \forall \ell \in \{0,1,\ldots,M\}
\]

Now we introduce the binary variable \( \nu_{i}^{k} \) for all \( i \in V \) and \( k \in \{1,2,\ldots,M\} \). We let \( \nu_{i}^{1} = 1 \) if and only if at least \( k \) EMS vehicles are stationed within time \( r_1 \) of vertex \( i \). Note that in particular \( \nu_{i}^{1} = \lambda_{i} \) and \( \nu_{i}^{2} = \mu_{i} \). This results in:

\[
E_i = \sum_{k=1}^{M} d_i(1-q)q^{k-1}\nu_{i}^{k} \text{ with } \nu_{i}^{k} = \mathbf{1}_{k \leq \ell}, \quad \forall i \in V
\]

To determine the \( q \)-values of the EMS vehicles, one needs to know the statistics. When the RAV is optimized by MEXCLP, it is likely that these \( q \)-values will change. Thus an input variable depends on the output. Optimizing via MEXCLP can be done iteratively.
Maximize: $\sum_{i \in V} \sum_{k=1}^{M} d_i (1 - q) q^{k-1} \nu_i^k$

Maximize the expected covered demand summed over all demand points.

Subject to: $\sum_{j \in W} y_j = M,$

Make sure that each EMS vehicle has a base assigned.

$\sum_{j \in W_i} y_j \geq \sum_{k=1}^{M} \nu_i^k \quad \forall i \in V,$

Set the $\nu$-values. The $\nu_i^{k-1} \leq \nu_i^k$ constraint is induced by the objective function.

$y_j \in \{0, 1\} \quad \forall j \in W,$

$\nu_i^k \in \{0, 1\} \quad \forall i \in V, \quad \forall k \in \{0, 1, 2, \ldots, M\}.$

---

Model 2.7: Maximum Expected Covering Location Problem

### 2.7.2 Maximum Availability Location Problem (MALPs, 1989)

The two MALP models, that are described in ReVelle and Hogan (1989), try to assign EMS vehicles to bases in such a way that as many demand points as possible can be reached within time $r_1$ by at least probability $\beta \in [0, 1]$. Thus if we take $\beta = .95$ and $r = 15$ minutes, this means that we try to find a coverage such that as many demand points as possible can be served within 15 minutes with probability 95% or higher.

Just like MEXCLP, we make use of a busy fraction $q$ for each EMS vehicle. Recall from MAXCLP that the probability that none of $k$ vehicles is available for dispatch to vertex $i$ equals $1 - q^k$. This results in the constraint

$$1 - q^k \sum_{j \in W_i} y_j \geq \beta$$

for each demand point $i \in V$. Taking the logarithm and ceiling yields

$$\sum_{j \in W_i} y_j \leq \lceil \log(1 - \beta) / \log(q) \rceil \quad \forall i \in V$$

Thus each demand point must be reachable by at least $b := \lceil \log(1 - \beta) / \log(q) \rceil$ EMS vehicles within time $r$. Only demand points by which this constraint is fulfilled may contribute to the objective function, i.e. we add $d_i$ to the objective function only if $\nu_i^b = 1$.

By MALP2 the $q$-values depend on the demand point’s location, i.e. $q = q_i$ and $b = b_i$ for all $i \in V$. Even more than by MEXCLP, the problem arises that it is hard to give good estimates for the values for $q_i$ because of its output-dependence. The Rel-P model tries to handle this inadequacy.
Maximize: $\sum_{i \in V} d_i \nu^b_i$

Maximize the number of demand points covered by $b$ EMS vehicles and thus a probability at least $\beta$ of being covered by at least one EMS vehicle.

Subject to:

$$\sum_{j \in W} y_j = M,$$

Make sure that each EMS vehicle has a base assigned.

$$\nu^{k-1}_i \leq \nu^k_i \quad \forall i \in V$$

What is covered by $k$ EMS vehicles is also covered by $k-1$ EMS vehicles.

$$\sum_{k=1}^b \nu^k_i \leq \sum_{j \in W_i} y_j \quad \forall i \in V,$$

Limit the $\nu$-values by the number of vehicles on the bases.

$$y_j \in \{0, 1\} \quad \forall j \in W,$$

$$\nu^k_i \in \{0, 1\} \quad \forall i \in V, \quad \forall k \in \{0, 1, 2, \ldots, M\}.$$

**Model 2.8: Maximum Availability Location Problem 1**

### 2.7.3 Reliability Perspective (Rel-P, 1993)

The Rel-P model from Ball and Lin (1993) is based on MALP2. The objective of Rel-P is to impose an upper bound $\beta_i \in [0, 1]$ on the probability that a call on the demand point $i \in V$ does not receive immediate service. Also, the model allows an upper bound for the number of EMS vehicles on each base. First we have a discussion how the constraints work, followed by a short discussion about the objective function.

**Constraints**

From statistics, it is easy to obtain an upper bound $T$ for the service time. If the amount of EMS vehicles that must depart from a base $j \in W$ within a time interval of length $T > k$, this means that not all calls can be served. Since Rel-P models the call generation by a poisson process on each demand point (whose parameter can be determined by statistics), it is possible to determine this probability given $T, k$ and $j$.

We define the binary variable $\xi^k_j = 1$ if and only if there are exactly $k$ EMS vehicles stationed at base $j \in W$. We take the probability that the number of EMS vehicles that depart from $j$ within a time interval of length $T$ exceeds $k$ to the power $\xi^k_j$:

$$P(\text{The number of EMS vehicles that depart from } j \text{ within a time interval of length } T > k) = \prod_{1 \leq k \leq M_j} P(\text{The number of EMS vehicles that depart from } j \text{ within a time interval of length } T > k)^{\xi^k_j},$$

where of course another constraint must be imposed to regulate that for at most one value of $k$ one can
have $\xi_j^{k} = 1$:

$$\sum_{1 \leq k \leq M_j} \xi_j^{k} \leq 1.$$ 

Then, the probability that a demand point $i \in V$ cannot be served by any EMS vehicle is given by

$$\prod_{j \in W_i} \prod_{1 \leq k \leq M_j} P(\text{The number of EMS vehicles that depart within a time interval of length } T > k)^{\xi_j^{k}}.$$ 

We have chosen $\beta_i \in [0, 1]$ in such a way that the probability that a call on $i \in V$ can be served is at least $\beta_i$, i.e. the probability that a call cannot be served is smaller than $1 - \beta_i$. This results in the following two equations:

$$\prod_{j \in W_i} \prod_{1 \leq k \leq M_j} P(\text{The number of EMS vehicles that depart from base } j \text{ within a time interval of length } T > k)^{\xi_j^{k}} < 1 - \beta_i \quad \forall \ i \in V,$$

$$\sum_{1 \leq k \leq M_j} \xi_j^{k} \leq 1 \quad \forall \ j \in W.$$ 

This is equivalent to

$$\prod_{j \in W_i} \prod_{1 \leq k \leq M_j} P(\text{The number of EMS vehicles that depart from base } j \text{ within a time interval of length } T \geq k)^{\xi_j^{k}} \leq 1 - \beta_i \quad \forall \ i \in V,$$

$$\sum_{1 \leq k \leq M_j} \xi_j^{k} \leq 1 \quad \forall \ j \in W.$$ 

Taking the logarithm yields

$$\sum_{j \in W_i} \sum_{1 \leq k \leq M_j} \xi_j^{k} \log(P(\text{The number of EMS vehicles that depart from base } j \text{ within a time interval of length } T \geq k))) \leq \log(1 - \beta_i) \quad \forall \ i \in V,$$

$$\sum_{1 \leq k \leq M_j} \xi_j^{k} \leq 1 \quad \forall \ j \in W.$$ 

Now, we define for all $i \in V$, $j \in W$ and $k \in \mathbb{N}^+$

$$a_{jk} := \log(P(\text{The number of EMS vehicles that depart } j \text{ within a time interval of length } T \geq k)),$$

$$b_i := \log(1 - \beta_i).$$

such that our system becomes

$$\sum_{j \in W_i} \sum_{1 \leq k \leq M_j} \xi_j^{k} a_{jk} \leq b_i \quad \forall \ i \in V,$$

$$\sum_{1 \leq k \leq M_j} \xi_j^{k} \leq 1 \quad \forall \ j \in W.$$
2 Literature review

Objective function

Usually, the objective function helps to acquire good coverage. By Rel-P this is already regulated by the constraints. However, we do not want too much EMS vehicles in our system, because this yields too high costs. Let $c_{jk}$ be the cost of having $k$ EMS vehicles on base $j \in W$. This can be predetermined as well. In that case, the total cost that we want to minimize will be given by

$$\sum_{j \in W} \sum_{1 \leq k \leq M_{j}} \xi_{j}^{k} c_{jk}.$$ 

Minimize: $\sum_{j \in W} \sum_{1 \leq k \leq M_{j}} \xi_{j}^{k} c_{jk}$ Minimize the total cost.

Subject to: $\sum_{j \in W, 1 \leq k \leq M_{j}} \xi_{j}^{k} a_{jk} \leq b_{i}$ $\forall i \in V$ Make sure that each demand point $i$ is covered by probability $\beta_{i}$.

$\sum_{1 \leq k \leq M_{j}} \xi_{j}^{k} \leq 1$ $\forall j \in W,$ Do not allow multiple amounts on EMS vehicles on the same base.

$\xi_{j}^{k} \in \{0,1\}$ $\forall j \in W,$ $\forall k \in \{0,1,2,\ldots,M_{j}\}$.

---

Model 2.9: Reliability Perspective

2.7.4 Two-Tiered Model (TTM, 1998)

The TTM can be seen as the static probablistic version of TEAM, where the distinction between ALS- and BLS-vehicles has been made, see Mandell (1998). An ALS-call must be handled within time $r_{A}$ by an ALS-vehicle, and a BLS-call can be handled by either an ALS- or a BLS-vehicle within time $r_{B}$. A base $j \in W$ can be occupied by at most one vehicle of each type.

TTM assumes that one can calculate the probability that a unit can reach a demand point $i \in V$ given the amounts of units in its vicinity; this probability is denoted by $\theta_{i}^{h,k,\ell}$. This can be done by a queuing model. Note that $\theta_{i}^{h,k,\ell}$ is an increasing function in $h$, $k$ and $\ell$ for all $i \in V$ in realistic situations.

$h$ : The number of ALS-vehicles within time $r_{A}$ of demand point $i \in V$.

$k$ : The number of ALS-vehicles within time $r_{B}$ of demand point $i \in V$.

$\ell$ : The number of BLS-vehicles within time $r_{B}$ of demand point $i \in V$.

The goal of TTM is to maximize the expected covered demand given $M_{A}$ ALS vehicles and $M_{B}$ BLS vehicles. The binary variable $\eta_{i}^{h,k,\ell} = 1$ if and only if demand point $i \in V$ is covered by $h$ ALS vehicles within time $r_{A}$, plus $k$ ALS vehicles within time $r_{B}$ and $\ell$ BLS-vehicles within time $r_{B}$. Now, the expected covered demand can be expressed as $\sum_{i \in V} \sum_{h=1}^{M_{A}} \sum_{k=0}^{M_{B}} \sum_{\ell=0}^{M_{B}} \theta_{i}^{h,k,\ell} \eta_{i}^{h,k,\ell}$. 

Page 48
2.7 Probabilistic Static Models

The constraints are designed to assign values to \( \pi_i^{h,k,\ell} \). For each demand point \( i \in V \) only one combination of \( h, k \) and \( \ell \) may yield \( \pi_i^{h,k,\ell} = 1 \). This effect can be described by the constraint

\[
\sum_{h=0}^{h_i} \sum_{k=0}^{k_i} \sum_{\ell=0}^{\ell_i} \pi_i^{h,k,\ell} \leq 1 \quad \forall \ i \in V.
\]

First we will obtain upper bounds for \( h, k \) and \( \ell \), denoted by \( h_i, k_i \) resp. \( \ell_i \). It is obvious that an upper bound for \( h_i \) and \( k_i \) is \( M^A \), and an upper bound for \( \ell_i \) is \( M^B \): a demand point cannot be reached by more vehicles than the number available in the RAV. Also, a demand point cannot be reached by more EMS vehicles of a specific type than it has bases that can reach the demand point within the time limit \( r^A \) or \( r^B \), since each base has at most one vehicle of each type. This gives the upper bounds \( |W_i^A| \) for \( h_i \) and \( |W_i^B| \) for \( k_i \) and \( \ell_i \). Concluding, we take the following upper bounds:

\[
\begin{align*}
  h_i & := \min(M^A, |W_i^A|) \quad \forall \ i \in V \\
  k_i & := \min(M^A, |W_i^B|) \quad \forall \ i \in V \\
  \ell_i & := \min(M^B, |W_i^B|) \quad \forall \ i \in V
\end{align*}
\]

We obtain constraints of the form stated below, and only give an explanation for \( h_i \) since \( k_i \) and \( \ell_i \) go similar. Recall that only for one value of \( h \), we still have \( \pi_i^{h,k,\ell} = 1 \). The objective value will be higher if the value of \( h \) increases. However, \( h \) is restricted by \( h_i \) and by the number of ALS-vehicles that will be stationed within time \( r^A \) from \( i \in V \). The former can be found as upper bound of the sum, and the latter is the right hand side of the constraint.

\[
\begin{align*}
  \sum_{k=0}^{h_i} \sum_{\ell=0}^{\ell_i} \sum_{h=0}^{k_i} \pi_i^{h,k,\ell} & \leq \sum_{j \in W_i^A} y_j^A \quad \forall \ i \in V \\
  \sum_{k=0}^{h_i} \sum_{\ell=0}^{\ell_i} \sum_{h=0}^{k_i} \pi_i^{h,k,\ell} & \leq \sum_{j \in W_i^B} y_j^B \quad \forall \ i \in V
\end{align*}
\]

Now we will adapt these four constraints and objective function to meet some other requirements. First of all we want to have at least one ALS-vehicle within time \( r^A \) of each demand point, so we set an lower bound \( h = 1 \). This is easily done by replacing all occurrences of \( h = 0 \) by \( h = 1 \) in these constraints.

Also, we do not condone combinations where \( k = 0 \) and \( \ell = 0 \) at the same time, because this means that a demand point is not covered within time \( r^B \) with a vehicle that can handle a BLS-call; recall that \( r^B \leq r^A \). Where this situation might occur in the constraints and objective function, we let \( \ell = 1 \{k=0\} \), which simply means that \( \ell = 0 \) if \( k > 0 \) and \( \ell = 1 \) if \( k = 0 \).

Furthermore, we must have \( h \geq k \). This means that the number of ALS vehicles that can reach a demand point within time \( r^A \) is at least the amount that can reach it within time \( r^B \). We will adapt the constraints and objective function to take this into account. If we first sum over \( h \), we let the value of \( h \) be an upper bound for \( k \). On the other hand, if we first sum over \( k \), we let \( k \) be an upper bound for \( h \).

This all results to the Two-Tiered Model as shown in Model 2.10 with objective function

\[
\sum_{i \in V} \sum_{h=1}^{h_i} \min(h,k_i) \sum_{\ell=1}^{\ell_i} \sum_{k=0}^{k_i} d_i \theta_i^{h,k,\ell} \pi_i^{h,k,\ell}.
\]
Maximize: \[ \sum_{i \in V} \sum_{h=1}^{h_i} \min(h,k_i) \sum_{k=0}^{\ell_i} \sum_{\ell' \in 1(k=0)} d_i \theta_i^{h,k,\ell} \pi_i^{h,k,\ell} \]

Maximize the expected covered demand.

Subject to:

\[ \sum_{h=1}^{h_i} h \sum_{k=0}^{\min(h,k_i)} \sum_{\ell=1(k=0)}^{\ell_i} \pi_i^{h,k,\ell} \leq \sum_{j \in W_i^A} y_j^A \quad \forall i \in V, \]

Set the number of ALS vehicles on each base within travel time \( r^A \).

\[ \sum_{k=1}^{k_i} k \sum_{h=k}^{h_i} \sum_{\ell=0}^{\ell_i} \pi_i^{h,k,\ell} \leq \sum_{j \in W_i^B} y_j^B \quad \forall i \in V, \]

Set the number of ALS vehicles on each base within travel time \( r^B \).

\[ \sum_{\ell=1}^{\ell_i} \sum_{h=1}^{h_i} \min(h,k_i) \sum_{k=0}^{\ell_i} \pi_i^{h,k,\ell} \leq \sum_{j \in W_i^B} y_j^B \quad \forall i \in V, \]

Set the number of BLS vehicles on each base within travel time \( r^B \).

\[ \sum_{h=1}^{h_i} \min(h,k_i) \sum_{k=0}^{\ell_i} \sum_{\ell=1(k=0)}^{\ell_i} \pi_i^{h,k,\ell} \leq 1 \quad \forall i \in V, \]

Only one combination of \( h, k \) and \( \ell \) may yield \( \pi_i^{h,k,\ell} = 1 \).

\[ \sum_{j \in W} y_j^A \leq M^A, \]

Make sure that each EMS vehicle of type A has a base assigned.

\[ \sum_{j \in W} y_j^B \leq M^B, \]

Make sure that each EMS vehicle of type B has a base assigned.

\[ \pi_i^{h,k,\ell} \in \{0,1\}, \quad y_j^A, y_j^B \in \{0,1\} \quad \forall i \in V, \quad \forall j \in W, \quad 0 \leq h \leq h_i, 0 \leq k \leq k_i, 0 \leq \ell_i, \]

Model 2.10: Two-Tiered Model

### 2.7.5 Summary of Probabilistic Static Models

Probabilistic static models are IP-formulations that take the probability into account that an EMS vehicle is busy. We have seen four of these models. MEXCLP finds a best coverage given the amount of EMS vehicles by maximizing the expected covered demand. MALP does the same, but only rewards after a certain probability of being covered. Rel-P finds a coverage such that each demand point is covered with a certain (chosen) probability, which implies a minimum amount of EMS vehicles. The TTM allows two types of vehicles, and finds a good way to allocate them: it is a probabilistic version of TEAM.

Maxwell et al. (2009) cites modern approaches as discussed by Ingolfsson on the INFORMS Conference in the year 2006 and personal communications with Goldberg in the year 2007 where static models are used to create lookup tables for every amount of available EMS vehicles in which a maximal coverage has been achieved. EMS dispatchers try to keep their EMS configuration as close to these tables by relocating idle EMS vehicles.

All static models find a way to allocate EMS vehicles such that some kind of coverage has been guaranteed.
2.8 Dynamic Models

However, when an EMS vehicle departs for a call, which happens quite often in a realistic situation, the coverage will likely to get gaps. Thus perhaps it is best to relocate one or several other vehicles such that the coverage will improve. Advanced models taking this effect into account, are called dynamic models. We will consider them in the next subsection.

2.8 Dynamic Models

Modern research in EMS deployment is mainly focussed to dynamic EMS modeling. Dynamic models help to relocate idle EMS vehicles such that a maximum of calls can be reached within the time threshold. In relation to the previous models, dynamic models are not searching for a static equilibrium but rather contribute to real life relocation systems. In this subsection we shall discuss two models: DDSM\textsuperscript{t} and ADP.

2.8.1 Dynamic Double Standard Model (DDSM\textsuperscript{t})

The double standard model is an adapted version of DSM, see Gendreau et al. (2001). In DSM we relocate every EMS vehicle to a base at the moment a call is received and a vehicle gets dispatched. The costs induced by a relocation of EMS vehicle $\ell$ on time $t$ to base $j \in W$ is $c_{t,\ell,j}^t$. EMS vehicles that are relocated to the base at their current position equals zero: this prevents too many relocations. We only consider available EMS vehicles for relocation. The amount of relocatable EMS vehicles is given by the variable $m$. The total costs of relocations is given by the expression

$$\text{Total relocation costs} = \sum_{j \in W} \sum_{\ell=1}^m c_{t,\ell,j}^t u_{\ell,j}.$$

The DSM model is displayed in Algorithm 2.4.

2.8.2 Approximate Dynamic Programming, (ADP, 2005)

The ADP formulation as described in Restrepo (2008) and Maxwell et al. (2009) captures the random evolution of a system we mentioned earlier in Section 2.8.2. We will give a compact description of the way ADP works.

As stated before, the trajectory of the system can be written as $\{s_k | k = 1, 2, 3, \ldots\}$. ADP only considers a relocation for an EMS vehicle if the event on its state $e(s)$ is ‘finished at a hospital’. Let $i \in V$ be the vertex where the hospital is located, thus $i := o_a(s) = d_a(s)$. We let the binary variable $u_{ij} = 1$ only if we relocate an EMS vehicle from $i$ to base $j \in W$. Defining $\mathbf{u}(s) := \{u_{ij}(s) : j \in W\}$, all feasible relocation decisions are given by

$$\mathcal{U}(s) := \left\{ \mathbf{u}(s) \in \{0, 1\}^{|W|} : \sum_{j \in W} u_{ij}(s) = 1 \right\}.$$

For other states $e(s)$ we do not relocate, which implies $\mathcal{U}(s) = \emptyset$. With only small a amount of bases, it is possible to enumerate over all possible relocations.
Maximize: \( \sum_{i \in V} d_i \mu_i - \sum_{j \in W} \sum_{\ell=1}^{m} c_{\ell j}^t u_{\ell j} \)  

Maximize the population covered (at least) twice and subtract the costs of relocating.

Subject to: \( \sum_{j \in W} u_{\ell j} = 1 \) \quad \forall \ell \in \{1, \ldots, m\},

\( \sum_{\ell=1}^{m} u_{\ell j} \leq M_j \) \quad \forall j \in W,

\( \sum_{j \in W} \sum_{\ell=1}^{m} u_{\ell j} \geq 1 \) \quad \forall i \in V,

\( \sum_{i \in V} d_i \lambda_i \geq \alpha \sum_{i \in V} d_i \)

\( \mu_i - \lambda_i \leq 0 \) \quad \forall i \in V,

\( \left( \sum_{j \in W \setminus i} \sum_{\ell=1}^{m} u_{\ell j} \right) - \lambda_i - \mu_i \geq 0 \) \quad \forall i \in V,

\( y_j \in \mathbb{N}^+ \) \quad \forall j \in W,

\( \lambda_i, \mu_i \in \{0, 1\} \) \quad \forall i \in V.

Algorithm 2.4: Dynamic Double Standard Model.

ADP models the call generation in the RAV according to a Poisson point process with a known arrival intensity. If a call cannot be served because there are not any available EMS vehicles, the call is placed into a waiting queue which is handled on a first in first out basis. ADP only considers one type of priority.

When taking decisions into account, the trajectory can be written as \( \{(s_k, u_k(s_k)) | k = 1, 2, 3, \ldots\} \), where \( u_k(s_k) \) may be an empty set. The dynamic model includes various stochastic processes, which are included in a probability function \( w(s_k, u_k) \). When including these uncertainties, we can calculate the next state \( s_{k+1} \) by some transfer function \( f \):

\( s_{k+1} = f(s_k, u_k(s_k), w(s_k, u_k)) \)

We introduce a binary cost function \( c(s_k, u_k, f(s_k, u_k(s_k), w(s_k, u_k))) \) equalling 1 only if the event \( c(s_{k+1}) \) is ‘ambulance arrives on scene’, the call has a high priority, and the response time exceeds time threshold \( r_1 \). Since ADP models only for a finite planning horizon \([0, T]\), we let the cost also equal zero if the starting time \( \tau(s_k) \) of the event exceeds \( T \).

Decisions are made by following certain policy \( p(s) \in \mathcal{U}(s) \), which states the decision that is getting made while the system is in state \( s \). The trajectory of the system can than be described by \( \{s_k^p := \begin{align*}
\end{align*} \)
(s_k, p(s_k) | k = 1, 2, 3, ...), and furthermore we can say $s_{k+1}^p = f(s_k^p, p(s_k^p), w(s_k^p, p(s_k^p)))$. The goal of ADP is to find policy $p^*$ which minimizes the expected costs given by

$$J^p(s) = \mathbb{E} \left[ \sum_{k=1}^{\infty} \alpha^k c(s_k, p(s_k), f(s_k^p, p(s_k), w(s_k^p, p(s_k^p)))) | s_1^p = s \right].$$

Here $\alpha \in [0, 1)$ is a fixed discount factor that exists to let expected lost calls in the far and uncertain future weight less than expected lost calls in the near future which we can predict much better. Rewriting it, and taking the minimizer over its policy yields:

$$J(s) = \min_{u \in U(s)} \left\{ \mathbb{E} \left[ c(s, u(s), f(s, u(s), w(s, p(s)))) + \alpha^{\tau(f(s, u(s), w(s, u(s)))) - \tau(s)} \cdot J(f(s, u(s), w(s, u(s)))) \right] \right\}$$

The difficulty is to find the minimizer $p^*(s)$ for this expression, because the state space is of infinite dimension. To find a good approximation for $p^*(s)$ we iteratively try to improve values for the policy $p(\cdot)$ and $J(\cdot)$, denoted by an iteration number in a superscript.

We approximate $J(\cdot)$ with $J(s, r) = \sum_{\ell=1}^{L} r_{\ell} \phi_{\ell}(s)$, where $r_{\ell} \in \mathbb{R} := \{r_1, r_2, \ldots, r_L\}$ are tunable parameters and $\phi_{\ell} \in \phi := \{\phi_1(s), \phi_2(s), \ldots, \phi_L(s)\}$ are $L$ fixed basis functions which approximate the effect of unreachable calls, uncovered call rates, missed calls and future uncovered call rates. See Maxwell et. al (2009) for more details for the choice of basis functions.

**Initialization**

At initialization, we set the iteration counter $n = 1$ and initialize $\{r_{\ell}^n | \ell = 1, 2, \ldots, L\}$ at random. The ADP optimization algorithm can be found in Algorithm 2.5.

**Algorithm**

The algorithm is as displayed in Algorithm 2.5:

---

1. **Do**
2. **Let $p^n$ be** the greedy policy induced by $J(\cdot, r^n)$.
3. **Simulate** the trajectory of policy $p^n$ over the time interval $[0, T]$ for $Q$ replications.
   
   This produces the state trajectories $\{(s_k^{n,q} : k = 1, 2, \ldots, K_q), q = 1, 2, \ldots, Q\}$. We denote the induced costs of following policy $p^n$ starting from $s_1^{n,q}$ by $C_{k,n}^{q}$.
4. **Compute** the tunable parameters by solving the linear square regression

   $$r^{n+1} = \arg\min_{r \in \mathbb{R}^K} \left\{ \sum_{q=1}^{Q} \sum_{k=1}^{K_q} (C_{k}^{n,q} - J(s_k^{n,q}, r^n))^2 \right\}$$

5. **Increase** $n$ by 1.
6. **Loop until** a predetermined amount of iterations has been made.

---

**Algorithm 2.5: Approximate Dynamic Programming.**

In this stage of development ADP only considers one type of call priority and does not include the possibility of having EHGV calls. Testing the ADP approach on the city of Edmonton which has approximately as many EMS vehicles as Amsterdam, gave an improvement of 2% on a static policy that was applied.
2.8.3 Summary of Dynamic Models

Dynamic models consider real time relocations. We have described the only two dynamic models: DDSM\textsuperscript{t} which is an extension of DSM, and ADP.

DDSM\textsuperscript{t} relocates only at the moment a vehicle gets dispatched, and ADP relocates when an EMS vehicle comes available. DSM allows multiple vehicles in one dispatch, whilst ADP only considers the relocation of a single EMS vehicle.

2.9 Simulations

Simulations are an important part in EMS research. Because they provide graphical information that is very illustrative, they are highly appreciated by EMS managers. Also, the effect of certain decisions is easily understandable when a simulation is used. Simulations provide a means to see the effect of certain decisions on a real time basis, see Henderson and Mason (2005).

The effect of all models that are mentioned before, are tested on a real life environment by using some real data such as previous call records to check their performance. However, there are simulation packages by other authors worth mentioning. We take a closer look to BartSim and its successor SIREN.

2.9.1 BartSim

BartSim is a simulation package developed by Henderson and Mason (2005) for St. Johns Ambulance Service in Auckland, New Zealand, to assist during policy making.

EMS vehicles have a computer-aided dispatch (CAD) system that logs all call data such as travel times, treatment time and transfer time, like the mobilophone does in The Netherlands (see Subsection 3.2.2 and Appendix B). This simulation engine is the first of its kind that uses real data for modeling the calls. In this way, data does not have to be recorded manually as in EMS studies before; Henderson and Mason state that during the survey of Swoveland et al. (1973) data was gathered manually for a period of two weeks. They also point out that using a GIS-system is relatively new in EMS planning.

Auckland was described by a graph containing 2200 vertices and 5000 directed arcs. Accidents are generated on the vertices via a bootstrapping procedure (see Subsection 3.2.2) process. Some of these vertices are not intersections nor dead ends. Leaving them out of the graph results in 765 vertices called decision vertices.

The travel speed of the EMS vehicles is time dependent in BartSim. During pre-processing, the all-to-all shortest path between the 765 decision vertices is calculated by using the Floyd-Warshall algorithm for times 8:00, 12:00 and 17:00, see Subsection 2.4.2. Travel times between these three times are calculated by interpolation. The shortest path calculation between two vertices are done by a heuristic that is claimed to have a good level of accuracy. Though the Floyd-Warshall is exact, the travel times are done by a heuristic and thus not exact.

The BartSim simulator works as a discrete-event simulator. Though the article does not mention how the discrete-events are implemented, we think that it has similarities with TIFAR’s speedsim modus, see Section 3.2.1.
2.9.2 SIREN

SIREN is the successor of BartSim, and simulates EMS movements as well. This software package was used on the Melbourne area, too. It is hard to find good information about SIREN, since the official website that is mentioned in the paper is offline (we tried to access several times in 2010). Now, it is integrated in commercial packages such as Optima Predict and Optima Life, see http://www.theoptimacorporation.com.

Some information about SIREN can be found in Henderson and Mason (2005) and Mason (2005). SIREN uses better base locations and it considers to move bases for improved response times. SIREN includes real data and simulations yield an improvement up to 9% on the previous strategy Melbourne used.

SIREN also includes stochastic travel times and non-homogeneous call generation. The simulation package can dispatch more vehicles to a call, too. SIREN can handle up to 6 000 vertices and 14 000 arcs, and contains arc specific travel times for EMS vehicles that drive with optical and auditory signals.

2.10 Chapters Conclusions

This chapter contains our literature review. In Section 2.2.1 we described three ways to model the general dispatch procedure, discussed the three parts in which we can divide the response time and gave a swift discussion about the road plan.

Section 2.2.3 described the current situation of The Netherlands and RAV Agglomeratie Amsterdam in particular. We introduced the call priorities (A1, A2 and B), gave the current location of bases, hospitals and the number of inhabitant of RAV AA, had a discussion about traffic exemptions of EMS vehicles, and concluded with statistics of EMS dispatch in the year 2009.

In Section 2.4 we discussed ways to calculate shortest paths. We have two possibilities to do this in an overall model: all-to-all shortest path at precessing or the on-demand one-to-one shortest path in processing. We discussed two algorithms for all-to-all shortest paths (Floyd-Warshall and Johnson), and two algorithms to do it on-demand (GSP and A*). Using a bidirectional search in A* improves the speed of finding a shortest path.

The succeeding sections discussed existing models for EMS dispatch.

Section 2.5 gave a short introduction of the classes of models and introduces variables that are used in multiple models. Section 2.6 described six deterministic static models (LSCM, MCLP, BACOP1, BACOP2, DSM and TEAM), Section 2.7 did the same for five probabilistic static models (MEXCLP, MALP1, MALP2, Rel-P, TTM), Section 2.8 handled two dynamic models (DDSM and ADP), and Section 2.9 concluded the chapter by discussing two simulation packages (BartSim and SIREN).

In the next chapter we will use our literature review to develop our own simulation package, TIFAR.
Chapter 3

TIFAR simulation package

3.1 Introduction

In this thesis we develop a C++ program called TIFAR (Testing Interface For Ambulance Research) that simulates EMS movements. With a simulation package, the effect of certain dispatch decisions gets visualized.

In the second section we discuss the way TIFAR works and we explain the choice for several decisions while developing TIFAR.

3.2 Program description

3.2.1 Program Overview

TIFAR is programmed in ansi C++ such that it is easily transferable over several platforms and has an easy maintenance. The object based program is inspired by ADP, see Section 2.8.2. An overview of some important classes can be found in Appendix B.

The program consists of three loops that are hooked into each other: The main loop (ML), renewal loop (RL) and accident generation loop (AG Loop) There are two modes in which TIFAR can run: visual mode or speed simulation mode. The only way that these two modes differ is they way they set the next time stamp at which the (new) state of the system gets calculated, and the way they perform output. Visual mode uses a GUI for output and the internal computer clock to determine the next time in which the current system state gets calculated. In this way one can see the EMS movements at real time, or with a speedfactor. When the speedfactor is set to 60, it means that one minute of run time simulates one hour of EMS movements. This mode stops when the ESC key on the keyboard is pressed.

Speed simulation mode holds a ordered list with the times that an event ends. At the moment an EMS vehicle departs, we can calculate the moment that the vehicle arrives, this gives a new time stamp. The moments when a time stamp enters the list is when a call enters the system, a vehicle departs from a location or a vehicle arrives at a location. The next time at which the state of the system is calculated is the first one in this ordered list. When a predetermined end time has ended, this modes terminates
the program and displays statistics in the terminal. Speed simulation mode is faster than visual mode, because only at necessary time stamps the renewal loop gets called. For each of the two modes there is a separate main loop included in the program.

The second loop is the renewal loop. This loop calculates the next state of the system given a new time stamp, see Section 3.2.3. The accident generation loop (which is called by the renewal loop) creates the accidents, and will be discussed in the next subsection.

3.2.2 Accident Generation Loop

Introduction

The accident generation loop (AG Loop) has come into existence to simulate the calls in TIFAR. In this subsection we will discuss the way the AG loop works.

A call has a few properties:

1. The model time when the accident occurs, called the start time.
2. The location on the map where the accident occurs, called the origin.
3. The treatment time, i.e. the time the EMS personnel must spend at the accident scene.
4. The transfer time, i.e. the time the EMS personnel must spend at the hospital to transfer the patient into hospital care. Only if the call is not EHGV, see also Section 2.2.1.
5. The priority of the call.
6. Whether the call is EHGV or not. Recall that EHGV means that the patient does not require transportation to the hospital.

A small percentage (2.8% in RAV Agglomeratie Amsterdam in the year 2009) of the calls is false alarm (Dutch: loze ritten). The EMS vehicle goes to the location, but there is no patient to treat. In TIFAR, false calls are modeled as EHGV: instead of treating the patient at the location, EMS personnel is searching for a non-existing person. The time intervals of false alarms are similar to those with an EHGV status.

At the moment that a call is generated by the AG, all these variables get a value. However, the dispatch center (DC) of TIFAR will only consider the current situation when making a decision, and will not make use of information that will become available in the future, just like reality.

Generating a call’s start time

There are two ways calls can be generated: one can choose to give each demand point its own distribution, or one can choose to have one distribution for the moment when a call in the RAV occurs and a separate distribution to determine the location within the RAV of the calls. TIFAR makes use of the latter.

Calls are generated by a Poisson process, which has an intensity parameter. The value of this intensity parameter states the average amount of time between two successive calls. Călinescu (2009) states that an inhomogeneous Poisson process seems an appropriate model for our call arrival process. The
inhomogeneousness is a result of daily and hourly patterns. In our simulations, we omit these patterns, and hence use a homogeneous process for call generation.

Let us determine the value of the parameter. In total there are 63901 A1 and A2 calls in the region, see Table 2.10. Now we can calculate the frequency of call generation:

\[
\frac{60 \text{ minutes per hour} \cdot 24 \text{ hours per day} \cdot 365.242199 \text{ days per year}}{63901 \text{ call per year}} = 8.23068 \text{ minutes per call.}
\]

Thus we generate an A1 or A2 call with a Poisson process with a parameter equally to 8:14 minutes. (When including B calls and perform a similar calculation, we get a value of 5:13 minutes for the parameter.)

Generating a call’s origin

There are a couple of possible choices for generating the location of the accident:

- **Using RD-coordinates**: The Netherlands has its own cartesian coordinate system called the rijkdriehoekscoördinaten, where the unit is 1 meter. Every location in the country has an x- and y-coordinate. When knowing the border of the RAV in RD-coordinates, one can generate a call on a random position uniformly distributed within the RAV. **Advantage**: An accident can happen on every location within the RAV, even on water. One has to keep in mind that this location must be mapped upon the road network since we have to send an EMS vehicle. **Disadvantage**: The population density is not included.

- **Using postal codes**: Every address with a mail box has one postal code assigned. Such a postal code consists of four digits and two letters, for example 1011AA. Postal codes are not unique: the part of a street can have the same postal codes. However, the combination of a postal code and house number forms an unique combination. The four digits are forming the neighborhood, whilst the two letters specify the location within this neighborhood. People involved in route planning in The Netherlands therefore make the distinction between these so called 6PP- and 4PP-postal codes. A rule of thumb states that the cumulative post for all houses with the same 6PP postal code is the quantity that a post man can hold in his hands. One can map a postal code onto RD-coordinates. **Advantage**: The population density is included. Using the rule of thumb, one can say that the amount of people on each postal code is almost equal. One must keep in mind that streets with only one mail box and not many inhabitants have their own postal code, while on the other hand nursing homes with one postal code as well have a high potential that an EMS vehicle should be called. **Disadvantage**: Forests, water and highways do not have postal codes because there are no mail boxes. Still, accidents can happen at these places.

- **Using bootstrapping**: Using EMS data from the past, one knows where an accident has happened, and thus where an accident can happen again. In The Netherlands, all EMS data from 2007 until now is known. Accidents in the model can be generated using a bootstrap procedure from this data. **Advantage**: Places where a lot of accidents happen in reality, will be represented very accurately. **Disadvantage**: There are a lot of places where no accident has happened before, but are available for new accidents. Furthermore, new neighborhoods are not included by this way of accident generation. The use of historical EMS data can also involve privacy concerns, though when one can afford to lose some precision, the location can be anonymized by mapping it onto the 4PP postal code.

One can take a linear combination of these ways to generate a good procedure, though in this case one should find a way to get good linearization parameters.
TIFAR simulation package

<table>
<thead>
<tr>
<th>Total call handling time:</th>
<th>46.25 minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driving time:</td>
<td>5.97 minutes</td>
</tr>
<tr>
<td>Time for transport to hospital:</td>
<td>6.47 minutes</td>
</tr>
<tr>
<td>Time for returning to base:</td>
<td>11.40 minutes</td>
</tr>
</tbody>
</table>

| Treatment time + Transference time | 22.41 minutes |

Table 3.1: Total time for treatment and transference, for A1 calls.

TIFAR reads a list consisting of all 6PP postal codes (within the RAV) from a file, and makes each postal code \( i \) a demand point with density \( d_i = 1 \). Calls are generated uniformly over all postal codes.

The reason that we have made this choice is that it is not easy to obtain a clean list with historical EMS data.

For further research, it is recommended to look into better ways to approximate the population density by contacting CBS or RIVM, and include the effect of demand points that have a higher density, such as nursing homes. In this way, calls can be generated more realistically.

Generating treatment time and transference time

At the moment of writing, not much research has been done into the distribution of treatment time and transference time. Therefore, we model the time to be a constant for calls of the same priority. For further research, it is recommended to look into these times.

A1 calls
From Zuidhof (2010) we know that handling an A1 call takes 46.25 minutes (46:15 minutes). This is the time from the moment the EMS vehicle gives status one on the mobilophone, until the designated EMS vehicles arriving back at the base or gets dispatched to a new call. The mean driving time of an EMS vehicle to the call scene is 5.97 minutes (5:58 minutes), see Boers (2010). From Călinescu (2009) we can see that in 2008 the driving time (from base to call location) was 5.90 minutes (5:54 minutes), and the time an EMS vehicle takes from the call location to the hospital was 6.40 minutes (6:25 minutes). We assume this ratio to be the same for calls in 2009. This gives us a transportation time from the call location to the hospital of \( \frac{5.90}{5.90} \cdot 5.97 \) minutes = 6.47 minutes (6:28 minutes). Also, Călinescu (2009) states that the time from a hospital to a base is 11.40 minutes (11:24 minutes). One might expect a lower amount of time, but sometimes EMS personnel goes to a store or takes a detour to the base: this is called cruising. Putting this data into a table gives Table 3.1.

Now, Section 4.1 of Călinescu (2009) says that the mean time for treatment is 19.23 minutes (19:14 minutes), and mean time for transference is 11.68 minutes (11:41 minutes). Her data includes A1, A2 and B calls. We assume this ratio to be the same for only A1 calls. This results in a treatment time of \( \frac{19.23}{19.23+11.68} \cdot 22.41 = 13.94 \) minutes (13:56 minutes) and a transfer time of \( \frac{11.68}{19.23+11.68} \cdot 22.41 = 8.47 \) minutes (8:28 minutes)

A2 calls
Calculating the treatment time and transference time for A2 calls goes similar to those of priority A1. We assume the ratios to be the same as before, and do calculations analog to those of A1 calls, see Table 3.2

This results in a treatment time of 9.33 minutes (9:20 minutes), and a transfer time of 5.67 minutes (5:40 minutes).
3.2 Program description

<table>
<thead>
<tr>
<th>Total call handling time:</th>
<th>50.10 minutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driving time:</td>
<td>11.37 minutes</td>
</tr>
<tr>
<td>Time for transport to hospital:</td>
<td>12.33 minutes</td>
</tr>
<tr>
<td>Time for returning to base:</td>
<td>11.40 minutes</td>
</tr>
</tbody>
</table>

- Treatment time + Transference time 15.00 minutes

Table 3.2: Total time for treatment and transference, for A2 calls.

General note
For declarable calls the ratio between treatment time and transfer time does not matter, because the EMS vehicle is just as long unavailable for redeployment as for a different ratio. However, for EHGV calls it does matter. EHGV calls do not have a transfer time.

Generating priority EHGV status

The distribution of the priority and the EHGV status can be determined by using available data of the past. In this stadium, B-calls are not taken into account by TIFAR because necessary data about the distribution became available too late for us to be able to implement it. In this way both priority and status can be modeled with a Bernoulli trial.

In Table 2.11 we can see that 25.7% of the calls is EHGV or loss. Each generated call has a probability of 25.7% of being EHGV, and 74.3% are declarable. Recall that loss calls are modeled as EHGV.

3.2.3 Renewal Loop

Instead of discussing the initialization first, we will look at the most important part of the program: the renewal loop (RL). The RL is exactly the same in both Visual Mode and Speedsimulation Mode.

Purpose

The main purpose of the renewal loop is to update the current state of the entire model, given the current model time. In every single time step the RL is called. How the time between two time steps $\Delta t$ is calculated will be part of a later discussion: this is the only point where the two modes differ as said in Section 3.2.1.

The renewal loop has the purpose to calculate:

- The current status of the calls, and it looks for newly generated calls in the accident generator.
- The current position of the EMS vehicle in the environment, and it changes the current status of the ambulances when one has arrived at its destination.
- The best assignment of available EMS vehicles to calls, and the RL lets the dispatch center (DC) confirm it.

Every time an EMS vehicle gets dispatched or becomes available for new dispatch, the relocation routine will be run. A more detailed discussion about relocations will be performed in Section 3.2.5.


3 TIFAR simulation package

3.2.4 Assignment EMS vehicles to calls, hospitals and bases

When a call occurs, we have to assign an EMS vehicle to the call. In this section we describe how the call handling works.

TIFAR has a queue that contains all calls. We assign EMS vehicles to calls in order of priority: first we assign vehicles to A1 calls on a first in first out basis, and if all A1 calls are served we start assigning vehicles to A2 calls on the same basis. The DC always assigns the closest available EMS vehicle to a call. A similar approach has been applied by Restrepo (2008). Other dispatch procedures are not found in the literature, and it is advised for future research to look into them.

Once an EMS vehicle is assigned to a call, the call will not be handled by another EMS vehicle. Not even when the other EMS vehicle gets available for dispatch while being closer to the call than the already assigned vehicle. In practice it rarely occurs that an EMS vehicle gets assigned to another call, so we do not think that this is a limitation while modeling the real situation. For further research we can see what the effect is on response times when including reassignments of calls to other EMS vehicles.

Each call is handled by exactly one EMS vehicle. Request for backup from other EMS teams or the Mobile Medical Team helicopter is not included. Although we do not have available data about the frequency of backups, EMS personnel has told us that this rarely occurs.

When driving to an A1 call we assume the vehicle has auditory and visual signals, and when driving to an A2 call we assume that the vehicle drives without them. Sometimes a vehicle drives with these signals to an A2 call, but this is not included in TIFAR. When driving to a hospital, we assume that the EMS vehicle goes with A2 speeds. In reality an EMS vehicle may drive with signals on to an hospital. We do not have statistics about the frequency by which either of them occur.

When a call is declarable, the patient will be brought to the nearest hospital. We assume that this corresponds well with the real situation, although there are cases in which a specialized hospital should be chosen, see Section 2.3.2.

When an EMS vehicle leaves a hospital, it will head to the nearest base unless a relocation decides otherwise. In the next section we discuss the relocation protocol.

3.2.5 Relocations

To relocate EMS vehicles over all bases, we use a heuristic. In this section we describe how our heuristic works. The real heuristic used by RAVAA is known to the researcher. However, for confidentiality reasons we create our own. Of course, these two heuristics have similarities.

An EMS vehicle is called involved with a base if one of the following conditions hold:

• The EMS vehicle is waiting at the base until a new call arrives.
• The EMS vehicle is driving to the base and is available for redispacth.

The function $\text{inv}(j)$ returns the amount of EMS vehicles that is involved with base $j \in W$. The function $\text{relocateAmbulance}(j_1, j_2)$ relocates an EMS vehicle that is involved with base $j_1 \in W$ to base $j_2 \in W$. When no EMS vehicle is involved with $j_1$, it sends an EMS vehicle with a minimal distance to $j_2$. When there are zero EMS vehicles involved with $j_1$ the function will not relocate an EMS vehicle.

While relocating, we make a distinction between three regions:
Relocations are designed by the following rules:

- North: Zaandam, Purmerend and Monnickendam.
- Center: Karpersweg and Valckenierstraat.
- South: Aalsmeer, Amstelveen and AMC.

If EMS vehicles do not get relocated and always go to the nearest base to wait until a new call arrives, they get drawn to the center. In the north, there are two hospitals. When an EMS vehicle delivers a patient to the BovenIJ hospital, it will return to the closest base which is Valckenierstraat in the center. In the south, there is a region where the closest base is Amstelveen, and the closest hospital is VUMC. After delivering a patient to the VUMC hospital the EMS vehicle will find that base Karpersweg is nearest, and will go to that place to wait until a new call arrives. Relocations are necessary to keep available EMS vehicles distributed well over the RAV.

Relocations are designed by the following rules:

- If there are not enough EMS vehicles available in the north, fill from center (if possible).
- If there are not enough EMS vehicles available in the south, fill from center (if possible).
- Try to keep EMS vehicles evenly distributed in the north, where Purmerend has a highest priority, followed by Zaandam and later on by Monnickendam. When an EMS vehicle from the center gets relocated to Purmerend, it will almost visit Zaandam.
- We try to have an equal amount of EMS vehicles on bases Karpersweg and Valckenierstraat.
- In the south: if there is only one EMS vehicle available we try to have one EMS vehicle available in Amstelveen or Aalsmeer. If there are more vehicles in the south, we try to keep Aalsmeer and AMC approximately evenly occupied.

3.2 Program description

```plaintext
// Distribute EMS vehicles in the north over the bases
if (inv("Zaandam") >= 0 & & inv("Purmerend") >= 0 & & inv("Monnickendam") >= 1) relocateAmbulance("Monnickendam", "Purmerend");
if (inv("Zaandam") >= 1 & & inv("Purmerend") >= 0 & & inv("Monnickendam") >= 0) relocateAmbulance("Zaandam", "Purmerend");
if (inv("Zaandam") >= 1 & & inv("Purmerend") >= 1 & & inv("Monnickendam") >= 0) relocateAmbulance("Zaandam", "Monnickendam");
if (inv("Zaandam") >= 2 & & inv("Purmerend") >= 2 & & inv("Monnickendam") >= 0) relocateAmbulance("Purmerend", "Monnickendam");
if (inv("Purmerend") >= 2 & & inv("Purmerend") >= 2) relocateAmbulance("Purmerend", "Zaandam");
if (inv("Purmerend") >= 1 & & inv("Purmerend") >= 1 & & inv("Monnickendam") >= 0) relocateAmbulance("Purmerend", "Monnickendam");
```

3.2 Program description

```plaintext
// South: try to keep Aalsmeer or Amstelveen occupied.
if (inv("Aalsmeer") >= 1 & & inv("Amstelveen") >= 0 & & inv("AMC") >= 0 & & inv("Karperweg") >= 1 & & inv("Valckenierstraat") >= 0) relocateAmbulance("Valckenierstraat", "Zaandam");
if (inv("Aalsmeer") >= 2 & & inv("Amstelveen") >= 0 & & inv("AMC") >= 0) relocateAmbulance("Aalsmeer", "Amstelveen");
if (inv("Aalsmeer") >= 0 & & inv("Amstelveen") >= 0 & & inv("AMC") >= 0) relocateAmbulance("Aalsmeer", "Amstelveen");
if (inv("Aalsmeer") >= 1 & & inv("Amstelveen") >= 0 & & inv("AMC") >= 0 & & inv("Karperweg") >= 0) relocateAmbulance("Amstelveen", "Aalsmeer");
if (inv("AMC") >= 1 & & inv("Amstelveen") >= 1 & & inv("AMC") >= 0) relocateAmbulance("Amstelveen", "Aalsmeer");
if (inv("AMC") >= 1 & & inv("Amstelveen") >= 0 & & inv("AMC") >= 0 & & inv("Karperweg") >= 1) relocateAmbulance("Karperweg", "Aalsmeer");
```

3.2 Program description

```plaintext
// If the north gets empty, fill from the city center (if possible)
if (inv("Zaandam") + inv("Purmerend") + inv("Monnickendam") < 1 & & inv("Karperweg") + inv("Valckenierstraat") >= 3 & & inv("Karperweg") > 1 & & inv("Valckenierstraat") < 1) relocateAmbulance("Karperweg", "Amstelveen");
```

3.2 Program description

```plaintext
// If the south gets empty, fill it from the city center (if possible)
if (inv("Karperweg") >= 1 & & inv("Valckenierstraat") >= 1 & & inv("Karperweg") >= 1 & & inv("Valckenierstraat") >= 1) relocateAmbulance("Valckenierstraat", "Amstelveen");
```

3.2 Program description

```plaintext
// Try to keep Aalsmeer and AMC approximately evenly occupied.
if (inv("AMC") + inv("Aalsmeer") >= 1 & & inv("AMC") + inv("Amstelveen") >= 1 & & inv("AMC") + inv("Aalsmeer") >= 1 & & inv("Amstelveen") >= 1) relocateAmbulance("Aalsmeer", "Amstelveen");
```

3.2 Program description

```plaintext
// Center: try to keep Karpersweg almost as occupied as Valckenierstraat.
if (inv("Karpersweg") >= 1 & & inv("Valckenierstraat") >= 1) relocateAmbulance("Valckenierstraat", "Karpersweg");
if (inv("Karpersweg") >= 2 & & inv("Valckenierstraat") >= 2) relocateAmbulance("Valckenierstraat", "Karpersweg");
```
3.2.6 Initial amount of EMS vehicles

For the simulation we need a good estimate for the amount of EMS vehicles that handle the A1 and A2 calls. In this section we will find a suitable estimator for this amount.

In Kommer (2008) the advise for the amount of vehicles in RAV AA can be found, see Table 3.3

<table>
<thead>
<tr>
<th></th>
<th>Weekdays</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0:00-7:59</td>
<td>8:00-15:59</td>
<td>16:00-23:59</td>
<td></td>
</tr>
<tr>
<td>Zaanstreek-Waterland</td>
<td>5</td>
<td>8</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Amsterdam-Amstelland</td>
<td>9</td>
<td>31</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>RAV AA</td>
<td>14</td>
<td>39</td>
<td>24</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Saturday</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0:00-7:59</td>
<td>8:00-15:59</td>
<td>16:00-23:59</td>
<td></td>
</tr>
<tr>
<td>Zaanstreek-Waterland</td>
<td>5</td>
<td>7</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Amsterdam-Amstelland</td>
<td>9</td>
<td>18</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>RAV AA</td>
<td>14</td>
<td>25</td>
<td>19</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Sunday</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0:00-7:59</td>
<td>8:00-15:59</td>
<td>16:00-23:59</td>
<td></td>
</tr>
<tr>
<td>Zaanstreek-Waterland</td>
<td>6</td>
<td>7</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Amsterdam-Amstelland</td>
<td>10</td>
<td>15</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>RAV AA</td>
<td>16</td>
<td>22</td>
<td>19</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.3: Advised amounts of EMS for three time-intervals of a day by RIVM, see Kommer (2008).

The average amount of EMS vehicles is 23.8 over the day. We can calculate that 49.8% of the capacity is in use for A1 and A2 rides by using call amounts from Table 2.9, total call handling times from Table 3.1 and Table 3.2 and the fact that the mean time for a priority B call is 76.8 minutes (Zuidhof (2010), page 20).

This yields that an average of 11.81 EMS vehicles is in use to handle calls with priority A1 or A2. We do need a realistic number for the amount of EMS vehicles in TIFAR. For an approximation of the real number, we will perform an interpolation between the results for 11 and 12 EMS vehicles.

For the initial distribution of the vehicles we approach the quotient 5 : 9 for Zaanstreek-Waterland and Amsterdam-Amstelland.
3.2 Program description

3.2.7 Services from Connexxion

TIFAR makes use of third party services at Connexxion for the graphical user interface and route planning software. In this section we give a short overview of all external services used.

**Map server**
The map server has as input the RD-coordinates of the bottom left corner, output image dimensions and a scale factor, and returns the corresponding map in JPG format. It makes use of Tele Atlas data. This service is the only one that does not communicate through telnet: it uses http instead.

**Positionplanner**
The position planner has as input a postal code and house number, and returns the corresponding RD-coordinates. This is used to determine the location of the predetermined bases and hospitals, and to display it on the map.

**Geo-projector**
The geo-projector has as input RD-coordinates, and it returns a geo-projection that holds a position on an arc from Tele Atlas which is closest to the RD-coordinates. The geo-projector is used to determine the location of an available EMS vehicle that is not on a base.

**AB-planner**
The AB planner returns the shortest route between two points in RD coordinates. As input one gives the type of vehicle, start point and end point. This input can be an adres, a postal code or a geo-projection. TIFAR uses the latter two options.

**Matrix-planner**
The matrix planner has two modes: one-to-many and many-to-one. The many to one determines the distance and travel time from many locations to only one location. This is used to determine the closest available EMS vehicle: from many vehicles to one call location. The one to many determines the distance and travel time from one location to many locations. This is used to determine the closest hospital from a call location, or to determine the closest base from a hospital when an EMS vehicle returns to base. Input can be both a postal code or geo-projection.

Both the AB-planner and Matrix-planner use the bidirectional A* algorithm to determine shortest paths, and Tele Atlas data for road network information.
### 3.2.8 The Renewal Loop in pseudocode

**CALLS**

- `dispatchcenter::renewDispatch()` → `accidentgenerator::renewCallList()`
  - If (there are no calls in the future)
    - Make 2 future calls.
  - For each future call
    - If (time of the call ≤ current model time)
      - Add this future call to the list of calls of the DC.
      - Delete this future call of the list with future calls.

**AMBULANCES**

- For every ambulance
  - `ambulance::renewPosition()` → `ambulance`
    - If (the ambulance’s origin = the ambulance’s destination)
      - `ambulance::switchStatus()`
    - If (route is clear) or (route.firstvertex = route.lastvertex)
      - Update xPos = ..., yPos = ...
      - Clear the route.
    - Else
      - Let the ambulance drive for `timenow − starttime` µsec.
      - Update xPos = ..., yPos = ... and location=...
      - If (the ambulance reached [past] its destination)
        - Clear the route.
      - `ambulance::switchStatus()`

**ASSIGNMENT**

- For every A1 call in the calllist
  - If (the call’s status = QUEUED_FOR_SERVICE)
    - Look for the nearest ambulance which is available.
    - If (there is an available ambulance [that is the closest].
      - Dispatch this ambulance to the call.

- For every A2 call in the calllist
  - If (the call’s status = QUEUED_FOR_SERVICE)
    - Look for the nearest ambulance which is available.
    - If (there is an available ambulance [that is the closest].
      - Dispatch this ambulance to the call.

- For every ambulance.
  - If (waiting time at hospital or at call location has finished).
    - Assign the ambulance to a new call or the nearest base.

- Make future calls.
- Parse future calls to the DC if they should not be in the future any more.
- Check if the EMS vehicle its status must change.
- If the EMS vehicle is on a vertex, the position can be easily obtained.
- If the EMS vehicle is on an arc, we will calculate its position, and at arrival update its status.
- Dispatch an available EMS vehicle to each A1 call.
- Dispatch an available EMS vehicle to each A2 call.
- Check if the treatment or transfer time of a vehicle has passed.
Chapter 4

Results

We will provide our results in the following manner. First we look into the data generated with the simulation package TIFAR. Later we look into the coverage per base. We end the chapter by simulating the effect of additional bases in the RAV.

4.1 Percentage in time

First we study the effect of the amount of EMS vehicles on response times. We run eleven simulations with TIFAR where we let the amount of EMS vehicles vary from 5 to 15. Each simulation will run for at least one hour, that corresponds with approximately 3 days of EMS movements where a total close to 500 A1 or A2 calls are handled. The simulation with 12 EMS vehicles has run for over 2 hours and simulates five days of EMS movements. The results of several simulations will be put in Table 4.2. The real statistics, with a mean amount of 11.81 EMS vehicles, is displayed in italic.

If we take the weighted average of our simulation results we get Table 4.1.

<table>
<thead>
<tr>
<th></th>
<th>Mean served (A1)</th>
<th>Percentage on time (A1)</th>
<th>Mean severed (A2)</th>
<th>Percentage on time (A2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weighed mean</td>
<td>6m 10s</td>
<td>91.0%</td>
<td>9m 49s</td>
<td>98.1%</td>
</tr>
<tr>
<td>Real statistics</td>
<td>5m 58s</td>
<td>91.7%</td>
<td>11m 22s</td>
<td>87.8%</td>
</tr>
</tbody>
</table>

Table 4.1: The weighted statistics from the simulation (11 and 12 vehicles) and real EMS statistics.

Our results for A1 calls comes close to the real statistics. Our mean driving time differs 12 seconds from the actual mean. In percentage of vehicles in time we see that our results differ 0.7% from the real data. This is also close to the actual data.

For A2 calls, we see that our results differ a lot from reality. The driving time of the simulation is two minutes lower than the actual driving time. This result in a difference of 10.3% with the actual statistics. Our current version of TIFAR should not be used to predict effects on A2 calls because these differences are too large.

In Chapter 5 we shall discuss the validity of these results and give ideas for improvements.
## 4 Results

<table>
<thead>
<tr>
<th>Nr. EMS vehicles</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total time (µs)</strong></td>
<td>242 921 535 490</td>
<td>267 758 276 400</td>
<td>261 557 200 260</td>
<td>256 175 635 680</td>
</tr>
<tr>
<td><strong>Total time (textual)</strong></td>
<td>02d 19h 28m 42s</td>
<td>03d 02h 22m 38s</td>
<td>03d 00h 39m 17s</td>
<td>02d 23h 09m 36s</td>
</tr>
<tr>
<td>Calls served:</td>
<td>420</td>
<td>439</td>
<td>427</td>
<td>449</td>
</tr>
<tr>
<td>Mean time:</td>
<td>15m 57s</td>
<td>11m 54s</td>
<td>9m 28s</td>
<td>8m 11s</td>
</tr>
<tr>
<td>Too late:</td>
<td>278</td>
<td>198</td>
<td>140</td>
<td>107</td>
</tr>
<tr>
<td>Percentage on time:</td>
<td>33.81%</td>
<td>54.90%</td>
<td>67.21%</td>
<td>76.17%</td>
</tr>
<tr>
<td>Calls served:</td>
<td>57</td>
<td>68</td>
<td>76</td>
<td>71</td>
</tr>
<tr>
<td>Mean time:</td>
<td>18m 01s</td>
<td>14m 41s</td>
<td>11m 45s</td>
<td>12m 32s</td>
</tr>
<tr>
<td>Too late:</td>
<td>23</td>
<td>13</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>Percentage on time:</td>
<td>59.65%</td>
<td>80.88%</td>
<td>93.42%</td>
<td>87.32%</td>
</tr>
<tr>
<td>Calls served:</td>
<td>477</td>
<td>507</td>
<td>503</td>
<td>520</td>
</tr>
<tr>
<td>Mean time:</td>
<td>16m 12s</td>
<td>12m 17s</td>
<td>9m 47s</td>
<td>8m 47s</td>
</tr>
<tr>
<td>Too late:</td>
<td>301</td>
<td>211</td>
<td>145</td>
<td>116</td>
</tr>
<tr>
<td>Percentage on time:</td>
<td>36.90%</td>
<td>58.38%</td>
<td>71.17%</td>
<td>77.69%</td>
</tr>
</tbody>
</table>

### A1

<table>
<thead>
<tr>
<th><strong>Calls served:</strong></th>
<th>560</th>
<th>516</th>
<th>531</th>
<th>1 year</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean time:</strong></td>
<td>7m 27s</td>
<td>6m 51s</td>
<td>6m 21s</td>
<td>5m 58s</td>
</tr>
<tr>
<td><strong>Too late:</strong></td>
<td>75</td>
<td>52</td>
<td>61</td>
<td>4 294</td>
</tr>
<tr>
<td><strong>Percentage on time:</strong></td>
<td>82.19%</td>
<td>88.10%</td>
<td>89.87%</td>
<td>91.7%</td>
</tr>
</tbody>
</table>

### A2

<table>
<thead>
<tr>
<th><strong>Calls served:</strong></th>
<th>85</th>
<th>79</th>
<th>123</th>
<th>10 608</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean time:</strong></td>
<td>10m 39s</td>
<td>9m 43s</td>
<td>10m 9s</td>
<td>11m 22s</td>
</tr>
<tr>
<td><strong>Too late:</strong></td>
<td>43</td>
<td>3</td>
<td>6</td>
<td>1 294</td>
</tr>
<tr>
<td><strong>Percentage on time:</strong></td>
<td>95.29%</td>
<td>96.20%</td>
<td>95.52%</td>
<td>87.8%</td>
</tr>
</tbody>
</table>

### Total

<table>
<thead>
<tr>
<th><strong>Calls served:</strong></th>
<th>56</th>
<th>51</th>
<th>63</th>
<th>518</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean time:</strong></td>
<td>7m 59s</td>
<td>7m 18s</td>
<td>6m 51s</td>
<td>6m 52s</td>
</tr>
<tr>
<td><strong>Too late:</strong></td>
<td>79</td>
<td>55</td>
<td>67</td>
<td>5 717</td>
</tr>
<tr>
<td><strong>Percentage on time:</strong></td>
<td>84.39%</td>
<td>89.34%</td>
<td>90.90%</td>
<td>91.1%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Nr. EMS vehicles</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th><strong>11.81</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total time (µs)</strong></td>
<td>242 741 034 780</td>
<td>246 927 201 480</td>
<td>352 430 568 000</td>
<td>31 556 926 000</td>
</tr>
<tr>
<td><strong>Total time (textual)</strong></td>
<td>02d 19h 25m 41s</td>
<td>02d 20h 35m 27s</td>
<td>04d 01h 53m 51s</td>
<td>1 year</td>
</tr>
<tr>
<td>Calls served:</td>
<td>421</td>
<td>437</td>
<td>602</td>
<td>53 293</td>
</tr>
<tr>
<td>Mean time:</td>
<td>7m 27s</td>
<td>6m 51s</td>
<td>6m 21s</td>
<td>5m 58s</td>
</tr>
<tr>
<td>Too late:</td>
<td>75</td>
<td>52</td>
<td>61</td>
<td>4 294</td>
</tr>
<tr>
<td>Percentage on time:</td>
<td>82.19%</td>
<td>88.10%</td>
<td>89.87%</td>
<td>91.7%</td>
</tr>
</tbody>
</table>

### A1

<table>
<thead>
<tr>
<th><strong>Calls served:</strong></th>
<th>85</th>
<th>79</th>
<th>123</th>
<th>10 608</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean time:</strong></td>
<td>10m 39s</td>
<td>9m 43s</td>
<td>10m 9s</td>
<td>11m 22s</td>
</tr>
<tr>
<td><strong>Too late:</strong></td>
<td>43</td>
<td>3</td>
<td>6</td>
<td>1 294</td>
</tr>
<tr>
<td><strong>Percentage on time:</strong></td>
<td>95.29%</td>
<td>96.20%</td>
<td>95.52%</td>
<td>87.8%</td>
</tr>
</tbody>
</table>

### A2

<table>
<thead>
<tr>
<th><strong>Calls served:</strong></th>
<th>506</th>
<th>516</th>
<th>531</th>
<th>63 901</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean time:</strong></td>
<td>7m 59s</td>
<td>7m 18s</td>
<td>6m 51s</td>
<td>6m 52s</td>
</tr>
<tr>
<td><strong>Too late:</strong></td>
<td>79</td>
<td>55</td>
<td>67</td>
<td>5 717</td>
</tr>
<tr>
<td><strong>Percentage on time:</strong></td>
<td>84.39%</td>
<td>89.34%</td>
<td>90.90%</td>
<td>91.1%</td>
</tr>
</tbody>
</table>

### Total

<table>
<thead>
<tr>
<th><strong>Calls served:</strong></th>
<th>977</th>
<th>609</th>
<th>481</th>
<th>523</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mean time:</strong></td>
<td>6m 41s</td>
<td>6m 47s</td>
<td>5m 46s</td>
<td>5m 44s</td>
</tr>
<tr>
<td><strong>Too late:</strong></td>
<td>74</td>
<td>41</td>
<td>15</td>
<td>14</td>
</tr>
<tr>
<td><strong>Percentage on time:</strong></td>
<td>92.43%</td>
<td>93.27%</td>
<td>96.88%</td>
<td>97.32%</td>
</tr>
</tbody>
</table>

![Table 4.2: Statistics generated by TIFAR simulations for 5 to 15 EMS vehicles, inclusive the actual statistic (italic).](image-url)
4.2 Coverage per base

The result for the simulation with 13 EMS vehicles can be found in Figure 4.1. We see that in the Zaandstad area a high percentage of calls exceed the maximum allowed latency. Also all A1 calls of Oosthuizen are just on time, and we see a weak coverage in part of west Amsterdam.

![Simulation result for the simulation with 13 EMS vehicles.](image)

This raises the question which parts of the RAV are covered by each base. In the remaining part of the section we will look into each separate base and discuss which parts of the RAV it can and cannot cover. This will be followed by a discussion which parts of the RAV are not reachable within the time standard of 15 minutes, and which bases are covered by only one base.

We do not have real data containing the locations where response times were exceeded.
4 Results

Purmerend

Figure 4.2: Base Purmerend. Reachable locations within 4:00 (red), 11:00 (blue) and 22:24 (green) minutes driving time.

Not Reachable: Because of the northern location within the city, it is hard to reach the southern part of the city or the most northern part of the RAV. Locations above and including the line Westbeemster, Noordbeemster, Beets and Warder cannot be reached within a response time of 15 minutes. Also municipalities Oostzaan and Landsmeer (including Purmerland) cannot be reached within the 15 minutes time standard.

Reachable: Purmerend, Middenbeemster, Oosthuizen, Edam, Volendam, Monnickendam, Ilpendam and Het Schouw. The A7 and A8 highways let an EMS vehicle reach Het Kalf and the northern part of Zaandam within 15 minutes. For A2-calls, everything north of the A9 highway can be reached.

Other RAVs: Only De Rijp (RAV NHN) can be reached within 15 minutes from base Purmerend.
Monnickendam

![Map of Monnickendam coverage]

**Figure 4.3:** Base Monnickendam. Locations that can be reached within 4:00 (red), 11:00 (blue) and 22:24 (green) minutes driving time.

**Not Reachable:** Because of the rural vicinity of base Monnickendam, it cannot reach a large proportion of the RAV’s population. The western part of Purmerend cannot be reached with a response time of 15 minutes. Municipality Landsmeer is not reachable within 15 minutes, neither is Ransdorp or Holysloot. Middenbeemster can also not be reached.

**Reachable:** Edam, Volendam, Purmer, Broek in Waterland, Zuiderwoude, Uitdam and Marken can be reached with a response time of 15 minutes. Also the major part of Amsterdam-Noord can be reached within 15 minutes: everything east of the Kadoelenweg and Buiksloterdijk and south of the A10 highway.

**Other RAV’s:** Other RAV’s cannot be reached with a response time of 15 minutes.
Zaandam

Figure 4.4: Base Zaandam. Locations that can be reached within 4:00 (red), 11:00 (blue) and 22:24 (green) minutes driving time.

Not Reachable: Base Zaandam is near the A7 and A8 highways, which has an advantage to get southwards of the Noordzeekanaal water. On the other hand, the western part of the RAV cannot be reached: polder De Buitenlanden that lies between Beverwijk and Assendelft. Also Krommeniedijk cannot be reached with a response time of 15 minutes.

Reachable: Zaandam, Krommenie, Wormerveer, Wormer, Jisp, Neck, Den Ijp, Watergang, Zunderdorp, Schellingwoude, Zuideinde, Nauerna, Assendelft, Weestzaan, Koog aan de Zaan, Oostzaan and Landsmeer can be reached with a response time of 15 minutes. Also the west part of Purmerend can be reached within 15 minutes of driving time: everything west of the EMS base. A major part of Amsterdam is reachable from base Zaandam. Zaandam is the only base that can reach the entire RAV for A2 calls.

Other RAV’s: Markenbinnen (RAV NHN) and Busch en Dam (RAV Kennemerland) can be reached with a response time of 15 minutes.
4.2 Coverage per base

Valckenierstraat

Figure 4.5: Base Valckenierstraat. Locations that can be reached within 4:00 (red), 11:00 (blue) and 22:24 (green) minutes driving time.

Not Reachable: Base Valckenierstraat lies within the city center of Amsterdam, which results in coverage of the entire city, except for Holysloot, Driemond and a region around Ruigoord. However, area’s south of the A9 highway cannot be reached from base Valckeniersstraat. Zaandam cannot be reached with a response time of 15 minutes, either. Neither can a part of Overdiemen.

Reachable: Oostzaan, Landsmeer, Watergang, Zunderdorp, Broek in Waterland, Ransdorp, Durgerdam, Diemen, Duivendrecht, Ouderkerk aan de Amstel and most parts of Amsterdam can be reached with a response time of 15 minutes. For A2 calls, everything south of Middenbeemster can be reached.

Other RAV’s: Schiphol Oost and Badhoevedorp (RAV Kennemerland) can be reached with a response time of 15 minutes. The same holds for Muiden (RAV Gooi en Vecht) and Abcoude (RAV Utrecht).
Karperweg

Figure 4.6: Base Karperweg. Locations that can be reached within 4:00 (red), 11:00 (blue) and 22:24 (green) minutes driving time.

Not reachable: EMS base Karperweg is located at the south-west part of the city of Amsterdam. Since it is close to the A10 highway, it can cover a large area. However, there are city parts that cannot be covered with a response time of 15 minutes: north of Ruigoord, Driemond, Overdiemen, IJburg Zuid, Ransdorp, Zunderdrop and Holysloot.

Reachable: Zaandam, Oostzaan, Landsmeer, Diemen, Duivendrecht, Ouderkerk aan de Amstel, Amstelveen, Bovenkerk and most parts of Amsterdam can be reached with a response time of 15 minutes. The entire RAV except for Warder and everything north of Oosthuizen can be reached within time standard A2.

Other RAV’s: Schiphol, Badhoevedrop, Lijnden, Zwanenburg and Halfweg (RAV Kennemerland), and Abcoude (RAV Utrecht), can be reached within the 15 minutes time standard.
AMC

**Figure 4.7:** Base AMC. Locations that can be reached within 4:00 (red), 11:00 (blue) and 22:24 (green) minutes driving time.

**Not Reachable:** Base AMC lies at the southeastern part of Amsterdam near the intersection of the A2 and A9 motorways. Not all of Amsterdam can be reached from base AMC. City parts Westpoort, Westpark, IJburg Zuid, (small part of) Centrum, (most of) Osdorp and (most of) Noord cannot be reached within the time standard of 15 minutes. Municipalities Uithoorn and Aalsmeer cannot be reached with a response time of 15 minutes, either.

**Reachable:** Municipalities Amstelveen, Ouder-Amstel and Diemen (except a small part of Overdiemen) can be reached within 15 minutes. Also most parts of Amsterdam (except northern and western parts) are reachable within the time standard of 15 minutes. For A2 calls, everything south and including Purmerend can be reached.

**Other RAV’s:** Abeoude, Baarnbrugge, Loenersloot, Vreeland, Nigtevecht, Loenen aan de Vecht and parts of Vinkeveen (RAV Utrecht), Schiphol-Oost (RAV Kennemerland) and Weesp, Muiden, Hakkelasbrug and the easten part of Muiderberg (RAV Gooi en Vecht) can be reached within 15 minutes.
Amstelveen

**Figure 4.8:** Base Amstelveen. Locations that can be reached within 4:00 (red), 11:00 (blue) and 22:24 (green) minutes driving time.

Not Reachable: Base Amstelveen is near the A9 motorway. This makes it easy to dispatch to both west and east. It is not possible to get far south: everything below the line Jan Ploegensluis and harbor Kudelstaart, Overdiemen, Zeeburg, Westpoort and everything above the water Noordzeekanaal cannot be reached within the time standard of 15 minutes.

Reachable: Amstelveen, Diemen, Ouderkerk aan de Amstel, Duivendrecht, Oosteednde, Aalsmeer, Vrouwenstroost, De Kwakel, Uithoorn, Nes aan de Amstel, Driemond and most of Amsterdam can be reached with a response time of 15 minutes.

Other RAV’s: Acoorde, Baarnbrugge, Loenersloot and Vreeland (RAV Utrecht) and Waarderpolder, Haarlemmerliede, Halfweg, Zwanenburg, Lijnden, Badhoevedorp, Schiphol, Rozenburg, Aalsmeerderbrug and a small part of Hoofddorp (RAV Kennemerland) can be reached with a time standard of 15 minutes.
Aalsmeer

Figure 4.9: Base Aalsmeer. Locations that can be reached within 4:00 (red), 11:00 (blue) and 22:24 (green) minutes driving time.

**Not Reachable:** Though base Aalsmeer is located south of the RAV, it is not possible to reach the municipality of *Ouder-Amstel*. Also, *Amsterdam* (and everything north of Amsterdam) cannot be reached, except for a small area in south-west Slotervaart.

**Reachable:** The time limit of 15 minutes for A1 calls can be reached in the following areas: Municipalities *Aalsmeer, Uithoorn, Amstelveen* (except for a small neighbourhood near Het Loopveld). For A2 calls, everything south of *Wormerveer, Broek in waterland* and *Ransdorp* can be reached with a response time of 30 minutes.

**Other RAV’s:** *De Hoef, Mijdrecht* (RAV Utrecht), *Nieuwveen, Noordeinde, Zevenhoven, Papenveer, Papenveer, Leimuiden* (RAV Hollands Midden), and *Burgerveen, Rijsenhout, Hoorddorp* and *Schiphol* (RAV Kennemerland) can be reached with a time standard of 15 minutes.
4.3 Coverage of Agglomeratie Amsterdam

In the previous section we discussed the covering of each independent base. In this section we will look to the complete picture, and analyze high and low covered areas in the RAV. First we discuss the non-covered areas and single covered areas with a response time of 15 minutes. Later we discuss what is well covered. Next we discuss the coverage for A2 calls and well covered areas in other RAV’s.

4.3.1 Not covered potential A1 call origins with a response time of 15 minutes

It is highly important to know which areas are not covered, and hence cannot be reached by an EMS vehicle within the given time standard. Areas that do not meet the standard, can be found in Table 4.4. All demand points that are not covered within 15 minutes are covered with only a few minutes more.

<table>
<thead>
<tr>
<th>Location</th>
<th>Municipality</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>De Buitenlanden</td>
<td>Zaanstad</td>
<td>The west part of the polder between Beverwijk and Assendelft cannot be reached within 15 minutes from any base.</td>
</tr>
<tr>
<td>Westbeemster</td>
<td>Beemster, and Zeevang</td>
<td>Also rural areas north to these towns cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Noordbeemster</td>
<td>Zaanstad</td>
<td>The entire town of Purmerland cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Beets</td>
<td>Landsmeer</td>
<td>The entire town of Krommeniedijk cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Etersheim</td>
<td>Amsterdam</td>
<td>The entire town Holysloot cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Warden</td>
<td>Zaanstad</td>
<td>Only a small part of Overdiemen cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Purmerland</td>
<td>Zaanstad</td>
<td>The entire town of Krommeniedijk cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Krommeniedijk</td>
<td>Zaanstad</td>
<td>The entire town of Krommeniedijk cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Holysloot</td>
<td>Amsterdam</td>
<td>The entire town Holysloot cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Overdiemen</td>
<td>Diemen</td>
<td>Only a small part of Overdiemen cannot be reached within 15 minutes.</td>
</tr>
<tr>
<td>Westpoort</td>
<td>Amsterdam</td>
<td>The Northwestern areas of the port of Amsterdam cannot be reached within 15 minutes.</td>
</tr>
</tbody>
</table>

Table 4.3: Demand points that cannot be reached with a response time of 15 minutes.

4.3.2 Single covered potential A1 call origins with a response time of 15 minutes

Single covered demand points are not covered when all EMS vehicles of a base get dispatched. A common way to solve this is trying to cover each demand point at least twice, see DSM and DDSM in Chapter 2.
4.3 Coverage of Agglomeratie Amsterdam

We shall see later on, that single covered demand points have a greater probability to get served too late. Bases with many single covered demand points should have a high priority to get full time occupied.

We split the single covered demand points per base:

<table>
<thead>
<tr>
<th>Base</th>
<th>Towns</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purmerend</td>
<td><em>Middenbeemster</em>, and <em>Oosthuizen.</em></td>
<td>These two towns are only covered by Purmerend.</td>
</tr>
<tr>
<td>Monnickendam</td>
<td><em>Uitdam</em>, and <em>Marken.</em></td>
<td>These two towns are only covered by Monnickendam.</td>
</tr>
<tr>
<td></td>
<td><em>Krommenie</em>,</td>
<td>These towns are only covered by base Zaandam.</td>
</tr>
<tr>
<td></td>
<td><em>Wormer</em>,</td>
<td>Only the center part of Koog aan de Zaan is single covered.</td>
</tr>
<tr>
<td></td>
<td><em>Assendelft</em>, <em>Nauerna</em>, and <em>Koog aan de Zaan (partly).</em></td>
<td></td>
</tr>
<tr>
<td>Zaandam</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Valckenierstraat</td>
<td>None.</td>
<td></td>
</tr>
<tr>
<td>Karperweg</td>
<td><em>Ruigoord.</em></td>
<td>This part of Amsterdam can only be reached from base Karperweg.</td>
</tr>
<tr>
<td>AMC</td>
<td><em>Overdiemen</em> (partly).*</td>
<td>The part of Overdiemen that is covered, is only covered by base AMC.</td>
</tr>
<tr>
<td>Amstelveen</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Aalsmeer</td>
<td><em>Kudelstaart.</em></td>
<td>The southern half of the town and rural areas south of Kudelstaart are only reachable from Aalsmeer.</td>
</tr>
</tbody>
</table>

Table 4.4: Demand points that can only be reached by one base with a response time of 15 minutes, while driving at A1 speeds.

We can see that Zaandam should be covered at all times in order to cover the northern and western part of municipality Zaanstad.

4.3.3 Well covered areas in RAV Agglomeratie Amsterdam

In Figure 4.10 we see areas that can be covered within a response time of 15:00 minutes. Green parts are covered by at least 4 EMS bases. We see that this holds for *Amsterdam* (except IJburg Zuid), *Zaandam, oostzaan, Landsmeer, Ilpendam, Broek in Waterland, Amstelveen, Ouderkerk aan de Amstel and Diemen*.

Figure 4.11 shows us that most parts inside the ring road can be reached within 4:00 minutes of driving (a response time between 6 and 8 minutes). Also *Zaandam, Amstelveen, Aalsmeer* and the center and north of *Purmerend* can be reached within this time. There are not much areas which are double covered.
4.3.4 Coverage for A2 calls

In Figure 4.12 we see the coverage of EMS vehicles over RAV Agglomeratie Amsterdam. Every demand point in RAV Agglomeratie Amsterdam is covered by at least three EMS bases. As stated before, base Zaandam is the only base that can cover the entire RAV. Please note that this figure is created with A1 speeds.
speeds, and in reality it is rare that an EMS vehicle rides with auditory and visual signals for A2 calls.

One can see that an A2 call is handled on time if at least one EMS vehicle is available for dispatch in:

- Zaandam, or;
- At least one of the following bases: Purmerend and Monnickendam, and;
  At least one of the following bases: Valckenierstraat, Karperweg, Amstelveen and Aalsmeer.

Figure 4.11: Coverage of the road network of RAV Agglomeratie Amsterdam with a driving time of 4 minutes, while driving at A1 speeds.
Figure 4.12: Coverage of the road network of RAV Agglomeratie Amsterdam with a response time of 30 minutes, while driving at A1 speeds.

4.3.5 Coverage for other RAV’s

If a RAV has not got enough EMS vehicles at its disposal, it may ask for backup from another RAV. In this section we look into areas where RAV Agglomeratie Amsterdam can help neighbouring RAVs. In this thesis, we do not consider the effect of help on RAV Agglomeratie Amsterdam. Locations of bases in other RAVs can be found in Mulder (2009).

Locations in other RAVs reachable with a response time of 15 minutes from a base in RAV Agglomeratie Amsterdam can be found in Table 4.5. Especially Schiphol Airport is well covered. This can be explained because its municipality Haarlemmermeer was part of RAV Agglomeratie Amsterdam until 2008. Also Abcoude and vicinity was once part of RAV Agglomeratie Amsterdam, and is therefore well covered.
### 4.3 Coverage of Agglomeratie Amsterdam

#### Table 4.5: Towns in other RAVs that can be reached from a base in RAV AA within 15 minutes.

<table>
<thead>
<tr>
<th>Towns</th>
<th>Other RAV</th>
<th>Bases</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Burgerveen, Rijsenhout, and Hoofddorp</strong></td>
<td>Kennemerland</td>
<td>Aalsmeer.</td>
</tr>
<tr>
<td>Schiphol Airport</td>
<td>Kennemerland</td>
<td>Aalsmeer, Amstelveen, Karperweg, and Valckenierstraat (only Schiphol-Oost).</td>
</tr>
<tr>
<td><strong>Lijnden, and Badhoevedorp</strong></td>
<td>Kennemerland</td>
<td>Amstelveen, Karperweg, AMC (partly), and Aalsmeer (partly).</td>
</tr>
<tr>
<td><strong>Halfweg, and Zwanenburg</strong></td>
<td>Kennemerland</td>
<td>Amstelveen, and Karperweg.</td>
</tr>
<tr>
<td><strong>De Rijp</strong></td>
<td>Noord-Holland Noord</td>
<td>Purmerend.</td>
</tr>
<tr>
<td><strong>Weesp, Muiden, and Hakkelaarsbrug</strong></td>
<td>Gooi &amp; Vecht</td>
<td>AMC.</td>
</tr>
<tr>
<td><strong>Abcoude</strong></td>
<td>Utrecht</td>
<td>Karperweg, Valckenierstraat AMC.</td>
</tr>
<tr>
<td><strong>Baarnbrugge, Loenersloot, Vreeland, and Loenen aan de Vecht.</strong></td>
<td>Utrecht</td>
<td>AMC, and Aalsmeer (not Loenen aan de vecht).</td>
</tr>
<tr>
<td><strong>De Hoef, and Mijdrecht</strong></td>
<td>Utrecht</td>
<td>Aalsmeer.</td>
</tr>
<tr>
<td><strong>Nieuwveen, Noordeinde, Zevenhoven, Papenveer, and Leimuiden.</strong></td>
<td>Hollands Midden</td>
<td>Aalsmeer.</td>
</tr>
</tbody>
</table>
4.3.6 Nearest base to each location in RAV Agglomeratie Amsterdam

When the dispatcher decides which EMS vehicle should be dispatched to a call, it is important to know (or estimate) the distance in time from each available EMS vehicle to the call. In Figure 4.13 we see for each demand point what the nearest base is for A1 speeds.

Let us note some interesting facts. West Osdorp is closer to base Amstelveen than to base Karperweg. A quarter of Amsterdam Noord is closer to base Zaandam than to any base in Amsterdam. Edam-Volendam is closer to base Monnickendam than to base Purmerend. Neck is closest to Zaandam, whilst Jisp is closest to Purmerend.

Figure 4.13: Nearest Base to each location in RAV Agglomeratie Amsterdam.
4.3 Coverage of Agglomeratie Amsterdam

4.3.7 Nearest hospital from each location in RAV Agglomeratie Amsterdam

A patient often needs transportation to the nearest hospital. We can see for each location in RAV Agglomeratie Amsterdam what the nearest hospital in the RAV is. We assume that we drive with A2 speeds.

Again, we note some facts: Marken and Monnickendam are closer to BovenIJ than to Waterland. IJburg is closer to BovenIJ than to OLVG. Aalsmeer is closest to VUMC, whilst Uithoorn is closest to Amstelland Ziekenhuis.

![Map of nearest hospitals](image)

**Figure 4.14:** Nearest Hospital from each location in RAV Agglomeratie Amsterdam.
4.4 Adding additional bases

A valid question that rises is: where to locate new bases? In consultation with GGD we decided to add the bases from Table 4.6 to our simulation package.

<table>
<thead>
<tr>
<th>Nr.</th>
<th>Name</th>
<th>EMS provider</th>
<th>Inhabitants Code</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Wandelweg 5</td>
<td>1521 AA</td>
<td>1521AA00005</td>
</tr>
<tr>
<td></td>
<td>Statenjachtstraat 1</td>
<td>1034 CS</td>
<td>Amsterdam 1034CS00001</td>
</tr>
</tbody>
</table>

Table 4.6: Added bases in RAV Agglomeratie Amsterdam

The locations of the other nine bases can be found in Table 2.1. A complete overview of all hospitals and bases is displayed in Figure 4.15.

Figure 4.15: Hospitals and bases in RAV Amsterdam including the two additional bases.
4.4 Adding additional bases

We add the following relocation rules that state that Wormerveer gets used if there is enough capacity in Purmerend and Zaandam, and BovenIJ gets used if there is enough capacity in Valckenierstraat and Karperweg.

```plaintext
// Include base Wormerveer.
if (inv("Zaandam") + inv("Purmerend") >= 3 & inv("Zaandam") >= 2 & inv("Wormerveer") == 0) relocateAmbulance("Zaandam", "Wormerveer");
if (inv("Purmerend") + inv("Wormerveer") >= 3 & inv("Purmerend") >= 2 & inv("Wormerveer") == 0) relocateAmbulance("Purmerend", "Wormerveer");

// Include base BovenIJ.
if (inv("Valckenierstraat") >= 2 & inv("Karperweg") >= 1 & inv("BovenIJ") == 0) relocateAmbulance("Valckenierstraat", "BovenIJ");
if (inv("Karperweg") >= 2 & inv("BovenIJ") == 0) relocateAmbulance("Karperweg", "BovenIJ");
```

When we run a simulation with a length of approximately 4 days (model time), we get the results as displayed in Figure 4.16.

![Simulation result for the simulation with 13 EMS vehicles, including the two newly added bases.](image)

**Figure 4.16:** Simulation result for the simulation with 13 EMS vehicles, including the two newly added bases.

We notice a slight improvement of the amount of calls getting reached in the Zaanstad area. Instead of a mean driving time of 5:50 minutes we now have 5:47 minutes, which is 3 seconds less. A response time of at most 15 minutes was be reached in 92.68% of the A1 calls. Now this amount is 94.53%, which is an improvement of 1.85%.

In the next chapter we will discuss the validity of our models and the results of our models.
Chapter 5

Validity and Discussion

Using a model has advantages and disadvantages. In this chapter we will discuss the validity of the TIFAR simulation package. Later we have a discussion in which we give advice for future research.

5.1 Validity

5.1.1 Regional borders

In reality, RAV Agglomeratie Amsterdam is divided in three regions: Zaanstreek-Waterland (region 11), Amsterdam and Amstelland (region 13 without Amsterdam). EMS vehicles stay most of the time within their own region, and only cross these borders if it is really necessary. TIFAR tries to approximate this by considering regions north, center and south when doing relocations, see Section 3.2.1. We do think that this relocation effect is quite similar to practice and will not alter our results significantly.

5.1.2 Agreement between GGD and VZA

Although GGD and VZA do have their own regions, TIFAR views all EMS vehicles as equal. In reality GGD vehicles enter VZA territory and the same happens the other way around. TIFAR has a relocation rule implemented that regulates a equal distribution of EMS vehicles on Karperweg (VZA) and Valckenierstraat (GGD). We assume that this simulate the effect we wished for.

5.1.3 Assistance from other EMS teams

When an accident is severe enough, like at a drowning or reanimation, two vehicles are dispatched. It is also possible that a mobile medical team (helicopter) provides assistance. TIFAR does not include backup, which means that each call will be handled by exactly one EMS vehicle. Although we do not have real statistics about these situations, it is said by EMS personnel that they are rare. Therefore we think that it will be insignificant for our results.
5 Validity and Discussion

5.1.4 Assistance from other RAVs

In total, Amsterdam provided help for 516 A1 calls and 273 A2 calls to other RAVs, and got assistance for 172 A1 calls and 53 A2 calls from other RAVs, see Boers et al. (2010) page 74 and Section 2.3.2. This amount is a very small percentage of the total amount of calls, and thus we think that we may neglect the effect of them. Assistance from or to other RAVs is not implemented in TIFAR.

5.1.5 No B calls

TIFAR does not simulate B calls. We did subtract the average amount of EMS vehicles handling B calls from the average total amount of EMS vehicles to obtain a good estimation for the amount of vehicles that are available for A1 and A2 calls.

Zuiddorf (2010) states referring to GGD data (excluding VZA data): ‘Since hospitals plan the most surgeries during weekdays and hence admit less patients during the weekend, the number of B rides per day shows a clear difference between weekdays and weekend, while the A1 rides occur more frequently during the weekend. This can explain the negative correlation of -0.19 between the daily A1 and daily B calls. While the A2 calls have a correlation of 0.18 with the B rides.’ Note that these correlations are close to zero thus the effects are rather small. Hence, we do not think that these correlations can give us any information about the outcome of our simulations.

When there is not enough capacity, EMS dispatchers will in the first place reduce B calls, in the second place reduce A2 calls and if really necessary, A1 performance will be reduced (and other RAV’s will be asked for assistance). Because B calls are not included in our simulations, we expect that estimations of A2 calls are not close to realistic values. The effect on A1 calls may be assumed smaller than the effect on A2 calls, but it is hard to say what the effect will be.

For future research, it is advisable to look into the effect of decreasing and increasing amount of B calls on the response times of A1 and A2 calls.

5.1.6 Inaccurate speeds and road plan

In Amsterdam, there are special lanes for public transportation that are also used by EMS personnel, see Section 2.3.2. Also additional exemptions are in place. However, TIFAR uses the road plan of TeleAtlas. This map does not include these exemptions, and therefore the EMS vehicles will follow all rules in traffic. We expect a decrease performance due to this fact. We therefore expect that real performance is better than the performance induced by TIFAR.

Also, the travel speeds are deterministic and time-independent. Some days when there are traffic congestions, speeds will be lower. The A1 travel speeds we used are CityGIS EMS speeds for the province Noord-Holland. These speeds are also used by RIVM and Rijkswaterstaat, and they are assumed to be the most reliable available speeds for EMS vehicles. We use linear interpolation to determine A2 speeds that are equal to the rest of the traffic: A1 speeds are 20 kph higher when the rest of the traffic drives at most 60 kph, and A1 speeds are 40 kph higher than the rest of the traffic when they drive at speeds exceeding 80 kph. We use linear interpolation between 60 kph and 80 kph. We base this scale on the experience of the research at the EMS service.

The effect on the performance of A1 and A2 calls is hard to say. The not-included road exemptions cause worse performance, while the effect of the approximated deterministic speeds can both increase or
decrease performance. Making the model time-dependent will produce more accurate results. This will also an advice for future research.

5.1.7 Distribution of accidents

In TIFAR, accidents are distributed by a homogeneous Poisson process, where every 6 position postal code in the RAV has the same probability of having an accident. In reality there are postal codes where an accident occurs with a higher probability, for example nursing homes. There are also more possibilities to choose an accident generator, see Section 3.2.2 for possible choices. A better accident generator will produce more accurate results.

5.1.8 Choice of EMS hospitals by an EMS vehicle

In Section 2.3.2 we stated the protocol by which EMS personnel chooses the hospital where a patient will be brought to. In practice this will be the closest hospital in most cases. (We do not have numbers to support this claim.) TIFAR always brings the patient to the closest hospital.

While driving to an hospital, it rarely occurs that an EMS vehicle goes with optical and auditory signals. We have modeled the speed from the call location to the hospital by A2 speeds.

Choosing the closest hospital will have a positive effect on the mean response time and performance, since the EMS vehicle will become sooner available for a new dispatch. Not driving with A1 speeds to the hospital will have a negative effect on the mean response time and performance. We assume that these effects will counteract each other. In future research it is advisable to include actual data about the amounts of people brought to each type of hospital and the frequency a patient is brought to a hospital with A1 speeds.

5.1.9 No staffing issues

In TIFAR we assume there is enough staff available to occupy as many EMS vehicles as we require. In practice this might not always be the case which results in less available EMS vehicles and thus a decrease of performance. This will result in a slight increase of response times in our results produced with TIFAR.

5.1.10 No refueling

In TIFAR, refueling is not included. Each eight hour shift an EMS vehicle must refuel two times, and thus is unavailable for new dispatch. Not including refueling will result in a slight increase of response times in our results produced with TIFAR.

5.1.11 Deterministic treating time and transfer time

Due to a lack of available data, we made both treating time (i.e. time spent at patient location) and transfer time (i.e. time spend to transfer the patient from EMS care into hospital care) deterministic. For more details about the calculations for these constants, see Section 2.3.2. Using real statistics or probabilistic approach will make TIFAR more accurate.
5.1.12 No cruising

When an EMS team may head back to a base, they often take a detour or visit a shop to buy some food. Instead of going to a base, an EMS team can request the DC to go to a square and wait there until a new call arrives. This is all called cruising.

In TIFAR, cruising is not implemented. When an EMS team goes to a supermarket, their response time will increase when a call arrives, which results in smaller response times in TIFAR. On the other hand, due to cruising, EMS vehicles might be better distributed over the area which can cause better response times.

5.1.13 Time-independency

TIFAR works time-independent. This holds for call volumes, travel speeds and amount of vehicles. During the night there are significantly less calls and vehicles than during the day time. Due to this time-independency there is no traffic congestion. The model will become more realistic if time dependency will be included.

5.1.14 Dispatch and Relocation rules

We dispatch on a first in first out basis, where all A1 calls must be handled before we start assigning EMS vehicles to A2 calls. This approximates the reality well because we hardly have calls that are queued for service. Also it happens quite frequently that an EMS vehicle that becomes available is much closer to a call than the already assigned EMS vehicle. In practice the call might get reassigned, but TIFAR will not apply a reassignment.

We used a set of relocation rules without proper mathematical background, see Section 3.2.5. We believe that these relocation rules are good, but we also think that they may be improved. In practice, relocation goes approximately the same way as our model, thus we do believe that this models the real situation quite accurately.

5.1.15 Initialization

At the start of the simulation, all EMS vehicles are available at their bases. In practice this will not occur very often. This results in low response times when the model starts. Too short simulation times will yield too optimistic results.

5.1.16 Influence of randomizer and stop condition

For the generation of calls we use a randomizer. When we do another simulation with the same static values, the randomizer gives us other call data and thus other results. We can counteract this effect when we seriously extend the simulation time, see the law of large numbers in Dekking et al. (2004). We now stop after 1 clock hour for each simulation, which is approximately 3 days of EMS movements in model time. When we extend the simulation time, TIFAR might crash due to memory errors. A solution to solve this problem is known, namely another way to store and clear paths, but it takes too much time
to implement this and we do not have enough time to run many long time simulations during the MSc project.

5.1.17 Concluding

We mentioned effects that are not included in TIFAR but do take place in reality, or effects that can be better implemented. Although TIFAR cannot help with predicting A2 performance because B calls are not implemented, our simulation results for A1 calls are very close to the actual statistics.

In our opinion, three following issues are influencing the results the most:

- No B calls, see Section 5.1.5. Not having B calls has a disastrous effect on our results of A2 calls.
- Inaccurate speeds, see Section 5.1.6. If the speed of EMS vehicles is not accurate, neither are our results.
- Time independency, see Section 5.1.13. Because the amount of vehicles is time-independent, we use a mean amount of EMS vehicles. Perhaps latencies occur more often during certain peeks.

TIFAR has several great features:

- It can hold large amounts of vertices (demand points, bases and hospitals).
- The graphic user interface can display effects of decisions at real time.
- It is easy to implement relocation rules.
- It is easy to vary the amount of EMS vehicles.

5.2 Future research

In Section 5.1 we already mentioned advise for future research. Let us mention some of those effects:

- Include B calls, see Section 5.1.5.
- Research to better fitting travel speeds for EMS vehicles, see Section 5.1.6.
- Making call volumes and travel speeds time dependent, see Section 5.1.13 and Section 5.1.7.
- Include transportation to various types of hospitals, see Section 5.1.8.
- Include cruising, see Section 5.1.12.

It is advisable for further research to investigate generic relocation rules. At this moment, TIFAR’s relocation rules are manually adjusted to the Amsterdam area. It is possible to create lookup tables for certain amounts of EMS vehicles, and relocate to approximate these schemes. These lookup tables can be created with static models as discussed in Chapter 2.

In the next chapter we will give a summary of the thesis.
Chapter 6

Summary

In this thesis we developed a simulation package called ‘Testing Interfaces For Ambulance Research’, or rather TIFAR, to evaluate EMS policies. Such a generic simulation tool is highly valued by EMS managers because the effect of certain decisions will become easy explainable by the illustrative graphic user interface.

In the second chapter we report on our literature review. Many governments measure the quality of EMS dispatch in the percentage of acute calls that is covered within a fixed time interval. In The Netherlands we have two acute priorities: the high priority A1 calls and lower priority A2 calls. Of all A1 calls 90% must be covered with a response time of at most 15 minutes. For A2 calls we have the time standard of 30 minutes. Because EMS vehicles have exemptions on the traffic laws, it is advisable not to rely on regular route planners but include additional lanes and maximal allowed driving times.

The EMS region RAV Agglomeratie Amsterdam has eight fixed EMS bases and nine hospitals for 1 246 872 inhabitants. A total 100 831 calls has been handled in the year 2009: of these calls 52.9% was classified as priority A1, 10.5% was classified as A2. Of these calls, 73.9% needed transportation to a hospital.

We discuss four possible ways for a route planner for TIFAR. Two of them are all-to-all shortest path algorithms (Floyd-Warshall and Johnsons), and the other two are one-to-one shortest path algorithms (GSP and A*). In early versions of TIFAR, Floyd-Warshall is used, later versions make use of a bidirectional A*.

Of all mathematical models we discuss, the dynamic model comes closest to the architecture design of TIFAR. Relocations of this model occur only at given moments in time, and the decision that is made depends on the system state of the model. We have made an effort to describe the models as clearly as possible. All mathematical models use the earlier discussed shortest path algorithms.

The third chapter focuses on the program design and the choices that are made during its design. There are two modes in which the program can run: the visual mode and the faster but non-visual speedsim mode. Accidents occur at a random postal code by a Poission point process, by the same frequency as the RAV’s year mean. Also other real stochastic processes are included.

Chapter 4 shows us that the results for A1 calls come very close to the actual data, both in driving time (12 seconds difference) and percentage of calls in time (0.7% difference). For A2 calls it is not recommended to use TIFAR because this data differs too much: the percentage of calls in time differs more than 10% from the real statistics. This can be explained by the not included B calls.
The current coverage of RAV Agglomeratie Amsterdam is good, but can be improved. There are only a few places that cannot be reached from a base with a response time of 15 minutes. Amongst them is the far northern part of the RAV, a region around the western border, and two very small villages in the center. Including two extra bases at Wormerveer and at the BovenIJ hospital improve the overall results slightly. Zaandam is the only base that can cover the entire RAV within the time standard for A2 calls.

The fifth chapter gives some points that can be improved. In order to predict A2 calls better, we should include B calls in TIFAR. The speeds seem to be fine, but it is recommended to look for better time-dependent travel time predictions. An inhomogeneous poisson point process fits better for call generation. Making the call generation and amounts of vehicles time dependent will improve the model as well. At this moment the model works with Amsterdam specific relocation rules. It is recommended for future research to determine good RAV unspecific relocation rules.
### Appendix A

#### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Augmentation</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALS</td>
<td>Advanced Life Support</td>
<td>Of the type A1.</td>
</tr>
<tr>
<td>BLS</td>
<td>Basic Life Support</td>
<td>Of the type A2.</td>
</tr>
<tr>
<td>EMS</td>
<td>Emergency Medical Service</td>
<td>An EMS vehicle is also known as an ambulance.</td>
</tr>
<tr>
<td>MKA DC</td>
<td>(Dutch:) MeldKamer Ambulancezorg Dispatch Center</td>
<td>(EMS) Dispatch Center. The control room which handles the calls and EMS vehicle movements. Also known as Control Room.</td>
</tr>
<tr>
<td>AG</td>
<td>Accident Generator</td>
<td>The model module which handles the creation of new calls.</td>
</tr>
<tr>
<td>RAV</td>
<td>(Dutch:) Regionaal Ambulance Voorziening</td>
<td>Region with one MKA assigned. There are 24 ambulance RAV’s in The Netherlands.</td>
</tr>
<tr>
<td>EHGv</td>
<td>(Dutch:) Eerste Hulp, Geen Vervoer</td>
<td>Type of accident in which the patient does not need transportation. (First Aid, No Transport)</td>
</tr>
<tr>
<td>VWS</td>
<td>(Dutch:) Voorwaarde scheppende rit</td>
<td>A relocation.</td>
</tr>
<tr>
<td>WAV</td>
<td>(Dutch:) Wet Ambulancevervoer</td>
<td>Law on EMS transportation.</td>
</tr>
<tr>
<td>WAZ</td>
<td>(Dutch:) Wet Ambulancezorg</td>
<td>Law on EMS care.</td>
</tr>
<tr>
<td>CBS</td>
<td>(Dutch:) Centraal Bureau voor de Statistiek</td>
<td>Statistics authority.</td>
</tr>
</tbody>
</table>
### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Augmentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSCM</td>
<td>Location Set Covering Model</td>
</tr>
<tr>
<td>MCLP</td>
<td>Maximal Covering Location Problem</td>
</tr>
<tr>
<td>BACOP</td>
<td>Backup Coverage Problem</td>
</tr>
<tr>
<td>DSM</td>
<td>Double Standard Model</td>
</tr>
<tr>
<td>TEAM</td>
<td>Tandem Equipment Allocation Model</td>
</tr>
<tr>
<td>MEXCLP</td>
<td>Maximum Expected Covering Location Problem</td>
</tr>
<tr>
<td>Rel-P</td>
<td>Reliability Perspective</td>
</tr>
<tr>
<td>TTM</td>
<td>Two-Tiered Model</td>
</tr>
<tr>
<td>DDSM(^4)</td>
<td>Dynamic Standard Model</td>
</tr>
<tr>
<td>ADP</td>
<td>Approximate Dynamic Programming</td>
</tr>
</tbody>
</table>
Appendix B

States within the TIFAR package

This appendix shortly describes some major components of TIFAR. We start with the real mobilophone states and the coupled states that a vehicle in TIFAR can be in. Later we describes the states of a call, and the way vertices are implemented. We end with the standard instances that shapes the entire program.

B.1 EMS vehicles

<table>
<thead>
<tr>
<th>Status</th>
<th>MPS</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GOING_TO_CALL</td>
<td>1</td>
<td>The EMS vehicle is heading a patient.</td>
</tr>
<tr>
<td>AT_CALL_SCENE</td>
<td>2</td>
<td>The EMS vehicle arrived at the patient’s location.</td>
</tr>
<tr>
<td>GOING_TO_HOSPITAL</td>
<td>3</td>
<td>The EMS vehicle is transporting the patient to the hospital.</td>
</tr>
<tr>
<td>AT_HOSPITAL</td>
<td>4</td>
<td>The patient is being transferred into the care of hospital staff.</td>
</tr>
<tr>
<td>RETURNING_TO_BASE</td>
<td>5</td>
<td>For modeling purposes, the returning to base has its own status. In reality it is considered CRUISING.</td>
</tr>
<tr>
<td>IDLE_AT_BASE</td>
<td>6</td>
<td>The EMS vehicle waits at the base for a call. Also being used while having a lunch/dinner break.</td>
</tr>
<tr>
<td>NR_8</td>
<td>8</td>
<td>Speak request with the DC. Used to give notice to the DC that the EMS personnel want to be contacted at the DC's convenience.</td>
</tr>
<tr>
<td>NR_9</td>
<td>9</td>
<td>Emergency speak request with the DC.</td>
</tr>
<tr>
<td>CRUISING</td>
<td>5</td>
<td>The EMS vehicle drives through the environment, or waits outside a base for a call (on a square or something).</td>
</tr>
<tr>
<td>RELOCATION</td>
<td>5</td>
<td>The EMS vehicle gets a VWS.</td>
</tr>
<tr>
<td>OFF_SHIFT</td>
<td>6</td>
<td>If an EMS vehicle is off shift, they cannot be dispatched†.</td>
</tr>
</tbody>
</table>

† Unless otherwise set in `ambulance::isAvailable()`.

Table A.1: The states of an ambulance
In the table we can find the program status used by the C++ enum `ambulance::status`. In reality the ambulance have a mobilophone in which they set their status, referred in this document to as MPS, by pressing a number (from 1 to 9 inclusive). The dispatch center (DC), can see this status. Numbers 8 and 9 are not actual ambulance states but requests for verbal communications with the DC. MPS numbers 7 and 0 are currently not in use.

For every call either the status EMS vehicles status will follow the procedure $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ (for declarable calls) or $1 \rightarrow 2 \rightarrow 5 \rightarrow 6$ (for EHGV calls 2.2.1) where the AMS vehicle can be dispatched to a new call while have MPS 5. There is one exception in reality: MPS 8 and 9 can be inserted at the ambulance personals' convenience. This exception isn't modelled in TIFAR.

### B.2 Call

A call can have the following states. They are defined in the file `call.h`.

<table>
<thead>
<tr>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUEUED_FOR_SERVICE</td>
<td>The patient is waiting for an EMS vehicle to be assigned to the call.</td>
</tr>
<tr>
<td>ASSIGNED_TO_AMBULANCE</td>
<td>An EMS vehicle has been assigned to the call, and is driving to the call's location.</td>
</tr>
<tr>
<td>AMBULANCE_AT_SCENE</td>
<td>The EMS vehicle has been arrived at the scene.</td>
</tr>
</tbody>
</table>
| PATIENT_BEING_BROUGHT_TO_HOSPITAL | The patient is underway to the hospital.  
|                                | The patient did not need transportation at the hospital, and the EMS vehicle is returning to base. |
| PATIENT_AT_HOSPITAL            | The patient has arrived at the hospital, and the call might be considered as closed. Also used when the call was an EHGV and the EMS vehicle arrives at base. |

<table>
<thead>
<tr>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>The ambulance will go to the call scene with <code>ambulance::blue=ON</code>.</td>
</tr>
<tr>
<td>A2</td>
<td>The ambulance will go to the call scene with <code>ambulance::blue=OFF</code>.</td>
</tr>
<tr>
<td>B</td>
<td>The ambulance will go to the call scene with <code>ambulance::blue=OFF</code>.</td>
</tr>
</tbody>
</table>

The call states are programmed by the C++ enum `call::status`. In the program, the calls will take every status, from top until bottom.

In theory, a call with A2 priority may have both optical and audible signals, thus `ambulances::blue=ON`. *Ambulances in-zicht 2009* [9], page 69, says this happened only 9 600 times of the total of 1 041 966 calls made in The Netherlands over the year 2009 (= 0.92%). Since this influence is nearly neglectable on our results, we shall omit this possibility in the model.
### B.3 Vertices

<table>
<thead>
<tr>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEMAND_POINT</td>
<td>Every vertex that has no base is considered a demand point.</td>
</tr>
<tr>
<td>POTENTIAL_AMBULANCE_LOCATION</td>
<td>The vertex has a base on it.</td>
</tr>
</tbody>
</table>

Table A.4.1: The states of a vertex.

Every vertex is either an **DEMAND_POINT** or a **POTENTIAL_AMBULANCE_LOCATION**. Being an hospital is not defined in a struct, but in the private boolean variable `vertex::hospital`, which has both an public accessor method and public mutator method. The variable `vertex::vertextype` is stored as an C++ enum in `vertex.h`.

### B.4 Standard instances

In the `main()` function of TIFAR some global object are defined, which are needed in most classes. These classes all hold pointers to them. For improved readability, these instances all have the same name in all classes. Because there are a lot of reserved variable names, it wasn’t always as possible to take the most obvious choices. Below one can find an overview of the names of these instances. Keep in mind that C++ is case-sensitive.

<table>
<thead>
<tr>
<th>Class</th>
<th>Instance</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>environment</td>
<td>plan</td>
<td>Holds the streetplan: vertices and arcs. Also for shortest distance calculations.</td>
</tr>
<tr>
<td>timer</td>
<td>watch</td>
<td>Holds the current time, in both speed-modes. Can also convert time to various formats. Stores also the time stamps in the speedsimulation modus.</td>
</tr>
<tr>
<td>logbook</td>
<td>journal</td>
<td>Holds the logbook of all important events. Used by calculation of all bench marks.</td>
</tr>
<tr>
<td>accidentgenerator</td>
<td>accidentGenerator</td>
<td>Creates accidents by using a Poisson process. It also parses the calls to the dispatch center</td>
</tr>
<tr>
<td>dispatchcenter</td>
<td>dispatchCenter</td>
<td>The dispatch center (DC) has a list with all calls. It also assigns ambulances to the calls, and regulates the ambulance relocations (Dutch: VWS, voorwaardeschepende rit).</td>
</tr>
<tr>
<td>speedsimulator</td>
<td>speedSimulationEngine</td>
<td>When in speedsimulation mode, this function handles the jump to the next time step and the entire model renews its state. It also checks for the stopping criterium.</td>
</tr>
<tr>
<td>interface</td>
<td>gui</td>
<td>When in visual mode, this function handles the jump to the next time step and the entire model renews its state. It also plots the environment.</td>
</tr>
</tbody>
</table>
## Appendix C

### Used notations

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
<th>Used notations</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V$</td>
<td>Set of demand points.</td>
<td>LSCM, MCLP, BACOP1, BACOP2, DSM, TEAM, MEXCLP, MALP1, MALP2, Rel-P, and TTM.</td>
</tr>
<tr>
<td>$W$</td>
<td>Set of base locations.</td>
<td>LSCM, MCLP, BACOP1, BACOP2, DSM, TEAM, MEXCLP, MALP1, MALP2, Rel-P, MALP1, MALP2, TTM, and ADP.</td>
</tr>
<tr>
<td>$\mathcal{V} = V \sqcup W$</td>
<td>Set of vertices of the road network.</td>
<td>Subsection 2.2.4: Road plan and Section 2.4.1: Shortest Path Introduction.</td>
</tr>
<tr>
<td>$\mathcal{A}$</td>
<td>Set of arcs in the road network.</td>
<td>Subsection 2.2.4: Road plan</td>
</tr>
<tr>
<td>$G = (\mathcal{V}, \mathcal{A})$</td>
<td>The graph representing the road network.</td>
<td>Subsection 2.2.4: Road plan and Section 2.4.1: Shortest Path Introduction.</td>
</tr>
</tbody>
</table>

### Traveling through the graph

- $s \in \mathcal{V}$: Starting point of a shortest path that we try to find.
- $t \in \mathcal{V}$: End point of a shortest path that we try to find.
### C. Used notations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Equations</th>
<th>Related Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_i$</td>
<td>Population density around demand point $i \in V$.</td>
<td></td>
<td>MCLP, BACOP1, BACOP2, DSM, MEXCLP, TTM, MALP1, and MALP2.</td>
</tr>
<tr>
<td>$d[i,j] \in \mathbb{R}$</td>
<td>Arc weight of arc $(i,j) \in A$.</td>
<td></td>
<td>Section 2.4.1: Shortest Path Introduction.</td>
</tr>
<tr>
<td>$t[i,j] \in \mathbb{R}$</td>
<td>Length of shortest path from vertex $i \in V$ to vertex $j \in V$.</td>
<td></td>
<td>Section 2.4.1: Shortest Path Introduction, and Subsection 2.4.3: GSP.</td>
</tr>
<tr>
<td>$m[i,j] \in \mathbb{R}$</td>
<td>Temporary best found approximation for $t[i,j]$ during an algorithm.</td>
<td></td>
<td>Subsection 2.4.2: Floyd-Warshall.</td>
</tr>
<tr>
<td>$v[i,j] \in \mathbb{R}$</td>
<td>Predecessor of vertex $j$ on the shortest path from $i \in V$ to $j \in V$.</td>
<td></td>
<td>Subsection 2.4.2: Floyd-Warshall, and Subsection 2.4.3: GSP.</td>
</tr>
<tr>
<td>$v(i) := v[s,i]$</td>
<td>Predecessor of vertex $i \in V$ on the shortest path from $s$ to $i \in V$.</td>
<td></td>
<td>Section 2.4.3: GSP.</td>
</tr>
</tbody>
</table>

### Coverage of demand points

- $r$ Maximum allowed drive time for a call, when there is only one time constraint.  
  | LSCM, MCLP, BACOP1, BACOP2, DSM. |
- $r_1$ Maximum allowed drive time for a priority A1 call, when there are two time constraints.  
  | DSM. |
- $r_2$ Maximum allowed drive time for a priority A1 call, when there are two time constraints.  
  | DSM. |
- $r^A$ Maximum allowed drive time for an ALS vehicle, when there are two types of vehicles.  
  | TEAM, and TTM. |
- $r^B$ Maximum allowed drive time for an BLS vehicle, when there are two types of vehicles.  
  | TEAM, and TTM. |

#### Bases covering node $i \in V$ within time $r$.  
- $W_i = \{j \in W \mid t[i,j] \leq r\}$  
  | LSCM, MCLP, BACOP1, BACOP2, MALP1, and MALP2. |
- $W_i^1 = \{j \in W \mid t[i,j] \leq r^1\}$  
  | DSM, and DDSSM$^4$. |
- $W_i^2 = \{j \in W \mid t[i,j] \leq r^2\}$  
  | DSM, and DDSSM$^4$. |
- $W_i^A = \{j \in W \mid t[i,j] \leq r^A\}$  
  | TEAM, and TTM. |
- $W_i^B = \{j \in W \mid t[i,j] \leq r^B\}$  
  | TEAM, and TTM. |
Binary variable that equals 1 iff \( i \in V \) is covered by at least one EMS vehicle.

Binary variable that equals 1 iff \( i \in V \) is covered by at least two EMS vehicles.

Binary variable that equals 1 iff \( i \in V \) is covered by at least \( k \in \mathbb{N} \) EMS vehicles.

Binary variable that equals 1 iff \( i \in V \) is covered by exactly \( k \in \mathbb{N} \) EMS vehicles.

Amount of EMS vehicles on base \( j \in W \)

Amount of EMS vehicles of type ALS on base \( j \in W \)

Amount of EMS vehicles of type BLS on base \( j \in W \)

Number of EMS vehicles

Number of ALS vehicles

Number of BLS vehicles

Maximum amount of vehicles base \( j \in W \) can hold.

Number of calls in the system

Number of vertices in shortest path problems.

Number of demand points.

Number of EMS vehicles dispatch

Status of the EMS vehicle.

Origin of the current (or: last) EMS movement.

Destination of the current (or: last) EMS movement.

Starting time of the current EMS movement.
C Used notations

\[ \sigma_c \] Status of the call. \hspace{1cm} \text{Section 2.2.1: Dispatch Process, ADP}

\[ o_c \] Origin of the call. \hspace{1cm} \text{Section 2.2.1: Dispatch Process, ADP}

\[ t_c \] Time the call came into the system. \hspace{1cm} \text{Section 2.2.1: Dispatch Process, ADP}

\[ p_c \] Priority of the call. \hspace{1cm} \text{Section 2.2.1: Dispatch Process, ADP}

\[ a := (\sigma_a, o_a, d_a, t_a) \] Tuple describing one EMS vehicle. \hspace{1cm} \text{Section 2.2.1: Dispatch Process, ADP}

\[ c := (\sigma_c, o_c, t_c, p_a) \] Tuple describing one call. \hspace{1cm} \text{Section 2.2.1: Dispatch Process, ADP}

\[ A := (a_1, a_2, \ldots, a_M) \] Vector holding all EMS vehicles. \hspace{1cm} \text{Section 2.2.1: Dispatch Process, ADP}

\[ C := (c_1, c_2, \ldots, c_N) \] Vector holding all calls. \hspace{1cm} \text{Section 2.2.1: Dispatch Process, ADP}

\[ \tau \text{ (or: } \tau(s) \text{)} \] Current time of the state \( s \) of the system. \hspace{1cm} \text{ADP, and DDSM four}

\[ e \text{ (or: } e(s) \text{)} \] Event that triggered the state \( s \) of the system. \hspace{1cm} \text{ADP, and DDSM four}

\[ s := (\tau, e, A, C) \] The state the system is in. \hspace{1cm} \text{ADP, and DDSM four}

\[ \{s_k | k = 1, 2, 3, \ldots\} \] State trajectory. \hspace{1cm} \text{ADP, and DDSM four}

\[ u_{ij} \text{ (or: } u_{ij} \text{)} \] Binary variable only equalling one if there is a relocation of vehicle \( 1 \leq i \leq M \) to base \( j \in W \). \hspace{1cm} \text{A relocation scheme.}

\[ u(s) := \{u_{ij} : j \in W\} \] Set containing all feasible relocations. \hspace{1cm} \text{ADP}

\[ \mathcal{U}(s) := \left\{ u(s) \in \{0, 1\}^{|W|} : \sum_{j \in W} u_{ij}(s) = 1 \right\} \] Policy we follow while the system is in state \( s \). \hspace{1cm} \text{ADP}

\[ p(s) \in \mathcal{U} \] Optimal policy we can follow while the system is in state \( s \). \hspace{1cm} \text{ADP}

\[ p^*(s) \in \mathcal{U} \] \hspace{1cm} \text{ADP}

\[ \{(s_k, u_k(s_k)) | k = 1, 2, 3, \ldots\} \] System trajectory while following policy \( u \). \hspace{1cm} \text{ADP}

\[ w(s, u) \] Probability function that influences a decision taken. \hspace{1cm} \text{ADP}

\[ f(s, u(\cdot), w(\cdot, \cdot)) \] Transfer function producing the next state of the system. \hspace{1cm} \text{ADP}

\[ \alpha \] Discount factor. \hspace{1cm} \text{ADP, DSM, and DDSM four}

\[ \alpha \] Percentage of calls that must be within a given fixed time standard. \hspace{1cm} \text{ADP}

\[ c(s, u, f(\cdot, \cdot, \cdot)) \] Binary cost function of following a relocation. \hspace{1cm} \text{ADP}

\[ c^j_k \] Costs of having \( k \) EMS vehicles on base \( j \in W \) at time \( t \). \hspace{1cm} \text{DDSM four}
\( J^p(s) \) & Expected costs when following policy \( p \). & ADP  
\( J(s) = J(p^*) \) & Minimized expected costs. & ADP  
\( K_q \) & Length for each simulation. \( K_q \) is large enough to simulate the time interval \([0, T]\) & ADP  
\( L \) & Number of basis functions we use to approximate \( J(s) \) & ADP  
\( Q \) & Number of replications while determining the trajectory of a policy. & ADP  
\( r := \{r_1, r_2, \ldots, r_K\} \) & Vector with tunable parameters to approximate \( J(s) \). & ADP  
\( \phi = \{\phi_1, \phi_2, \ldots, \phi_K\} \) & Vector with basis functions to approximate \( J(s) \). & ADP  
\( J(s, r) := \sum_{k=1}^{K} r_k \phi_k \) & Approximation of \( J(s) \). & ADP  
\( \{s^m_{n,q}|k = 1, 2, \ldots, K; q = 1, 2, \ldots Q\} \) & State trajectories. & ADP  
\( \{C^n_{q}|k = 1, 2, \ldots, K; q = 1, 2, \ldots Q\} \) & Total expected costs when following a policy. & ADP  

**TTM Specific**

\( h \) & The number of ALS-vehicles within time \( r_A \) of demand point \( i \in V \). & TTM  
\( k \) & The number of ALS-vehicles within time \( r_A \) of demand point \( i \in V \). & TTM  
\( \ell \) & The number of BLS-vehicles within time \( r_B \) of demand point \( i \in V \). & TTM  
\( h_i \) & Upper bound for \( h \). & TTM  
\( k_i \) & Upper bound for \( k \). & TTM  
\( \ell_i \) & Upper bound for \( \ell \). & TTM  
\( \theta_i^{h,k,\ell} \) & Probability that demand point \( i \in V \) is covered with the amounts of vehicles in its vicinity given by \( h, k \) and \( \ell \). & TTM  
\( \pi_i^{h,k,\ell} \) & Binary variable equal to one only if for demand point \( i \in V \) the amounts \( h, k \) and \( \ell \) are correct. & TTM  

**Other Model Specific**

\( E_k \) & Expected coverage demand. & MEXCLP  
\( a_{jk} \) & Logarithm of the probability that the number of EMS vehicles that depart from base \( B \) within a time interval of length \( T \) exceeds \( k \). & Rel-P  
\( c_{jk} \) & Costs of having \( k \) EMS vehicles on base \( j \in W \). & Rel-P  
\( b := \left\lceil \log(1 - \beta)/\log(q) \right\rceil \) & Minimal amount of vehicles which should cover each demand point. & MALP1  
\( b_i := \left\lceil \log(1 - \beta)/\log(q) \right\rceil \) & Minimal amount of vehicles which should cover demand point \( i \in V \). & MALP2  
\( b_i := \log(1 - \beta_i) \) & Constant variable for each \( i \in V \). & Rel-P
### C Used notations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Used in</th>
</tr>
</thead>
<tbody>
<tr>
<td>(q)</td>
<td>Probability that an EMS vehicle is in use.</td>
<td>MEXCLP, MALP1 and MALP2</td>
</tr>
<tr>
<td>(q_j)</td>
<td>Probability that an EMS vehicle on base (j \in W) is in use.</td>
<td>MALP2</td>
</tr>
<tr>
<td>(\beta)</td>
<td>Fixed probability that acts like an lower bound for the response time on each demand point.</td>
<td>MALP1</td>
</tr>
<tr>
<td>(\beta_i)</td>
<td>Fixed probability that acts like an lower bound for the response time on demand point (i \in V).</td>
<td>MALP2</td>
</tr>
<tr>
<td>(\theta \in [0, 1])</td>
<td>Weight factor for single covered demand points. BACOP2.</td>
<td>BACOP2.</td>
</tr>
</tbody>
</table>

#### Shortest Path Specific

- **Distance from the added vertex to vertex \(v \in V\).** \(g(v)\)
- **Length of the shortest path** \(\ell\)
- **Labels for length of the shortest path** \(d_1, d_2, \ldots, d_n\)
- **Graph after a certain amount of iterations** \(G^1, G^2, \ldots\)
- **Deleted nodes from a graph after a certain amount of iterations** \(i', i'', i''', \ldots\)
- **Estimated length (by a heuristic) for the distance (in time) between \(v \in V\) and end point \(t\).** \(h_v\)
- **Candidate list.** \(C\)
- **Candidate list in bidirectional search, from \(s \to t\).** \(C^s\)
- **Candidate list in bidirectional search, from \(t \to s\).** \(C^t\)
- **Estimated length (by a heuristic) for the distance (in time) between \(i \in V\) and end point \(t\) while performing a bidirectional search.** \(h_i^t\)
- **Estimated length (by a heuristic) for the distance (in time) between \(i \in V\) and start point \(s\) while performing a bidirectional search.** \(h_i^t\)
- **Modified arc lengths in a bidirectional search, from starting point \(s\).** \(d'[i, j]\)
- **Modified arc lengths in a bidirectional search, from end point \(t\).** \(d''[i, j]\)
- **Average of \(d'[i, j]\) and \(d''[i, j]\).** \(\bar{d}[i, j]\)
- **Set of permanently marked vertices in a bidirectional search, from starting point \(s\).** \(R^s\)
- **Set of permanently marked vertices in a bidirectional search, from end point \(t\).** \(R^t\)
Bibliography


BIBLIOGRAPHY


