EFFICIENT PLANNING OF AMBULANCE SERVICES

theory and practice

Martin van Buuren

Efficient Planning of Ambulance Services

Theory and Practice

Martin van Buuren

This work was supported by the Dutch Technology Foundation STW, which is part of the Netherlands Organisation for Scientific Research (NWO), and partly funded by the Ministry of Economic Affairs (grant no. 11986).

Cover artwork by: Nicole van den Berg
Printing by: Ipskamp Printing

ISBN: 978-94-028-1010-3

VRIJE UNIVERSITEIT

Efficient Planning of Ambulance Services

Theory and Practice

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad Doctor aan
de Vrije Universiteit Amsterdam,
op gezag van de rector magnificus
prof.dr. V. Subramaniam,
in het openbaar te verdedigen
ten overstaan van de promotiecommissie
van de Faculteit der Bètawetenschappen
op woensdag 16 mei 2018 om 9.45 uur
in de aula van de universiteit,
De Boelelaan 1105

door

Martin van Buuren

geboren te Velsen

promotoren:	prof.dr. R.D. van der Mei
	prof.dr. S. Bhulai

CONTENTS

DANKWOORD

Dit proefschrift is een mijlpaal die volgt op negen jaren werkzaam zijn in de wonderlijke wereld van de wiskunde achter de ambulancezorg. Dit werk had nooit van deze kwaliteit kunnen zijn zonder de inhoudelijke terugkoppeling en morele steun van veel mensen om mij heen. Aanvullend op de acknowledgements van elk hoofdstuk, bedank ik daarom de volgende personen.

Allereerst zijn dat mijn promotoren. Rob, dank je voor de vele gezellige besprekingen die richting gaven aan het onderzoek, en ook je gedetailleerde blik op de daaruit volgende papers en dit proefschrift. Maar de echte baas blijf je voor mij in situaties zoals in Berlijn, toen we in de sneeuwstorm terechtkwamen en ons uit bittere noodzaak urenlang in de kroeg moesten ophouden. Sandjai, heel veel dank voor de kritische kijk op de modellen, de bewijzen en de resultaten. Ik heb veel plezier beleefd aan het praten over onze gemeenschappelijke passie voor iSpul, en het nachtelijke wachten voor de nieuwste iPad. Bovendien wil ik de hele commissie bedanken voor het doornemen en het beoordelen van het proefschrift. Mijn paranimfen Jochem en Peter wil ik bedanken voor het bijstaan op de verdediging.

In het REPRO-onderzoeksproject had ik het genoegen om samen te mogen werken aan de ambulancemodellen. Pieter, je kritische blik is bewonderenswaardig, waar je in je carière volgens mij veel aan zal hebben. Als een ambassadeur van Oman heb je tot diep in de avond rake punten bij me weten te scoren. Thije, ik wil je in het bijzonder bedanken voor de data-analyses van de pilot en zal het uitwisselen van onze zomerkampervaring missen. Caro, ik vind het echt super knap hoe jij je kan focussen op wat je wil bereiken en dat altijd voor elkaar krijgt. Ik vond het onder andere erg fijn om samen met je de feedback van de pilot te analyseren. Theresia, dank je voor regelmatige sparren dat aan de basis staat aan het derde hoofdstuk. Met veel plezier denk ik ook terug aan ons dagje waarin we langs alle standplaatsen in de buurt hebben gereden. Geert Jan, heel erg bedankt dat je altijd bereid was om je enorme kennis over de rekenmodellen en de statistieken in de ambulancezorg met me te delen. Door je inzet is in het tweede hoofdstuk een goed meldkamermodel beschreven. Karen, heel erg bedankt dat je me met dit onderszoeksgebied in aanraking hebt laten komen.

Doordat, tot mijn grote teleurstelling, de omvang van dit dankwoord aan begrenzingen onderhevig is, zal het niet mogelijk zijn om per lotgenoot van de PNA2/ST/OBP/A&O (pick to your liking) een betoog te produceren dat recht doet tot je karakter. Ik prijs het rappe tempo waarmee jullie onzin tot je kunnen nemen, om het vervolgens te verwerken tot (ongetwijfeld) zinnigheid. Of was het andersom. Alleen daarvoor zou aan jullie al met terugwerkende kracht een (tweede) doctorsgraad verstrekt moeten worden. Japhne, Maria, J-P.L., Mr. Dingetje, Mondwater, Frol, DD, CFG, Karin, Frank, Guido, BB, Pet(e)r(a')s, Cré, Arnoud, Marijn, Ruben, Jaap, Bart, Ewan, April, Asparuh, Dennis, Rudi, Mellie, Paulien, Alwin, Jacky en jij die ik ongetwijfeld vergeten bent te noemen, voor elk van jullie kan ik vele argumenten verzinnen waarom je een geweldige toevoeging was aan mijn promotietijd. De gedachtewisselingen bij de koffie, de taalkundige hoogstandjes, de borrels, de etentjes naar werktijd en de meerdere reizen richting de Alpen, zal ik niet vergeten. René, dank je voor het aanreiken van het laatste puzzelstukje in de bewijzen van de *adjusted queuing*. Er zijn behoorlijk veel mensen bij het CWI, onder meer binnen de BHV en andere ondersteunende functies, waar ik veel gezelligheid mee beleefd heb en die ik daarvoor wil bedanken. Suus, dank je ook voor de meerdere hand en spandiensten die het promotieleven verlicht hebben.

De aansluiting met de praktijk is dankzij de inzet van velen die werkzaam zijn bij meerdere ambulancediensten en andere organisaties binnen de sector. Ik wil elk van jullie bedanken voor je inzet binnen het onderzoek, die ook ten goede is gekomen aan mijn proefschrift. Bij enkele ambulancediensten heb ik langere tijd intern mogen werken. Om het beginnen van het hele team van Future Technology, waar ik al voordat het onderzoek begon al een eerste versie van de simulatiesoftware kon maken. Deze versie gaf een opzet tot het pakket dat hier beschreven wordt in het laatste hoofdstuk.

Met veel plezier bracht ik de eerste jaren van mijn promotie een dag door bij Ambulance Amsterdam, waar ik me altijd welkom voelde. In het bijzonder wil ik Peter bedanken voor zijn dagelijkse begeleiding en voor het delen van zijn scherpe visie op de ambulancelogistiek. Naat, het is bijzonder om een goede vriendin op de werkvloer te treffen, dank je voor de gezelligheid daar en de etentjes thuis.

De afgelopen jaren heb ik veel tijd door mogen brengen op GMK Flevoland. Woorden kunnen geen recht geven aan de warmte die ik bij jullie heb mogen ontvangen. Door jullie inzet is de pilotstudie, de kroon op m'n werk die hier vertegenwoordigd is in het zesde hoofdstuk, mogelijk geworden. Heel veel dank voor het doorgeven van moeilijk te verklaren inzetten, en verbeteringen voor het interface. In de fijne samenwerking met de centralisten, de beheersorganisatie en het management heb ik erg veel kennis kunnen opdoen over het vak. Het liefste zou ik jullie hier allemaal bij naam noemen. André, heel erg bedankt voor alles tijdens m'n hele promotietijd. Robert, dank voor alle technische ondersteuning.

Ook wil ik m'n collega's bij Stokhos bedanken voor de afgelopen twee jaar waarin ik het proefschrift heb af kunnen ronden. In het bijzonder Coen, voor de goede samenwerking aan de dynamic routing, die in het vijfde hoofdstuk uitvoerig besproken wordt. Tess, dank je voor het af en toe meedenken bij de latere iteraties van het proefschrift, en het daarbij oprakelen van willekeurige sci-fi-weetjes.

Verder wil ik mijn vrienden van Star Trek Dark Armada/Batavia bedanken. Naast alle leuke activiteiten in binnen en buitenland hebben jullie ook een bijdrage geleverd aan m'n onderzoek in de vorm van leuke promotiefilms over het onderzoek. Nicole, dank je voor het ontwerpen van de mooie kaft. Dit maakt het boekje echt af.

Bovendien wil ik iedereen van EHBO Diemen bedanken. De gezellige winterse bijscholingsavonden die we om de week met elkaar hebben, zorgen ervoor dat ik in de praktijk veel makkelijker begrijp wat er aan de hand is. Deze kennis kwam tijdens het werken aan publicaties goed van pas bij gesprekken en bij de interpretaties van de modellen en hun resultaten.

Ik wil iedereen van Vierkant voor Wiskunde bedanken. In mijn jonge jaren gaven de zomerkampen het zetje in de rug om voor de toegepaste wiskunde te kiezen. Tegenwoordig haal ik veel voldoening en plezier uit het overbrengen van kennis aan de jongeren. De meerdere vriendschappen, de lijst met namen is te lang om hier te vermelden, die ontstaan zijn bij Vierkant, hebben behoorlijk geholpen met het vormen van ideeën. Ook heb ik de bezoekjes tussendoor op 't kantoor altijd erg fijn gevonden.

En dan heb ik ook nog enkele goede vrienden... Dank voor al jullie interesse, ook al snapten jullie vaak weinig van waar ik allemaal mee bezig was. Het was altijd een opluchting om te weten dat ik (op de PS4) op jullie kon rekenen, alwaar we meermaals de dag doornamen terwijl we op een iets andere manier de wereld aan het redden waren. Mijn kennis over deuren, en wel meer unieke uitspattingen, is door jullie de laatste jaren fors toegenomen. Hier zal ik vast profijt van hebben.

Tot slot wil ik mijn hele familie bedanken voor alle belangstelling en steun de afgelopen jaren. Ruud, je doorlezingen en de vele feedback hebben de teksten op een hoger niveau gebracht, waarvoor ik je erg dankbaar ben. Hans, heel erg bedankt voor je verbale duwtjes die me gevormd hebben tot de persoon die ik nu ben. Irene, dank je voor al je hulp de afgelopen jaren die het zoveel makkelijker gemaakt heeft om dit proefschrift te schrijven en voor de vele reizen die we gemaakt hebben.

Martin van Buuren
maart 2018

1

INTRODUCTION

In serious life-threatening situations a fast ambulance response is required. Late arrivals can have a serious impact on the well-being of the patients, and can also have consequences regarding to policy making. Ambulance service providers (ASPs) face a broad range of challenges in order to realize fast response times. While remote rural areas face the challenge to provide care for an aging population that decreases in size, urban areas are making efforts to cope with the increasing demand on the number of calls. In the meantime, regional characteristics are subject to constant change in regulations, changes in hospital openings, and alterations in the ambulance fleet's composition and size.

There are many questions to be addressed in realizing high quality ambulance care: What are good locations for ambulance bases? How many ambulances should be positioned at each base location? How should a dispatcher reposition ambulances that are free and available over the region to maintain an optimal coverage level? How should the dispatch center be staffed? What would be the effect on the response times of a given policy change?

In the ever-changing landscape of ambulance care there is a trade-off between costs and quality care. This raises the need for methods and tools that effectively enable decision makers to make optimal use of the resources at hand. The efficiency of decision making can constantly be improved by these increasingly sophisticated models in combination with emerging data technology. Topics in this thesis help practitioners and scientists to bridge real-life problems and mathematical models.

In this thesis we propose and study a range of models that address the questions above—models that function on both the tactical and the operational level.

1.1 Emergency medical services

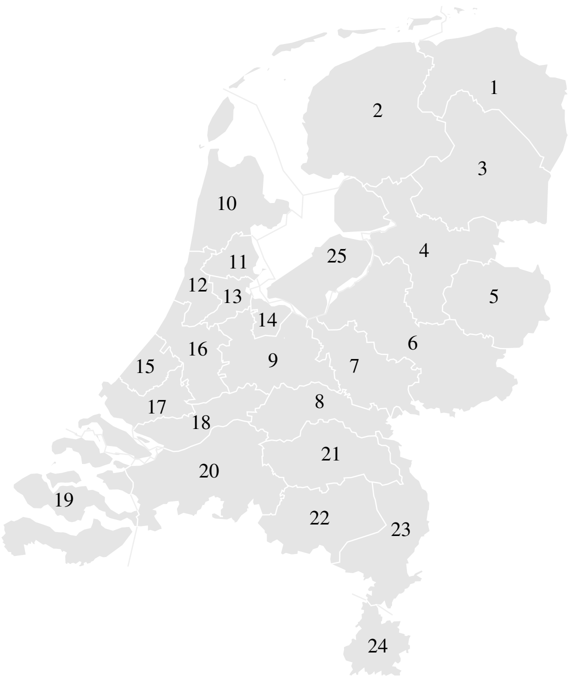
In this section we give a brief outline of emergency medical services (EMS) systems, with a primary focus on the Netherlands. The structure of ambulance care is largely similar within the Europen Union [29, 57, 140] and the rest of the world [43].

1.1.1 Regions

Ambulance care is spatially partitioned into multiple regions in the fast majority of countries; each region has an ambulance fleet and is coordinated by one dispatch center. Currently, the Netherlands is partitioned into 25 of these EMS regions; see Figure 1.1. For each region, the ASPs work operationally together as one organization called a RAV (in Dutch: regionale ambulancevoorziening) [D6]. Each RAV is responsible for its business processes, performance and finances.

1.1.2 Dispatch centers

A request for ambulance care starts by an applicant at an incident location who believes that professional medical aid from emergency medical technicians is required; see Figure 1.2. In most countries, a request for EMS is executed by calling a nation-wide emergency telephone number, such as 1-1-2 or 9-1-1. The



1	Groningen
2	Friesland
3	Drenthe
4	IJsselland
5	Twente
6	Noord- en Oost Gelderland
7	Gelderland-Midden
8	Gelderland-Zuid
9	Utrecht
10	Noord-Holland Noord
11	Zaanstreek-Waterland
12	Kennemerland
13	Amsterdam-Amstelland
14	Gooi en Vechtstreek
15	Haaglanden
16	Hollands Midden
17	Rotterdam-Rijnmond
18	Zuid-Holland Zuid
19	Zeeland
20	Midden- en West Brabant
21	Brabant Noord
22	Brabant Zuidoost
23	Limburg-Noord
24	Zuid-Limburg
25	Flevoland

Figure 1.1 The Netherlands is partitioned into 25 ambulance regions.

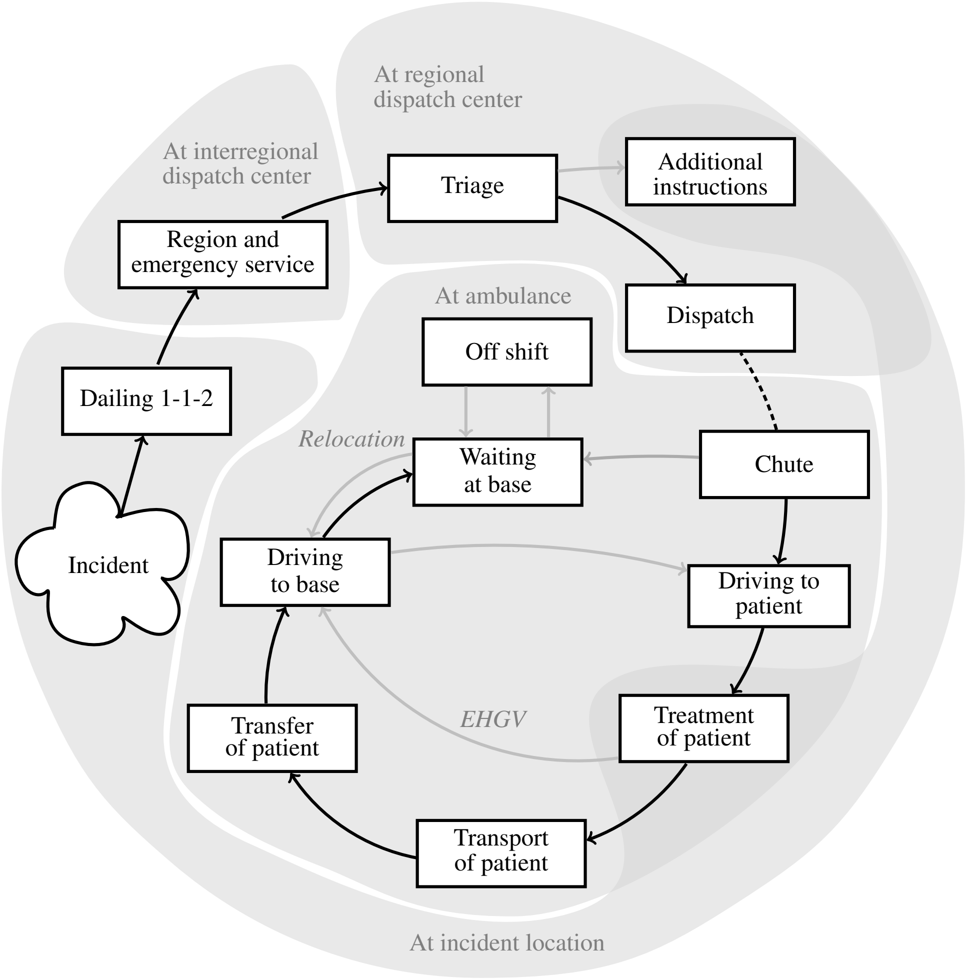


Figure 1.2 The ambulance service process as it usually takes place consists of various steps: at the incident, in the dispatch centers, and at the ambulance. [Variant Default]

request is answered by a *call center agent*. Depending on the country's system, this *call center* handles the entire request, or the agent determines the region and emergency service one needs, whereafter the request is forwarded to the correct regional call center. This may be a general call center for emergency services, or a specific medical or EMS call center. Medical professionals and other emergency services have a separate telephone number that directly connects them to the right dispatcher. The call center is called the dispatch center in the case it is the last of multiple call centers that an applicant speaks to in order to get access to ambulance care.

The *dispatch center* has the task to determine the level of the injury in the so-called *triage* procedure, that is the assignment of degrees of urgency to wounds through performing a questionnaire. Consequently, an ambulance is sent to incidents only in the case the patient really needs this service. The number of ambulances is limited, and it is important to have an ambulance available for dispatch when needed. If ambulance care is required, i.e., the request for EMS is *honored*, an available ambulance is sent to the incident during the dispatch procedure, and care is provided when the ambulance arrives at the incident location. More information on the choice of the ambulance is provided in Section 1.1.4.

To bridge the time gap until the medical professionals arrive, the dispatch center may provide the applicant with *additional instructions* on first aid actions. Reanimation instructions are a clear example of these additional instructions to an applicant who has no medical know-how. If the patient needs hospital care, transportation to a hospital is provided. The dispatch center is in control when it comes to *coordination* of the EMS services. EMS call center agents communicate with hospitals and ambulance teams and if necessary also with police, firefighters, and other EMS or homeland security call centers.

1.1.3 Urgency levels

Each honored call is assigned an urgency level. The Dutch EMS system distinguishes three levels: A1, A2, and B.

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 patient as soon as possible. Examples: heart attack, reanimation or serious traffic incidents.

A2 A 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 have discussed this with the dispatch center, but this only happens on rare occasions. Examples: a broken leg or a general practitioner asks for transportation to a hospital.

B A *non-urgent* call 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 a scheduled transport from an EMS vehicle to his or her home, it will be classified as a B-call as well. Calls with this urgency level are commonly referred to as *ordered* transportations.

A major difference between calls that are labeled with urgencies A1 and A2 on the one hand and calls with urgency B on the other is that the occurrence of calls with A1 or A2 urgency are not known beforehand, while calls with urgency B can be planned.

Internationally, it is common practice to differentiate between urgency levels, with only Hong Kong being the known exception. Most countries have a completely separated system for ordered transportations [29].

1.1.4 The ambulance service process

Once a dispatch center agent has determined the urgency and incident location, the search for the best ambulance to dispatch starts. Usually, this is the ambulance that can reach the incident location in minimal time, although in practice many other variables play a role, such as the vehicle transportation capabilities and the remaining shift time. Not all vehicle types can attend any incident type. For instance, in a fleet with multiple ambulance types the cheaper to operate basic life support ambulances can only attend ordered transportations. When a shift is about to end, or when emergency medical technicians (EMTs) just have attended a severe incident, a dispatcher can decide to send another ambulance that travels a bit longer, but is better for the overall fleet's morale.

When the pagers of the EMTs are activated at dispatch, some time passes before the ambulance starts moving, since it takes time to get into the vehicle. This time period is called the *chute time*. The entire duration of a call entering the system until the ambulance starts moving is called the *pre-trip delay*.

Next, the EMTs *drive to the patient*. When the ambulance arrives at the incident location, the response time is known. The response time is defined as the length of the time interval between the moment that the call enters the queue at the call center agent until the moment the EMS vehicle arrives at the incident location.

After driving to the incident follows the *treatment of the patient*. In some cases, the patient is treated at the incident location, after which the ambulance returns to the base location. In other cases, the patient is transported to the hospital. In some countries a patient is always transported to the hospital, either because ASPs are paid per transport, or because the country's legislation states that the ASP must provide fastest means to the hospital and the medical professionals start treatment there instead of at the incident location. Literature is not unambiguous about whether 'stay and treat' outperforms 'scoop and run'; while [116] says 'scoop and run' gives a 38% reduction in the odds of dying, [69] says that it is not clear which policy has the preference. The Netherlands uses the 'stay and treat' policy.

Each call is classified as *declarable*, *EHGV* or *loss*. A call is classified as *declarable* if transportation is required: either a patient is brought to a hospital or to his home. An *EHGV* call (in Dutch: eerste hulp, geen vervoer) is a type of call within the 'stay and treat' policy where the EMS team can provide care locally, and the patient does not require transportation to a hospital. EMS personnel determines this status at the incident location. For EHGV calls ambulances *drive back to the base* after treating the patient. Sometimes an EMS team arrives at the scene and no patient is present. This might be the result of a patient who left after the call was made or a prank call to the emergency number. This type of call can be classified as a *loss* call. In case a patient in a hospital is not ready for transportation at the moment the EMS vehicle arrives, the call will be classified as a loss call as well.

After the *transport of the patient* to the destination follows the *transfer of the patient* that usually bridges pre-hospital care with hospital care. Sometimes the transfer time is referred to as the *turn-around time*. After finishing up the transfer, the ambulance is available for handling the next incident. If there is no nearby patient to attend, the last stage of the ambulance trip consists of *driving back to its base location*. Here it waits for a new call, or until the shift ends.

For basic life support (BLS) transport, and occasionally for advanced life support (ALS) transport, it is possible that the incident location is a hospital. In that case, the destination may be another hospital or a home address. This can happen if a patient is brought home after a day treatment, or when another hospital is specialized in the patient's medical condition. Since our primary focus is on ALS transport with random arrivals, we keep respecting the definitions mentioned before, even if the patient's destination is not a hospital.

1.1.5 Response times

The public perception with respect to ambulance care dictates that every second counts. Throughout the world, the response times are used as a proxy for the quality ambulance care. Multiple times questions have been asked about the quality and origins of this proxy, which is the motivation for this section. Does *every* second count?

The key performance indicator (KPI) is the *fraction of late arrivals* for potentially life-threatening incidents; that is, the number of calls that exceed a given response time threshold divided by the total number of calls in the system. The response time threshold is set to 92% in 12 minutes for Hong Kong, and Australia uses 50% within 10 minutes. The Netherlands [D2, D7] and parts of Germany use 95% within the 15 minutes. Larger countries differentiate response times and have other KPIs for urban, rural and wilderness [43]. Too many late arrivals can result in a penalty or can have an adverse impact on the reputation of the EMS provider. Traditionally, this KPI is calculated for an entire ambulance region, over a whole year.

The origin of the common response time threshold in North American cities, being 90% in 8:59 minutes, can be found in [33], which was published in 1976. This research shows that eight minutes is the case of cardiac resuscitation survival limit, which is only a small fraction of the high urgency calls. The 59 seconds are added on top as to accommodate for the inaccuracy of the punch clocks used at the time [43]. The Dutch performance requirement of 95% within 15 minutes for high urgency calls was put in place in the late nineties, a time period when politicians wanted to have more control on health care costs, and accordingly, performance indicators were developed [70].

But is there a better alternative? Scientific literature provides some insight. Articles [38, 90] show that there are only a few validated indicators of the effectiveness and quality, which mostly relate to patient satisfaction and general system processes, each providing an indirect measure of quality which is difficult to relate to outcomes in patient care. Two other studies [92, 95] conclude that no statistical evidence can be found that short responses increase the probability of survival.

Ambulance crews also find this measurement overly simplistic and thereby put patients and ambulance crews at risk [101]: “You see, it’s an unfortunate situation. With this eight minutes, if you arrive in seven minutes and the patient dies it’s a success. If you arrive in nine minutes and the patient lives, and it’s a good outcome, you’ve failed. Which to me is absolute rubbish. And we are now treating the clock and not the patient. The patient care, in my view, is gone, absolutely. Well it’s terrible. It’s awful. (Andy)”.

However, there are studies showing the opposite. Especially in the field of heart attacks fast responses do save lives [74, 79, 134, 139]. Furthermore, there is no chance of survival if the response time exceeds 30 minutes [135]. The one-year survival rate increases even more if bystanders or police officers provide external cardiac massage and artificial respiration, also known as CPR [79, 136]. Also, for roadside incidents it is shown that a response time reduction from 25 minutes to 15 minutes causes a third fewer mortalities [117]. Finally, [3] indicates that response times, amongst others, is a valid indicator of the quality of care for cardiac arrests.

Based on the heart attack survival curves, ambulances location models are created that maximize the probability of survival [40, 89]. As a result, it is shown that maximizing response time thresholds can actually serve as effective proxies for patient survival if this limit is set to seven or eight minutes. These models can even incorporate survival functions for other incident types. So perhaps a response time threshold is not so bad as it first looks likes after all.

In the year 2008, the Dutch Parliament asked the Minister of Health Care about lowering the response time threshold from fifteen to eight minutes. He answered that this would significantly increase the costs, which he could not justify, because scientific research has not provided strong evidence on the effect of increased health outcomes [D4].

In the end, some things are not contradicted. More research is needed to get survival curves of other injury classes to which ambulances respond. Also, it does not negatively influence the medical outcome if the ambulance arrives faster, because of efficient pre-dispatch ambulance positioning. And last but not least, it feels reassuring for the patient and bystanders if professional help can assist within a short time span after a potentially life-changing incident has happened.

1.2 Outline of the thesis

There are numerous interesting mathematical challenges in ambulance care. Fields include the development and analysis of dispatch center models, facility location problems, ambulance allocation models, and operational control methods that real-time distribute available vehicles through dynamic ambulance management (DAM) models. Topics in this thesis touch several of these challenges. Various results presented in this thesis have already found their way to ambulance service providers.

In Chapter 2 we propose simulation models for evaluating the performance of ambulance dispatch centers. So far, *dispatch center simulation* has been fairly untouched in the literature. The models address dispatch center formats that are encountered in practice, in which the agents can have one or more roles: only call taking, only dispatching, or the generalist specialism that can do both. Characteristic for this dispatch center model is that it can be seen as a call center *network*, in which a call can enter the same queue multiple times through a feedback mechanism. This mechanism models the multiple emergency situation update contacts between the dispatch center and ambulance team. The model enables EMS planners to better understand the impact of these features on the response time performance of EMS dispatch centers. Extensive simulations show that there is not one model that outperforms all others. These models found their way to practice by its inclusion in the report *Models for the national ambulance plan in the Netherlands* [D5].

The distribution of ambulances over the bases resulting from most existing allocation models is out of balance for regions with combinations of urban and rural demand, concerning the fraction of late arrivals for each area within a region. In so-called *regional coverage location problem* models that focus on region-wide coverage, ambulances are more likely to be positioned in urban areas due to the large call volume, while the set coverage models try to spread the ambulances as much as possible and provide relatively too much coverage in rural areas. To be able to meet local needs, one wants to allocate ambulances such that there is a guarantee of the minimal performance in each area of the region. The existing so-called *minimal reliability* and *maximal availability* models address this problem, but require homogeneity in the call arrival rate, service time and minimum required reliability level. It is known that these minimal reliability models lead to major over-estimations on the required number of ambulances if they are applied to real inhomogeneous ambulance regions—against their original design. Maximal availability models have similar issues that result in fewer people being covered than necessary. As a result, they are hardly used in practice. Chapter 3 provides fundamental new insight into why this overstaffing takes place and solves the issues by replacing two assumptions by their respective generalizations in what is called the *adjusted queuing approach*. This approach allows for fluctuations in the call arrival rate, the service time and the minimum required reliability level, and thereby make regional coverage location problems applicable to real ambulance regions.

Chapter 4 proposes two minimal reliability models that follow the adjusted queuing approach that was introduced in Chapter 3. The first model formulates a *mixed integer program* that solves the minimal reliability problem. The main drawback of this model is that it can only be applied to relatively small model instances because of its calculation complexity. The second model is a *heuristic* that uses a post-processor that adds ambulances on top of a relatively simple IP-formulation until the reliability condition is satisfied, and can be applied to larger model instances. The model outcomes of both models are compared to the best-known model from literature. The results show a strong reduction in the required number of ambulances, and come close to the realistic numbers used by the ambulance service providers. To our best knowledge, these models are the first of its kind that is ready to be used in practice.

Chapter 5 focuses on the real-time relocations of ambulances. Dynamic ambulance models give dispatchers advice on how to redistribute their available ambulances over a region to minimize late arrivals. Usually, the request to an ambulance is to relocate to a base location, without real-time usage on the other available ambulance on *how* to drive, i.e., what route to follow. Most time ambulance drivers take the fastest or shortest route. Taking an *alternative route* has an impact on the coverage during the relocation. Chapter 5 formulates a method to obtain a good alternative route, and shows improved intra-regional fairness for three real ambulance providers.

In Chapter 6 we evaluate two dynamic ambulance management policies in practice through a *pilot study*. The models used, numerical results and user experiences from the dispatchers and management are shown. During the pilot the number of late arrivals was reduced by a third, resulting in the first time of ASP GGD Flevoland's history that the requirement of 95% of the high urgency calls was responded to within the response time threshold of 15 minutes. Based on the success of this pilot the software is still used in daily operation and under continuous refinement. Current developments prepare the software for a roll-out in other dispatch centers.

Chapter 7 gives an overview on the structure of the *testing interface for ambulance research*, TIFAR—a powerful software framework that can be used to evaluate what-if scenarios in ambulance care. The *TIFAR* software framework is omnipresent in this thesis, as it enables us to create simulation models and solve optimization problems; these are the so-called *refinements*. This chapter provides detailed descriptions for four refinements: (1) the dispatch center simulation model of Chapter 2, (2) an ambulance allocation model of Chapter 4, (3) the road domain simulation engine that has been used to generate the results for Chapter 5, and (4) the operational version that is presented in Chapter 6. In order to solve mixed integer problems, TIFAR is linked to the optimization suite Coin-OR [O2]. TIFAR has extensive reporting possibilities, such that a wide range of analysis can be made. The TIFAR simulation engine for the road domain is used to evaluate new model updates, and to provide advice to actual problems that ambulance providers encounter in their daily life.

A list of all abbreviations used can be found on pages 195–199, and a full variable listing is provided on pages 201–204.

Acknowledgements
This Ph.D. thesis results from the REPRO research project. REPRO, short for 'From *re*active to *pro*active planning of ambulances services,' is a collaboration between research partners, ASPs, and organizations that provide support and feedback. The goal of REPRO was to develop state-of-the-art mathematical models for a broad range of problems in the field of ambulance services. The research partners were Centrum Wiskunde & Informatica, Delft University of Technology and Vrije Universiteit Amsterdam. Participating ambulance service were Ambulance Amsterdam, RAVU, GGD Flevoland, RAV Gooi & Vechtstreek and Connexxion Ambulancezorg. Supportive organizations that participated are RIVM, Brandweer Amsterdam-Amstelland, Politie Amsterdam, ANWB, Ambulancezorg Nederland, Oranje Kruis, Achmea, and City-GIS. The project was partly financed by STW (currently TTW), which is part of the Netherlands foundation for scientific research NWO.

2

DISPATCH CENTER SIMULATION MODELS

In pre-hospital health care the dispatch center plays an important role in the coordination of EMS. A dispatch center handles inbound requests for EMS and dispatches an ambulance if necessary. The time needed for triage and dispatch is part of the total response time to the request, which, in turn, is a key performance indicator for the quality of EMS. The call center agents of the dispatch center should perform the triage efficiently, so that incoming calls have short waiting times, and the dispatch of ambulances must be adequate and swift to get a fast EMS response. This chapter presents and compares three discrete event simulation models for dispatch centers. The first has two different call center agent classes between whom communication tasks are split, while the second has one class of call center agents that share all tasks. The third model is a combination of both. The models provide new insight into the dispatch center processes and can be used to address strategic issues, such as capacity and workforce planning. The analysis and simulations of urgent communication and decision processes in this chapter are also valuable to other emergency call centers.

This chapter is based on the following publications:

[A1] M. van Buuren, G. J. Kommer, R. D. van der Mei, and S. Bhulai. "A Simulation Model for Emergency Medical Services Call Centers". *Proceedings of the 2015 Winter Simulation Conference*. Dec. 2015, pp. 844–855

[A2] M. van Buuren, G. J. Kommer, R. D. van der Mei, and S. Bhulai. "EMS Call Center Models With and Without Function Differentiation: A Comparison". *Operations Research for Health Care* 12 (Mar. 2017), pp. 16–28

2.1 Introduction

The dispatch center has the task to perform the triage and dispatch adequately. That is, to determine the need and urgency level for EMS properly, so that an ambulance is sent to incidents only in the case the patient needs this service. The dispatch center is also the central communication hub between the ambulances and other partners in the emergency process.

The total time needed for taking the request, performing the triage, and dispatching an ambulance is called the *call center time*, which is part of the total response time. In the case of life-threatening situations short response times are crucial. Hence, it is essential to have short waiting, triage, and dispatch times at the dispatch center.

The three discrete event simulation (DES) models for a dispatch center developed in this chapter provide insight into these crucial variables. They use the policy of the dispatch center and the number of call center agents as an input. The models simulate the communication processes at the dispatch center in detail, and they can be used to evaluate and predict the dispatch center's performance. Results are shown for all three models for realistic call volumes and durations, where the input contains call record data provided by a real dispatch center.

In Section 2.1.1 we give an overview of the dispatch center's processes. Section 2.1.2 concludes the current section by stating the contribution of our models.

2.1.1 Overview

Before going in-depth we provide a high-level model overview that discusses some typical model characteristics.

Three call center agent classes

Most dispatch centers have two classes of call center agents, who work in cooperation. The first class is the *call takers*. They handle the inbound

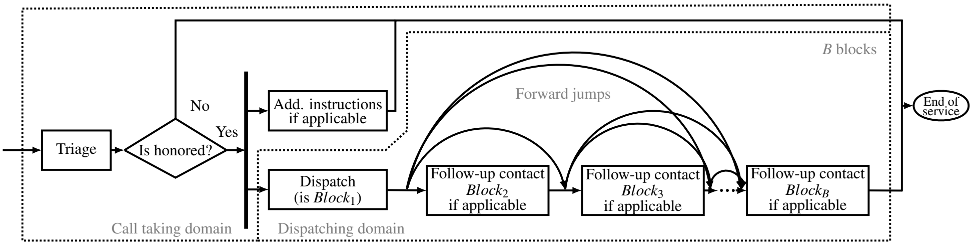

Triage
Is honored?
No
Yes
Add. instructions if applicable
Dispatch (is $Block_1$)
Follow-up contact $Block_2$ if applicable
Follow-up contact $Block_3$ if applicable
Follow-up contact $Block_B$ if applicable
Forward jumps
B blocks
End of service
Call taking domain
Dispatching domain

Figure 2.1 High-level description of the dispatch center. The separation of the call taking and dispatching domains is shown by dotted contours.

	Model name	Call takers	Dispatchers	Generalists
1	Function differentiation	✓	✓	
2	Solely generalists			✓
3	Mixed model	✓	✓	✓

Table 2.1 The call center agent classes for each model.

requests and perform communication with the caller, also referred to as the *applicant*. *Dispatchers* are the second call center agent class. They take care of the dispatch process, the communication with the ambulance team and the hospitals in various follow-up contact moments of the service. An example of a follow-up call is a request from the ambulance team for additional health condition information while driving to the incident. The dispatcher has logistic skills and can be supported by decision support software (DSS) to determine the most appropriate ambulance to send to the incident.

The just mentioned type of dispatch centers make a distinction between call takers and dispatchers. This is called *function differentiation*. Other call centers have one class of call center agents, called *generalists*, that do both tasks. One can consider a generalist as a call taker and dispatcher embedded in one person.

Figure 2.1 illustrates the processes of the call taking domain and the dispatching domain. The call taking domain contains a triage procedure, in which is decided if a request is honored. Sometimes, when the need for an ambulance has been determined, additional instructions are given by the call taker while at the same time the dispatcher is assigning a call to an ambulance. The dispatch process and follow-up contacts are modeled through a block sequence, which is described in Section 2.3.3.

The three dispatch center models use different call center agent classes, which is displayed in Table 2.1.

Prioritizing calls

Requests can be made by several disjunct applicant classes, such as civilians, general practitioners, or police officers. The priority of an incoming request depends on this applicant class, i.e., an ambulance service provider can choose to give civilians a higher priority than hospitals to prevent call center agents from taking a hospital line if there are civilians waiting in a life-threatening situation.

The applicant's class may affect the service time distributions in the call taking domain. Requests applied for by a general practitioner or a police officer generally do not require extensive triage and therefore may have a shorter service time than requests from civilians. Requests from civilians always need to be triaged in order to determine the need and urgency of the service.

Requests for EMS are often categorized into urgency levels. This chapter uses high urgency in the case of a potentially life-threatening situation, to low urgency where a patient is stable and an immediate life-saving response is not required. Low urgency requests are for instance planned transports from and to a hospital. This corresponds to the Dutch urgency levels A1, A2, and B, respectively.

Only requests that are honored by the call taker—or generalist, depending on the model—get an urgency assigned. The urgency is used for prioritizing in the dispatch domain: dispatchers only address low urgency tasks if all potential life-threatening high urgency tasks are completed.

Note the difference between priority and urgency: the priority for an incoming request is used to decide what telephone line to answer when there are multiple waiting calls in the telephone system, while an urgency, which is assigned during the triage procedure, gives information as regards the seriousness of the injuries, and consequently how the EMS should respond in the dispatch domain.

2.1.2 Contribution of our models

This chapter presents the first simulation models for dispatch centers that: (1) contain multiple call center agent classes, as can be found in a modern dispatch center, (2) contain follow-up contact moments, (3) simulate dispatch center processes in high detail, (4) use real dispatch center data sets as simulation input to gain insights, and (5) provide a comparison between multiple dispatch center models.

Regular call centers versus dispatch centers

There are two main differences between a regular call center and a dispatch center. By regular call centers we mean call centers that handle incoming calls for a service of a company, with questions on a particular subject. The first is the urgency at which calls need to be taken and processed. Requirements on the quality of service of dispatch centers are much stricter than regular call centers due to the urgency. A noticeable subject is the respectable uncertainty of demand in combination with the fact that a request for EMS may not wait too long, because it might involve a life-threatening situation. As a result, we take in the numerical analysis the strict requirement of at most six seconds waiting time as a key performance parameter which is common in Dutch dispatch centers, whereas in a regular call center a typical response time requirement is often not that strict.

The second difference between a dispatch center and a regular call center is the fact that the dispatch center has more communication tasks in the coordination of the EMS services. Dispatch center agents communicate with hospitals and ambulance teams and if necessary also with police, firefighters, and other EMS or homeland security call centers. These tasks are also done under time pressure. The number and characteristics of these communication processes are uncertain and difficult to include in an analytical model. Simulation models do not have the difficulty that an analytic solution to a mathematical problem needs to be determined. As analytic modeling is not possible for dispatch centers, simulation techniques seem the best ways to analyze and evaluate dispatch center systems.

Contribution to practice
The implemented simulation models can be used by decision makers, managers of dispatch centers, and other policy makers in managing the operational subjects of the dispatch center. All processes are assumed to be stochastic, the simulations make use of the uncertainties in call volumes and lengths of the communications. The models give insight into the variance of the workload in different situations. It is possible to study economies of scale in the case of increasing call volumes. The outcomes of individual simulation runs include the workload of the system and the waiting times for different applicant classes. These performance indicators can be examined to explore optimal staffing strategies, staffing levels, and service level requirements.

This chapter compares the performance of the two most common staffing policies from practice: with function differentiation and with only generalists. A third model we propose and analyze, the so-called *mixed model*, is a generalization of the other two policies.

In Section 2.2 we provide a literature overview. Section 2.3 describes our models from a queuing theoretical perspective. Section 2.4 shows realistic parameters that were obtained from a real dispatch center database and expert guesses, used to generate the results of Section 2.5. In the results section we also provide new insights, which are concluded and discussed in Section 2.6.

2.2 Literature

The current literature review makes a distinction between general call center simulation models and dispatch center simulation models. For simulation models that focus on the ambulance domain we refer to Chapter 7.

Regular call center simulation

The literature on the more general call centers is broad and vast, in contrast to dispatch centers. Topics of research are arrival processes, optimal staffing of call centers, estimation of demand and expected future demand for services, and routing of specific types of demands. Koole and Mandelbaum give a good introduction to this subject [71], while other papers review research on customer call centers [47, 72]. The surveys show that call centers are mainly analyzed within the framework of queuing theory. In order to keep the queuing models tractable, most papers deal with a single call type with a homogeneous pool of agents. The extension to multiple call types and a heterogeneous pool of servers leads to complex and intractable models [23, 114]. Because of this complexity, simulation is an appropriate tool to analyze more complex call centers, such as dispatch centers.

Dispatch center simulation

There are limited simulation models that focus on the call center domain of the EMS process. Kozan and Mesken have modeled the dispatch center within a simulation context [73]. They developed a simulation model to analyze the effects of varying call volumes, personnel resources, and workload distributions on the performance of the call center. Ross [113] studies the Toronto dispatch center and develops a simulation model to examine the effect of changes of the dispatch processes on the workload of the call center staff. For this research, communication flows at the call center are identified and different dispatch systems are evaluated. Other preliminary work on dispatch center simulation can be found in [37, 102].

2.3 Models

This section describes the three models in detail. In Section 2.3.1 we formulate our assumptions on the ambulance practice, followed by the model with function differentiation in Section 2.3.2. All models contain a block sequence, the inner working of which is explained in Section 2.3.3. Section 2.3.4 describes the second model holding the classic regime with solely generalists who do both call taking and dispatching. We conclude with the mixed model in Section 2.3.5.

2.3.1 Assumptions

We assume that an ambulance region has exactly one regional dispatch center that is responsible for both the call taking and the dispatch of ambulances. As stated before, we assume that there are three classes of dispatch center agents, each with their own set of skills and costs: *call takers, dispatchers*, and *generalists*. Another assumption is that there are no call abandonments in the dispatch center.

Each honored request gets an urgency assigned. We assume that this urgency is not subject to change during the remainder of the call handling. In practice, urgency mutations do occur but often seldom. We assume that an ambulance service provider uses the totally ordered set of urgency classes $\mathcal{U}$.

We assume that to each honored request exactly one ambulance is dispatched. After dispatch, it can generate follow-up calls. In practice, there are situations where multiple vehicles are dispatched to a request, each generating follow-up calls.

Performance indicators

The quality standard and volume of care differ per country, and they depend on legislation, tradition, culture, and prosperity level. In some countries ambulances are staffed with paramedics, while other countries employ specially trained nurses. Hoogeveen provides a European overview [57]. Hence, various countries use different KPIs for EMS care, as is discussed in Section 1.1.5. Most countries use a constraint for the fraction of late arrivals. KPIs of dispatch centers are not always well defined for lower urgency calls.

The KPIs used in this chapter are the waiting time before a call is taken, the average workload for a call center agent, and the total call center time. These KPIs are inspired by the service requirements that are applicable to most Western countries. Recall that, by definition, the *call center time* is the time elapsed between the moment when an incoming request enters the system and the moment the ambulance has been dispatched by a dispatcher. Follow-up contacts and the additional instructions are not included in the call center time, but they are, however, part of the workload of an agent.

Formalized, we limit ourselves to the following performance indicators for all models in the remainder of this chapter:

1. The fraction α_1 of the calls that are picked up by the call taker (or generalist) within at most r_1 time units.
2. The fraction $\alpha_2(u)$ of the honored calls of urgency $u \in \mathcal{U}$ that have a call center time at most $r_2(u)$ time units.
3. The average workload on a call center agent, if applicable split by call center agent class.

Our goal is to gain insight into the required number call takers $n_{calltaker}$ and dispatchers $n_{dispatcher}$ in the model with function differentiation, or generalists $n_{generalist}$ in the model with solely generalists, to reach certain performance indicator thresholds. In the mixed model we use all these variables. Let n denote the total number of agents in the system.

2.3.2 Dispatch center model with function differentiation

Our first model describes the communication processes at a dispatch center with function differentiation in detail; a schematic for this model is displayed in Figure 2.2. Inspired by the literature, the dispatch center is modeled as a queuing model; it encapsulates all communication moments and delays in queuing systems, in which call center agents are the servers and communication moments are the tasks. Using this approach, each domain can be modeled as a queuing system with priority queues and a server pool, conditional routing statements, and assignment blocks.*

Priority queues at the two queuing systems

The priority queues in our models all have similar behavior. All tasks are non-impatient. This means that once they have entered the system, they wait until they are served. All priority queues have an infinite capacity and no overflow regulations. Tasks are handled on a preemptive priority first-come first-served basis by a fixed number of servers. This means that when a task enters the highest priority queue and finds all servers busy, a task of the lowest priority in service is put on hold and the respective server starts serving the task from the highest priority queue. This task resumes service from the point at which it was put on hold when a server becomes available and stays in service with a duration of the remaining service time, but only if there are no tasks to serve with a higher priority. As a result, higher priority tasks are not influenced by the presence of lower priority tasks.

Call taking domain

Requests from applicant classes whose requests have a similar behavior are bundled into one incoming stream. It is allowed that an incoming stream consists of only one applicant class.

Inbound requests are modeled as new tasks for the queuing system of the call taking domain. The request arrival process is assumed to be dependent on time and the applicant class.

Depending on the incoming stream of a task, the call is routed to one of the priority queues at the call taker. Tasks originating from the same incoming stream are led to the same priority queue of the call taker queuing system.

The call taker queuing system has a fixed number M_1 of priority queues, denoted by $Q_1^C, Q_2^C, \dots, Q_{M_1}^C$. The priorities are strictly decreasing, i.e., Q_1^C is the priority queue with calls of the highest priority. These priority queues are also filled by a feedback loop containing the extra communication moments in the case the call taker gives additional instructions.

*Technically, the dispatcher domain contains multiple queuing systems, since each block has one, as we show in Section 2.3.3.

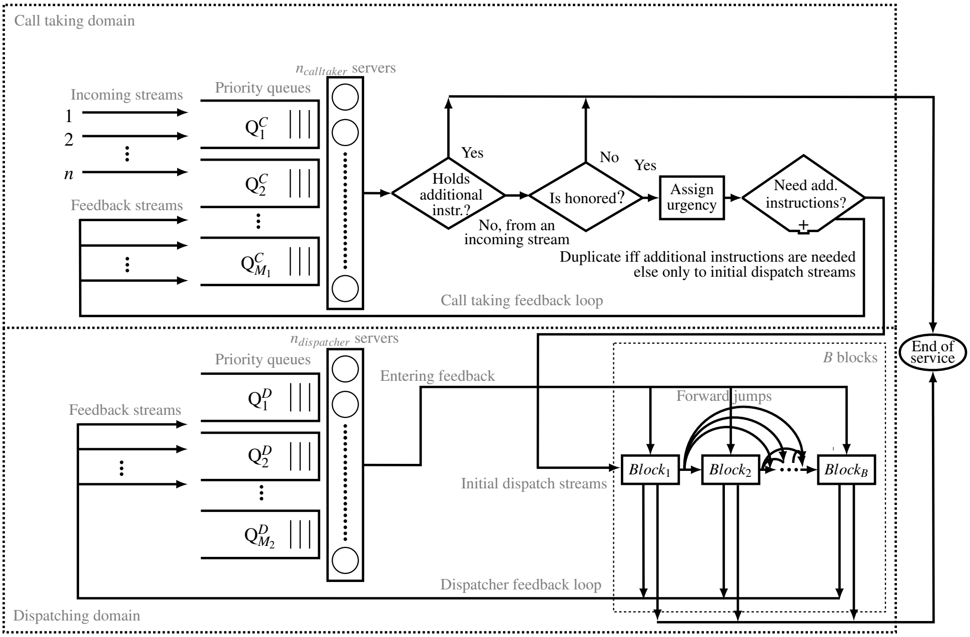


Figure 2.2 Representation of the call taker and dispatching domains of the dispatch center.

When a server is available the task immediately receives service. The service time distribution depends on the applicant class and the urgency. In practice the urgency is assigned during the service, and the call taking duration depends on that urgency. Therefore, we actually calculate the urgency at the start of the service, though in the schematic we have not found a way to clearly denote it and we choose to display it at service completion.

The first time a task arrives at the *'Holds additional instructions?'* statement, the answer is no. In the next routing statement, either with probability $1 - ph$ the call is not honored and the request ends service. Otherwise, with probability ph an ambulance is to be sent to the incident. In the latter case the request gets randomly an urgency $u \in \mathcal{U}$ assigned and we say that a request is honored. The probability ph depends on the incoming stream.

With probability $pa(i, u)$ the call taker gives additional instructions to the applicant of a call with urgency u that originates from incoming stream i; see the *'Need additional instructions?'* conditional fork in Figure 2.2.

When the applicant receives additional instructions in the model, the task splits into two separate tasks. One goes to the priority queues at the call takers using a feedback stream, while the other directly goes to the dispatcher. Using this conditional fork both a call taker and dispatcher can work simultaneously. If the priority queue at the dispatcher is of the highest priority, Q_1^C, it acts as an uninterrupted call at the call taker. An additional instruction task ends service when the agent puts down the telephone.

If there are no additional instructions required, the task directly enters the dispatcher domain. This happens through an initial dispatch stream and $Block_1$ to one of the priority queues of the dispatcher. The routing to the priority queue at the queuing system in the dispatching domain depends on the urgency of the request.

Dispatching domain
Task handling at the dispatcher server pool is done by a priority queuing policy, just like at the call taker server pool. The M_2 priority queues $Q_1^D, Q_2^D, \dots, Q_{M_2}^D$ are filled by honored requests of the call taker and communication moments with ambulance and hospital, represented by blocks. The service time distribution depends on the call's urgency, incoming stream, and if applicable the block it originates from.

The model contains a sequence of blocks representing the interarrival times, contact probabilities, routing, and the service time of follow-up contact moments with ambulance teams of the dispatching domain.

A block starts with a period of waiting, advances to a possible contact with the dispatcher in a feedback contact, and concludes with the continuation to a succeeding block or the end of service. A more detailed description of the block sequence is given below.

2.3.3 Block description

All models have a *block sequence* that consists of a fixed number of B blocks; a schematic picture of a block is displayed in Figure 2.3.

The number of priority queues open to receive feedback from the blocks is denoted by M. Note that $M = M_2$ for the model with function differentiation, and a similar expression can be found for the other two models.

A special case is $Block_1$. This block is always present and on all occasions leads to feedback; this represents the dispatch of an ambulance where the priority and service time of the dispatch equal the priority and service time by the assigned server in the dispatch server pool.

Without loss of generality we describe the structure of $Block_b$, $b \in \{1, 2, \dots, B\}$, and we illustrate its behavior for a request with urgency $u \in \mathcal{U}$. The incoming tasks originate from previous blocks (for $b > 1$) or the newly honored calls that enter the dispatch domain (for $b = 1$). To mimic the behavior of a time interval in which no communication occurs, tasks are handled by an infinite-server pool

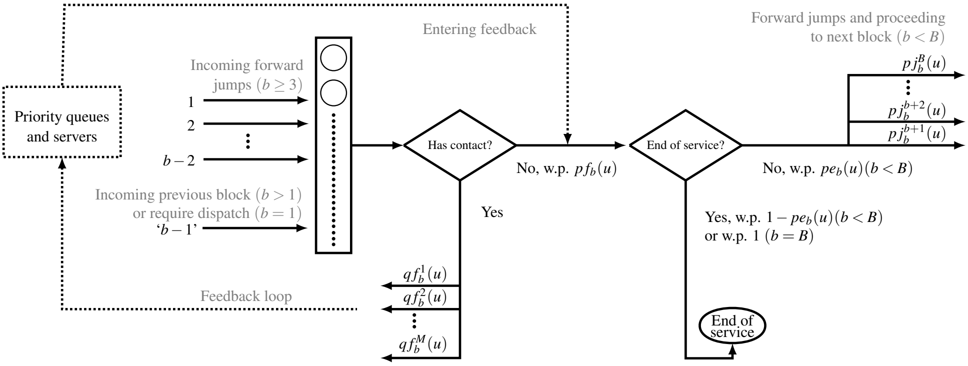

Priority queues and servers
Entering feedback
Incoming forward jumps ($b \geq 3$)
1
2
$b-2$
Incoming previous block ($b>1$) or require dispatch ($b=1$)
'$b-1$'
Has contact?
No, w.p. $pf_b(u)$
Yes
$qf_b^1(u)$
$qf_b^2(u)$
$qf_b^M(u)$
Feedback loop
End of service?
No, w.p. $pe_b(u)(b<B)$
Yes, w.p. $1-pe_b(u)(b<B)$ or w.p. 1 ($b=B$)
End of service
Forward jumps and proceeding to next block ($b<B$)
$pj_b^B(u)$
$pj_b^{b+2}(u)$
$pj_b^{b+1}(u)$

Figure 2.3 The structure of a block in detail.

with generally distributed service times. This service time can be interpreted as a delay by a driving time or contact moment with a patient at the incident location or hospital.

At this point in time the decision has to be made whether the ambulance team and call center agent have a contact moment using a priority queue in $\{Q_1^D, Q_1^D, \ldots, Q_{M_2}^D\}$, which is indexed by m. With probability $qf_b^m(u)$ a contact occurs through the feedback loop mechanism to priority queue $Q^{\cdot}_m$, and with probability $pf_b(u)$ no contact occurs in this block and the call moves forward to the next decision moment. The service time is generally distributed, and the parameters may depend on b and u. Note that $pf_b(u) + \sum_{1 \leq m \leq M} qf_b^m(u) = 1$, for all b and u.

Immediately after service completion, the feedback task enters $Block_b$ again. Both the tasks from the feedback loop and tasks that had no feedback continue to the next decision moment. Now we determine whether the task moves to a succeeding block, with probability $pe_b(u)$, or alternatively if there is an end of service. If $b = B$ we take $pe_b(u) = 0$; this leads to an end of service. The last decision moment of $Block_b$, ($b \neq B$) tells us at which block the task continues, the so-called forward jump. With probability $pj_b^{b'}(u)$ we redirect to $Block_{b'}$. For $b' \leq b$ we have $pj_b^{b'}(u) = 0$, and $\sum_{b'>b} pj_b^{b'}(u) = 1$ for all $b \neq B, u$. Notice that "End of Service" can also be implemented as a special case of a forward jump to a dummy block $B+1$.

2.3.4 Dispatch center model with solely generalists

In this section we describe the dispatch center model with solely generalists. Figure 2.4 provides a schematic overview of this model.

There is a major difference between this model and the model with function differentiation of Section 2.3.2. In this model with only generalists there is

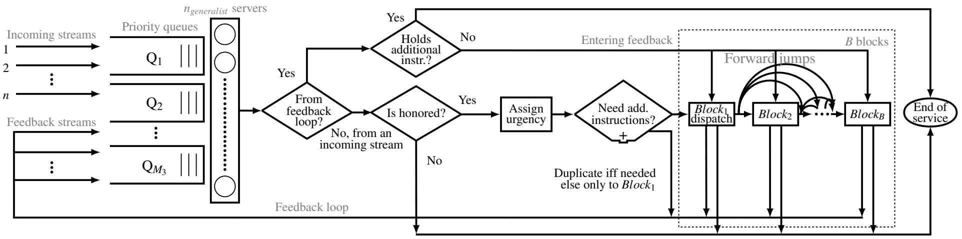


Figure 2.4 Representation of the model with solely generalists.

only one finite server pool with agents, thus the feedback from blocks and additional instructions is mixed with the incoming demand. In this section we build upon the assumptions of Section 2.3.1 and the blocks from Section 2.3.3.

A new request generates an incoming task from an incoming stream into one of the M_3 priority queues for the server pool with $n_{generalist}$ generalists. The priority queue assignment is based on the incoming stream. The service time distribution depends not only on the incoming stream, but also on the temporary hidden urgency of the request, like in the model with function differentiation. Recall that this choice is made because in reality the urgency is determined during the service. Similar to the call taker queue in the model with function differentiation, tasks waiting to be served by a generalist are put in a priority based preemptive FCFS queue with an infinite capacity.

At service completion of the call taking process, the call has probability $ph(i)$ to be honored by the call center agent, similar to the model with function differentiation. An urgency $u \in \mathcal{U}$ gets assigned to each honored request, based on a categorical distribution. Its parameters depend only on the incoming stream. Not honored requests exit the system. Directly after the urgency assignment, there is a conditional fork that generates an extra task in the case there is an additional instructions contact. Whether this is the case is determined by a Bernoulli distribution that depends on the incoming stream.

Further distribution parameters depend on the urgency, the incoming stream, and the block last visited. The extra task is led to one of the priority queues with the highest priority. This additional instruction task exits the system at service completion. The other task leaves the conditional fork to $Block_1$, which contains the dispatch.

A block sequence mimics the behavior of the dispatch, driving to an incident, taking care of the patient at the incident, et cetera. The distribution can be determined using a best-fit approach from real data sets. After the last block the call is ended, but it is also possible that another block already ended the call.

2.3.5 Mixed model

The *mixed model* contains call takers, dispatchers, and generalists. It is similar to the model with function differentiation; see Section 2.3.2. The major difference between the two models is that there is an extra server pool with $n_{generalist}$ generalists. A generalist starts to provide service to a task when there are no call takers or dispatchers available to take it, i.e., it handles the overflow from both the call takers and the dispatchers. Generalists respect the priority of jobs like any other call center agent class. If an agent of another class becomes available, the generalist finishes the task that he services instead of giving the remaining service to the available centralist.

2.4 Input

In this section we discuss the input data and parameter estimations. We use two data sources of the dispatch center in the city of Utrecht in the Netherlands over a period of three months. The first is the telephone information (TI) data set that originates from a server that monitors all in- and outbound telephone calls. The second contains call record details of the ambulance services in this period, which include status updates from the ambulance teams. Every inbound request to the ambulance service provider has its own row in the ambulance services call record details data set. Parameters that could not be determined from the data sets are estimated through expert opinion.

We group the input parameters as Bernoulli distributions and service time distributions. The parameters that are discussed in Sections 2.4.1 to 2.4.5 are the same for both models.

In Section 2.4.1 we describe the data sets that we used as an input source. Section 2.4.2 explains our aggregation to reduce multiple applicant classes into incoming streams. Sections 2.4.3 and 2.4.4 describe parameters for the call taking and dispatching domains, respectively. Section 2.4.5 gives the service time distributions for all call taking and dispatching processes. Finally, Section 2.4.6 discusses the routing policies toward the priority queues for both models separately.

2.4.1 Processing the data sets

The TI data set contains the timestamps for the telephone communication, i.e., the moment the dispatch center agent lifted the handset, when the call ended, and if it was an in- or outbound call. This data set consists of 109,000 inbound and outbound telephone calls, of which 79% are inbound. The TI data set does not include any information on the content of the calls, such as the applicant class (e.g., civilian or police), and whether the request was honored. The records do not indicate if the call was a newly incoming call or a follow-up call by an ambulance team.

Incoming stream	Applicant classes	Percentage
Civilian	1-1-2 and 9-1-1	24.4
Hospital–low	Hospitals*	17.4
Others	Unspecified and others	15.3
Dispatch center	Dispatch center agents	13.9
GP–low	GPs* and GP centers*	11.7
GP–high	GPs* and GP centers*	10.2
Health care institutions	Psychiatric, midwives, and homecare	3.3
Police	Police and fire fighters	2.6
Hospital–high	Hospitals*	1.3

Table 2.2 The distribution of new requests over the incoming streams.

Additional information was linked to the TI records by matching it to the database of the call record details. In this time period there were 41,000 requests for ambulance care. The matching is done probabilistically, in the sense that we coupled the start of the ambulance service request to the most likely corresponding telephone contact in the TI data set. In the matching process, usually one inbound call was matched to one request and, thereby, to one service. In some cases an inbound call was matched to multiple services; we omitted these multiple matches in the estimation of the parameters. A fraction of 92% of the call record details were matched to a TI record. From these matched data records we identified applicant classes, urgencies, priorities, and service times. Unmatched TI records were classified as follow-up calls.

2.4.2 Aggregation to incoming streams

Every applicant class is mapped onto one incoming stream. For example, the incoming stream *health care institutions* contains every request from the applicant classes *psychiatric, midwives,* and *home care*. This grouping is based on applicant classes with similar substantive grounds, service time distributions and priorities. Table 2.2 lists the call volume per incoming stream as percentages of the total call volume. It also shows which applicant classes are contained in each stream.

We assume the arrival process for applicant class k to be a Poisson process with rate f_k. Then the arrival process for an incoming stream i can be modeled as a Poisson process with rate

$$\lambda_i = \sum_{\substack{k \text{ is included} \\ \text{in stream } i}} f_k, \qquad\qquad i \in \{1,\ldots,n\}$$

for a fixed number of incoming streams n, where applicant k is included in incoming stream i.

* Differences between starred entries are addressed in Section 2.4.5.

Incoming stream		Urgency	
	High	Medium	Low
Civilian	60.1	37.8	2.1
Hospital–low	0	0	100
Others	20.1	64.6	15.3
dispatch center	5.1	84.1	10.8
GP–low	0	23.7	76.3
GP–high	72.6	27.4	0
Health care institutions	7.1	17.7	75.2
Police	48.7	49.7	1.6
Hospital–high	38.9	61.1	0

Table 2.3 The urgency distribution for every incoming stream (in %).

The call flow matches the Dutch practice. An underlying assumption is that civilians have inadequate knowledge of what to do in emergency situations and GPs can determine well when they require ambulance services on short notice, yielding the result that the two streams are answered directly. A call from these classes even may force ongoing calls of a lower priority ‘on hold’. All other incoming callers have some basic knowledge of how to keep the patient stable. Therefore, they have a somewhat lower urgency, and are routed to the medium priority queue. The low priority calls are made from hospitals, and often contain a request for patient transfer to another hospital or house address.

2.4.3 Bernoulli parameter estimations for the call taking domain

The call taking domain holds Bernoulli distributions for a request being honored, call urgency assignment, and receiving additional instructions.

Only civilian requests qualify for not being honored; this happens with probability 22%, a value obtained from the call records data set. Every honored request gets an urgency assigned. We use three urgencies: u_1 = High, u_2= Medium, and u_3 = Low. Table 2.3 shows the urgency distribution for each incoming stream.

With probability 10% a request from a civilian gets additional instructions, which is independent of the urgency. This value is obtained from expert guesses. The rest of the applicants do not need additional instructions.

2.4.4 Bernoulli parameter estimations for the dispatching domain

The Bernoulli distributions in the dispatching domain are structured in blocks: for each block these parameters consist out of the follow-up contact, end of service, and forward jump probabilities. Let us describe these parameters for the $B = 6$ blocks that we use in our simulation runs.

The probabilities for a follow-up contact are listed in Table 2.4. They are all based on expert opinion, and they are addressed for each block individually in high detail. The probability distribution for the infinite server agents for each block is obtained from the service detail records; these are the durations for the travel times, treatment duration, transfer duration at the hospital, etc. We fitted a log-normal, normal, and exponential distribution and used the mean squared error to determine the best fit. These distributions depend only on the urgency and status, and in particular not on the incoming stream. Another choice one could have made is to use an empirical distribution. The best-fit distributions and their parameter values from our data set are listed in Table 2.5.

Forward jumps only occur from *Block*$_4$ *(During patient treatment)* to *Block*$_6$ *(Transfer at the hospital)*; that is when a patient does not require transport to a hospital. The probability that a forward jump $pj_4^6(u)$ occurs is 41.87% for high urgency, 40.34% for medium urgency, and 15.29% for low urgency requests. We only use a redirect to *End of Service* in the last block.

We look at each of the blocks individually to set the remaining routing parameters. Parameters that are not mentioned explicitly are zero-valued.

Block$_1$ *(dispatch)*

This block includes the dispatch process of choosing the right vehicle for every honored request, and the contact moment from the dispatch center to the ambulance team to give initial instructions. Also time for optimizing the coverage of the fleet using dynamic ambulance management (DAM) methodology, see Chapter 6, is included in this block. Giving the initial instructions often is done digitally by sending a notification to the team's pagers. Note that the block's infinite-server pool has a zero service time for all calls, because there is no time delay between call taking and dispatching other than the queues of the dispatcher pool. There is a feedback loop which directs to the dispatch-queue whose name is similar to the calls urgency $qf_1^{D:High}(High) = qf_1^{D:Medium}(Medium) = qf_1^{D:Low}(Low) = 1$; see Table 2.7 in Section 2.4.6 for details on the queues. Based on the assumption that enough

Block	Status	Urgency (in %)		
		High	Medium	Low
Block$_1$	Dispatch	100	100	100
Block$_2$	Leaving the base location	15	15	15
Block$_3$	Driving to patient	30	10	10
Block$_4$	During patient treatment	10	0	0
Block$_5$	Transport of patient	0	0	0
Block$_6$	Patient transfer at hospital	100	100	100

Table 2.4 The probability $\sum_{i=1}^{M} qf_b^i(u)$ that a follow-up contact occurs (in %).

Block	Status	Urgency	Distribution	μ	σ	Mean	SD
Block$_1$	Dispatch	H/M/L	Deterministic	0	*N.A.*	00:00	00:00
Block$_2$	Leaving the base location	High	Log-normal	3.926	0.624	01:02	00:42
	Chute time	Medium	Log-normal	4.244	0.717	01:30	01:14
		Low	Exponential	145.280	*N.A.*	02:25	02:25
Block$_3$	Driving to patient	High	Log-normal	5.836	0.482	06:25	03:16
		Medium	Normal	622.342	300.994	10:22	05:01
		Low	Normal	734.297	444.314	12:14	07:24
Block$_4$	During patient treatment	High	Log-normal	7.102	0.471	22:55	11:16
		Medium	Log-normal	6.799	0.683	18:53	14:33
		Low	Log-normal	6.783	0.480	16:30	08:53
Block$_5$	Transport of patient	High	Log-normal	6.518	0.549	13:07	07:46
		Medium	Normal	847.788	411.833	14:08	06:52
		Low	Log-normal	6.886	0.536	18:50	10:52
Block$_6$	Patient transfer at hospital	High	Log-normal	6.917	0.491	18:59	09:55
		Medium	Normal	975.745	439.378	16:16	07:19
		Low	Normal	955.944	470.430	15:56	07:50

Table 2.5 The service time distribution for the infinite-server pool of every block in seconds, or min:sec.

ambulances are available, a request cannot end at this stage: $pe_1(u) = 1$. When an ambulance is already on the road, there is no departure from the base, but the EMS team has motivation for similar questions to the dispatch center and $Block_2$ is not skipped. Notice that in this case its block name is not fully accurate, although the behavior can be included in the service time distribution for the infinite-server pool in $Block_2$. There is always a redirect to $Block_2$: $pj_1^2(u) = 1$ for all $u \in \mathcal{U}$.

$Block_2$ (leaving the base location)
When an ambulance team reads the incident description on the on-board monitor, there may be pressing questions about medical uncertainties or special equipment that must be taken on board. Another reason for contact is that the incident location might be unclear to the EMS team. Under the assumption that a request is not canceled at this stage, the request is passed to $Block_3$: $pe_2(u) = 1, pj_2^3(u) = 1$ for all $u \in \mathcal{U}$.

$Block_3$ (driving to patient)
When the ambulance arrives at the incident location, there can be a contact moment with the dispatch center for various reasons: the EMS team can find the patient or there is a request for assistance by another team. In most cases the dispatch center specifies the location in more detail. Because the EMS team is assumed to search for the patient or start treating, there is a forward jump to $Block_4$: $pj_3^4(u) = 1$ for all $u \in \mathcal{U}$.

$Block_4$ (during patient treatment)
Depending on the findings at the incident locations, there are multiple possible outcomes. When a patient is treated and needs transportation to a hospital, the crew can contact the dispatch center to ask them to notify the hospital's emergency department. When a patient is not found, not yet ready for transportation, or can be treated at the incident location, the crew becomes available again. In that case they can give a situation update to the dispatch center and go to a base location to wait for a new request being assigned to them. When a patient needs transport to a hospital or other destination we redirect to $Block_5$. If a patient is treated at the incident location, forward jumps occur with probabilities $pj_4^6(u)$ as discussed earlier.

$Block_5$ (transport of patient)
It is unlikely that a contact occurs upon arrival at a hospital. Ambulance providers whose dispatch center agents provide extra motivation to the EMS team to become available again on short notice may include those contact moments in this block. We include this stage as a delay, although it could be merged with $Block_6$ in our case. The only non-zero parameters are $pe_5(u) = 1$ and $pf_5(u) = 1$ for all $u \in \mathcal{U}$.

Block$_6$ (transfer at the hospital)

At the hospital the patient is transferred to another health care provider, and the EMS team take a few minutes to refresh themselves. Ambulances in some EMS regions refill medical materials at the hospital. When the ambulance has handed over the patient, the dispatch center is notified that they are available for dispatch again. Since this is the last block in line, it results by definition in an *End of Service*: $pe_5 = 0$.

Incoming stream	Priority	μ	σ	Mean	SD
Civilian	High	4.604	0.695	2:07	1:40
	Medium	4.579	0.697	2:04	1:38
	Low	4.689	0.616	2:11	1:29
Police	High	4.432	0.761	1:52	1:39
	Medium	4.417	0.689	1:45	1:22
	Low	4.744	0.496	2:10	1:09
Others	High	4.559	0.746	2:06	1:49
	Medium	4.645	0.667	2:10	1:37
	Low	4.620	0.606	2:02	1:21
Health care institutions	High	4.518	0.538	1:46	1:01
	Medium	4.821	0.502	2:21	1:15
	Low	4.615	0.599	2:01	1:19
Hospital–high	High	4.471	0.582	1:44	1:06
	Medium	4.695	0.500	2:04	1:06
	Low	*N.A.*	*N.A.*	*N.A.*	*N.A.*
GP–high	High	4.538	0.579	1:50	1:10
	Medium	4.744	0.390	2:04	0:50
	Low	*N.A.*	*N.A.*	*N.A.*	*N.A.*
GP–low	High	*N.A.*	*N.A.*	*N.A.*	*N.A.*
	Medium	4.744	0.390	2:04	0:50
	Low	4.715	0.540	2:09	1:15
Hospital–low	High	*N.A.*	*N.A.*	*N.A.*	*N.A.*
	Medium	*N.A.*	*N.A.*	*N.A.*	*N.A.*
	Low	4.644	0.605	2:05	1:23
EMS dispatch center	High	4.471	0.737	1:55	1:37
	Medium	4.626	0.687	2:09	1:40
	Low	4.579	0.657	2:01	1:29
Additional instructions	All	3.926	0.624	1:02	0:43
Dispatch	High	3.926	0.624	1:02	0:43
	Medium	4.244	0.717	1:30	1:14
	Low	3.988	0.570	1:03	0:39
Follow-up call	Medium	3.389	0.846	0:42	0:43

Table 2.6 The log-normal service time distribution for the dispatch center agents is shown below. Units are in seconds, or min:sec.

2.4.5 Service time distributions

We discuss the service time distributions for the call taking and dispatching domains separately.

Call taking domain

In line with related papers in the literature [27, 37, 144], we assume that the service time distribution at the call taker is log-normal for each incoming stream and urgency couple. For all models we take the same service time distributions. The parameter values for every incoming stream and priority couple are listed in Table 2.6.

Hospitals and GPs each have two separate lines to reach the dispatch center and are able to prioritize their request using these lines. For hospitals we assume that the high urgency line is used if and only if it leads to a high urgency call. For GPs we assume that high and low urgencies are from the high priority and low priory lines, respectively. For the medium urgency we assume that the calls were evenly distributed over the high and medium urgency lines.

The service time distribution for the additional instructions equals the chute time of high urgency calls, because we have no data to support a better assumption, and this choice feels reasonable. Recall that the chute time is the time it takes the EMS team to leave the base location.

Dispatching domain

The service time distribution for the dispatch time is also not contained in the data sets provided, and hence is assumed to be the same as the service time distributions of the chute time for high and medium urgency calls. Because the dispatch of low urgency calls require less time and are assumed to be log-normally distributed, we have omitted outliers that are over three standard deviations above the mean, and made a best fit log-normal distribution.

The service time distribution of the dispatcher's follow-up contacts is obtained from unmatched TI records. Because we were unable to distinguish between the status in the process, priority or urgency, we used the same best-fit log-normal distribution for every follow-up contact with a mean of 42 seconds.

2.4.6 Routing policies toward the priority queues

Regardless of the model, all agent pools handle tasks on a priority based first-come first-served policy. This means that tasks with a higher priority are handled first, and for tasks with the same priority the call takers handle them on a first-come first-served basis. Notice that the infinite-server pools in the blocks have no queues and priorities because there are enough agents to start any incoming service directly.

Call taking domain		
Priority queue	Name	Task originates from
Q_1^C	C: Ultra high	Additional instructions.
Q_2^C	C: High	Incoming streams: civilian, GP–high.
Q_3^C	C: Medium	Incoming streams: police, others, health care institutions, hospital–high, GP–low, dispatch center.
Q_4^C	C: Low	Incoming stream: hospital–low.

Dispatching domain		
Priority queue	Name	Task originates from
Q_1^D	D: High	High urgency, from $Block_1(dispatch)$.
Q_2^D	D: Medium	Medium urgency, from $Block_1(dispatch)$. All feedback from $Block_2(leaving\ the\ base\ location)$ up to and including $Block_6(transfer\ at\ the\ hospital)$.
Q_3^D	D: Low	Low urgency, from $Block_1(dispatch)$.

Table 2.7 Priority queues for dispatch center agents for the model with function differentiation.

Model with function differentiation

The priority queues in the model with function differentiation are listed in Table 2.7. In the call taking domain, the civilian calls and high urgency general practitioners calls are of high priority. Giving their additional instructions a slightly higher priority leads to an uninterrupted call: these additional instruction tasks are directly picked up by a call taker after finishing the triage process and the number of additional instructions cannot exceed the number of call takers, thus all additional instructions have zero waiting time.

In the dispatching domain, the dispatch of high urgency calls have the highest priority, and the dispatches of low urgency, often ordered patient transports, are done when there are no remaining tasks left. Relocations are considered part of the dispatch procedure.

Model with solely generalists

Table 2.8 describes the priority queues of the model with solely generalists: the two priority queues of the model with function differentiation are zipped together. The main idea is that dispatching an ambulance to a request with a certain urgency is more important than taking a call with compatible priority. In fact, communication with a team that may need additional assistance may be more important than taking a new request from the police.

2.5 Results

To assess performance, and to gain insight into the optimal staffing decisions in dispatch centers, we have performed extensive simulation experiments based on the models with and without function differentiation described above. Recall that the KPIs are:

1. the *waiting time* before the call is taken, for high, medium and low priority classes,
2. the *average call center time*, i.e., the time duration from call entering the call taker queue until a call is dispatched, and
3. the *average* workload for a call center agent, for call takers, dispatchers and generalists.

In general, the cost of hiring different agent classes depend on the required level of education. The total yearly cost for call takers, dispatchers, and generalists is assumed to be €70*k*, €55*k*, and €90*k*, respectively; these numbers are representative for the Netherlands. For convenience, we denote the *staffing policy* by the triple

$$\underline{n} := (n_{calltaker}, n_{dispatcher}, n_{generalist}), \qquad (2.1)$$

where $n_{calltaker}$, $n_{dispatcher}$, and $n_{generalist}$ are the numbers of call takers, dispatchers, and generalists, respectively.

Solely generalists versus function differentiation

Table 2.9 shows the results of extensive simulations for the models with solely generalists and function differentiation. For each arrival rate between 100 and 3000, and for both models, we determined what the lowest cost and corresponding policy are for which the KPI requirements are met. For brevity

Priority queue	Name	Task originates from
Q_1	C: Ultra high	Additional instructions.
Q_2	D: High	High urgency, from $Block_1(dispatch)$.
Q_3	C: High	Incoming streams: civilian, GP–high.
Q_4	D: Medium	Medium Urgency, from $Block_1(dispatch)$. All feedback from $Block_2$ to $Block_6$.
Q_5	C: Medium	Incoming streams: police, others, health care institutions, hospital–high, GP–low, dispatch center.
Q_6	D: Low	Low urgency, from $Block_1(dispatch)$.
Q_7	C: Low	Incoming stream: hospital–low.

Table 2.8 The priority queues for the model with solely generalists.

λ	Best model	$\underline{\mathbf{n}}$	n	Costs	Fraction within 6 sec			Call center time			Busy fraction		
				(in k€)	(in %)			(in min:sec)			CT	Disp	
					High	Medium	Low	High	Medium	Low			Gen (in %)
100	FD	(1, 1, 0)	2	125	96.3	87.4	85.9	3:09	3:22	3:45	14.5	11.1	
		(0, 0, 2)	2	180	99.8	98.1	96.9	2:59	3:04	3:15			12.8
200	SG	(0, 0, 2)	2	180	99.4	92.9	89.8	3:02	3:09	3:35			25.5
		(2, 1, 0)	3	195	99.7	97.2	96.3	3:01	3:10	3:42	14.5	22.1	
300	SG	(0, 0, 2)	2	180	98.7	85.6	80.2	3:06	3:17	4:13			38.3
		(2, 1, 0)	3	195	99.3	94.2	92.5	3:03	3:16	4:12	21.7	33.2	
400	FD	(2, 1, 0)	3	195	98.9	90.2	87.8	3:07	3:24	4:54	28.9	44.3	
		(0, 0, 3)	3	270	99.8	94.4	91.5	3:00	3:06	3:29			34.0
500	FD	(2, 2, 0)	4	250	98.2	85.3	82.0	3:07	3:19	3:47	36.2	27.7	
		(0, 0, 3)	3	270	99.7	90.4	84.9	3:02	3:09	3:50			42.6
600	FD	(2, 2, 0)	4	250	97.6	79.7	75.6	3:12	3:27	4:09	43.4	33.2	
		(0, 0, 3)	3	270	99.4	85.3	77.0	3:04	3:14	4:22			51.1
700	FD	(2, 2, 0)	4	250	96.8	73.7	68.0	3:17	3:37	4:37	50.6	38.7	
		(0, 0, 4)	4	360	99.90	93.6	88.8	3:00	3:06	3:36			44.7
800	FD	(3, 2, 0)	5	320	99.5	90.9	88.7	3:02	3:12	3:58	38.6	44.2	
		(0, 0, 4)	4	360	99.8	90.4	83.2	3:01	3:08	3:53			51.0
900	FD	(3, 2, 0)	5	320	99.4	88.0	84.8	3:03	3:15	4:16	43.4	49.8	
		(0, 0, 4)	4	360	99.8	86.4	77.0	3:02	3:11	4:19			57.4
1000	FD	(3, 2, 0)	5	320	99.2	84.8	80.3	3:05	3:18	4:38	48.2	55.2	
		(0, 0, 4)	4	360	99.7	81.3	68.7	3:04	3:15	4:57			63.8
1200	FD	(4, 3, 0)	7	445	99.90	92.7	89.7	3:01	3:08	3:37	43.4	44.2	
		(0, 0, 5)	5	450	99.96	88.0	77.6	3:01	3:09	4:12			61.2
1400	FD	(4, 3, 0)	7	445	99.8	88.7	84.4	3:02	3:11	3:56	50.6	51.6	
		(0, 0, 6)	6	540	100	92.4	83.5	3:00	3:06	3:49			59.5
1600	FD	(4, 3, 0)	7	445	99.6	83.6	77.0	3:05	3:16	4:26	57.8	58.9	
		(0, 0, 6)	6	540	99.98	87.1	72.5	3:01	3:08	4:29			68.0
1800	FD	(5, 3, 0)	8	515	99.90	91.7	87.3	3:01	3:09	4:35	52.0	66.3	
		(0, 0, 7)	7	630	99.98	91.7	79.1	3:00	3:06	4:00			65.6
2000	SG	(0, 0, 7)	7	630	99.98	88.0	69.4	3:01	3:08	4:44			72.8
		(6, 4, 0)	10	640	99.97	95.8	92.8	3:00	3:05	3:38	48.2	55.2	
2200	FD	(6, 4, 0)	10	640	99.96	94.0	89.2	3:00	3:07	3:53	53.0	60.7	
		(0, 0, 8)	8	720	99.99	92.1	76.3	3:00	3:06	4:10			70.1
2400	FD	(6, 4, 0)	10	640	99.95	91.6	84.9	3:01	3:08	4:14	57.7	66.2	
		(0, 0, 8)	8	720	99.99	88.3	65.7	3:01	3:08	4:57			76.4
2600	FD	(6, 4, 0)	10	640	99.95	88.7	80.1	3:02	3:11	4:45	62.5	71.7	
		(0, 0, 9)	9	810	99.98	92.1	73.2	3:00	3:06	4:18			73.6
2800	FD	(7, 5, 0)	12	765	99.97	93.8	87.6	3:00	3:06	3:46	57.7	61.8	
		(0, 0, 10)	10	900	99.99	94.6	79.3	2:59	3:05	3:56			71.3
3000	FD	(7, 5, 0)	12	765	99.96	91.6	84.2	3:01	3:08	4:00	61.8	66.1	
		(0, 0, 10)	10	900	100	92.3	71.7	3:00	3:06	4:26			76.3

Table 2.9 Simulation results for the optimal policies for function differentiation (FD) and solely generalists (SG) for various rates.

of the table we only show results in steps of 200 after $\lambda = 1000$, although we consider every λ that is a multiple of 100 in our analysis. We require that at least 95% of the calls must be taken within six seconds, and low urgency calls must have a call center time of at most five minutes. We do not pose a boundary on the busy fractions. We call a policy better than another one when it has a lower cost. Various interesting insights can be gained from this data.

First, there is not one model that outperforms the other for all rates. For the lowest arrival rate we see that we need two agents, though we can do it at a lower cost with function differentiation. For $\lambda = 300$ two generalists can reach the required performance. From that point, the lowest cost model with function differentiation is slightly cheaper than the model with solely generalists. The only exception can be found at $\lambda = 2000$, where the two models are nearly equal, since solely generalists are €10k cheaper on annual basis.

Second, the latencies for high priority calls are exceeding 99.5% in nearly all of the cases, especially for higher arrival rate values λ. For lower and medium priority calls we can see a clear, though not completely unanimous, the difference in favor of the model with function differentiation.

Third, busy fractions of call center agents do not exceed 76.4%. In general, one can say that generalists have a higher workload than call center agents in the model with function differentiation. The latter model has at most two call center agents more, which helps to explain this difference.

Fourth, notice that the best policy with function differentiation in the cases considered has at least the same number of dispatch center agents as the model with solely generalists. Thus the number of work stations will not decrease when switching from a solely generalists policy to function differentiation.

We observe that good prioritizing of tasks by the EMS call center agents, in combination with a good KPI requirement for the low urgency and low priority calls, leads to good results for medium and high urgency and priority calls. This is due to the fact that KPIs of higher urgency calls are not affected by the lower priorities.

Mixed model
Table 2.10 shows the results of extensive simulations of mixed policies, combined with the other two models. More precisely, it shows the KPIs for those combinations $\underline{n}$ as defined in Equation (2.1) on page 42, for which no additional agent (of any class) can be hired within the budget constraint, and for which also the system is stable. We call a model stable when every queue length stays within bounds. In the simulations, the total annual budget for hiring agents is €500k. The table shows a KPI for generalists, in addition to the three already mentioned:

4. The work distribution, i.e., the average fraction of the time that the generalists spend taking calls versus dispatching.

To gain insight into the implications of staffing decisions on the KPIs, we have performed extensive simulations. The results are outlined below. In the examples discussed below the call arrival rate was taken to be $\lambda = 2000$, i.e., a rate of 83.33 requests per hour, and the service time distributions were taken from Table 2.6. The results lead to a number of observations.

$\underline{\mathbf{n}}$	n	Fraction within 6 sec (in %)			Call center time (in min:sec)			Busy fraction (in %)			Work distribution gen. (in %)	
		High	Medium	Low	High	Medium	Low	CT	Disp	Gen	Call taking	Dispatching
(0, 4, 3)	7	95.8	29.9	2.5	05:02	07:01	33:51	*N.A.*	53.5	98.7	98	2
(0, 2, 4)	6	99.3	57.9	27.0	03:24	03:50	10:27	*N.A.*	77.3	88.8	81	19
(1, 4, 2)	7	95.8	30.3	3.3	05:03	07:08	22:34	98.9	53.6	98.2	97	3
(1, 2, 3)	6	99.3	58.9	28.8	03:24	03:51	10:21	91.2	78.4	87.3	76	24
(1, 1, 4)	6	99.9	70.6	39.1	03:10	03:26	09:37	88.5	84.1	84.3	59	41
(2, 4, 1)	7	95.8	30.6	4.8	05:05	07:06	06:02	98.2	54.0	97.3	95	5
(2, 3, 2)	7	99.3	66.8	46.5	03:18	03:40	05:56	83.7	64.1	75.2	81	19
(2, 1, 3)	6	99.9	72.4	42.5	03:10	03:26	10:56	84.7	86.7	84.6	47	53
(2, 0, 4)	6	99.9	77.2	46.6	03:06	03:20	11:26	83.7	*N.A.*	85.7	36	64
(3, 5, 0)	8	95.8	31.2	8.0	05:07	06:57	42:30	96.3	44.2	*N.A.*	*N.A.*	*N.A.*
(3, 3, 1)	7	99.3	68.0	49.9	03:18	03:39	05:57	79.1	66.4	73.4	70	30
(3, 2, 2)	7	99.9	82.2	66.0	03:05	03:16	05:30	73.9	75.0	69.1	49	51
(3, 0, 3)	6	99.9	78.7	51.0	03:07	03:21	24:08	77.7	*N.A.*	92.3	20	80
(4, 4, 0)	8	99.3	70.7	57.3	03:15	03:35	05:02	72.2	55.2	*N.A.*	*N.A.*	*N.A.*
(4, 2, 1)	7	99.9	83.9	71.1	03:04	03:16	07:33	67.0	82.3	77.2	27	73

Table 2.10 The simulation results for the mixed policies.

First, comparing the results for the cases (0, 4, 3) and (4, 2, 1), it is quite remarkable that we can observe that the fraction of calls that meets the 6-second target is higher for *all* priority classes in the case (4, 2, 1). This seems rather counter-intuitive, since it would be natural to say that swapping agents between different classes would favor at most one or two classes at the expense of another class. Note also that the total cost for the case (4, 2, 1) is 480, which is less than the cost of 490 for the case (0, 4, 3). Looking at the call center times, we see that the case (4, 2, 1) outperforms (0, 4, 3) for all urgency classes. To understand why this is the case, we observe from the work distribution results that in the case (0, 4, 3) the generalists are heavily loaded (in fact, 98.7% utilization), of which 98% is spent on call taking and only 2% on dispatching, whereas in the (4, 2, 1) case the workload is much more balanced.

Second, the results show that in none of the cases considered only generalists were chosen, whereas intuitively it would make sense to do so, because of the facts that a generalist is flexible and hiring a generalist is less expensive than hiring both a call taker and a dispatcher. Consider for example the case (0, 2, 4) and compare this with the case (0, 0, 5). In the former case, in which six persons are hired, the system is stable, whereas in the latter case the system is unstable.

Third, there is an important difference between general call centers and dispatch centers with respect to the utilization of specialists versus generalists. This is so because in general call centers generalists are expensive, and hence, call center planners will tend to maximize their busy fraction, because idle time of generalists is costly. On the contrary, in dispatch centers there is an incentive for planners to keep the utilization of generalists low, because the generalists' idle times can be used to support other EMS services. For example, compare the case (3, 3, 1) to the case (3, 2, 2). Their performances are similar, while in the case (3, 2, 2) the generalists are less heavily loaded.

Technical details

This study contains 2676 unique simulations runs: 2200 for a call center with function differentiations, 450 for a call center with solely generalists, and 26 for the mixed policy study. For each simulation run we simulated 50,000 incoming calls. If there were over a thousand calls in the system, that is the total of all queues, we said that the simulation is unstable and excluded it from further analysis.

The simulation engine is written in a C++ and MariaDB using the TIFAR-framework that we developed [A7], see Chapter 7. The correct working of the software was validated by intensively tracing individual tasks and call center agents. We validated the input distributions and parameters against the output database.

The simulations have run on a Calleo Application Server 2260 that is dedicated to this project. This machine contains two Intel Xeon Processor E5-2640v2 (8 cores, 20MB cache, 2.0GHz) and hyper-threading enabled, resulting in maximal 32 cores. The RAM equals 256GB at 1600MHz. Running all 2676 required simulations took 37,871 wall seconds (10h 31m), and during this time period on average 14.12 CPU's where in use simultaneously. TIFAR makes use of multi-threading and was limited to 16 cores maximal, as the database and operation system also needed processing power.

2.6 Conclusion

In this chapter we presented three performance and capacity simulation models for dispatch centers. The first model, with the function differentiation policy includes two classes of dispatch center agents: call takers and dispatchers. The second model contains solely generalists, a call center agent class that can do both call taker and dispatcher tasks. The third, mixed model, allows for a mixture of call takers, dispatchers and generalists.

A key feature of the models is that it includes follow-up calls from EMS teams and hospitals. The model also discriminates between multiple types of applicants which differ in priorities. The model enables EMS planners to better understand the impact of these features on the response time performance of dispatch centers.

In the comparison between the function differentiation and solely generalists models, we concluded that there is not one model that outperforms the other for all arrival rate. Another interesting conclusion is that if the two minimal requirements (1) ALS calls are answered within six seconds and (2) low urgency calls are answered within five minutes are met, then in most cases the in 99.5% of the cases the telephone is answered within six seconds. In that case the busy fraction of the agents does not exceed the 96%.

In the analysis of the mixed model our primary finding is that it can be beneficial to have one generalist working alongside multiple call takers and dispatchers: the generalist can than take the overflow from one of the other policies.

We continue with some remarks.

We assume that the service time distributions of both call takers and dispatchers are independent of the workload. In reality, the time used for servicing a call may depend on the workload and service time decreases when the workload increases. The inclusion of workload dependent service time distributions may have a significant impact on the response time performance of dispatch centers.

The input originates from actual call center databases. There are some estimations in our results; they cannot directly be used for management decisions without further research. Probability distributions and parameters may differ for various ambulance regions.

Our model assumes that switching occurs instantaneously. However, in practice switching between tasks takes time, which is a motivation for the six-second response time threshold. The inclusion of non-negligible switching times is an interesting subject.

There are some secondary advantages and disadvantages in function differentiation. In a dispatch center with function differentiation a general assumption is that dispatchers work faster than generalists in doing the same tasks. This is because the full focus is on the logistics domain. An advantage of generalists is that they are easy to deploy in the case of sickness absence.

The models presented in this chapter are also applicable to other call centers that take calls, dispatch and perform coordination, in situations where short response times are required. These include, but are not limited, to police, firefighters, taxi service, and roadside assistance. However, before proper use in another context the model's input parameters should be adapted, amongst others, the arrival distributions, the number of urgencies, priorities, queues, and blocks.

This chapter opens up interesting topics for further research.

One may consider a dispatch center as a multi-skilled call center. This means that there are people, the so-called junior call takers, who can only take the easy calls from selected incoming streams, e.g., from police, firefighters and hospitals. In addition, newly introduced senior call takers are able to serve life-threatening calls. We can even introduce a third call taker class, equipped to handle all calls. This also opens the possibility of other combinations, such as dispatch center agents who do both dispatching and low priority call taking, which is a combination that also exists in practice.

Another interesting extension of the models is to include so-called *ramping*, which occurs when the emergency departments of the hospitals are over capacity, which implies that ambulances have to wait in line to transfer a patient. Regions suffering from occasional ramping may want to include extra contact moments with a dispatch center to cope with this load. We have not included ramping in the current model since this is not seen in the ambulance region that provided us with the data sets.

It is interesting for future research to evaluate the use of these models in other, non-EMS, emergency call centers, and conclude if the claims we make also hold in these contexts.

As a final remark, note that the effect on the choice of triage protocols, which is also in focus in various parts of the world, is not considered in this study. This choice is part of the service time distributions, and therefore it is considered as an input for our models.

Acknowledgements
We thank Regionale Ambulance Voorziening Utrecht (RAVU) for sharing the dispatch center data. We thank numerous EMS drivers and nurses participating for their expert guesses.

3

THE ADJUSTED QUEUING FRAMEWORK

Minimal reliability and maximal availability models guarantee a minimum performance level at *each* demand point, in contrast to the majority of facility location and allocation models that guarantee a minimum performance that is *aggregated* over the entire ambulance region. As a consequence, existing models generally lead to overstaffing, particularly in 'mixed' regions with both urban and rural areas, which leads to unnecessarily high costs. This chapter addresses this strategic problem. To this end, we first introduce the concept of *demand projection* to give fundamental insight into why this overstaffing takes place. Next, we overcome the overstaffing by the so-called *adjusted queuing* (AQ) approach that provides generalizations of the existing models. We provide proofs for the correctness of the AQ-approach.

This chapter is based on the following publication:

[A3] M. van Buuren, R. D. van der Mei, and S. Bhulai. "Demand-point Constrained EMS Vehicle Allocation Problems for Regions with Both Urban and Rural Areas". *To appear in Operations Research for Health Care* (2018)

3.1 Introduction

Ambulance server providers are interested in what the best locations for ambulance bases are, and what the optimal number of ambulances is for each of these bases. Overcapacity of ambulances leads to unnecessarily high costs, while extensively reducing the costs can lead to dangerous situations.

Most systems try to express the performance of an ambulance region in a single value, e.g., by calculating the fraction of late arrivals aggregated over the entire ambulance region for a duration of one year. When maximizing this covered fraction, it may lead to low coverage in rural areas in favor of densely populated cities. This problem is addressed in the so-called *regional coverage location problem* (RCLP) class of models; a name that stresses the regionally aggregated key performance indicator.

Another approach is to evaluate the performance of each subarea in the region individually, and satisfy at least a minimum required performance threshold that is set for each subarea in the best possible way. This can be achieved by giving a minimum performance constraint to every subarea.

This either means to find an ambulance allocation that maximizes the total demand over the subareas that receive the minimum required performance when limited resources are available (maximal availability), or to determine the minimum number of ambulances and the resulting allocation such that the minimum requirement for every subarea is satisfied (minimal reliability). This leads to the *minimal reliability and maximum availability* (MR-MA) class of models. The current chapter has its focus on this class.

Currently, RCLP is mostly used in practice, although we note an emerging balance shift in favor to MR-MA. Erkut, Ingolfsson, and Budge [39] wrote in what they call a critique on the MR-MA models that “the objective functions of the models in this class are not the same as the expected coverage performance measure that typically drives EMS system design”. This is still a valid argument though it loses its strength as time goes on. The trend shift from ambulance practice moving away from purely focusing on this RCLP class objective is due to pressure from local intra-regional governments. Although the key performance indicator for the regions that we consider in the results section in practice still is the fraction of calls covered within a time threshold of 15 minutes measured on a yearly basis, ASPs have to deal with mayors of rural municipalities in their region who insist on a minimum coverage level for their own population. As a result, every subarea (or: municipality) within the ambulance region must also receive coverage under a minimum reliability constraint. The current MR-MA models in the literature are not suited for regions that have both rural and urban areas. The practical need for MR-MA models that are suitable for these so-called mixed regions provides the motivation for this research.

The existing high-end MR-MA models in the literature Q-PLSCP and Q-MALP [82, 83], throughout this thesis referred to as the Q-models, are made for regions where the demand is fairly homogeneously spread. If they are applied to actual regions with inhomogeneous demand—against their original design—they generally lead to over-estimations for the required number of vehicles [20, 22]. As a consequence, ambulance providers have an unnecessarily high cost.

In this chapter we propose a new MR-MA approach that is applicable to regions with both urban and rural demand: the so-called *mixed regions*. Our approach is based on many concepts that can be found in the Q-models, to such an extent that we call our solution the *adjusted queuing* approach.

The present chapter can be divided into two parts. First, we provide an *in-depth explanation* of why the over-estimation takes place when Q-models are directly applied to mixed regions. Secondly, we propose the *AQ-approach* that leads to credible results. We use the minimal reliability model Q-PLSCP for illustration purposes throughout the current chapter, although the findings are not limited to this model.

The *adjusted queuing* (AQ) approach improves on the queuing approach as follows.

- Q-models tend to project urban demand to rural areas that may lead to major local overstaffing in rural areas. The AQ-approach that we propose solves this problem, leading to better staffing for these rural areas.

- Contrary to Q-models, we allow for major differences in demand and service time.

- In AQ-models, the required reliability or availability level is demand point dependent instead of a system wide constant.

We provide proofs that the AQ-approach works. Existing papers use simulation studies to illustrate that their approach work [82, 83]. Because the models in these papers are special cases of AQ-models, this chapter also includes the proofs of these models.

The remainder of this chapter is organized as follows. In Section 3.2 we give an extended literature review on the line of models that leads to the Q-models. Section 3.3 starts with some definitions and assumptions on the ambulance practice, and it provides a detailed description of the Q-PSCLP model that is required later on in the chapter. We show in Section 3.4 why the Q-models give over-estimations on the number of ambulances needed in an ambulance region. In Section 3.5 we replace the main assumption of previous models with the more general workload condition. In Section 3.6 we give a solution to the over-estimation. Section 3.7 contains the conclusion and gives advice for future research.

Chapter 4 is a continuation of this chapter, that shows the extent of the improvements for minimal reliability models through numerical results.

3.2 Literature

This section starts with an overview of the RCLP models, and elaborate on the model MEXCLP that is referred to on multiple occasions throughout this thesis. Subsequently, we describe the MR-MA models. We end this section by mentioning a few other related research topics in the field of EMS logistics.

Good reviews on EMS facility location and ambulance allocation models are available [31, 78, 81]. Facility location and staffing take place in the strategic and tactical domain of EMS. Golberg [51] discusses the properties, advantages, and disadvantages of the various models in his review paper. Recent review papers are [17] and [107].

3.2.1 Regional coverage location problems

Early proposed ambulance location models were linear integer programming formulations, such as the *Maximal Covering Location Problem* (MCLP) [32]. This model discretizes an ambulance region into a set of *demand points*, and maps each incident onto the nearest demand point. The expected workload in a demand point is called the *demand.* MCLP positions a given number of ambulances such that the demand that is covered at least once is maximized. The RCLP model called *Maximum Expected Coverage Location Problem* (MEXCLP) [35] extends on MCLP, and is amongst the first that can be used to link EMS facility location to the stochastic nature of EMS logistics, which is the largest shortcoming of MCLP. Many papers can be found in the literature that are in some sense extension of MEXCLP.

A drawback of a single coverage function as in MCLP is that once the ambulance gets dispatched, people in the same area of the previous incident can become out-of-reach of ambulance care. To address this issue, the notion of double coverage is introduced in the *Backup Coverage Problems* (BACOP) [56] and the *Double Standard Model* (DSM) [49]. An additional feature of DSM is that it allows for two different response time thresholds. A fleet can contain multiple vehicle types, which is addressed in the TEAM and FLEET models that are also based on MCLP [120]. The main difference is that the FLEET model allows ambulances of different types to be positioned on separate base locations, whereas the TEAM model can only position the second ambulance type where a unit of the first type is present. The *Two-tiered Model* (TTM) brings the stochastic nature of EMS in the [80] is the TEAM model's objective function. The MOFLEET model combines MEXCLP and FLEET [24]. A time-dependent version of MEXCLP, called TIMEXCLP, runs MEXCLP once for every time interval [105]. More recently, Rajagopalan et al. [103] have tried four multiple meta-heuristic search approaches to find good solutions in case MEXCLP becomes hard to solve.

3.2.2 The maximum expected coverage location problem

The MEXCLP, as published in [35], is an IP-formulation that maximizes the expected covered demand. We use the model in Chapters 5 and 6 as well. The model defines the coverage of a region in terms of a *busy fraction*, which it denotes as q. The busy fraction can be estimated by dividing the expected workload of the system by the total number of available ambulances. This busy fraction is predetermined, and is assumed to be the same for all vehicles. Furthermore, vehicles are assumed to operate independently. Consider a demand point i, which is within the range of k_i ambulances. Using expected travel times we can directly determine this number k. The travel times should be taken as estimates for movements, which are faster because ambulance sirens are on. The probability that at least one of these k_i ambulances is available at any point in time is then given by $1-q^{k_i}$. Denote the demand by d_i, that is a proxy for the fraction of the total workload of the ambulance region that is aggregated to demand point i. The expected covered demand of this vertex is $E_i = d_i(1-q^{k_i})$. The MEXCLP policy positions the ambulances such that the total maximum expected covered demand, summed over all demand vertices, is reached. This method has two major disadvantages that other newer methods still inherit:

1. Many regions have both rural and urban areas. A constant system-wide busy fraction q for each ambulance is not realistic because demand points with fewer demands will most likely have a lower busy fraction.

2. Rural areas have a lower population density, so the method decreases coverage in rural areas in favor of densely populated urban population. Major differences in ambulance care between a region's population can occur. From an equity perspective this may be deemed unfair. For further work on fairness in EMS logistics we refer to [64].

Although the MEXCLP model has some limitations, most notably the assumption of the ambulances being independent, it is still widely used as starting model for extensions. For instance, in [16], the hypercube correction factors proposed in [75] were incorporated in the MEXCLP model to relax this independence assumption. Various extensions on the hypercube model are provided in the literature [16, 46, 65, 76, 103]. In [40], the MEXCLP model is extended to a model in which survival probabilities and probabilistic response times are incorporated.

3.2.3 Minimal reliability and maximal availability models

The MR-MA class of models has a completely different approach at which a minimum local coverage performance level is set for every demand point, called the *minimum reliability level*. Minimal reliability (min-rel) guarantees that every demand point has the minimum required coverage, and minimizes the total number of ambulances in the system to achieve this. Maximal availability (max-av) swaps these objective function and constraints, thus allocating a fixed number of ambulances, such that a maximum demand is covered under the minimum reliability level.

The *Set Coverage Location Problem* (SCLP) [132] is the first minimal reliability model that optimizes the number of facilities such that each demand point can be reached by at least one ambulance within a given response time threshold. The PLSCP model [110] poses an upper-bound on the reliability of every demand point, and it ensures that every demand point is covered by a minimum reliability level. The *Reliability Perspective* (Rep-P) [11] provides a generalization of the objective function of PLSCP that allows for various cost functions and blocking probabilities.

The *Maximal Availability Location Problem* (MALP) [109] is the first maximal availability model. A demand point is covered when the probability that it can be reached within the response time (or distance) threshold by at least one ambulance exceeds a constant α. In the first MALP model there is a system-wide busy fraction for the ambulances used, and a binomial approach similar to MEXCLP calculates for each demand point the minimum number of ambulances that is required to satisfy the required minimum required coverage probability constraint. MALP allocates a fixed number of ambulances, such that the total covered demand is maximized. The same paper [109] proposes an extension where the ambulance's busy fraction depends on the demand point.

MALP and Rel-P are adjusted by [28] such that the busy fraction of each base location depends on a preference list that each demand point holds. This way, the busy fraction becomes more realistic, and the authors show that it leads to a reduction in the number of vehicles. The LR-MEXCLP [126] is a combination of MEXCLP's maximum coverage objective with the reliability constraints of MALP. This is achieved by using a slope with the reliability by which a demand point is covered instead of the Boolean coverage constraint by demand point. A multi-objective extension of Q-MALP is presented in [54].

Q-PLSCP and Q-MALP by Marianov and Revelle are the queuing versions of PLSCP and MALP [82, 83]. Instead of a binomial approach, they use an Erlang B formulation. Our AQ-models extend these models. Subsection 3.3.4 describes the Q-PLSCP in detail. Because the literature is rather limited on the subject, the difference in assumptions and outcomes between the binomial approach of PLSCP and MALP and the queuing approach of Q-PLSCP and Q-MALP is described in Section 3.3.5.

3.3 Preliminaries

This section starts by giving definitions and abbreviations for the ambulance process. Next, we discuss the assumptions on the ambulance practice. Thereafter, we give a detailed description of the Q-PLSCP model that is our baseline for illustration purposes. Since the queuing approach for ambulance allocation models is not generally known, we end this section by sharing the differences between this approach and the binomial approach found in many papers.

3.3.1 Definitions

We briefly repeat the ambulance service process in order to introduce abbreviations that are used in the remainder of this thesis, for an extensive description we refer to Section 1.1.4. Figure 3.1 illustrates the stages and time intervals of the EMS process.

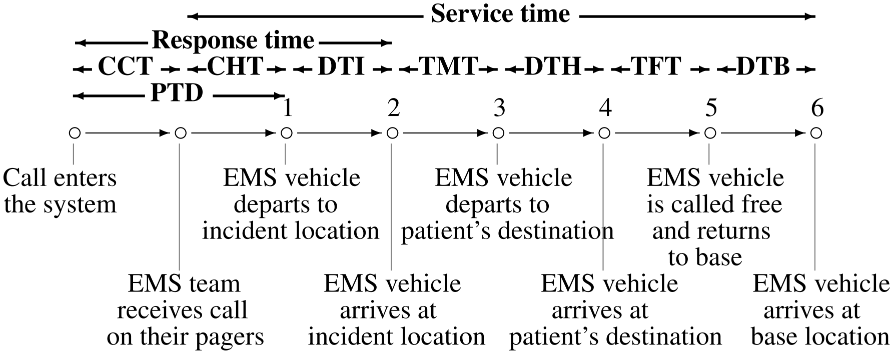

Service time
Response time
CCT
CHT
DTI
TMT
DTH
TFT
DTB
PTD
1
2
3
4
5
6
Call enters the system
EMS vehicle departs to incident location
EMS vehicle departs to patient's destination
EMS vehicle is called free and returns to base
EMS team receives call on their pagers
EMS vehicle arrives at incident location
EMS vehicle arrives at patient's destination
EMS vehicle arrives at base location

Figure 3.1 Trace of call statuses and the corresponding time intervals.

When a call enters the system, an agent at the dispatch center needs time to perform the triage procedure, and if required, dispatch an ambulance; this is the so-called *call center time* (CCT). When the pagers of the EMTs are activated at dispatch, some time passes before the ambulance starts moving, since they have to get to the vehicle. This time period is the *chute time* (CHT). The entire duration from a call entering the system until the ambulance starts moving is the *pre-trip delay* (PTD).

The next stage is the *driving time to the incident* (DTI), which is followed by the *treatment time* (TMT). In the case of a declarable call the patient is brought to the hospital. We call this time interval the *driving time to the hospital* (DTH), after which we have a *transfer time* (TFT) that bridges pre-hospital care with hospital care. The last stage of the ambulance trip consists in *driving back to its base location* (DTB).

The *service time* is the time interval during which an ambulance is busy and cannot respond to newly incoming incidents. This starts when the EMS team receives a notification from the dispatch center, and it stops when the ambulance arrives back at its base location. Note that some authors define the service time as the moment that the EMS vehicle becomes available at the hospital. The choice that ambulances become available at their base location guarantees that rural demand points at the region borders directly receive coverage when an ambulance becomes available again, since hospitals are most often located in urban areas.

The *response time* starts from the moment the call enters the system, and it ends when the ambulance arrives at the patient.

3.3.2 Assumptions

Incidents only occur at demand points. The number of demand points should be large enough to give an adequate representation of the region, but small enough to perform calculations within an acceptable time. Postal code areas with several thousand inhabitants are a good candidate for a demand aggregation.

We have a fixed and bounded set of demand points, potential base locations, and hospitals in each ambulance region. These locations are all assumed to be known a priori. It is possible to give a model free choice of base locations by making every demand point a potential base location. Every demand point can be reached by at least one potential base location within the response time threshold. We assume that a region contains at least one hospital.

Arrivals occur according to a Poisson process. For every demand point the frequency of incident arrivals is given. There is only one urgency class, and there is one type of ambulance that can handle all calls. The service time may depend on the incident location, base location, and ambulance allocation, and it is determinable and finite for all calls.

Parameters PTD, TMT, and TFT are assumed constant in this chapter; they are the same for every call. The driving times DTI, DTH, and DTB can be obtained from a lookup table or by navigation software. Travel times are assumed to be symmetric, i.e., swapping origin and destination of one route has no effect on the driving times.

Once assigned to a base location, the ambulance starts and ends every service on its base location. There are no relocations of ambulances between base locations. Every hospital has unlimited capacity and any patient can be brought to any hospital. When necessary one can include ramping, i.e., the waiting duration until the hospital's emergency department (ED) has free space, by including its mean time in the TFT.

Any call that does not receive immediate service from an available ambulance within the given response time standards is lost. There is no queue for waiting calls. In practice, a lost call is either served by a neighboring ambulance service provider or by another base location that has the capacity.

For the Q-models the assumption is required that ambulances from base locations that have an overlap in demand points do not have major differences in call arrival rate, mean service time and minimum reliability level. This assumption is not required for the proposed AQ-models.

3.3.3 Variables

Let $\mathcal{H}$ be the finite set of hospital locations, let $\mathcal{I}$ be the set of demand points, and let $\mathcal{J}$ be the set of potential base locations. If a demand point is located at the same location as a hospital, we add two elements with the same coordinates; one for $\mathcal{H}$ and one for $\mathcal{I}$. The three sets are not empty. Denote $\mathcal{V} := \mathcal{H} \cup \mathcal{I} \cup \mathcal{J}$.

Denote the minimum driving time from point k to point ℓ by $t_{k,\ell}$ $(k,\ell \in \mathcal{V})$. Recall that symmetric driving times are assumed, thus $t_{k,\ell} = t_{\ell,k}$. We denote the *response time* by

$$r_{i,j} := \mathrm{PTD} + t_{j,i} \qquad (i \in \mathcal{I}, j \in \mathcal{J}).$$

If $r_{i,j} \leq R$ we say that i is *covered by j*, for a constant system-wide *response time threshold R*.

For each demand point $i \in \mathcal{I}$ we require a minimum reliability level of at least $\alpha_i \in [0,1)$, i.e., the probability that an ambulance is available to reach a patient within R time units must be at least α_i. A typical value one can take is $\alpha_i = \alpha = 0.95$ for all demand points $i \in \mathcal{I}$.

In minimal reliability models, the total number of ambulances in the system is denoted by Z as this is the variable we want to minimize, and the variable $x_j \geq 0$ denotes the utilized capacity of potential base location $j \in \mathcal{J}$. The utilized capacity of a potential base location is the number of ambulances that are allocated to this base.

For each demand point $i \in \mathcal{I}$, we denote the call arrival frequency by $f_i \geq 0$. This is the number of calls per time unit that enters the system at this demand point.

The mean service time of a demand point is the average time an ambulance is busy to a call that takes place at that demand point and therefore is not available for dispatch to a new incident. We denote the mean service time of demand point i by β_i. Usually, there are no major differences in the mean service times between demand points because differences in driving times are rather small compared to the entire service duration.

Denote the set of demand points that can be reached, i.e., have a response time within R time units from $i \in \mathcal{I}$ by

$$\mathcal{N}_i := \{i' \in \mathcal{I} \mid r_{i,i'} \leq R\},$$

throughout the thesis referred to as the *neighborhood* of demand point i. We can interpret this as the set of all demand points that would be covered if there were a base location collocated to i. For referral to demand points near a base location $j \in \mathcal{J}$ we extend the notation with $\mathcal{N}_j := \{i' \in \mathcal{I} \mid r_{j,i'} \leq R\}$, i.e., the set of demand points that can be reached from base location j. Similarly for the base locations near demand points or base locations, we define the set

$$\mathcal{M}_i := \{j' \in \mathcal{J} \mid r_{i,j'} \leq R\}.$$

3.3.4 Queuing probabilistic location set coverage problem

The Queuing Probabilistic Location Set Coverage Problem (Q-PLSCP) [83] is a minimal reliability model that minimizes the total number of ambulances in an ambulance region such that the minimum reliability level requirements are met for all demand points. This is done in two phases: the first calculates right-hand side values for the constraints of a mixed integer program (MIP), and the second phase solves the resulting MIP.

This model needs an extra assumption, the so-called *isolation assumption*, which is used to justify the application of the Erlang B formula later on.

Assumption 3.1 (isolation assumption). For each neighborhood the ambulance allocation can be considered as an isolated problem.

In Q-PLSCP, every neighborhood gets a number of ambulances b_i assigned ($i \in \mathcal{I}$). In the case that all b_i ambulances in $\mathcal{N}_i$ are busy, ambulances from bordering neighborhoods can respond. Similarly, ambulances from bordering neighborhoods may receive assistance from ambulances of $\mathcal{N}_i$. The influx and outflux cancel out, because there are minor differences between bordering neighborhoods in the call arrival rate, mean service time, and minimum reliability level. Thus each individual neighborhood may be treated as an isolated area; this addresses the core of isolation assumption. The average number of assigned ambulances that is available or serving calls in $\mathcal{N}_i$ is on average close to b_i.

By assumption we have independent Poisson arrivals, and we denote its arrival rate by $f_i \geq 0$ for all $i \in \mathcal{V}$. The service time duration is taken constant at β time units for all calls, and the minimum reliability level α is a fixed system-wide constant.

The arrival rate in neighborhood $\mathcal{N}_i$ is calculated by $\lambda_i = \sum_{k \in \mathcal{N}_i} f_k$. Using the isolation assumption, the Erlang B blocking formula gives the probability that no ambulance is available in neighborhood $\mathcal{N}_i$ if it has n_i serving ambulances:

$$\text{Erlang}_{\text{B}}(\lambda_i, \beta, n_i) = \frac{\frac{(\lambda_i \beta)^{n_i}}{n_i!}}{\sum_{\iota=0}^{n_i} \frac{(\lambda_i \beta)^{\iota}}{\iota!}}. \tag{3.1}$$

Because values for λ_i, β, and α are determined or given, the Erlang B formula provides a lower-bound on the number of required ambulances in neighborhood $\mathcal{N}_i$, which is denoted by $b_i \in \mathbb{N}$:

$$b_i = \text{argmin}_{n \in \mathbb{N}_{\geq 0}} \{\text{Erlang}_{\text{B}}(\lambda_i, \beta, n) \leq 1 - \alpha\}.$$

Due to the assumption of symmetric travel times, the number of ambulances in neighborhood $\mathcal{N}_i$ equals the number of ambulances that can reach demand point $i \in \mathcal{I}$ from their base location. Thus it poses a constraint of the form $\sum_{j \in \mathcal{M}_i} x_j \geq b_i$ to every demand point i when allocating ambulances to bases, which concludes the first phase. Recall that x_j denotes the number of ambulances at base $j \in \mathcal{J}$.

In the second phase an *integer program* is solved to ensure that every neighborhood gets coverage by at least the minimum number of required ambulances.

$$\begin{aligned} \min Z = \sum_{j \in \mathcal{J}} x_j & \\ \text{s.t.} \sum_{j \in \mathcal{M}_i} x_j \geq b_i, & \qquad (i \in \mathcal{I}) \\ x_i \in \mathbb{N}_{\geq 0}. & \end{aligned}$$

After solving the IP-problem x_j holds the number of ambulances allocated to base location $j \in \mathcal{J}$, and Z equals the total number of ambulances in the ambulance region.

The authors of [83] chose a fixed service time for all calls. The frequency f_i is taken proportional to the population at $i \in \mathcal{I}$. Instead of a travel time constraint, they took an action radius in miles from the base location to the demand points.

Reference [83] proves that the Erlang B approach works for exponential service times, although their result is just as valid for general service times. Hence, their method is much stronger than suggested by the paper.

3.3.5 Comparison of the binomial and queuing approaches

In this section we compare the Erlang Blocking approach used in the Q-models to the binomial approach that is used in PLSCP [110] and MALP [109]. The authors of the Q-models show with computational results that these models require fewer ambulances than PLSCP and MALP for the same coverage constraints. We consider a theoretical point of view and discuss the advantages and disadvantages of both approaches, and we conclude that generally the queuing approach outperforms the binomial approach when all its assumptions can be satisfied.

In calculating the blocking probabilities in facility location and allocation problems in the rather limited literature on this subject, we encounter two approaches: (1) the binomial, and (2) queuing approaches. In this subsection we compare the two approaches for a classical toy example: the island model. This comparison provides the motivation why we continue research on the queuing approach, instead of the binomial approach.

Consider a small enough island with one base location $j \in \mathcal{J}$, which covers the entire island, i.e., $|\mathcal{J}| = 1$. Particularly, this base location is not influenced by other base locations; see Figure 3.2. Subsequently, the set of the (potential) serving bases $\mathcal{M}_i$ is the same singleton for all demand points $i \in \mathcal{I}$. Also the arrival frequency f_i for each demand point $i \in \mathcal{I}$ is given, and thereby the total demand that must be served by ambulances that are stationed at base location j.

We pose some assumptions that are required by the models. Arrivals are independent and require service from one ambulance. We assume for this island is that every demand point can reach every other demand point within the response time constraint, i.e., $\mathcal{N}_i = \mathcal{I}$ for all $i \in \mathcal{I}$. The island contains one hospital and differences between the travel times are assumed negligible compared with the mean service time of a call, hence we take $\beta_i = \beta$ constant. We assume that the minimum reliability level requirement is constant: $\alpha_i = \alpha$ for all $i \in \mathcal{I}$. The last assumption is that any call that cannot be immediately assigned to an available ambulance is lost.

We take a number of ambulances $Z = x_j$ at $j \in \mathcal{J}$ and calculate the blocking probability for each approach.

Binomial approach

The binomial approach uses a fixed busy fraction q for every ambulance, that may be estimated from historically recorded data. Under this assumption, the probability that an ambulance is not available at call arrival is q. Hence, the probability that none of the x_j ambulances at this base is available is given by the following probability:

$$P_{Bin}(\text{no ambulance is available}) = q^{x_j}.$$

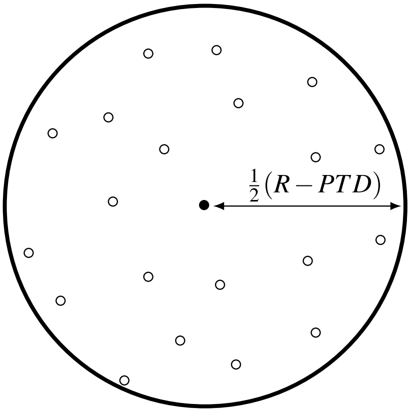


Figure 3.2 In the island model there is only one base location (denoted •). Every demand point (denoted ○) can reach every other demand point within the response time norm.

When determining the required number of ambulances such that a minimum reliability level of at least α is met, we get constraints of the form $q^{\sum_{j\in\mathcal{M}_i} x_j} \leq 1-\alpha$ for every demand point i. Taking $b^{Bin} = b_i^{Bin} := \log(1-\alpha)/\log(q)$ yields constraints

$$\sum_{j\in\mathcal{M}_i} x_j \geq b_i^{Bin} \qquad (i \in \mathcal{I}).$$

The island model has $Z = x_j = b_i^{Bin} = \log(1-\alpha)/\log(q)$ for the binomial approach (for all $i \in \mathcal{I}, j \in \mathcal{J}$).

General assumptions of the binomial approach are (1) that the busy fraction of an ambulance is constant, and (2) that the number of ambulances $\sum_{j\in\mathcal{J}} x_j$ is a good approximation to handle the workload of the entire region. When using the binomial approach for a realistic situation with multiple base locations an IP-formulation fits the number of ambulances to at most this (acceptable) busy fraction q instead of to the region's KPI.

Queuing approach

The queuing approach considers each neighborhood as an isolated area and uses the Erlang B blocking formula for an M/G/c/c-loss system with Poisson arrivals, general service time distribution with finite mean and a server pool with c servers, with the assumption that calls that cannot be directly served are lost.

Because of the small size of the island we consider the arrival rate λ_i equally valued for every demand point: $\lambda_i = \sum_{i' \in \mathcal{N}_i} f_{i'} = \sum_{i' \in \mathcal{I}} f_{i'}$. Recall that the mean service time β_i and minimum reliability level α_i are assumed constant over all $i \in \mathcal{I}$.

Knowing λ_i, β_i, and α_i we can calculate the minimum number of required ambulances stationed at $\mathcal{M}_i$ ($i \in \mathcal{I}$), which is equally valued for all demand points:

$$b_i^{Erl} = \operatorname{argmin}_{b' \in \mathbb{N}_{\geq 0}} \{\mathrm{Erlang}_{\mathrm{B}}(\lambda_i, \beta_i, b') \leq 1 - \alpha_i\}.$$

This leads to the same family of constraints as the binomial approach, but with a different calculated value for b_i:

$$\sum_{j \in \mathcal{M}_i} x_j \geq b_i^{Erl} \qquad (i \in \mathcal{I}).$$

The island model has $Z = x_j = b_i^{Erl} = \operatorname{argmin}_{b' \in \mathbb{N}_{\geq 0}} \{\mathrm{Erlang}_{\mathrm{B}}(\lambda_i, \beta_i, b') \leq 1 - \alpha_i\}$ for the queuing approach (for all $i \in \mathcal{I}, j \in \mathcal{J}$).

Differences between the approaches

The main difference between the two approaches is that the queuing approach does not consider the busy fraction as an *input* parameter, whereas the queuing approach *outputs* the busy fraction through the relation $q = q_i = \beta / (b_i \lambda_i)$.

By taking the busy fraction as an input parameter the binomial approach has an unwanted side effect. To see this, we construct an example where the island has only one demand point. The busy fraction is kept fixed. In practice, when ambulances are added to the base location the busy fraction for each ambulance decreases as the workload gets shared between the ambulances. This does not happen in the binomial approach; instead we observe a strange effect. By keeping the busy fraction q fixed as the number of ambulances x increases, the binomial approach indirectly assumes that either the rate of the incoming calls λ or the mean service time β, or both, increase. This contradicts the fact that the number of incoming calls and mean service time are fixed input parameters. Especially at rural bases with low demand, such as one or two ambulances, adding an extra ambulance has a significant impact on the busy fraction of ambulances that are allocated at that base, which is not adequately incorporated in the binomial approach. This effect becomes smaller when more ambulances are allocated to the base locations. Therefore, the binomial models provide credible results when many vehicles are required per base location, that is, in highly populated areas.

When an ambulance region has both urban and rural areas the binomial approach may not be the best choice. In practice, in the case of a constant system-wide minimum reliability level, the busy fractions of ambulances positioned at a rural base location are much lower in contrast to ones at urban base locations. In the queuing approach this effect is incorporated into the Erlang B formulation. However, by design, the Q-models may not be applied to mixed regions because their demand may not significantly fluctuate. What happens the Q-models are applied, is discussed in detail in Section 3.4.

It is straightforward to change binomial models to their queuing counterpart, since they only differ in the way b_i is calculated ($i \in \mathcal{I}$).

We conclude that Erlang Blocking formulations are preferred over the binomial choice in minimal reliability and maximal availability models when all model assumptions can be satisfied.

3.4 Motivation

This section explains why the Q-models give over-estimations (min-rel) or undercoverage (max-av) when they are applied to mixed regions, and it turns these insights into a search direction for the solution that we follow in the remainder of this chapter.

It is generally agreed that the current minimal reliability and maximal availability models yield over-estimation and undercoverage, relatively, if they are applied to mixed regions. The effect is mentioned in [22] at multiple occasions, and computational comparisons with other facility location models that show the over-estimation can be found in [19, 20, 22] and computations the next chapter. We must note, however, that the Q-models are originally not designed to be used in mixed regions. This over-estimation provides motivation for the development of models that do not have this shortcoming.

Recall that Q-PLSCP (or MR-MA in general) consists of two phases: First we calculate the lower-bound for the required number of ambulances b_i ($i \in \mathcal{I}$), and second, we solve an IP-problem that provides the ambulance allocation. We show that the over-estimation is mainly caused in the first phase for regions with varying demand.

We illustrate the over-estimation for Q-PLSCP using two separate situations. In Subsection 3.4.1 we take a realistic situation and show why Q-PLSCP yields an over-estimation. Subsection 3.4.2 gives a theoretical example that demonstrates that the so-called *demand projection* effect can result in an over-estimation of any extent. The solution we propose is presented in the next two sections.

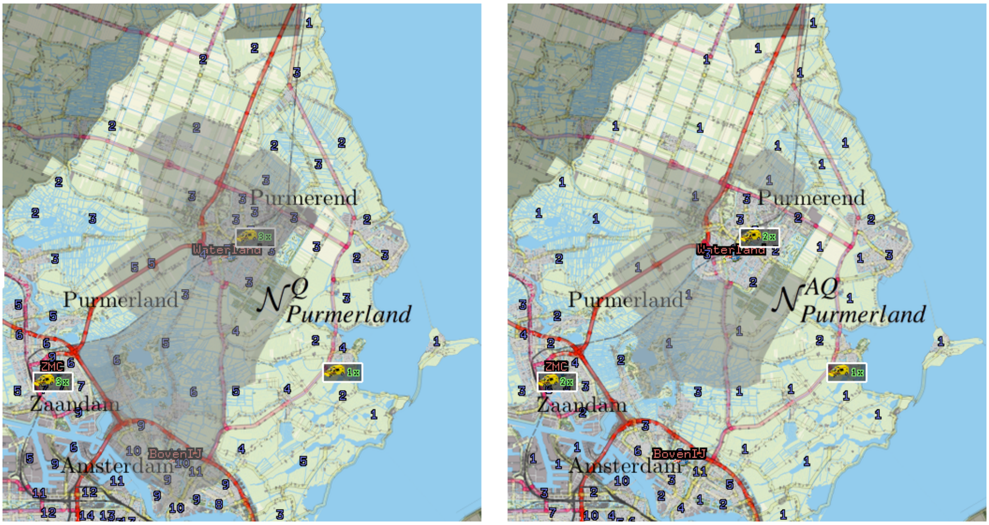


Figure 3.3 The required coverage b_i for each demand point $i \in \mathcal{I}$ *without* (left) and *with* (right) frequency adjustment is shown as numbers. The neighborhood $\mathcal{N}^{\bullet}_{Purmerland}$ is shaded for both methods.

3.4.1 Real life illustration

In this illustration, we highlight the small village of Purmerland with a population of approximately 500 people (just north of the city Amsterdam). Figure 3.3 illustrates that this approach results in too many ambulances that are required to cover most rural neighborhoods, up to three ambulances near Purmerland. After applying our proposed AQ-approach, we see that only one ambulance is sufficient for this village—a difference by a factor of three. The figure shows that this leads to an additional ambulance at base Purmerend.

In our calculation, we took the average demand at each demand point between 10:00 and 12:00 AM on working days measured over a period of five years, with $\alpha_i = 0.95$ and mean service time β_i calculated from the nearest actual base location to the nearest actual hospital with an emergency department for each demand point $i \in \mathcal{I}$. This mean service time approximation technique is explained in Section 4.4.

The source of the over-estimation on the number of ambulances can be found in the approximation of the arrival rates λ_i rather than in the mean service times β_i $(i \in \mathcal{I})$.

If the current queuing approaches are used for this region, $\mathcal{N}_{Purmerland}$ contains many demand points of the nearby cities of Amsterdam and Purmerend. Consequently, the arrival rate at Purmerland's neighborhood $\lambda_{Purmerland} = \sum_{k \in \mathcal{N}_{Purmerland}} f_k$ is not representative for the demand point Purmerland itself because the arrival rate of neighborhood Purmerland is dominated by urban demand, and so is the value $b_{Purmerland}$. Because Purmerend is the only base location that can provide coverage to Purmerland, the IP-formulation in the second phase of Q-PLSCP shows that base Purmerend gets $x_{Purmerend} \geq b_{Purmerland}$ ambulances allocated.

We have shown that base Purmerend provides coverage for non-existing demand that is projected from Amsterdam onto Purmerland. We call this effect *demand projection*. In general, we see that the same effect can occur for all rural base locations that have a driving time between response time threshold R and $2R$ from a large enough city's border.

The underlying cause is that the isolation assumption says that no major fluctuations in demand between the region's neighborhoods may occur when you apply Q-PLSCP. This is not realistic for ambulance regions in reality as demand fluctuations occur all-around. The 'faulty' use of the Q-models to the mixed region in this illustration provides insight what to do next. To make the Q-models applicable to mixed regions, the isolation assumption of the Q-models needs to be replaced by a mathematical structure that allows for major fluctuations in the arrival frequency. This structure is the *workload condition* that is introduced in Section 3.5. Next, using this workload condition, we can redefine the way the arrival rate of in neighborhood is calculated such that demand projection cannot occur; see Section 3.6.

3.4.2 Theoretical example

The theoretical example of the current section shows that demand projection, being the over-estimation effect of rural areas, can be extended to any magnitude.

To study this effect for Q-PLSCP, we consider a one-dimensional scenario with five demand points, of which the first three, A, B, and C, are in an urban area and the latter two, D and E, are rural. At the locations of central urban point B and outer rural point E are potential base locations; see Figure 3.4.

Our focus is on the arrival rates, and therefore we take the service time and minimum reliability level system-wide constants: $\beta = \beta_i$ and $\alpha = \alpha_i$ for all $i \in \mathcal{I} = \{A,B,C,D,E\}$. Hence, the minimum required staffing b_i through Erlang B only depends on the arrival rates λ_i, i.e., $b_i = b_i(\lambda_i)$ for all $i \in \mathcal{I}$. The scenario has driving times $r_{A,B} = r_{B,C} = r_{D,E} = R - PTD - \varepsilon$ and $t_{C,D} = 2\varepsilon$ for a small enough constant $\varepsilon > 0$, and an allowed response time threshold of $R > \varepsilon$ minutes.

The scenario is designed such that $\mathcal{M}_A = \mathcal{M}_B = \mathcal{M}_C = \{B\}$ and $\mathcal{M}_D = \mathcal{M}_E = \{E\}$.

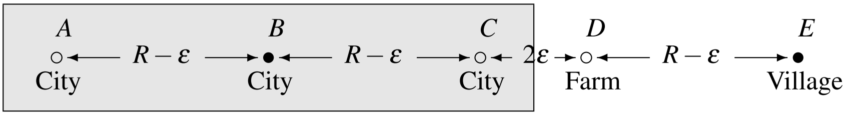


Figure 3.4 Our theoretical region contains urban and rural demand points. The filled points also have both demand and a base.

The substitution of $\lambda_i = \sum_{i' \in \mathcal{N}_i} f_{i'}$ yields:

$$\begin{aligned} b_A &= b_A(f_A + f_B), \\ b_B &= b_B(f_A + f_B + f_C), \\ b_C &= b_C(f_B + f_C + f_D), \\ b_D &= b_D(f_C + f_D + f_E), \text{ and} \\ b_E &= b_E(f_D + f_E). \end{aligned}$$

In a desirable situation, Q-PLSCP allocates ambulances such that demand of A, B, and C is covered by base B, and demand of D and E is served by the ambulances at base location E. Particularly, we do not want to staff base location E for any urban demand.

This fails if $f_A = f_B = f_C = M$, $f_D = \varepsilon'$, and $f_E = 0.1$ for small enough $\varepsilon' \in \mathbb{R}_{>0}$ and a large enough $M \in \mathbb{R}_{>>\varepsilon'}$. After all, using the constraints of the IP-formulation, we can approximate the number of ambulances assigned to base E by

$$\begin{aligned} x_E &= \max\{b_D(M + 0.1 + \varepsilon'), b_E(0.1 + \varepsilon')\} \\ &= b_D(M + 0.1 + \varepsilon') \\ &\approx b_D(M). \end{aligned}$$

However, we wish that our method returns a staffing of $b_E(0.1)$ for base location E. The *demand projection* of the urban demand point C onto demand point D results in an order of magnitude too high staffing at base location E, while demand point C is not even within reach of base location E. This is exactly the effect that causes major over-estimation in the Q-models for realistic regions, such as Purmerland.

The so-called *outer region* $U_j \subseteq \mathcal{I}$ of base location $j \in \mathcal{J}$ is the set of demand points that cannot be reached within response time threshold R from j, but that

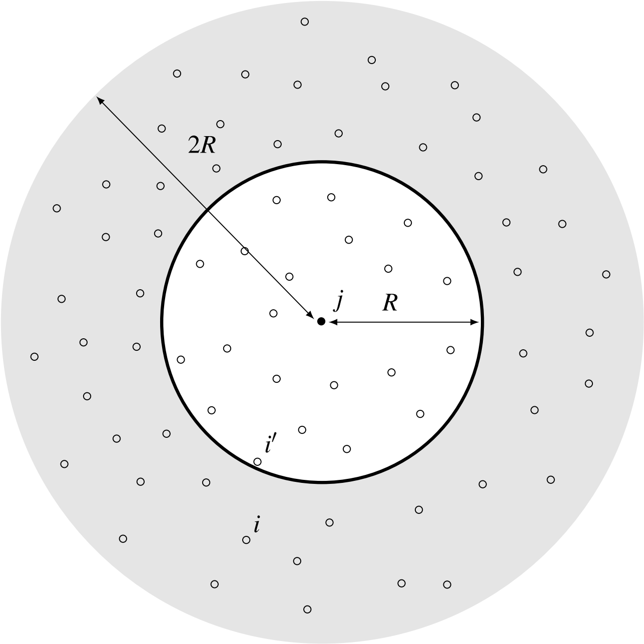


Figure 3.5 Demand point i is in the outer region $\mathcal{U}_j$ for base $j \in \mathcal{J}$, because there exists a demand point i' that can be reached from both i and j within R time units (and i itself cannot be reached from j). Likewise, outer region $\mathcal{U}_j$ consists of all demand points $(\circ)$ that are located in the gray area.

can be reached within R time units by a demand point i' that can be reached j within time threshold R, i.e.,

$$\mathcal{U}_j := \{i \in \mathcal{I} : \exists i' \in \mathcal{I}, r_{i',i} \leq R, r_{i',j} \leq R, r_{i,j} > R\}.$$

A good approximation of the outer region is the set of demand points that are contained in a torus with center j and a radius between R and $2R$; see Figure 3.5. In our example $U_B = \{D\}$ and $U_E = \{C\}$.

The number of ambulances allocated to a potential base location is only too high when the so-called density in the outer region of that base is significantly higher than any of the densities in the inner region $\mathcal{N}_i$ of that base. Various choices for the definition of the density of a demand point ψ_i can be made. For example: the density ψ_i of a demand point may be the population divided by the area that is mapped onto demand point i, or one may choose to define the density as the frequency f_i while assuming a constant area for each demand point. The example in this subsection uses the latter option, leading to $\psi_i = f_i$ for all $i \in \mathcal{I}$. In general we assume that demand points with a higher density also have a higher performance requirement: if $\psi_{i_1} \leq \psi_{i_2}$ then $\alpha_{i_1} \leq \alpha_{i_2}$ for all $i_1, i_2 \in \mathcal{I}$.

This can also be seen as follows: Fix all variables in our example except for the frequency at demand point D. Observe the staffing at base location B:

$$\begin{aligned} x_B &= \max(b_A(f_A + f_B),\, b_B(f_A + f_B + f_C),\, b_C(f_B + f_C + f_D)) \\ &= \max(b_A(2M),\, b_B(3M),\, b_C(2M + f_D)) \\ &= \max(b_B(3M),\, b_C(2M + f_D)). \end{aligned}$$

Only when f_D exceeds M it can have an unwanted influence by unnecessarily increasing b_B.

This has two consequences:

1. If the density of any demand point in $\mathcal{U}_j$ significantly exceeds the density of all demand points in $\mathcal{N}_j$, then base $j \in \mathcal{J}$ gets overstaffed through demand projection.

2. If all demand points in $\mathcal{N}_j$ have at least the density of all demand points in $\mathcal{U}_j$, there will be no overstaffing at base $j \in \mathcal{J}$.

3.4.3 Differentiating the minimum reliability level in the Q-models

It the Q-models it is not trivial to implement a differentiation of the minimum reliability level. That is, having a parameter α_i that significantly differs for bordering neighborhoods ($i \in \mathcal{I}$). The reason is that the isolation assumption of the Q-models loses its validity when one does.

Key element is the Erlang Blocking formula $\text{Erlang}(\lambda, \beta, c) \leq 1 - \alpha$ for an M/G/c/c-loss system where c represents the number of servers (ambulances). Recall that in these models λ, β, and α are input parameters, and the minimum value c for which the inequality is satisfied is the output parameter. For the isolation criteria to hold, the inflow and outflow over a neighborhood's border to adjacent neighborhoods must cancel out. Hence, both λ, β and α should be treated equally. If one of the three fluctuates while the other two are kept stable, it results in a non-zero flux over a neighborhood border. Thus α should also be stable across bordering neighborhoods for the isolation criteria to hold.

More in line with the literature, likewise to the call arrival rate, one could say in Q-PLSCP and Q-MALP that one allows for a demand point dependent minimum reliability level α_i, as long as α_i does not differ to a significant extent from the minimum reliability level of the neighborhoods that border i. Such a small difference, however, may not be manageable from an administrative point of view. In the literature we see that α does not vary over i [82, 83].

3.4.4 Concluding: necessary changes to the minimal reliability and maximal availability models for application to mixed regions

There are two challenges that must be addressed and solved to allow MR-MA models to be applied on ambulance regions with both urban and rural areas:

Challenge 1 The isolation assumption for a neighborhood should be generalized and made more explicit. In areas with both urban and rural demand the isolated neighborhood assumption is not realistic, in particular, the case when an urban located base location covers rural areas near the city that has no closer base locations, or when larger base locations are located near a neighborhood's border. A generalization of this condition is handled in Section 3.5.

Challenge 2 We need a new neighborhood definition such that demand projection cannot happen. The solution is discussed in Section 3.6.

It is shown in the literature that maximal availability models give urban areas a relatively high coverage at the cost of the rural areas [19]. Similar to our examples for min-rel models, it can be shown that this effect is also caused by the demand projection.

3.5 Workload condition

Section 3.4 showed for minimal reliability models that the demand projection effect can cause over-estimation on the required numbers of ambulances to any extent when Q-PLSCP is applied to a mixed region. This section discusses an approach that can replace the isolation assumption that was introduced in Assumption 3.1 on page 58. This poses a solution to Challenge 1. Another advantage of the workload condition is that it allows for a demand point dependent minimum reliability level requirement α_i instead of a system-wide constant.

From a theoretical point of view, it is not necessary that all ambulances at $\mathcal{M}_i$ must be able to serve demand at $i \in \mathcal{I}$: our guarantee is that by a given probability an ambulance within the driving time R is available for dispatch, and this is not necessarily the ambulance that can arrive at the incident the fastest. Recall that the concept of neighborhoods $\mathcal{N}_i$ is only used to get a notion of the workload if i would be the only base location within a response time radius of at most R.

Notation 3.1. We introduce the following notations ($i \in \mathcal{I}$):

a. Denote the set of ambulances in the system by $\mathcal{A}$.
b. Denote the set of ambulances that serve demand at i by $\mathcal{A}_i \subseteq \mathcal{A}$.
c. Denote the number of ambulances that serve demand at i by $n_i = |\mathcal{A}_i|$.
d. Denote the number of ambulances stationed at bases of $\mathcal{M}_i$ by $y_i = |\mathcal{A}_i|$.
e. Denote the number of ambulances that can reach i within response time R by x_i.

There is a relation between variables x_j en y_j in Q-PLSCP. Solving the IP-formulation yields values x_j $(j \in \mathcal{J})$, and equation $b_i \leq y_i = \sum_{j \in \mathcal{N}_i} x_j$ holds for all demand points $i \in \mathcal{I}$.

Note it is not necessary that each ambulance for which $t_{a,i} \leq R$ holds is an element of $\mathcal{A}_i$. This makes our adjustment also possible for max-avail.

3.5.1 Bounds on the busy fractions

In the remainder of the chapter the (offered) workload that is generated in a neighborhood, and the busy fraction of the ambulances play a central role.

Definition 3.1 (workload and Erlang blocking formula). We define the workload and Erlang B for the remainder of the chapter as follows.

a. Define for arrival rate λ and mean service time β the *workload*

$$\rho := \lambda\beta.$$

Similarly, the workload at demand point $i \in \mathcal{I}$ is denoted by $\rho_i := \lambda_i\beta_i$.

b. For the remainder of this chapter we redefine *Erlang B* as a function of ρ and b by substituting $\rho = \lambda\beta$, which is equivalent to Equation 3.1 on page 59:

$$\text{Erlang}_\text{B}(\rho, b) := \frac{\frac{\rho^b}{b!}}{\sum_{k=0}^{b} \frac{\rho^k}{k!}}$$

Similarly, we get $\text{Erlang}_\text{B}(\rho_i, b_i) := \frac{\rho^{b_i}}{b_i!} / \sum_{k=0}^{b_i} \frac{\rho^k}{k!}$ for $i \in \mathcal{I}$.

We define the bounds on the busy fraction of an ambulance, and in Section 3.5.2 we illustrate how they are used. A starred notation $(*)$ refers to the solution of the problem that we found, and hence depends on n_i; this is not necessarily a global optimum.

Definition 3.2 (bounds on the busy fraction). Consider an independent system with Poisson arrivals, workload ρ, and a fixed minimum reliability level α.

a. Define the *minimum required number of ambulances* at workload ρ by

$$b(\rho) := \operatorname{argmin}_{b' \in \mathbb{N}} \{\text{Erlang}_{\text{B}}(\rho, b') \leq 1 - \alpha\}.$$

For a neighborhood $\mathcal{N}_i$, $i \in \mathcal{I}$, we have a demand point dependent α_i and define it as

$$b_i(\rho_i) := \operatorname{argmin}_{b' \in \mathbb{N}} \{\text{Erlang}_{\text{B}}(\rho_i, b') \leq 1 - \alpha_i\}.$$

b. Define the *lower-bound on the busy fraction per ambulance* with $n > 0$ serving ambulances by

$$q^{low}(\rho, n) = \rho / n.$$

Furthermore, define $q^{low}(\rho) := \rho / b(\rho)$ and $q_i^{*,low}(\rho) := \rho / n_i$ for $i \in \mathcal{I}$ where we have $n_i = |\mathcal{A}_i|$ ambulances that serve demand point i. Denote $q_i^{low} := q^{low}(\rho_i)$, and $q_i^{*,low} := q^{*,low}(\rho_i, n_i)$ for $i \in \mathcal{I}$.

c. Define the *upper-bound on the busy fraction per ambulance* for a system with offered workload ρ by

$$q^{upp}(\rho) \quad := \quad \max_{\rho'}(\text{Erlang}_{\text{B}}(\rho', b(\rho)) \leq 1 - \alpha) / b(\rho).$$

Consequently, the *upper-bound on the busy fraction per ambulance for a neighborhood $\mathcal{N}_i, i \in \mathcal{I}$,* is the maximum on the busy fraction serving each ambulance in $\mathcal{N}_i$ such that any more workload leads to an additional ambulance:

$$\begin{aligned} q_i^{upp} &:= \max_{\rho'}(\text{Erlang}_{\text{B}}(\rho', b_i) \leq 1 - \alpha_i) / b_i, \\ q_i^{*,upp} &:= \max_{\rho'}(\text{Erlang}_{\text{B}}(\rho', n_i) \leq 1 - \alpha_i) / n_i. \end{aligned}$$

Denote the corresponding arguments by ρ_i^{upp} and $\rho_i^{*,upp}$, respectively.

The next section provides the intuition in words to the meaning of the lower and upper-bounds on the busy fraction.

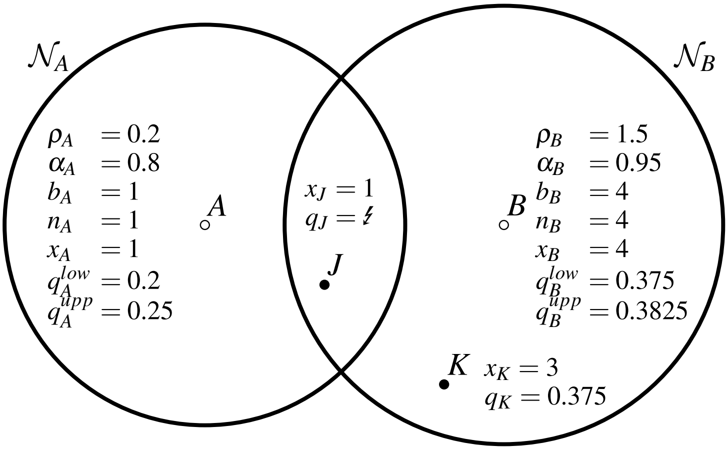


Figure 3.6 In this counterexample for a region where rural neighborhood $\mathcal{N}_A$ and urban $\mathcal{N}_B$ share one ambulance at J, the workload conditions of both neighborhoods cannot be fulfilled simultaneously.

3.5.2 The basic idea

The basic idea behind the workload condition consists of two insights.

1. Handling overcapacity
   We staff for an isolated neighborhood $i \in \mathcal{I}$ with workload ρ_i and minimum reliability level requirement α_i. This yields a minimum number of ambulances b_i required in the neighborhood $\mathcal{N}_i$. We do realize that due to b_i being an integer we have a slight overcapacity of ambulances in the neighborhood. This overcapacity can be used to serve outside neighborhood i.

2. Busy fraction per ambulance
   Neighborhoods share ambulances, while an ambulance $a \in A$ may only have one busy fraction at which it operates. Ambulances can only serve neighborhoods if these neighborhoods' have no conflicting constraints on the ambulance's busy fraction. We formalize this concept later in this section.

Clarification by a counterexample

We clarify the need for the various bounds on the busy fraction through a counterexample. In our counterexample, an assignment through Q-PLSCP with demand point dependent minimum reliability level requirements fails in the case that demand between adjacent neighborhoods differs significantly; see Figure 3.6.

Consider an ambulance region with two demand points A and B, with $\rho_A = 0.2$, $\rho_B = 1.5$, $\alpha_A = 0.8$, and $\alpha_B = 0.95$. We assume $\mathcal{N}_A$ to be an isolated region, that is using the isolation concept of Q-PLSCP where a neighborhoods where a neighborhoods receives as much help from bordering neighborhoods as it provides the other way around. Then, substituting these parameter values in Erlang B from Definition 3.2a yields $b_A = 1$. Because $\text{Erlang}_\text{B}(\rho = 0.25, b = 1) = 0.2$ we have $q_A^{low} = 0.2/1 = 0.2$ and $q_A^{upp} = 0.25/1 = 0.25$. Likewise, we have $b_B = 4$ and $\text{Erlang}_\text{B}(\rho = 1.53, b = 4) = 0.05$, hence $q_B^{low} = 1.5/4 = 0.375$ and $q_A^{upp} = 1.53/4 = 0.3825$. Figure 3.7 shows this behavior. If $y_i = n_i = b_i$ for all $i \in \mathcal{I}$, there is a slight overcapacity of workload $O_A = (0.25 - .2)1 = 0.05$ Erlang in place for $\mathcal{N}_A$, and $\mathcal{N}_B$ has an overcapacity of $O_B = (0.3825 - 0.375)4 = 0.03$ Erlang.

Let us now consider two base locations J and K, such that $\mathcal{M}_A = \{J\}$ and $\mathcal{M}_B = \{J, K\}$. An optimal solution to the IP of Q-PLSCP is $x_J = 1$ and $x_K = 3$. If we focus on the ambulance at J we see that neighborhood $\mathcal{N}_A$ says that its workload may not exceed $q_A^{upp} = 0.25$, because any extra workload yields the need of an additional ambulance at $\mathcal{M}_A$ to keep the guarantee that the minimum reliability level is at least $\alpha_A = 0.8$. On the other hand, neighborhood $\mathcal{N}_B$ requests from each of its ambulances to handle a workload of at least $q_B^{low} = 0.375$. If the busy fraction of an ambulance goes below this value, we cannot guarantee the minimum reliability level (for insight, see that, when n ambulances are put to work with busy fraction q^{low}, the maximum workload that they can serve with the minimum provided reliability level is $\rho = q_B^{low} n_B$, while the higher workload ρ_B is being offered). We see that the two neighborhoods have contradicting requirements for the workload of this ambulance. Hence, the ambulance can only satisfy the reliability requirement of one of these neighborhoods.

This illustrates why an extra condition on the workload for a valid ambulance allocation is required: we will call this condition the *workload condition.*

Intuition behind the various busy fractions

The counterexample shows that an ambulance allocation *cannot* guarantee the minimum reliability level α_i for demand point $i_1 \in \mathcal{I}$ if there is another demand point $i_2 \in \mathcal{I}$, such that:

1. either $q_{i_1}^{*,low} > q_{i_2}^{*,upp}$ or $q_{i_2}^{*,low} > q_{i_1}^{*,upp}$, and
2. demand points i_1 and i_2 share ambulances.

We adjust the queuing approach such that this cannot occur, which is a step forward in our quest to replace the isolation assumption by a more general structure.

From the negation of this statement we draw a hypothesis, that we later prove to be correct. Take any ambulance $a \in \mathcal{A}$ at random. If $q_{i_1}^{*,low} \leq q_{i_2}^{*,upp}$ holds for all combinations of demand points $i_1, i_2 \in \mathcal{I}$ that a serves, then we are able to guarantee the minimum reliability level α_i for demand point $i \in \mathcal{I}$.

This is only the case if there is a variable ${}^{a}q^{dum}$ such that $q_i^{*,low} \leq {}^{a}q^{dum} \leq q_i^{*,upp}$ holds for all demand points i that ambulance a serves. We call ${}^{a}q^{dum}$ the ambulance's dummy busy fraction. It can be easily shown that the dummy busy fraction is an upper-bound to the actual busy fraction that the ambulance gets using the allocation that follows from the solution. (To prove, add *only for ambulance a*, a minimum dummy demand of the type 'keep on waiting' to all demand points that a serves until all these demand points have a similar lower-bound on the busy fraction per ambulance.)

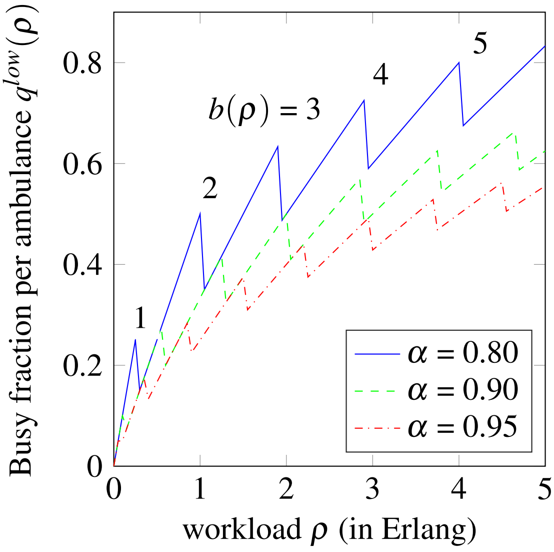


Figure 3.7 The lower-bound on the busy fraction per ambulance $q^{low}(\rho)$ is shown for a given workload $\rho = \lambda\beta$, such that the minimum reliability level α is met with a minimum number of ambulances b induced by Erlang B.

Definition 3.3 (dummy busy fraction). The *dummy busy fraction* of ambulance a, denoted by ${}^{a}q^{dum}$, is an approximation for the busy fraction of an ambulance, such that:

1. ambulance a can serve all demand offered by its assigned neighborhoods under the reliability constraints, when working at speed ${}^{a}q^{dum}$, and

2. it is an upper-bound on the busy fraction of a after allocation of the IP solution.

The overcapacity $O_i := (q_i^{*,upp} - q_i^{*,low})n_i$ may be used to serve calls outside $\mathcal{N}_i$.

3.5.3 Workload condition

This section formalizes the hypotheses from Section 3.5.2. In the AQ models we replace the isolation assumption of the Q-models by the *workload condition*.

Theorem 3.1 (workload condition). The minimum reliability level α_i is guaranteed for every demand point $i \in \mathcal{I}$ if there exists an assignment of dummy variable ${}^{a}q^{dum}$ for every $a \in \mathcal{A}$ and an assignment $\mathcal{A}_i \subseteq \mathcal{A}$ for every $i \in \mathcal{I}$ such that all these conditions hold:

$$q_i^{*,low} \leq {}^{a}q^{dum} \leq q_i^{*,upp}.$$

Proof. In the proof we need the following *property*: For each number of servers $s \in \mathbb{N}_{>0}$ and workload ρ, $\text{Erlang}_\text{B}(\rho, s)$ is continuous, strictly increasing in ρ, and surjective to the open interval $(0,1)$. This is a direct result of Theorem 1 of Jagers [62], and the notion that Erlang_B is a cumulative probability function.

Assume that there exists an assignment ${}^{a}q^{dum}$ for every $a \in \mathcal{A}$ and an assignment $\mathcal{A}_i \subseteq \mathcal{A}$ for every $i \in \mathcal{I}$ such that the condition holds. We need to show that the minimum reliability level α_i is guaranteed for every demand point $i \in \mathcal{I}$ by two steps: first, showing that the ambulances work hard enough that they can handle the workload at each demand point they serve and, second, other demand points outside the neighborhood $\mathcal{N}_i$ do not use an excess of capacity from $\mathcal{M}_i$.

a. Choose $i \in \mathcal{I}$ at random and keep it fixed. Take $q_i^{min,dum} = \min_{a' \in \mathcal{A}_i} {}^{a'}q^{dum}$ the minimum value of the assigned ambulances' dummy busy fractions. Because by assumption the workload condition holds, we have $q^{*,low} \leq {}^{a}q^{dum}$ for all $a \in \mathcal{A}_i$, thus $q^{*,low} \leq q_i^{min,dum}$. The workload at i can be served by its $n_i = |\mathcal{A}_i|$ serving ambulances since $\rho_i = q_i^{*,low} n_i \leq q_i^{min,dum} n_i \leq \sum_{a' \in \mathcal{A}_i} {}^{a'}q^{dum} =: \rho^{dum}$. In words, the workload that the n_i ambulances can handle within the minimum reliability level α_i (that is ρ^{dum}) exceeds the workload that the neighborhood $\mathcal{N}_i$ generates (that is ρ_i).

b. By assumption, the capacity of ambulances in $\mathcal{A}_i$ to serve demand points outside $\mathcal{N}_i$ is limited to $q_i^{*,upp}$ for all $a \in \mathcal{A}_i$. By Definition 3.2c,

$$\rho_i^{*,upp} := \operatorname{argmax}_{\rho'}(\text{Erlang}_{\text{B}}(\rho', n_i) \leq 1 - \alpha_i)/n_i$$

is the maximum workload that n_i ambulances can handle with a minimum reliability level of α_i. It corresponds to a maximum workload of $q_i^{*,upp}$. Because ${}^a q^{dum} \leq q_i^{*,upp}$ for all $a \in \mathcal{A}_i$ summation over all ambulances in $\mathcal{A}_i$ gives us $\rho_i \leq \rho_i^{*,upp}$. Combine this with the knowledge that $\text{Erlang}_{\text{B}}(\rho_i, n_i)$ is strictly increasing in the workload (see the *property*) and ${}^a q^{dum} n_i = \rho_i$ to conclude that if $\text{Erlang}_{\text{B}}(q_i^{*,upp} n_i, n_i) \leq 1 - \alpha_i$, then also $\text{Erlang}_{\text{B}}(\rho_i, n_i) \leq 1 - \alpha_i$ holds. Consequently, other demand points outside the neighborhood $\mathcal{N}_i$ do not use an excess of capacity from ambulances in $\mathcal{A}_i$. □

Hence, the workload condition provides inspiration for a new definition of coverage for a demand point. Using this definition, every demand point is covered if and only if the workload condition holds.

Definition 3.4 (coverage of a demand point). Given a set of ambulances $\mathcal{A}$, and a coupling $\mathcal{A}_i$ between a demand point and a subset of ambulances $(i \in \mathcal{I})$, such that every $a \in \mathcal{A}_i$ is stationed at any $j \in \mathcal{M}_i$ and has a fixed dummy busy fraction ${}^a q^{dum}$. We say that demand point i is *covered* if $q_i^{*,low} \leq {}^a q^{dum} \leq q_i^{*,upp}$ holds for all $a \in \mathcal{A}_i$.

The following theorem gives a relation between the workload condition and the isolation assumption.

Theorem 3.2 (generalization of isolation assumption). If the isolation assumption holds for an allocation through a Q-model, then the workload condition is satisfied.

Proof. The isolation assumption states that the demand over the neighborhood borders cancels out and that demand between neighborhood borders does not vary much over space. Because demand and mean service time does not differ much over space, and the minimum reliability level is the same for all demand points, this balance can only be upheld if the ambulances are evenly spread over the region. Because $\rho = \lambda\beta$, also the workload per demand point doesn't vary much over space. Hence, the upper-bound on the ambulance's busy fractions (q_i^{low} and q_i^{upp}) do not differ much in space. In fact, if the variations are small enough the upper-bound is equally valued for all demand points. Take the dummy busy factions of each ambulance $a \in \mathcal{A}_i$ serving in neighborhood $\mathcal{N}_i$ equal to the upper-bound on the busy fraction: ${}^{a}q^{dum} = q^{*,upp}$. Also, $q_i^{*,low} \leq q_i^{*,upp}$ if $n_i \geq b_i$, which is the case because of the Q-model's IP-formulation constraints. Combining these two statements yields $q_i^{*,low} \leq {}^{a}q^{dum} \leq q_i^{*,upp}$ for all $a \in \mathcal{A}_i$. The IP-formulation through a Q-model guarantees $b_i \leq n_i$. Since i is chosen at random it holds for all demand points. □

Theorem 3.2 shows that the workload condition is a generalization of the isolation assumption that Q-models use.

Corollary 3.1 (generalization of coverage). The demand point coverage in Definition 3.4 is a generalization for the definition of coverage found in the Q-models.

Proof. The Q-models say that an ambulance is covered if $y_i = n_i \geq b_i$. In Q-PSCLP the isolation assumption holds, Theorem 3.2 states that the workload condition is satisfied. Hence, every demand point is covered in Definition 3.4. Remark: for Q-MALP a prior step is required. If a demand point is not covered in Q-MALP, $n_i \leq b_i$, and thus it cannot be covered according to our definition. Remove all uncovered demand points from $\mathcal{I}$ so that we are left with the covered demand points. For these remaining demand points, we can use the same proof as for Q-PLSCP. □

The workload condition is a sufficient condition, not a necessary one. Using the actual average workloads of ambulances it may be possible to construct an example where the minimum reliability level is guaranteed for all demand points, while the workload condition is not satisfied. We leave the construction of a counterexample open for future research.

3.5.4 Satisfying the workload condition

Recall that we replace the isolation assumption of the queuing models by the workload condition, and drop the assumptions that the demand and minimum reliability levels between neighborhoods do not vary much over space. When the IP-formulation of a minimal reliability model is solved, it is not always the case that the workload condition holds for the resulting allocation.

Widening a demand point's acceptance gap

We show in the current paragraph for min-rel models that when enough ambulances are added we can fulfill the workload condition for any region. After applying the Q-model's IP to a possibly homogeneous region, the requirement on the various busy fractions is usually not fulfilled. We show that it is possible for any ambulance region that this 'workload condition' gets fulfilled when enough ambulances are strategically added to the base locations by a post-processor, on top of the allocation we already obtained from the IP. This lays the base for the proposed AQ-approach.

When n_i increases, $i \in \mathcal{I}$, both the lower and upper-bounds on the busy fractions for the ambulances in neighborhood $\mathcal{N}_i$ change: the lower-bound on the busy fraction *decreases* in n_i, while the maximum workload per ambulance *increases* in n_i. This gets formalized in the following proposition.

Proposition 3.1 (strictly increasing and decreasing busy fraction bounds). For $b \leq n$, the following two conditions hold for any $\rho \geq 0$ and any fixed $\alpha \in [0,1]$.

a. $q^{low}(\rho,n) \leq q^{low}(\rho,b)$. Equality holds if and only if $n = b$.

b. $q^{upp}(\rho,n) \geq q^{upp}(\rho,b)$. Equality holds if and only if $n = b$.

Proof. For this proof we need the following *statement*: For each $t \in \mathbb{R}$, number of servers $s \in \mathbb{N}_{>0}$ and workload $\rho > 0$, $\text{Erlang}_\text{B}(t\rho,ts)$ is strictly decreasing in t. A proof found by Burke is given in the appendix of [125].

a. $q_i^{low}(\rho,n) = \rho/n \leq \rho/b = q_i^{low}(\rho,b)$. Note that $\rho/n = \rho/b$ if and only if $n_i = b_i$.

b. That equality holds when $b = n$ is trivial. We consider the case $b < n$. The *property* in the proof of Theorem 3.1 implies that, for given s, there is a unique value of ρ for which $\text{Erlang}_\text{B}(\rho,s) = 1-\alpha$. This value of ρ is denoted by $\widehat{\rho}_s$. In this construct there is no overcapacity, hence we have $q^{upp}(\widehat{\rho}_s,s) = q^{low}(\widehat{\rho}_s,s) = \widehat{\rho}_s/s$ Also by definition, for $s \in \mathcal{N}_{>0}$, $\text{Erlang}_\text{B}(\widehat{\rho}_s,s) = 1-\alpha$. By substituting $t = \frac{s+1}{s}$ in Burke's statement, it follows that

$$\text{Erlang}_\text{B}\left(\frac{s+1}{s}\widehat{\rho}_s, s+1\right) < \text{Erlang}_\text{B}(\widehat{\rho}_s,s) = 1-\alpha.$$

Since $\text{Erlang}_\text{B}(\frac{s+1}{k}\widehat{\rho}_s,s+1) < 1-\alpha$, $\text{Erlang}_\text{B}(\widehat{\rho}_{s+1},s+1) = 1-\alpha$, and the *property* tells Erlang B increases in ρ, we know that $\widehat{\rho}_{s+1} > \frac{s+1}{s}\widehat{\rho}_s$. This directly leads to $\frac{\widehat{\rho}_{s+1}}{s+1} > \frac{\widehat{\rho}_s}{s}$. Induction shows that $q^{upp}(\widehat{\rho}_s,s) = \widehat{\rho}_s/s$ is increasing in s. The result follows directly. □

Proposition 3.1 shows that the gap between $q^{low}(\rho,n)$ and $q^{upp}(\rho,n)$ widens if n increases from two sides: both the lower-bound decreases and the upper-bound increases. For $q_i^{*,low}$ and $q_i^{*,upp}$ we call this the *acceptance gap*. As a result, once a neighborhood is covered, it stays covered when additional ambulances are included in $\mathcal{A}_i$.

As n increases, the lower-bound on the busy fraction for each ambulance goes to zero, and the upper-bound exceeds one. This is an important insight that is required to prove that the proposed AQ-approach always leads to a feasible solution for minimal reliability problems.

Proposition 3.2. For any $i \in \mathcal{I}$, $\rho_i > 0$, $\alpha_i \in [0,1)$ for $n_i \to \infty$, we have

a. $q_i^{*,low} \downarrow 0$, and

b. $q_i^{*,upp} \uparrow \frac{1}{1-\alpha_i} > 1$.

Proof. We address the two parts subsequently.

a. This part is a direct result of the fact that $\lim_{n_i \to \infty} q_i^{*,low} = \rho_i / n_i \downarrow 0$.

b. Let $n_i \to \infty$, while there is precisely enough work to keep all agents fully busy, the busy fraction of an agent approaches 1. Because the agent is allowed to miss a fraction α_i of its total offered load, we conclude that the maximum allowed workload that the agent receives equals $\frac{1}{1-\alpha_i}$. The inequality then follows directly from the assumption that $0 \le \alpha_i < 1$.□

Adjustment options

Theorem 3.1 shows a sufficient condition that proofs for an ambulance allocation that the reliability requirement is met for every demand point. An allocation from MR-MA models applied on a mixed region does not necessarily meet this condition. Proposition 3.2 shows that it is possible to meet the requirements by only adding ambulances on top of the solution provided.

Despite the fact that the workload condition is usually not fulfilled after solving the IP-formulation, there are multiple approaches possible to adjust the solution from the IP-formulation to satisfy this requirement on the various busy fractions:

1. Adjust the IP-formulation

   It is possible for small enough model instances to adjust the IP-formulation such that the workload condition always holds. This approach is followed in Section 4.2.

2. Alternative optimum
   Find an alternative optimum to the IP-formulation for which the workload condition holds. In the counterexample $x_J = 4$ and $x_K = 0$ would be such a solution with the same objective value $Z = 4$, while all constraints are still respected. This approach is especially relevant for max-av models.

3. Postprocessor: increase minimum required number of ambulances
   For min-rel, a post-processor on top of the existing IP-formulation is introduced to adjust the solution. In the case that a solution of a Q-model does not respect the workload condition, we demand a higher minimum coverage by stating $b_i \leftarrow b_i + 1$ for each neighborhood that cannot satisfy the condition, and solve the IP-formulation again.

4. Postprocessor: increase number of ambulances allocated to a base
   For min-rel, a post-processor on top of the existing IP-formulation is introduced to adjust the solution. Put extra ambulances at a base location in a neighborhood where the workload condition is not satisfied. This approach is followed in Section 4.3.

3.6 Demand aggregation

Section 3.4 illustrated the cause of over-estimation for the required number of ambulances in each neighborhood for realistic ambulance regions and showed that two challenges must be solved in order to resolve this. Section 3.5 proposed the workload condition as a generalization of the isolation assumption and provides a solution to Challange 1; see Theorem 3.2. This section proposes a solution to Challenge 2 that does not contain the demand projection effect: the *density-dependent demand aggregation*.

Our solution to the over-estimations lies primarily in the way the arrival rate λ_i is approximated for each demand point $i \in \mathcal{I}$. We propose an alternative calculation method. We also allow for a more precise way of approximation of the mean service time β_i; however, we leave the details for Chapter 4. The next section also contains two AQ-model variants of the Q-PLSCP that use the findings of this section as an input.

The AQ-approach uses another calculation methodology for the three Erlang B input parameters of demand point i's neighborhood $(i \in \mathcal{I})$: arrival rate λ_i, mean service time β_i, and minimum reliability level α_i. We discuss these input parameters one at a time.

The first difference is that the minimum reliability level α_i is a demand-point-based fixed input parameter instead of a system-wide constant. This is a trivial change, and needs no further elaboration.

The second difference is that the mean service time may be different for other demand points. From practice it is known that there are only small differences in mean service time between close by demand points, since differences in driving times are rather small compared to the total service time. It is also shown that models can be quite robust towards changes in the mean service time [66]. The mean service times can be calculated from the call record details. In this chapter we do not further elaborate.

The approximation of the arrival rate, the most important change, is discussed in the next section.

3.6.1 Arrival rate

In the AQ-approach we calculate λ_i differently than the Q-models, and (optionally) add a post-processing phase that guarantees a valid solution. Whether or not a post-processor is added in a min-rel model depends on the IP-formulation chosen. Directly after proposing our adjustments we provide theorems that prove that the proposed adjusted queuing approach guarantees the minimum reliability level at each demand point.

Method

The method consists of an initialization, solving an IP-formulation and finally (possibly) a post-processing phase that makes the solution satisfy the workload condition.

Initialization

To each demand point $i \in \mathcal{I}$ we assign a fixed density value $\psi_i \in \mathbb{R}$ that reflects an approximation of the fraction of total workload we can expect in neighborhood $\mathcal{N}_i$. Various choices for the density ψ_i can be made. One choice is to take $\psi_i^{FAQ} = f_i$ (FAQ-models), and another choice is to take the population per square meter for some system-wide constant $C > 0$ (DAQ-models):

$$\psi_i^{DAQ} = C \cdot \mathfrak{p}_{\mathcal{N}_i} / \mathfrak{a}_{\mathcal{N}_i}.$$

Here $\mathfrak{p}_{\mathcal{N}_i}$ denotes the number of inhabitants of neighborhood $\mathcal{N}_i$ (i.e., the population), and $\mathfrak{a}_{\mathcal{N}_i}$ denotes the size of the area of neighborhood $\mathcal{N}_i$ in square meters. The AQ-approach redefines the neighborhood $\mathcal{N}_i^{AQ}$ by only including demand points that have at most the same density value of demand point i ($i \in \mathcal{I}$):

$$\mathcal{N}_i^{AQ} := \{i' \in \mathcal{I} : t_{ii'} \leq R \text{ and } \psi_{i'} \leq \psi_i\}.$$

The arrival rate λ_i^{AQ} of neighborhood $\mathcal{N}_i^{AQ}$ is obtained by summation over its demand points of at most the same density:

$$\lambda_i^{AQ} := \sum_{k \in \mathcal{N}_i^{AQ}} f_k = \sum_{\substack{k \in \mathcal{N}_i^{Q} \\ \psi_k \leq \psi_i}} f_k \qquad (i \in \mathcal{I}).$$

Denote the neighborhood of the Q-models by $\mathcal{N}_i^Q$. Note that $\mathcal{N}_i^{AQ} \subseteq \mathcal{N}_i^Q$. The general concept of neighborhood is denoted by $\mathcal{N}_i$, and it will not be used in mathematical formulations from hereon because it has become ambiguous; instead we use $\mathcal{N}_i^Q$ or $\mathcal{N}_i^{AQ}$.

IP-formulation
Similar to the literature, Erlang B yields

$$b_i^{AQ} := \text{argmin}_{b' \in \mathbb{N}}(\text{Erlang}_\text{B}(\lambda^{AQ}, \beta^{AQ}, b') \leq 1 - \alpha_i),$$

which takes the places of b_i in the IP-formulation of the Q-models.

Post-processing
In the case of a minimal reliability problem, if the allocation from the IP-formulation does not satisfy the workload condition of Theorem 3.1, a post-processor can transform the allocation to a feasible solution. For example, through local search and putting additional ambulances at base locations. As discussed in Section 3.5.4 there are multiple options for a post-processor.

Correctness of adjusted queuing approach
Theorem 3.3 shows that the new adjusted neighborhood definition for demand aggregation may be used similarly to the ones in the Q-models.

Theorem 3.3 (correctness of the AQ-approach). The ambulance allocation using the AQ-approach's neighborhood definition guarantees minimum reliability level α_i for each demand point $i \in \mathcal{I}$ that is covered.

Proof. Consider an ambulance region and an ambulance allocation (also referred to as the *solution*) at this region $\{x_j, j \in \mathcal{J}\}$ that:

1. respects the workload condition, and
2. uses the adjusted queuing definition for the arrival rate λ_i^{AQ} $(i \in \mathcal{I})$.

We show that minimum reliability level α_i is guaranteed for each demand point i in this ambulance region (min-rel), or that the minimum reliability level is guaranteed for the demand points that are said to be covered (max-av). In particular, we show that it is allowed to omit demand points with a lower density in the calculation of the arrival rate, in the way that the AQ-approach does.

Only for maximal availability models, we start by discarding the demand points from $\mathcal{I}$ for which the solution does not provide coverage. This way, we have to proof for all other demand points that the remaining demand points receive the required coverage.

Consider the density measure $\psi = \{\psi_i : i \in \mathcal{I}\}$ on $\mathcal{I}$ that was used to generate the solution. Because there are a finite number of demand points, there is always a demand point with the largest density. Take a so-called *density ordering* $v : \mathcal{I} \mapsto \{0, 1, \ldots, |\mathcal{I}| - 1\}$, such that $v(i_1) \leq v(i_2)$ if $\psi_{i_1} \geq \psi_{i_2}$. In words, the demand point with the highest density value is mapped onto the 0, and the demand point with the lowest density value is mapped onto $|\mathcal{I}| - 1$. In the case of equally density valued demand points we choose an order of demand points and keep this order fixed, such that v is a bijective mapping. Denote the inverse of v by v^{-1}.

The mean service time β_i can be calculated, and the minimum reliability level α_i is given for each demand point $i \in \mathcal{I}$. Now, all input parameters that are required to calculate the minimum required number of ambulances b_i for each demand point $i \in \mathcal{I}$ are known. The calculation of the solution also provides values for ${}^{q}q^{dum}$. Consider all these values as known and fixed during the remainder of the proof.

Define the set of demand points that has at least density ψ_i by

$$\hat{\mathcal{I}}_i := \{i' \in \mathcal{I} \mid v_{i'} \leq v_i\} \text{ for all } i \in \mathcal{I}.$$

For the proof we use induction over the density ordering v.

Take in the base case $i = i_0 = v^{-1}(0)$ as the inductive variable; that is the demand point with the highest density. At i, we place a virtual base location to determine the arrival rate in its neighborhood, that is the arrival frequency over the demand points within the response time threshold: $\lambda_i^{AQ} := \sum_{k \in \mathcal{N}_i^{AQ}} f_k = \sum_{k \in \mathcal{N}_i^{Q},\ \psi_k \leq \psi_i} f_k$. Knowing also the mean service time β_i and minimum reliability level α, provides values for the minimum required number of ambulances b_i^{AQ}, $q_i^{low}(\rho_i, n_i)$ and $q_i^{upp}(\rho_i, n_i)$. By assumption ambulance allocation from the solution respects the workload condition for an allocation using arrival rate λ_i^{AQ}, hence there are at least b_i ambulances in $a \in \mathcal{A}_i$, such that $q_i^{low}(\rho_i, n_i) \leq {}^{q}q^{dum} \leq q_i^{upp}(\rho_i, n_i)$ for each of these ambulances. These ambulances work hard enough to guarantee with minimum reliability level α_i that ρ_i can be handled (the proof for this i equals the proof at Theorem 3.1a). Other remaining neighborhoods, that is neighborhoods $\mathcal{N}_{i'}$ with $v(i') > v(i)$ do not use an excess of capacity from the ambulances in $\mathcal{A}_i$ such that actually more ambulances are needed (the proof for this i equals the proof at Theorem 3.1b). Hence, reliability α_i is guaranteed for i.

The key insight is that demand points that are proven to be covered do not have to be taken into account in the coverage of the next demand points, because it is proven that ‘some process’ already insures that its minimum reliability level is met. Also, because the ambulances work hard enough to guarantee with minimum reliability level α, it is proven that demand point i does not require 'extensive service' from ambulance resources shared with demand points that remain to be proven covered. Hence, we do not have to consider demand point i in any further calculations.

Now take any other random demand point i. As induction hypothesis we assume that all other demand points than i in $\hat{\mathcal{I}}_i$ are proven to be covered. From hereon, we focus on the remaining demand points that did not receive coverage yet. Using the same arguments as in the base case, we can prove that i is covered.

Consequently, we can ignore i in all upcoming iteration steps. Induction over v gives the reliability guarantee for all demand points. □

Any density measure ψ works, though some will perform significantly better than others. This is the case because value λ_i depends on the set of density values $\{\psi_i, i \in \mathcal{I}\}$. Another choice for the density measure leads to other neighborhood definitions $\mathcal{N}_i$, and hence it can change arrival rate λ_i in neighborhood $\mathcal{N}_i$.

Illustration for the theoretical example

In Section 3.4.2 for the one-dimensional example of Figure 3.4.1 that, once a Q-model is applied to a mixed region, the demand projection can give a serious overstaffing for minimal reliability models. Using the same illustrative region, we show that the adjusted queuing models do not have this staffing.

Take $\alpha_A = \alpha_B = \alpha_C = 0.95$, $\alpha_D = \alpha_E = 0.80$, and $\beta_i = 1$ for all i. Recall that $f_A = f_B = f_C = M$, $f_D = \varepsilon'$ and $f_E = 0.1$ for large enough M and small enough $\varepsilon' > 0$. We take the frequency adjusted density function, that is $\psi = f_i$ yields

$$\begin{aligned}
b_A &= b_A(f_A + f_B) &&= b_A(2M),\\
b_B &= b_B(f_A + f_B + f_C) &&= (3M),\\
b_C &= b_C(f_B + f_C + f_D) &&= b_C(2M + \varepsilon'),\\
b_D &= b_D(f_D) &&= b_D(\varepsilon'), \text{ and}\\
b_E &= b_E(f_D + f_E) &&= b_E(0.1 + \varepsilon').
\end{aligned}$$

The allocation by the IP that follows is

$$\begin{aligned}
x_B &= \max\{b_A, b_B, b_C\} = b_B(3M), \text{ and}\\
x_E &= \max\{b_D, b_E\} = b_E(0.1 + \varepsilon').
\end{aligned}$$

This way, base location B gets staffed for the urban demand at A, B and C, and E gets staffed for rural demand at D and E. This allocation is exactly like the desired situation.

We check if the workload condition is fulfilled. First, we consider the three urban demand points. Take for example $M = 60$, i.e., every minute an incident that takes an ambulance on average an hour, then $b_B = 194$ and $b_A = b_C = 133$. Hence, $q_B^{low} = 0.928$, $q_A^{low} = q_C^{low} = 0.902$, $q_B^{low} = 0.936$, $q_A^{upp} = q_C^{upp} = 0.929$. Choose ${}^{a}q^{dum} = 0.929$ for all ambulances stationed at B, with $\mathcal{A}_A = \mathcal{A}_B = \mathcal{A}_C$ and the workload condition certainly holds for the urban subarea. For one ambulance and miniaml reliability level 0.80 stationed at the rural E, we can calculate that $q^{*,upp} = 0.25$. Because both D and E have a non-zero workload not exceeding the 0.25 ($= q^{*,upp}n$ with $n = 1$), and $q^{*,low} \leq q^{*,upp}$ if $n_i \geq b_i$, the workload condition gets fulfilled if we choose ${}^{a}q^{dum} = 0.25$ for the one ambulance at E that serves all demand at both D and E. Consequently, a 95% minimum reliability level for urban and an 80% minimum reliability level is guaranteed for the calculated allocation.

This shows that the adjusted queuing approach solved the problem of demand projection for the theoretical example.

The choice of density values: Frequency vs density adjusted queueing Choosing other orders than FAQ has two implications: (1) Because demand points with higher densities are included in more neighborhoods, they are counted multiple times. Hence, the mean demand over the area increases leading to a higher number of vehicles. Of course this is a valid bound to reach a minimum reliability level of α, but our proposed ordering yields a sharper bound. In this argument we used the symmetric travel times assumption. (2) The FAQ ordering guarantees that demand concentrates at the demand point of interest. Any other ordering is likely to lead to demand projection to some extent.

DAQ projects demand to some extent to regions where the population density is relatively high in relation to the historic incidents. This choice leads to more conservative solutions in the case of demographic changes due to an aging population.

3.6.2 Relation between Q-models and AQ-models

We conclude this chapter with statements about the relationship between the Q-models and AQ-models.

Theorem 3.4 (generalization of Q-models). The AQ-models are a generalization of the Q-methods.

Proof. Take $\psi_i = \psi$ and $\alpha_i = \alpha$ constant for all $i \in \mathcal{I}$ and assume the isolation assumption. Theorem 3.2 states that now the workload condition is satisfied. Because the density values are the same for every demand point $i \in \mathcal{I}$, we have equal neighborhood definitions for every demand point $\mathcal{N}_i^{AQ} = \mathcal{N}_i^{Q}$. □

Corollary 3.2 (correctness of the Q-models). The ambulance allocation using the queueing approach's neighborhood definition, the isolation assumption, and constant minimum reliability level α guarantees a minimum reliability level of at least α for each demand point $i \in \mathcal{I}$ that is covered.

Proof. Theorem 3.3 states that all AQ-models are correct. Theorem 3.4 states that the Q-models are a special case of the AQ-models. Hence, also the Q-methods are correct. □

Note that the proof of Corollary 3.2 does not rely on a cancellation argument like the Q-method papers, but instead uses the workload condition.

Proposition 3.3 shows that the minimum required number of ambulances for a demand point is an increasing function for the workload. The proposition is used in the theorem that follows.

Proposition 3.3 (more workload never implies fewer ambulances). Increased workload yields at least the same minimum required number of ambulances (for a constant minimum reliability level):

$$b(\rho_1) \leq b(\rho_2) \text{ if } \rho_1 \leq \rho_2.$$

Proof. Take $\rho_1 \leq \rho_2$, and minimum reliability level α fixed. By definition:

$$\begin{aligned} b(\rho_1) &= \operatorname{argmin}_{b'_1 \in \mathbb{N}}\{\text{Erlang}_\text{B}(\rho_1, b'_1) \leq 1-\alpha\}, \text{ and} \\ b(\rho_2) &= \operatorname{argmin}_{b'_2 \in \mathbb{N}}\{\text{Erlang}_\text{B}(\rho_2, b'_2) \leq 1-\alpha\}. \end{aligned}$$

Directly from the definition follows $\text{Erlang}_\text{B}(\rho_2, b(\rho_2)) \leq 1-\alpha$. Because $\rho_1 \leq \rho_2$ and Erlang B is increasing in ρ (see proof of Theorem 3.1), we have $\text{Erlang}_\text{B}(\rho_1, b) \leq \text{Erlang}_\text{B}(\rho_2, b)$ for b servers. Taking $b = b(\rho_2)$ yields

$$\text{Erlang}_\text{B}(\rho_1, b(\rho_2)) \leq \text{Erlang}_\text{B}(\rho_2, b(\rho_2)) \leq 1-\alpha.$$

By definition, $b(\rho_1)$ is the lowest value such that $\text{Erlang}_\text{B}(\rho_1, b(\rho_1)) \leq 1-\alpha$. Because the inequality holds for $b(\rho_2)$, and $b(\rho_1)$ is the lowest value for which it holds, our result follows immediately: $b(\rho_1) \leq b(\rho_2)$. □

The next theorem shows that, for minimal reliability problems, the number ambulances with the AQ-neighborhood definition will never exceed the number of ambulances using the Q-models' neighborhood definition.

Theorem 3.5 (**AQ never underperforms Q [min-rel]).** The required number of ambulances for a minimal reliability adjusted queuing model Z^{AQ} is at most the number of required ambulances Z^{Q}, i.e., $Z^{AQ} \leq Z^{Q}$, through Q-PLSCP for constant $\alpha_i = \alpha$ over all demand points $i \in \mathcal{I}$ if the solution of the IP-formulation respects the workload condition without the need of additional ambulances by a post-processor.

Proof. The arrival rate for each demand point in AQ-PLSCP is bounded by the arrival rate for the same demand point in Q-PLSCP:

$$\lambda_i^{AQ} = \sum_{k \in \mathcal{N}_i^{AQ}} f_k = \sum_{\substack{k \in \mathcal{N}_i^{Q} \\ \psi_k \leq \psi_i}} f_k \leq \sum_{k \in \mathcal{N}_i^{Q}} f_k = \lambda_i^{Q}.$$

For small enough variations in the service times this means $\rho_i^{AQ} \leq \rho_i^{Q}$. Proposition 3.3 then gives $b_i^{AQ} \leq b_i^{Q}$. Because the required number of vehicles on each demand point for AQ is at most the corresponding number for the queuing approach, it means that the IP-formulation yields for the number $Z^{AQ} \leq Z^{Q}$. Because the solution of the IP-formulation respects the workload condition, the post-processor does not have any effect on the outcomes. □

Similar, for maximal availability problems, we can show that the results of adjusted queuing are never worse when using the AQ-approach's neighborhood definition compared to the queueing approach's neighborhood definition.

Theorem 3.6 (**AQ never underperforms Q [max-av]).** The maximum covered demand for a maximal availability adjusted queuing model P^{AQ}, is at least the covered populated P^{Q} through Q-MALP for the same number of vehicles and constant $\alpha_i = \alpha$ over all demand points $i \in \mathcal{I}$, if the workload condition is satisfied for the solution of Q-MALP.

Proof. For the same arguments as in the proof of Theorem 3.5 we have $b_i^{AQ} \leq b_i^{Q}$. Introduce equal to MALP and Q-MALP [82, 109] the binary variable $\hat{y}_{i,k} = 1$ iff at least k ambulances serve i. Hence $\hat{y}_{i,k} \leq \hat{y}_{i,(k-1)}$ for $k = \{2, 3, \ldots b_i\}$. We show that the solution $\hat{y}_{i,k}$ of Q-MALP is also feasible for its adjusted queuing counterpart. If $\hat{y}_{i,b^Q} = 1$ then $\hat{y}_{i,b^{AQ}} = 1$, because $b_i^{AQ} \leq b_i^{Q}$ $(i \in \mathcal{I})$. For the objective function of (Q-)MALP we now have:

$$P^{Q} = \sum_{i \in \mathcal{I}} d_i \hat{y}_{i,b_i^Q} = \sum_{i \in \mathcal{I}} d_i \hat{y}_{i,b_i^{AQ}} = P^{AQ}.$$

By assumption, the workload condition is satisfied for Q-MALP. Then the workload condition is also satisfied for AQ-MALP (Theorem 3.4). Hence the solution of Q-MALP is a lower-bound on the solution of the adjusted queuing maximal availability problem, i.e., $P^{AQ} \geq P^{Q}$. This concludes the proof. □

The following property shows that the same solution is provided by AQ-PLSCP and Q-PLSCP if a region satisfies all requirements of the queuing models.

Property 3.1 (AQ equals Q if the isolation assumption holds). If a region satisfies all assumptions of a Q-model, then this Q-model and its frequency adjusted queuing counterpart give the same number of required vehicles (for min-rel) or the same demand covered (for max-av) if one chooses $\alpha_i = \alpha$ constant for all demand points $i \in \mathcal{I}$. Furthermore, if the solver has no random components, the facility location and allocation solutions are equal for both models. We assume that the same solver is used for both IP-formulations, and for min-rel that the post-processor stops if the workload condition can be met.

Proof. Take any region that respects all assumptions of the Q-models. Q-models use the isolation assumption that states that demand is evenly and well spread between adjacent neighborhoods. We can say that a constant $\beta = \beta_i$ corresponds to base locations and hospitals being evenly and well spread between adjacent neighborhoods. Because demand is evenly spread between adjacent neighborhoods, we see that $\varphi_i = f_i$ takes equal values, hence $\mathcal{N}_i^{FAQ} = \mathcal{N}_i^{Q}$. Thus λ_i is similar for all demand points i$\in \mathcal{I}$. Furthermore values β_i are similar in both the FAQ-models and the Q-models for all $i \in \mathcal{I}$. By assumption $\alpha_i = \alpha$ is constant, hence through identical computations b_i gets the same value for the queuing approach and frequency adjusted queuing approaches for all $i \in \mathcal{I}$. Now, note that b_i is similar valued for all neighborhoods. In other words, all neighborhoods get the same number of the minimum required ambulances. Consequently, the IP-formulations for the queuing and adjusted queuing approaches are identical, and consequently, also its solutions (we assume that the same solver is used and not an alternative solution is produced). Because the isolation assumption holds on the Q-model, also the workload condition holds for the solution for the Q-model (Theorem 3.2). Equal solutions yield that the workload condition holds for the solution of the AQ counterpart. In the case of a min-rel model with postprocessor, by assumption, this post-processor does not change the solution. We have shown that the queuing and adjusted queuing approaches give exactly the same facility location and allocation solutions. □

3.7 Conclusion

In this chapter we propose an adjusted queuing approach to the minimal reliability and maximal availability facility location and allocation model. To this end, we address and explain the cause of the over-estimation problem that existing queuing based MR-MA models have; e.g., a combination of so-called *demand projection* and conflicting neighborhood requirements on an ambulance's busy fraction. The adjusted queuing approach solves the later by generalizing the isolation assumption to the so-called *workload condition*. The former is solved by a newly proposed neighborhood definition that only includes demand points that are at least as busy as the considered neighborhood's defining demand point.

It is not necessarily hard to rewrite binomial and queuing MR-MA models into their adjusted queuing counterparts, as the *minimum required number of ambulances* b_i for demand point $i \in \mathcal{I}$ and neighborhood definitions $\mathcal{N}_i$ are usually core components of such a model, which can be calculated a priori.

Adjusted queuing MR-MA models allow for fluctuations in the arrival rate, mean service time and minimum reliability level, in contrast to the Q-models that do not. This means that urban subareas of an ambulance region can have another minimum reliability level than rural subareas.

The theoretical framework provided in the chapter provides proofs that the AQ-approach gives the guaranteed reliability for each covered demand point. We show that the adjusted queuing models lead to improved results for regions that have both urban and rural areas, both for the minimal reliability and the maximal availability problems.

We also showed that the Q-models are a special case of the AQ-models for regions in the case that an ambulance satisfies all assumptions of the Q-models: that is, homogeneous demand, service time and a constant reliability level for all demand points. For these regions the Q-models and frequency AQ-models give the same solution.

There are some interesting topics left open for future research.

A key performance indicator in practice is the fraction of late arrivals, which strongly depends on the travel time distribution. To this end, it would be interesting to extend the current deterministic travel time model toward stochastic travel times. Extending the current fixed travel time model toward models with stochastic travel times is an interesting topic for further research.

Whether an ambulance is on-time or late is a binary variable. An interesting extension is taking the probability that an ambulance is on-time. This can be done by taking the continuous counterpart of Erlang B, e.g., by allowing 2.5 ambulances to cover a demand point.

It is also interesting to research how we can incorporate the case when no available back-up EMS provider is available, and what effects this may have on the number of required ambulances. A good first approximation is to use Erlang C.

Another subject of interest is to research if the adjusted queuing models can be applied to a fleet with multiple vehicle types; see for example the FLEET model [120].

Our method aggregates at demand-point level. This can be extended by giving a minimal reliability or availability on a set of demand points, so that we could say that a municipality is covered with 80% certainty instead of a separate constraint for every single demand point in the municipality.

Models with a binomial approach also have a queuing approach equivalent. Hence, they have an adjusted queuing approach. Because the isolation assumption limitation has been overcome in this chapter, this opens up an entire new family of models, for example AQ-REL-P and AQ-MALP. Using these insights it might be even possible to find the adjusted queuing counterpart of MEXCLP where the AQ definitions of coverage and busy fraction are being used; this loosens the hard assumption of a system-wide busy fraction.

Chapter 4 uses the theory of the current chapter to develop two adjusted queuing models for the minimal reliability problem.

4

MINIMAL RELIABILITY ADJUSTED QUEUING MODELS

The *adjusted queuing probability location set coverage problem* discussed in Chapter 3 provides a solution for the overstaffing of existing MR-MA models on mixed regions by first providing a set of sufficient conditions that guarantee the minimum reliability level at each demand location. These conditions, better known as the workload condition, contain a mutual dependency between the ambulance busy fractions and the allocation that make the problem hard to solve. The current chapter proposes two models that provide a solution to the minimal reliability problem. The first is a mixed integer linear program formulation that does directly provide the optimal solution to the minimal reliability problem, which can be applied to small enough model instances. The second model is a heuristic that uses a basic postprocessor that can handle larger model instances. Numerical results show significant improvements as regards the number of required ambulances for four actual ambulance regions.

This chapter is based on the following publications:

[A3] M. van Buuren, R. D. van der Mei, and S. Bhulai. "Demand-point Constrained EMS Vehicle Allocation Problems for Regions with Both Urban and Rural Areas". *To appear in Operations Research for Health Care* (2018)

[A4] M. van Buuren, R. D. van der Mei, and S. Bhulai. "Ambulance Allocation in a Mixed Region with Guaranteed Performance at Every Urban and Rural Area". *Submitted* (2018)

4.1 Introduction

Currently, in rural areas late ambulance arrivals are a hot topic. In practice EMS providers frequently have to explain bad local performance to mayors of rural municipalities. In this chapter, we provide two minimal reliability models to address these concerns. Recall from the previous chapter that in minimal reliability models, instead of a system-wide busy fraction, a minimum performance constraint is posed on every demand point.

However, minimal reliability models, such as the *Probability Location Set Coverage Problem* (PLSCP) [108] and the *Queuing Probability Location Set Problem* (Q-PLSCP) [83] give substantial over-estimations when they are applied to regions with both urban and rural demand [20, 22]. These methods are designed to be used for homogeneous demand and service times, and a constant minimum reliability level. Recently, the cause for the over-estimation in mixed regions had been found and a solution was provided by first calculating the per demand point arrival rate differently, and second by the so-called *workload condition* (Theorem 3.1). The latter provides sufficient conditions on the relation between the ambulances busy fraction and the allocation for a minimum reliability level requirement to hold.

The workload condition, however, has a mutual dependency between the ambulance allocation and the ambulances dummy busy fractions, which complicates the development of an adjusted queuing MR-MA model: the ambulance allocation x_j and $\mathcal{A}_i$ is used to calculate values n_i, and consequently values $q_i^{*,low}$ and $q_i^{*,upp}$ $(i \in \mathcal{I}, j \in \mathcal{J})$; see Definition 3.2bc. The other way around, an ambulance may only be assigned to serve demand point i if its dummy busy fraction is between $q_i^{*,low}$ and $q_i^{*,upp}$, hence the dependency of $\mathcal{A}_i$ and x_j on the variables $q_i^{*,low}$ and $q_i^{*,upp}$. Section 3.5.4 provides four approaches that solve this issue.

The main contribution of this chapter is that it presents two models for the minimal reliability problem.

The first model is a mixed integer program that calculates an exact solution for this optimization problem; the so-called *Adjusted Queuing Mixed Integer Probability Set Coverage Location Problem* (AQ-MIPSCP).

The second model is a heuristic that iterative updates the busy fractions and the ambulance allocation, and stops when this workload condition is met. This model is called the *Adjusted Queuing Heuristic to the Probability Set Location Problem* (AQ-HPLSCP). The AQ-HPLSCP can be applied to larger model instances, and provides a lower bound on the required number of ambulances. This solution leads to a lower number of required ambulances than Q-PLSCP.

For both models, the model assumptions known from the previous chapter are in effect; see Section 3.3.2. For each demand point, the arrival frequency f_i, mean service time β_i and minimum reliability level α_i can be calculated or are given ($i \in \mathcal{I}$). The PTD, TMT, TFT are given, and driving times between every two points can be calculated (see page 55). Each demand point can be reached from at least one potential base location within the given response time threshold R, and the ambulance region has at least one hospital.

Both models are open to any choice for the density value that is used in the neighborhood definition. Taking $\psi_i = f_i$ the arrival frequency leads to the FAQ-MIPLSCP and FAQ-HPLSCP, while taking $\psi_i = d_i$ the population density leads to DAQ-MIPLSCP and DAQ-HPLSCP ($i \in \mathcal{I}$).

The remainder of this chapter is as follows. In Section 4.2 we propose the AQ-MIPLSCP model, and Section 4.3 introduces the AQ-HPLSCP. In Section 4.4 we describe our input data, which Section 4.5 uses to provide the results for four actual ambulance regions. A conclusion and proposals for future research are given in Section 4.6.

4.2 Model I: Mixed integer program

In this section we formulate a mixed integer program, the so-called *Adjusted Queuing Mixed Integer Probability Location Set Coverage Problem* (AQ-MIPLSCP), that fulfills the workload condition with a minimum number of ambulances.

After providing the model outline in Section 4.2.1, we discuss the preprocessor that calculates helper variables in Section 4.2.2. Sections 4.2.3 and 4.2.4 explain the objective function and the constraints in detail.

4.2.1 Model outline

Denote the set of ambulances $\mathcal{A}$. For the binary variable $\bar{y}_{a,i}$ we have $\bar{y}_{a,i} = 1$ if and only if ambulance $a \in \mathcal{A}$ provides coverage to demand point $i \in \mathcal{I}$. The allocation of ambulance a to base location $j \in \mathcal{J}$ is denoted by the binary variable $\bar{x}_{a,j} = 1$, and it is zero if it is not allocated to the base. We start with a large enough set of ambulances that may get relocated, and we try to allocate a minimum number of ambulances, i.e., we do allow ambulances to remain unallocated. In particular, we denote the binary variable $g_a = 1$ if and only if ambulance a is in use. We note that this is an uncommon approach for minimal reliability problems.

Take the upper-bound $M = \sum_{i \in V} b_i^{AQ}$ on the number of ambulances in the system. Other techniques may provide a lower value for M. Lower values for M make the solver run faster. Notice that, depending on the value M, many ambulances can be kept unallocated when an optimal solution is found.

$$
\begin{array}{llcll}
\min & Z & = & \sum\limits_{a\in\mathcal{A}} g_a - \left({}^a q^{dum}/(M+1)\right) & \\
s.t. & \sum\limits_{a\in\mathcal{A}} \bar{y}_{a,i} & \geq & k_i & i\in\mathcal{I} \\
 & \sum\limits_{j\in\mathcal{M}_i} \bar{x}_{a,j} & \geq & \bar{y}_{a,i} & a\in\mathcal{A}, i\in\mathcal{I} \\
 & \sum\limits_{j\in\mathcal{J}} \bar{x}_{a,j} & \leq & 1 & a\in\mathcal{A} \\
 & \sum\limits_{i\in\mathcal{I}} \bar{y}_{a,i} & \leq & |\mathcal{I}|\cdot g_a & a\in\mathcal{A} \\
 & \sum\limits_{k'=1}^{M} \bar{z}_{i,k'}k' & \leq & k_i & i\in\mathcal{I} \\
 & \sum\limits_{k'=1}^{M} \bar{z}_{i,k'} & \geq & 1 & i\in\mathcal{I} \\
 & \sum\limits_{k=1}^{M} \bar{z}_{i,k} q_{i,k}^{*,low} - (1-\bar{y}_{a,i}) & \leq & {}^a q^{dum} & a\in\mathcal{A},\ i\in\mathcal{I} \\
 & \sum\limits_{k=1}^{M} \bar{z}_{i,k} q_{i,k}^{*,upp} + (1-\bar{y}_{a,i}) & \geq & {}^a q^{dum} & a\in\mathcal{A},\ i\in\mathcal{I} \\
 & \sum\limits_{a\in\mathcal{A}} \bar{x}_{a,j} & \leq & C_j & j\in\mathcal{J} \\
 & \bar{x}_{a,j},\ \bar{y}_{a,i},\ \bar{z}_{i,k},\ k_i,\ g_a & \in & \{0,1\} & 1\leq k\leq C_j,\ i\in\mathcal{I},\ j\in\mathcal{J},\ a\in\mathcal{A} \\
 & {}^a q^{dum} & \in & [0,1] & a\in\mathcal{A}
\end{array}
$$

Mathematical Program 4.1 AQ-MIPLSCP

Recall from the workload condition that the minimum reliability level α_i is guaranteed for every demand point $i\in\mathcal{I}$ if there exists an assignment ${}^a q^{dum}$ for every $a\in\mathcal{A}$ and an assignment $\mathcal{A}_i\subseteq\mathcal{A}$ for every $i\in I$ such that the following condition holds:

$$q_i^{*,low} \leq {}^a q^{dum} \leq q_i^{*,upp}.$$

The following variables are introduced to formulate the workload condition.

Denote the lower and upper-bound on the busy fraction per ambulance for a neighborhood $\mathcal{N}_i$, $i \in V$, with $1 \leq k \leq M$ serving ambulances, by

$$\begin{aligned} q_{i,k}^{*,low} &= \rho_i/k \text{ and} \\ q_{i,k}^{*,upp} &= \text{argmax}_{\rho'}(\text{Erlang}_{\text{B}}(\rho',k) \leq 1-\alpha_i)/k, \end{aligned}$$

respectively. The last variable of the workload condition, the dummy busy fraction ${}^{a}q^{dum} \in [0,1]$ for each ambulance a, gets a value assigned by the mixed integer problem later on.

We assume that $\rho_i > 0$ for every demand point, hence $b_i > 0$. This can be achieved by simply omitting all demand points that generate no demand, before solving the mixed integer program.

The proposed mixed integer program provides a solution to the ambulance allocation problem that respects the workload condition with a minimum number of ambulances. The objective function and each constraint are addressed thoroughly in the remainder of this section.

4.2.2 Preprocessing

Calculate λ_i, β_i, ρ_i and b_i for each demand point $i \in \mathcal{I}$. In the calculation of the arrival rate we use the adjusted queuing neighborhood definition from Section 3.6.1: $\lambda = \lambda^{AQ}$.

Hence, in the pre-processing stage for every $k \in \{1,\ldots,M\}$ we can calculate the bounds on the busy fractions $q_{i,k}^{*,low}$ and $q_{i,k}^{*,upp}$. This means that the lower and the upper-bounds on the dummy busy fractions of all ambulances that serve demand point i when exactly k ambulances are assigned to provide coverage to this demand point.

The heuristic AQ-HPLSCP, which follows in Section 4.3, can be used to calculate good values for M and β_i.

4.2.3 Objective function

The objective is to minimize the number of ambulances in the system

$$Z = \min \sum_{a \in \mathcal{A}} g_a - ({}^{a}q^{dum}/(M+1)).$$

Within the solution space of number of ambulances we prefer a solution where the busy fractions of ambulances become realistic and low, i.e., we minimize the dummy workload. Hence, we include $\sum_{a\in\mathcal{A}}({}^{a}q^{dum}/(M+1))$. Notice that ${}^{a}q^{dum} \in [0,1]$ and $|\mathcal{A}| = M$; hence $0 \leq \sum_{a\in\mathcal{A}}({}^{a}q^{dum}/(M+1)) < 1$. In words, this results in a higher contribution to the objective function when an allocation can be done with an ambulance fewer, no matter what the dummy busy fraction of the ambulances is.

4.2.4 Constraints

The constraints are placed to ensure that ambulances provide coverage to their assigned areas, and the workload condition is respected, amongst others.

Ambulances providing coverage to each demand point

Denote the number of ambulances that are assigned to provide coverage to i by k_i, such that $\sum_{a\in\mathcal{A}}\bar{y}_{a,i} = k_i$, for all $i \in \mathcal{I}$.

Restrict by ambulances in the neighborhood

Only ambulances that can reach the patient in time may provide health care. Hence $\bar{y}_{a,i} = 0$ if $\sum_{j\in\mathcal{M}_i}\bar{x}_{a,j} = 0$. The constraints do not force $\bar{y}_{a,i} = 1$ if a provides coverage to i, but they leave room to allow for it: $\sum_{j\in\mathcal{M}_i}\bar{x}_{a,j} \geq \bar{y}_{a,i}$ for all $a \in \mathcal{A}, i \in \mathcal{I}$.

Unique allocation to a base location

An ambulance may be positioned on at most one base location. Hence, we get the constraints $\sum_{j\in\mathcal{J}}\bar{x}_{a,j} \leq 1$ for all $a \in \mathcal{A}$.

Notation of ambulances in use

The binary value g_a equals 1 if and only if ambulance a is used in the allocation. Hence, it must equal 1 if for any i, $\bar{y}_{a,i} = 1$ holds. We can realize that behavior through the constraints $\sum_{i\in\mathcal{I}}\bar{y}_{a,i} \leq |\mathcal{I}| \cdot g_a$ for all $a \in \mathcal{A}$. The objective value sets $g_a = 0$ when ambulance a is not providing coverage to any demand point.

Workload condition

We require $q_{i,k}^{*,low} \leq {}^{a}q^{dum} \leq q_{i,k}^{*,upp}$ to hold if there are k_i ambulances assigned to $i \in \mathcal{I}$, but only in the case that ambulance $a \in \mathcal{A}$ provides coverage to i. For all other values k_i, or when a does not provide coverage to i, this constraint is not required. We use two steps to meet this condition. First, we turn k_i into an array of binary variables $\bar{z}_{i,k'}$, such that only $\bar{z}_{i,k'} = 1$ if and only if $k' = k_i$. This can be obtained through

$$\begin{aligned}
\sum_{k'=1}^{M} \bar{z}_{i,k'}k' &= k_i && (i \in \mathcal{I}),\\
\sum_{k'=1}^{M} \bar{z}_{i,k'} &= 1 && (i \in \mathcal{I}).
\end{aligned}$$

The workload condition is rephrased through the following two sets of constraints:

$$\begin{aligned} \sum_{k=1}^{M} \bar{z}_{i,k} q_{i,k}^{*,low} - (1-\bar{y}_{a,i}) &\leq {}^{a}q^{dum} \qquad && (a \in \mathcal{A},\ i \in \mathcal{I}), \\ \sum_{k=1}^{M} \bar{z}_{i,k} q_{i,k}^{*,upp} + (1-\bar{y}_{a,i}) &\geq {}^{a}q^{dum} \qquad && (a \in \mathcal{A},\ i \in \mathcal{I}). \end{aligned}$$

Here we used the property that ${}^{a}q^{dum}$ takes values 0 and 1; hence the addition and subtraction of the term $(1-\bar{y}_{a,i})$ satisfy the latter two equations for any ambulance effective busy fraction if a does not provide coverage to i, i.e., the term $(1-\bar{y}_{a,i}) = 1$ (or equivalently $\bar{y}_{a,i} = 0$).

Base capacity
We can limit the maximum allowed number of ambulances $C_j \in \mathbb{N}$ assigned to base $j \in \mathcal{J}$ through $\sum_{a \in \mathcal{A}} \bar{x}_{a,j} \leq C_j$, for all $j \in \mathcal{J}$.

For computational ease the constraints are relaxed. Mathematical Program 4.1 summarizes the objective function and all relaxed constraints.

4.3 Model II: Heuristic

This section provides a model description for the AQ-HPLSCP.

The heuristic consists of applying two transformations to Q-PLSCP. First, we replace the neighborhood definition $\mathcal{N}_i^Q$ by the adjusted queuing version $\mathcal{N}_i^{AQ}$ $(i \in \mathcal{I})$. Second, we make sure that the workload condition is satisfied by applying a post-processor. We leave the IP-formulation of Q-PLSCP intact.

Section 4.3.1 provides the preprocessing of helper variables. Subsequently, Section 4.3.2 gives the model formulation.

4.3.1 Preprocessing

If a patient at demand point $i \in \mathcal{I}$ is served by an ambulance that departs from base location $j \in \mathcal{J}$ and brought to hospital $h \in \mathcal{H}$, the mean service time for a call, $\bar{\beta}_{h,i,j}$, is defined by

$$\bar{\beta}_{h,i,j} := r_{i,j} + t_{h,i} + t_{h,j} + TMT + TFT - CTT + addon_{h,i,j}.$$

To accommodate a demand point dependent mean service time β_i we make use of a weighted summation over variables $\bar{\beta}_{h,i,j}$. A full explanation of the calculation methodology follows in Section 4.4.1.

We explain how and why the value for $addon_{h,i,j}$ is calculated. When a patient is transported from a hospital, and there is a base location in the same postal area, we have $t_{j,i} = 0$ seconds. This is not realistic, because often the ambulance station is positioned in a separate building. For example, we can use a constant of five minutes driving time between these buildings. This includes the opening and closing of the garage doors. If we assume that a patient is brought to the nearest hospital and in the case an incident happens in a hospital, the nearest hospital would be the hospital itself, leading to a DTH of 0 seconds (see page 55). Because calls that originate in a hospital often concern a patient brought home or taken to another hospital, this is not realistic. From the call record details database we can calculate an average DTH for every demand point that contains a hospital and use that as $addon_{h,i,j}$.

4.3.2 Model formulation

The *adjusted queuing heuristic probability location set coverage problem* (AQ-HPLSCP) uses an iterative process to obtain a realistic approximation of the mean service time at each demand point. The heuristic model formulation is as follows.

Initialization

1. Calculate values $\lambda_i = \sum_{\substack{i' \in \mathcal{N}_i \\ \psi_{i'} \geq \psi_i}} f_{i'}$ for all $i \in \mathcal{I}$.

2. Calculate values $\bar{\beta}_{h,i,j}$ for all combinations of $h \in \mathcal{H}$, $i \in \mathcal{I}$, and $j \in \mathcal{J}$.

3. Set $x_j = 1000$ (or higher bound if required) for exactly one randomly chosen base location $j \in \mathcal{J}$. It also works for other choices of allocations with a huge upper-bound for the number of ambulances.

Iterations

1. Calculate the values of β_i for all $i \in \mathcal{I}$:

$$\beta_i = \max_{j \in \mathcal{M}_i} \left(\frac{\sum_{i' \in \mathcal{N}_j} d_{i'} \min_{h \in \mathcal{H}, j' \in \mathcal{J}, x_{j'} > 0} \overline{\beta}_{h,i',j'}}{\sum_{i \in N_j} d_i} \right).$$

   A full explanation is provided in Section 4.4.1.

2. Calculate b_i from Erlang for each $i \in \mathcal{I}$:

$$b_i \quad = \quad \operatorname{argmin}_{b' \in \mathbb{N}_{\geq 0}} \{1 - \mathrm{Erlang_B}(\lambda_i, \beta_i, b') \geq \alpha\}. \qquad (4.1)$$

3. Allocate x_j ambulances at base $j \in \mathcal{J}$ by solving the IP-formulation,

$$\begin{aligned} \min Z = & \sum_{j \in \mathcal{J}} x_j \\ \text{s.t.} & \sum_{j \in \mathcal{M}_i} x_j \geq b_i & (i \in \mathcal{I}), \\ & x_j \in \mathbb{N}_{\geq 0}. \end{aligned}$$

4. Run the post-processor that follows below on the allocation.

Stop condition

Stop when the number of vehicles $\sum_{j \in \mathcal{J}} x_j$ stays the same for two successive iterations. Although a rigorous proof is missing, in practice we noticed that this algorithm converges to its accumulation point within a few iterations.

Post-processor

The AQ-HPLSCP uses a basic post-processor; see Algorithm 1. This post-processor assumes that every used potential ambulance within the response time threshold must be able to respond to a call. It also assumes that the dummy busy fraction of ambulances that are stationed at the same base location is always the same. It prefers to add ambulances to base locations where it has the most added value.

It is not hard to see that the basic post-processor satisfies the workload condition: take ${}^{a}q^{dum} = q_j^{*,upp}$ if a is stationed at j and note that Line 13 in combination with the stop condition $w = \mathit{undefined}$ guarantees $q_i^{*,low} \leq q_j^{*,upp} (\leq q_i^{*,upp})$ for all $i \in \mathcal{I}$. Note that $n_i = y_i$ holds.

A property of this post-processor is that either all ambulances of a base location cover a demand point, or none does.

4.4 Input

In this section we compare results from the Q-PLSCP, AQ-MIPLSCP, and AQ-HPLSCP models. The same input data set is used for all models, that originates from actual call record databases from four actual ambulance regions. For the adjusted queuing models we consider both call arrival frequency adjustment (FAQ) and population density adjustment (DAQ), i.e., $\psi_i = f_i$ and $\psi_i = d_i$, respectively $(i \in \mathcal{I})$.

4.4.1 Parameter estimations

The calculation of the call arrival rate is straightforward. In this section we discuss an iterative method to get good estimates for the mean service time per demand point, and we explain our choice for the demand point dependent

Algorithm 1 Basic post-processor for minimal reliability models

```
repeat
    for all $i \in \mathcal{I}$ do
        update $q_i^{*,low}$ and $q_i^{*,upp}$
    end for
    for all $j \in \mathcal{J}$ do
        $q_j^{*,upp} \leftarrow \min\{q_i^{*,upp} : j \in \mathcal{M}_i$ and $i \in \mathcal{I}\}$
    end for
    for all $j \in \mathcal{J}$ do
        $m_j \leftarrow 0$                    ▷ uncovered demand $m_j$ through workload condition
        $m^{*,upp} \leftarrow 0$              ▷ $m^{*,upp}$ holds the maximum uncovered demand
        $w \leftarrow$ undefined       ▷ base location $w$ has the highest uncovered workload
        for all $i \in \mathcal{I}$ do
            if $j \in \mathcal{M}_i$ and $q_i^{*,low} > q_j^{*,upp}$ and $n_j > 0$ then
                $m_j \leftarrow m_j + f_i$
                if $m_j > m^{*,upp}$ then
                    $w \leftarrow j$          ▷ $j$ has the most uncovered demand so far
                    $m^{*,upp} = m_j$
                end if
            end if
        end for
    end for
    if $w \neq$ undefined then
        $x_w \leftarrow x_w + 1$    ▷ add an additional vehicle at $w$ and repeat this procedure
    end if
until $w = \textit{undefined}$   ▷ if $w \neq \textit{undefined}$, the workload condition is not satisfied
```

minimum reliability level. Model results are compared with the actual number of ambulances that the ambulance providers have in service. An ambulance may only be allocated to a base location that is currently in use.

First we determine a set of parameters that are input for all calculations, next we compare Q-PLSCP to FAQ-HPLSCP and DAQ-HPLSCP. These calculations give the minimum number of vehicles needed to provide coverage.

Dataset

Utrecht and Amsterdam-Waterland are urban regions with rural outskirts; Gooi & Vechtstreek is a small region with mid-sized villages, and Flevoland is a large rural area with two cities and multiple small-scaled towns. Tables 4.1

Region	Utrecht	Amsterdam-Waterland	Gooi & Vechtstreek	Flevoland
Demand points	220	103	41	94

Table 4.1 Number of demand points for every ambulance region.

Region	Urgencies	Number	PTD	CHT	TMT	TFT	Fractions	
		of calls	of HU				HU & MU	LU by ALS
Utrecht	Only ALS	28,109	2:51	1:35	17:58	16:52	53.74%	42.88%
	All calls	73,668		2:33	16:46	18:59	25.03%	
Amsterdam-Waterland	Only ALS	26,259	3:32	1:22	19:05	19:37	77.32%	54.59%
	All calls	59,060		2:54	17:23	20:06	30.56%	
Gooi & Vechtstreek	Only ALS	4061	2:37	1:04	15:57	10:49	75.48%	54.49%
	All calls	8930		2:08	14:14	12:13	34.32%	
Flevoland	Only ALS	6640	2:50	1:22	15:46	17:13	59.21%	40.59%
	All calls	12,725		2:25	14:54	16:58	33.04%	

Table 4.2 Constants calculated from the actual call record database, for weekdays 10:00–12:00 over the years 2008–2012.*

and 4.2 show the number of demand points, the number of base locations, the call volume, the mean values of pre-trip delay (PTD), the chute time (CHT), the treatment time (TMT), and the transfer time (TFT). Recall that the pre-trip delay is the time from the ringing of the telephone until the ambulance starts moving. To get a correct coverage region, we assume the pre-trip delay for high urgency (HU) calls. Because advanced life support (ALS) vehicles occasionally do low urgency (LU) transportations, it is too short-sighted to base the number of ALS vehicles only on the high urgency workload. Instead, we correct the number of ALS ambulances by subtracting the fraction of the time that they spend on LU calls. The historical fractions of HU and LU calls by ALS ambulances taken from the dataset are used for this correction.

For the evaluation of fixed base locations we use the actual location at the time that our dataset originates from. We did not pose any restriction on the base capacity through our choice $C_j = M$ for all $j \in \mathcal{J}$.

We limit ourselves to weekdays between 10 and 12 am. These intervals do not contain a shift change for any of the ambulance regions we considered, and they have a reasonably constant but substantial arrival rate. There are also no major fluctuations in demand between these weekdays.

Travel speeds

We aggregate to four position postal code level; see Table 4.1 for the number of demand points in every ambulance region. Because minimal reliability models assume that every demand point can be reached from at least one base location, we omit a few isolated demand points to be able to perform calculations. Table 4.2 displays all input constants, which are calculated from actual call center data records.

* These durations are not necessarily equal to the regions' official performance. For the official numbers we refer to [D2].

All travel speeds are deterministic and are being requested by navigation software. We correct the travel speeds for traveling with optical and auditory signals in the case the ambulance travels to a high urgency (HU) patient. This is the highest class of ALS urgency, and the only class where an ambulance always travels with active optical and auditory signals. Medium urgency (MU) calls are also classified as ALS, and all low urgency (LU) calls are considered BLS. In reality ALS ambulances can respond to both ALS and BLS calls, and BLS ambulances can only respond to BLS calls.

In our analysis we differentiate in two min-rel cases. The first case calculates the minimum required number of ALS ambulances to serve all HU and MU demand. This calculation requires a correction because in practice the ambulances in our data set do also respond to BLS calls. Details on this correction are discussed as Remark 1 in the results section. The second case has one ambulance type that responds to calls regardless of the urgency.

The arrival rate
The arrival frequency f_i for every demand point is calculated from the actual call detail records that were provided by each of the individual ambulance regions, over the years 2008–2012. We counted the number of dispatched calls and divided through the total duration over all these two-hour interval blocks.

Another differentiation we make in calculations is on the two density measures that were introduced in Section 3.6.1. We evaluate the frequency adjustment $\psi_i = f_i$, and the density adjusted $\psi_i = C\,\mathfrak{p}_{\mathcal{N}_i}/\mathfrak{a}_{\mathcal{N}_i}$ density measures. The latter divides the population through the area that has postal code i. The choice of the scaling parameter $C > 0$ does not influence the outcome of the methods.

The mean service time
We use a fixed-point method to calculate the mean service time β_i of an ambulance covering $i \in \mathcal{I}$. Every demand point $i \in \mathcal{I}$ may have multiple base locations by which it can be reached, i.e., $|\mathcal{M}_i| \geq 1$. It is reasonable to assume that ambulances that are allocated to the same base location have the same mean service time.

Differences in the mean service time between neighboring base locations are not that large due to the fact that PTD, TMT, and TFT are system-wide constants, and the differences in driving time are relatively small compared to the mean service time. Moreover, increasing the value of the service time leads to conservative solutions because we rather allocate more ambulances than less, i.e., a slight over-estimation of β_i is allowed to honor the reliability constraint, in contrast to an underestimation that may break this constraint.

We take a demand point $i \in \mathcal{I}$ at random and describe how we calculate its mean service time β_i. The values of β_i are calculated iteratively where we alternate the update of the allocation by solving the IP-formulation and values for β_i. During initialization the ambulances are allocated at random. Recall that we use a clear overcapacity for the AQ-HPLSCP model, e.g., we put 1000 ambulances on a single base. Recall our assumption that base locations are 'reasonably located' and demand points are most likely to be served by the nearest ambulance base location. Hence, we assume that the response time is dominated by the travel time from the nearest base location that has at least one ambulance stationed.

Consider only the demand at this demand point, and ignore for now that ambulances are shared with other demand points. We can approximate the mean service time of an ambulance serving exclusively this demand point i, thereby ignoring the fact that the service is influenced by all other demand points. The contribution of this exclusive mean service time $\tilde{\beta}_i$ to the mean service time of an ambulance serving demand point i can be approximated by the time from the nearest base location with at least one ambulance allocated to the nearest hospital. This value can be interpreted as the contribution of the demand point to the mean service time $\hat{\beta}_j$ of an ambulance at this base location $j \in \mathcal{J}$. We have

$$\tilde{\beta}_i \quad = \min_{h \in \mathcal{H},\ j \in \mathcal{J},\ x_j > 0} \bar{\beta}_{h,i,j}. \tag{4.2}$$

In words, that is the shortest round-trip from a manned base j to an incident i transporting the patient to hospital h, and returning to base j.

Recall that the ambulances are the servers in our Erlang B approach, and the service time depends on the base locations and the vertices they serve. For every base location, we now have to find a reasonable approximation of the mean service time by taking the weighted average mean service time over all exclusive mean service times in the base's neighborhood $\mathcal{N}_j$:

$$\hat{\beta}_j = \frac{\sum_{i' \in \mathcal{N}_j} \psi_{i'} \tilde{\beta}_{i'}}{\sum_{i' \in \mathcal{N}_j} \psi_{i'}}.$$

We do not condition the neighborhood on the density value in this calculation, since the allocation by the IP-formulation assumes that this base location can service any demand that it covers. Also recall that we may slightly overestimate mean service times. A corollary is that the mean service time of base locations with a large overlap in demand points is limited.

Because base locations with an ambulance stationed in the vicinity of i are more likely to cover demand points with a large response time from this base j than the demand point close to j, this $\hat{\beta}_j$ is conservative. Here we used the assumption that base locations are reasonably spread over the region.

Region	Urgencies	Actual		Q-PLSCP		FAQ-MIPLSCP	DAQ-MIPLSCP	FAQ-HPLSCP		DAQ-HPLSCP	
				Base locs		Base locs	Base locs	Base locs		Base locs	
		Act.	C/R	Fixed	Free	Fixed	Fixed	Fixed	Free	Fixed	Free
Utrecht	Only ALS	31	C: 21.0	34	28	21	23	18-27	16-30	23-25	18-27
	All calls	37	R: 35	65	48	31*	37*	25-33	25-46	35-40	28-45
Amsterdam-Waterland	Only ALS	35	C: 20.7	41	33	21	42	18-28	17-36	20-30	19-35
	All calls	40	R: 39	81	59	34*	42*	30-42	29-54	37-49	35-66
Gooi & Vechtstreek	Only ALS	7	C: 4.1	5	5	4	4	4-4	4-7	4-5	4-5
	All calls	7	R: 6	9	9	6	6	6-7	5-8	6-7	6-8
Flevoland	Only ALS	13	C: 9	17	13	13	13	12-13	11-12	13-13	10-13
	All calls	13	R: 11	21	18	15	16	14-17	13-17	16-16	14-18

Table 4.3 The required number of ambulances for weekdays 10:00–12:00 over the years 2008–2012 for the three reliability models.[†]

Staying conservative, for the mean service time of demand point i we can take the largest value $\hat{\beta}_j$ that can be reached from the base location. Most likely this is a sharp upper-bound for the actual mean service time of ambulances that serve i. We have

$$\beta_i = \max_{j \in \mathcal{M}_i} \hat{\beta}_j.$$

Q-models take $\beta_i = \beta$ as a system-wide constant [83, 109]. Also, a statement can be found about taking β_i equal to the average amount of work on the fictional server in the neighborhood $\mathcal{N}_i$, effectively saying

$$\beta_i = \frac{\sum_{i' \in \mathcal{N}_i} \min_{h \in \mathcal{H}} \lambda_{i'} \bar{\beta}_{hi'i}}{\sum_{i' \in N_i} \lambda_{i'}}.$$

For the Q-PLSCP model we use for all $i \in \mathcal{I}$ a mean service time $\beta_i = \tilde{\beta}_i$. In the evaluation of the AQ-MIPLSCP model we use the per-demand point mean service time that results from the AQ-HPLSCP model's iteration.

An improvement is made by giving the actual server, i.e., the ambulance(s) on a staffed ambulance base, a central role in the calculation of β_i.

Minimum reliability level

An ambulance service provider provided us with values $\alpha_i = 0.95$ for urban and $\alpha_i = 0.80$ for rural areas. After plotting the population density per postal code area on a map we choose a good threshold value between urban and rural areas. The population density does not necessarily depend on ψ_i. Although Q-PLSCP is not designed for a variable reliability value, we use it for a fair comparison; after all the real regions also do not meet the isolation assumption.

[†] Notation AQ-HPLSCP: Before–after applying the post-processor. (C) Actual number of ALS vehicles, when corrected for their BLS transportations. (R) Advised number of ambulances for 95% within $R = 15$ minutes subject to at least one ambulance at every base location by RIVM. (*) Starred values are the best-found results after two hours of solving.

4.5 Results

Table 4.3 shows results for Q-PLSCP, AQ-HPLSCP, and AQ-MIPLSCP.

The model is used to solve various scenarios. We made a distinction between calls of ALS urgency only, and all calls. A second distinction we made was between a free choice of base location and fixed locations. In the free base location, also known as a greenfield scenario, a potential base location is placed at the location of each demand point. For the fixed locations we used the actual base locations that were in use during the time where our data originates from. With the hardware provided it is not possible to calculate the AQ-MIPLSCP with a free choice of bases in an acceptable time. We use a constant response time threshold of $R = 15$ minutes.

We make various observations.

First, we see that FAQ-MIPLSCP outperforms all other models, especially for the larger regions. This confirms that the AQ-MIPLSCP is better suited for ambulance regions with urban and rural demand (Theorem 3.5), and shows the extent of the improvements.

Second, the AQ-MIPLSCP usually finds a better solution than the best solution found by the post-processor provided by AQ-HPLSCP. The only exceptions are when the lower bound and the best solution found are equal, and for the FAQ on Flevoland for *Only ALS*.

Third, for AQ-MIPLSCP the frequency adjustment never gives worse results than the density adjustment. This was not the case for AQ-HPLSCP; in some instances the best solution found for FAQ-HPLSCP exceeded that of DAQ-HPLSCP.

Fourth, comparing the *Only ALS* results of FAQ-MIPLSCP with the corrected number (C) of ambulances, we see that AQ-MIPLSCP provides a perfect fit for all regions with the only exception of Flevoland. This shows that the model can be used for real applications. In Flevoland there are a few rural outskirts that do not have an 80% coverage in practice. Using AQ-MIPLSCP these bases should have two ambulances, whereas in practice only one is used. As of the year 2017 ambulances were added to this region as the only-one ambulance per base policy was not sustainable.

Fifth, comparing the *All calls* results of AQ-MIPLSCP with the actual number of ambulances, DAQ-MIPLSCP yields results closer to reality than FAQ-MIPLSCP. A possible explanation can be found in the fact that ambulances in reality are distributed for quick response to ALS demand, as hospitals can provide help for life-threatening situations. For *All calls* the population density adjusted DAQ represents incidents without direct medical intervention better than FAQ, which has peaks around hospitals.

Also some interesting findings can be made for the heuristic.

Sixth, for all scenarios with fixed base locations, the required number of ambulances in Q-PLSCP exceeds FAQ-HPLSCP, and that the actual numbers for the fixed base locations are much closer to the actual (corrected) numbers than Q-PLSCP. In the case the model instance becomes too large for AQ-MIPLSCP, AQ-HPLSCP has a preference over Q-PLSCP.

Seventh, for *Only ALS* we see that the corrected actual number of ambulances for Utrecht and Amsterdam-Waterland lies between the FAQ-HPLSCP numbers with free and fixed bases. This is realistic, because in practice dynamic ambulance management yields an optimization on the scenario with fixed base locations, and a free choice of base locations can be seen as optimal locations, which provides a lower bound. For *All calls* we see that FAQ-HPLSCP gives a slight underestimation for these two regions. We assume the main cause is the way BLS calls differ from ALS calls, and BLS calls are a major part of the total volume.

Eighth, Gooi & Vechtstreek is the smallest region of the country, and it has only ALS vehicles. The Q-PLSCP model gives over-estimations, while the other models give realistic numbers.

Ninth, Flevoland is a rural area with two large cities. We see that all minimal reliability models require more ambulances than what Flevoland currently has. In reality we see that the urban areas have a good score, while the rural areas underperform. The minimal reliability models allocate an additional ambulance on rural bases to meet performance which explains the slightly higher numbers. A clear difference between Q-PSCLP and AQ-HPLSCP can be observed.

We end this section with remarks about the implications of the models introduced.

Remark 1: Correction on advanced life support vehicles for basic life support load
Regions Utrecht and Amsterdam-Waterland have both ALS and BLS vehicles. ALS vehicles are capable of handling BLS load, but this does not hold the other way around. To compare our results with the actual numbers, we have to correct the realistic number of ALS vehicles for their BLS load.

We know that BLS vehicles have a higher busy fraction than ALS vehicles because they handle the load that can be planned in advance. We assume that ALS vehicles will only do BLS within their own region, and that the longer inter-regional rides are all handled by the dedicated BLS vehicles. This yields similar driving times for all calls handled by ALS vehicles: the variance is in the order of minutes rather than hours—the mean service time of is approximately one hour.

Our correction is done through an estimation. Realistic numbers for the busy fractions are 0.8 for BLS and 0.6 for ALS for dense regions like Utrecht and Amsterdam-Waterland. The times used for these estimates include the driving back to base. When ALS vehicles handle BLS calls they are more time efficient than handling ALS calls. This is because first of all the stochasticity of the arrival rate is reduced by introducing planned transportations, and second, because the ALS vehicles only do BLS load when there is an overcapacity in ALS vehicles. When dealing with BLS load, we assume that ALS vehicles are as effective as BLS vehicles doing the same work.

This means that when an ALS vehicle handles BLS load, it has $0.8/0.6 = 4/3$ times the effectiveness. If we know the fraction of BLS load done by an ALS vehicle, we can use this fraction and the effectiveness measure to correct the actual number of vehicles for the BLS work they do. This yields a corrected number of 21.0 vehicles for Utrecht and 20.7 vehicles for Amsterdam-Waterland.

Remark 2: Practical implications
The actual numbers are not necessarily the optimal allocation, because the regions optimize another objective function than we do. On the one hand they try to minimize the number of late arrivals, while keeping local communities satisfied. Also, it may be that due to shift lengths they have a temporary over- or undercapacity, and they correct this in another time block to meet the constraints and objective.

Remark 3: Dynamic ambulance management
All regions perform some kind of dynamic ambulance management (DAM). In our calculations we assumed that every ambulance returns to its home base. It is generally known that DAM increases performance.

Remark 4: Response radius
We assumed that every call should be available as if it were a high urgency call. The area in the calculation of λ_i equals that of a high urgency call. When considering other ALS calls that have a larger response radius and BLS vehicles that have no response time threshold, we see that our method may yield a slight over-estimation.

Remark 5: Late arrivals limit versus mean response time
There are two extremes in facility locations; one can either minimize the number of late arrivals, or one can minimize the mean response time. We pose a lower bound on the late arrivals for every demand point. The result is that in some areas, the mean arrival time may approach the response time limit for various bounds. Especially when these demand points have a considerable demand, ambulance providers may choose to position more ambulances than strictly needed by our response time threshold, such that the mean response time decreases.

Remark 6: Spiking
The demand point dependent minimum reliability level requirement α_i, $i \in \mathcal{I}$, can result in an effect that we call *spiking* in small towns marked as urban in an otherwise rural environment. If there is a small area with relatively high demand and large α_{urban} encapsulated by a rural area with a low α_{rural}, the arrival rate $\lambda_\bullet$ of urban demand points can be dominated by the summarized arrival frequency of the rural demand points. However, α_i depends only on the urban area. Hence, through this effect the rural demand is rated under the minimum reliability level of urban demand. This combination may lead to relatively high values b_i for these urban demand points. An overcapacity of about three ambulances in the ambulance region Flevoland is caused by this effect.

Technical details
We have implemented Q-PLSCP, FAQ-HPLSCP, and DAQ-HPLSCP in the TIFAR framework (see Chapter 7 and [A7]), and used the Coin-OR CBC solver through the CoinMP interface [O2] to solve the IP-formulation. These problems are solved on a MacBook Pro (17-inch, Mid 2009) with a 2.8 GHz Intel Core 2 Duo, and 8GB 1067 MHz DDR3. Each model instance was solved within minutes.

The AQ-MIPLSCP is implemented in AIMMS 4.28.1.0 and solved with CPLEX 12.6.3 on a Windows 7 PC with an Intel Core i7-4770K LGA1150 processor (8 hypercores at 3.5GHz) and 16GB Corsair Vengeance Pro 2400MHz (2x8GB) RAM on an Asus Z87-K motherboard. All Gooi & Vechtstreek and Flevoland computations were done within 18 seconds. Computation times of *Only ALS* scenarios for Utrecht and Amsterdam-Waterland ranged from 10 minutes to 1.5 hours. *All calls* scenarios for the latter two ambulance regions were all interrupted after two hours.

4.6 Conclusion

This chapter addresses the minimal reliability ambulance allocation problem, that aims to find a static ambulance allocation such that an on-time ambulance response can be guaranteed by a given minimum reliability level. The Q-

PLSCP model that is known from literature is not suitable for so-called mixed regions, which have both urban and rural demand; applying it nevertheless provides an unrealistic over-estimation of the number of required ambulances. The (theoretical) adjusted queuing framework that we proposed in the previous chapter solves this over-estimation problem.

This chapter proposes two minimal reliability models that use this theoretic adjusted queuing framework, i.e., the adjusted queuing neighborhood definition and the workload condition. The first model, a mixed integer program, gives an exact solution to this problem. Calculation times, however, limit the application of this model to small model instances. The second model is a heuristic that is derived from Q-PLSCP that can be applied to larger model instances.

Application of these two models to four actual ambulance regions shows improved results for the minimum required number of ambulances. The AQ-MIPLSCP is the first minimal reliability model that is designed for application in mixed regions, and it solves it to optimality. Results show that FAQ-HPLSCP finds realistic values for *Only ALS*, as does DAQ-HPLSCP for *All calls*.

However, it should be noted that the objective for actual ambulance regions and minimal reliability models are not necessarily the same.

There are some issues open for further research.

When the numbers of potential ambulances and demand points grow, so does the number of binary variables of the mixed integer program. If the problem has around a hundred demand points and fifty ambulances, CPLEX can solve AQ-MIPLSCP within a couple of hours, but for regions with hundreds of demand points and ambulances this may be too much to ask for current state-of-the-art computers and solvers. Mixed integer programs are NP-hard. However, heuristics can be found to provide good approximations of the optimum.

The AQ-HPLSCP leads to a wide optimization gap between the lower bound and the solution found that leaves room for improved heuristics.

Although this chapter is motivated by the EMS context, the models are useful for a wide spread of applications that statically position high valuable assets that must be available for an unplanned emergency by at least a given probability at any pre-determined location. Examples include specialty units of a police department, positioning valuable tools that need post-processing after every use, or choosing what volunteers to provide with basic life-saving education in the case that there are limited training spaces available.

An open question that helps the applicability to larger model instances is how a better post-processor can be designed for the heuristic. The question if FAQ always outperforms (or equals) DAQ, in the case the (heuristic) model solves to optimality, is an interesting open question for further research.

The present chapter focuses on minimal reliability models, although a similar mixed integer adjusted queuing technique as the AQ-MIPLSCP may be applied to maximal availability location models that have a limited number of ambulances to provide maximum coverage. It is an interesting question to what extent the maximal availability models can be improved by the adjusted queuing framework.

In the calculation of $\tilde{\beta}_i$ we assume that the ambulance is dispatched from the nearest opened base location. An interesting enhancement would be to adapt this such that all staffed base locations that can reach the demand point are included in the calculation. We can do that by taking the probability that a base location is the one that sends the ambulance, multiplied by the driving time. One can also estimate a good distribution for the busy time at each demand point using call center records details.

Acknowledgements
Our thanks go to RAVU, Ambulance Amsterdam, GGD Gooi & Vechtstreek and GGD Flevoland for the call center records. We thank Connexxion NV for the area per postal code used in the density adjustment, and RIVM for the lookup table with driving times between every two postal codes in the Netherlands.

5

ON-ROUTE COVERAGE BY AVAILABLE AMBULANCES THROUGH ROUTE OPTIMIZATION

Dispatchers usually relocate ambulances by the fastest route. However, there are cases when the fastest relocation is not the best one, as the transitory coverage while driving over the fastest route can be relatively low. When a fast relocation happens over the highway with a limited number of exits the ambulance is less flexible, while a longer route through villages can lead to a higher performance due to transitory coverage. In this chapter, we study the effect of taking alternative routes in ambulance relocations, which is an unaddressed problem in the literature. Results for four actual ambulance regions show that this so-called *dynamic routing* can help at an operational level to obtain a more fair distribution of ambulance coverage.

This chapter is based on the following publication:

[A5] M. van Buuren, C. W. P. Huibers, and R. D. van der Mei. "On-route Coverage by Available Ambulances Through Route Optimization". *Submitted* (2018)

5.1 Introduction

Dynamic ambulance management, DAM, distributes available ambulances over an ambulance region to minimize late arrivals. When an ambulance finishes a service at a hospital, or when a dispatch to a new incident occurs, it is generally beneficial to ask available ambulances to drive to another location than a fixed base location, such that the coverage increases. An ambulance movement for improved coverage is called a *relocation*. Relocations may only go to a limited number of predefined relocation points—often these are base locations.

The DAM models known from literature often start by determining for each ambulance to what base locations it can be relocated, and consequently, what configuration scores best. After calculating the optimal configuration, the output for each ambulance from what origin to what destination it must move; this is called an *OD-pair*. These models, however, do not specify *how* an ambulance should drive to the destination. Usually it is assumed that the ambulances take the fastest route.

Ambulances in densely populated areas such as the Netherlands are distributed for fairness: no matter where in the country you are, there must be a reasonable probability that an ambulance is nearby. Because the performance indicators are aggregated over the year and the entire ambulance region, there is an incentive to concentrate ambulances around cities with high demand. Recall that Chapters 3 and 4 address this issue and propose facility and allocation models to increase fairness.

The main contribution of the current chapter is that we focus on the question *what route* the ambulance should take to drive to its destination. While an available ambulance is moving to its destination, the route choice has an influence on the coverage, and thereby, both on the volume and the spatial distribution of late arrivals. This so-called *dynamic routing* can be highly effective in increasing fairness over an ambulance region, particularly for regions where subareas are not covered from existing base locations. We show results for three ambulance regions.

A literature review is provided in Section 5.2. Next, in Section 5.3, we propose our dynamic routing model. Results for actual ambulance regions are provided in Section 5.4. This chapter ends with a conclusion in Section 5.5.

5.2 Literature

Many methods address the issue of *how* to choose the ambulance movement, see Section 6.2 for a literature overview on this topic. We first discuss ways for alternative routing. Directly after, we give a brief overview of the *Dynamic*

Maximum Expectated Coverage Location Problem (DMEXCLP) that is used in this chapter.

Generating a routeset, i.e., a set containing route alternatives, in combination with a route choice model is well studied in the literature. An early review of this topic is given in [99]. Further extensive literature can be found in [104] and [112].

Various techniques are proposed to deal with uncertainty in travel costs [34, 42, 98]. Stochastic approaches calculate iteratively each route alternative in two phases. In the first phase, the edge costs are drawn from a distribution, and in the second phase a shortest path is calculated [26, 36]. Doubly stochastic models are an extension that also randomize the objective function, in the case it consists of multiple weighted link properties [30, 93]. The labeling approach generates a routeset for multi-criteria objective by including one route for each of the criteria [18]. Link elimination alternatively executes two procedures: a shortest path algorithm and a path deletion algorithm that deletes the characteristic link [9, 100]. A breath-first search can be appended to link elimination to speed up the process of generating a high diversity of paths [112].

The models mentioned above are designed to find a good fastest route. Our objective, however, differs because we wish to incorporate the coverage provided *by a route*.

The route selection problem for hazardous material transports is a relatable problem to ambulance routing, since both involve coverage. Instead of taking routes with the *least* coverage, the ambulance context wants us to take a route that generates the *most* coverage in their objective function. A known approach is to first define the edge risk for each road segment (that is the probability that an incident occurs on an edge multiplied by the damage), and thereafter to calculate a minimal path [1, 41, 115, 123].

There are differences between hazardous materials and real-time ambulance allocation is the required calculation speed. Whereas hazardous material routes can be calculated weeks in advance, the ambulance dispatch centers need a faster method that can provide an answer within seconds. Also, in ambulance care the route choice depends on the locations of other available ambulances, which is not the case for hazmat transports.

In this chapter we use the DMEXCLP model for the calculation of the OD-pairs, and base our route choice on the MEXCLP model (see Section 3.2.2 and [35] for more details). An outline of DMEXCLP follows (an extensive description can be found in Section 6.3.1 and [63]). The basic idea of DMEXCLP is to take multiple coverage into account, and consequently, sending an ambulance to the base location where its marginal contribution is the highest. To this end, a constant busy fraction q is introduced, that is the average fraction of the time that an ambulance attends an incident. The marginal contribution to the coverage of the k^{th} ambulance on demand point i (denoted by k_i) is given by $C_i = d_i(1-q)q^{k_i-1}$. Summation over all demand points yields the total contribution of an ambulance movement.

5.3 Model

The DMEXCLP-model from the previous section, or any other DAM-model, provides the origin O and destination D for ambulance a to send. The goal of our model is to optimize *the route* for ambulance a, while taking transitory coverage into account.

The ambulance region is discretized in a set of demand points $\mathcal{I}$ (where demand point i has demand demand d_i) and base locations $\mathcal{J}$, similar to Chapter 3 and 4. The set of *waypoints*, denoted by $\mathcal{Y}$, is the set of the locations of all road intersections and all locations where a road changes direction—a curve in the road can be modeled by placing multiple waypoints. Take $\mathcal{L} := \mathcal{Y} \cup \mathcal{I} \cup \mathcal{J}$ as the set containing the waypoints, the demand points, and with the base locations. Denote the minimum travel time between two points $\ell_1, \ell_2 \in \mathcal{L}$ by t_{ℓ_1,ℓ_2}. The road network is modeled as a directed complete graph with nodes $w \in \mathcal{W}$ and the minimum travel times as the edge weights. We assume that the grid of demand points is dense enough in relation to the travel speeds.

A *route* is modeled by a finite sequence $r = (\ell_0, \ell_1, \ldots, \ell_{|r|-1})$ that has taken all its elements from $\mathcal{L}$. A set of routes is referred to as a *routeset*, and is denoted by $\mathcal{R}$.

The proposed dynamic routing method consists of two phases. First, we generate a routeset that contains a fixed number of alternative routes from O to D (see Section 5.3.1). Subsequent, we evaluate each of the routes in this routeset individually (see Section 5.3.2). The ambulance is sent over the highest valued route alternative.

5.3.1 Generating routesets

In this section we show how to calculate the routeset $\mathcal{R}$ for a given OD-pair. In this context, there is a trade-off: on the one hand we need enough routes in the routeset to make a good choice, but on the other hand lead too many routes to unacceptably long calculation times during the evaluation. Therefore, the limited number of routes must be sufficiently different to cover the entire area between O and D.

From practice we get two constraints on a route:

1. It should be easy to explain the route in words over the telephone to an ambulance driver, e.g., a route alternative may not meander through residential areas.

2. EMTs do not accept 'large detours' to reach their destination.

In any case, we want to evaluate the fastest route between O and D as a route alternative. Hence, this is the first route that we include in the routeset.

Decision points

We do not follow the approach from most models in the literature that changes the arc properties for the entire road network in the calculation of a route alternative, as this is too calculation intensitive. Instead, we introduce *decision points* on the road network that help us generate more routes, denoted by $v \in \mathcal{Z}$ $(\mathcal{Z} \subseteq \mathcal{Y})$. Decision points are waypoints that lay on the road network where a main road splits, and are used as *via-points*. That is, a route alternative is a combination of the fastest route from the origin to the decision point, and the fastest route from the decision point to the destination. A method that calculates the decision points follows in Section 5.4. We allow route alternatives to be outside the ambulance region for a limited time. The (partial) travel time matrix with entries t_{wi} is precomputed $(w \in \mathcal{W}, i \in \mathcal{I})$.

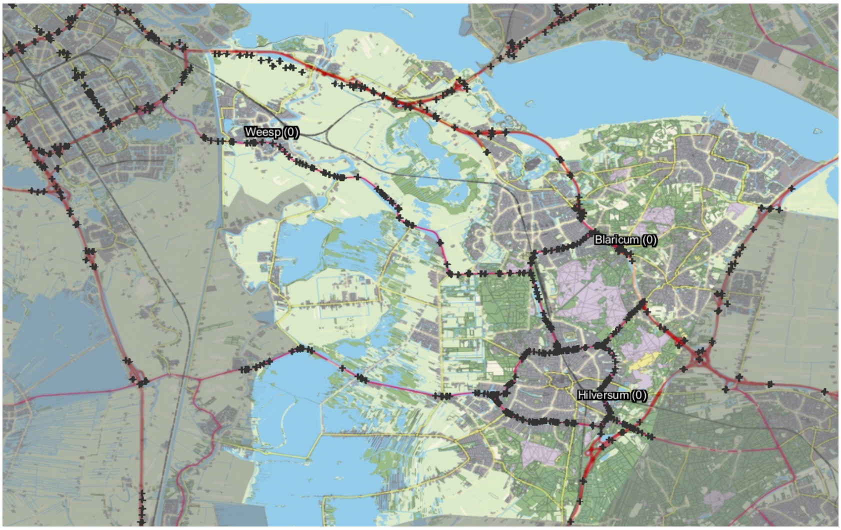


Figure 5.1 The decision points for ambulance region Gooi & Vechtstreek are indicated by black crosses.

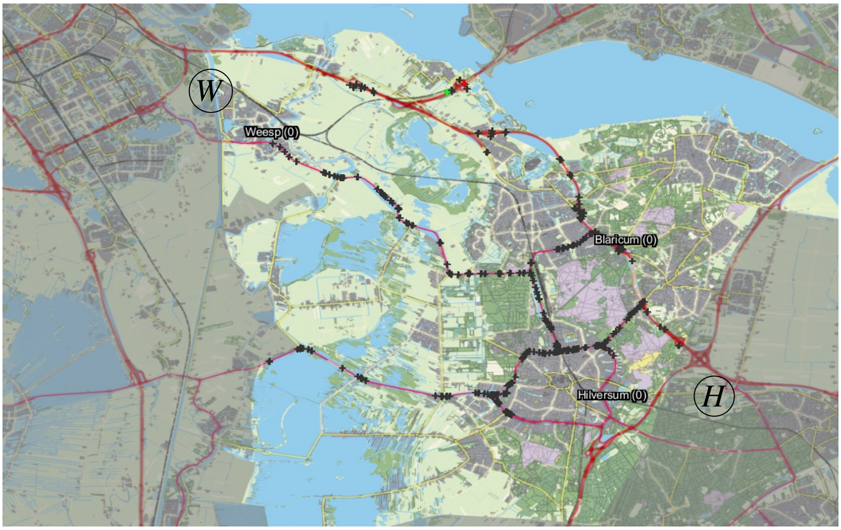


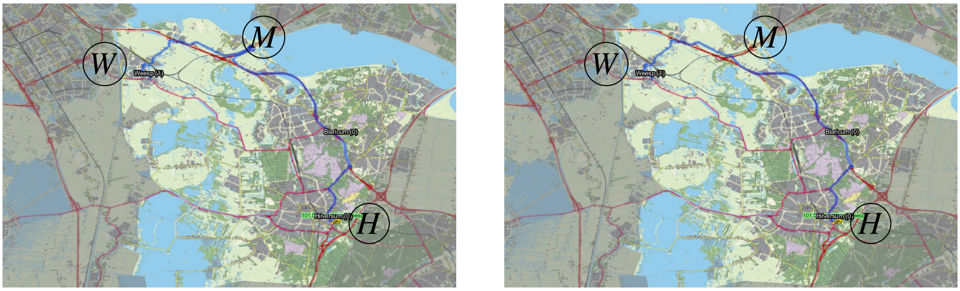
Figure 5.2 The decision points between Hilversum (H) and Weesp (W).

This satisfies the first constraint: decision points are easy to explain. As an illustration Figure 5.1 shows the decision points for ambulance region Gooi & Vechtstreek.

The second constraint motivates to take a subset of the decision points between O and D, and only to consider these decision points as a via-point:

$$\mathcal{Z}_{O,D} := \{z \in \mathcal{Z} : t_{O,z} \leq t_{O,D} \text{ and } t_{z,D} \leq t_{O,D}\}. \tag{5.1}$$

Figure 5.2 shows the resulting decision points between Hilversum and Weesp.



(a) Route without snapping.

(b) Route with snapping.

Figure 5.3 Illustration of snapping on a route from Hilversum (H) to Weesp (W) through Muiderberg (M).

Algorithm 2 The snapping algorithm

```
$r_1 \leftarrow$ routepart of $r$ from (not incl) $O$ to (not incl) $p$
$r_2 \leftarrow$ routepart of $r$'s inversed route from (not incl) $D$ to (not incl) $p$
for all $\ell_1 \in r_1$ do                       ▷ visit each waypoint $\ell_1$ on $r_1$ in order
    for all $\ell_2 \in r_2$ do           ▷ visit each waypoint on the inversed route in order
        $d_{snap} \leftarrow$ dist($\ell_1$, $\ell_2$).                       ▷ Euclidean distance in meters
        if $d_{snap} \leq \Delta$ then        ▷ points are within the snapping distance threshold
            if $t_{z_1 p} \leq S$ and $t_{p z_2} \leq S$ then                    ▷ pivot $p$ remains covered
                return append(route$(O, \ell_1)$, route$(\ell_1, \ell_2)$, route$(\ell_2, D)$)
            end if                          ▷ the route function gives the fastest route
        end if
    end for
end for
```

Snapping

We apply the so-called *snapping* to a route alternative. This procedure is designed to prevent going back and forth over the same path when we create the shortest path through the decision point by removing paths that we travel over in opposite directions. We limit snapping to S minutes: if the original via-point is not reachable by its resulting route alternative, we communicate the point of snapping in its place.

Figure 5.3 illustrates snapping. While generating route alternatives between Hilversum (H) and Weesp (W), a decision point next to Muiderberg (M) is taken as a pivot. Figure 5.3a shows the route alternative without snapping. This route alternative is not acceptable for EMTs because it (locally) is a large detour: the same road is traveled twice while entering and leaving Muiderberg. In Figure 5.3b we applied snapping, and we do not go into Muiderberg, but we still cover the area because the ambulance drives past it.

Algorithm 2 describes the snapping process. We assume two waypoints on opposite sides of the road if their difference is at most Δ meter.

Generation algorithm

From hereon we explain the workings of our *routeset generation algorithm*.

For each decision point in $\mathcal{Z}_{O,D}$ we keep track whether there is already a route that visits the decision point: in the case that any route from the routeset visited a decision point, this decision point is marked as *visited*.

After adding the fastest route and marking all the decision points along its way as visited, we take an unvisited decision point as the pivot p. The next route alternative we consider to add to the routeset is the fastest route from O via p to D. We mark all decision points on this new route alternative as visited. For each route alternative r that is already in the routeset, we calculate the *fraction of overlap* with the new route. That is, the number of decision points that is both in the new route and r divided by the total number of decision points that the new route has. If the fraction of overlap is below a given threshold δ for all routes in the routeset, we consider the new route to be sufficiently unique. Only then, we add this route to our routeset. We repeat this process until all decision points in $\mathcal{Z}_{O,D}$ are marked visited.

We use a method to speed up routeset generation, which prevents picking multiple decision points on a road facing outward. Figure 5.2 illustrates the motivation for the so-called *counter-clockwise pivot picking strategy*. If one first takes the route through the green decision point, one will not reach the red decision point, and thus we have to look at a route through the red decision point at a later time as well, since it is still unvisited. This route will be very similar to the route through the green decision point, and we want to avoid similar routes. Thus, by picking the more outward decision points first, we mark more decision points and prevent looking at too many similar routes.

The counter-clockwise pivot picking strategy is as follows. We first choose the unvisited decision point with the lowest y-coordinate as our pivot. As the next pivot we take the unvisited decision point with the highest x-coordinate. Next we take the highest y-coordinate, and at last, the lowest x-coordinate. Then we look again at the unvisited decision point with the lowest y-coordinate, and we repeat this procedure until all decision points are visited. Hence, we mark the decision points in a counter-clockwise fashion, starting from the outside and working towards the inside of the ambulance region.

Lower values of δ result in smaller routesets. Figures 5.4a, 5.4b and 5.4c show multiple routesets that are generated between Hilversum and Weesp for different values of δ. The thickness of the black line indicates the number of route alternatives that use the road segment.

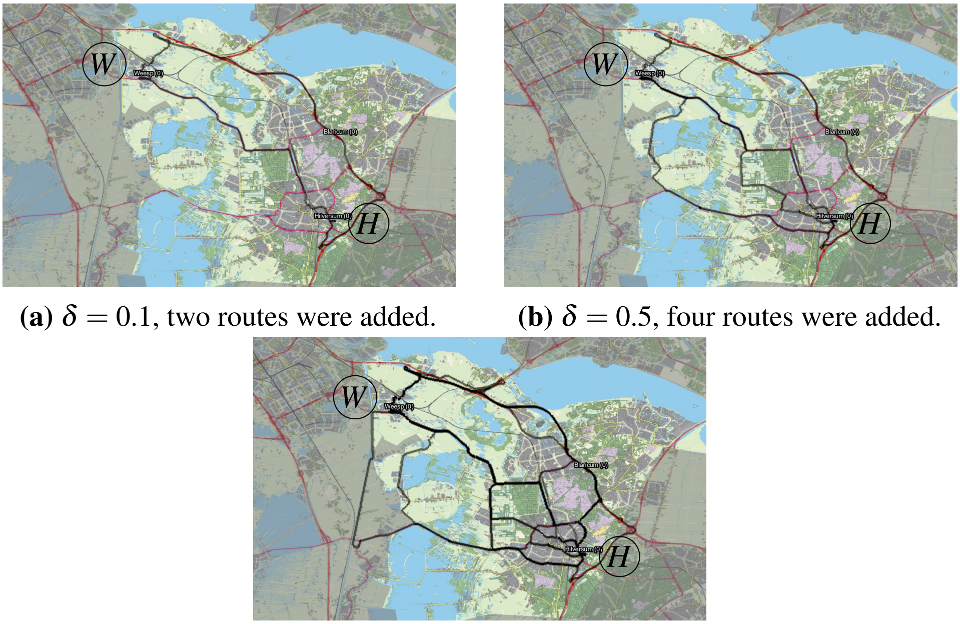
Algorithm 3 concludes the routeset generation algorithm.

5.3.2 Route choice model

The previous section provided a routeset $\mathcal{R}$. This section shows how a coverage value can be assigned to each route alternative. The route alternative with the largest value is advised to the ambulance.

Algorithm 3 Routeset generation

```
calculate $\mathcal{Z}_{O,D}$                               ▷ use the definition from Equation (5.1)
mark all decision points in $\mathcal{Z}_{O,D}$ as not visited
calculate route($O$, $D$)                                   ▷ fastest route from $O$ to $D$
mark all decision points on Route($O$, $D$) as visited
put route($O$, $D$) in the routeset $\mathcal{R}$
**repeat**
    $p \leftarrow$ first unvisited decision point according to the pivot picking strategy
    $PivotRoute \leftarrow$ append(route$(O,p)$, route$(p,D)$)
    mark all decision points on $PivotRoute$ as visited
    $PivotRoute \leftarrow$ snapping($PivotRoute$)                ▷ using Algorithm 2
    $FractionOfOverlap \leftarrow \max_{r\in\mathcal{R}}\{|PivotRoute \cap r|/|PivotRoute|\}$
    **if** $FractionOfOverlap \leq \delta$ **then**
        put $PivotRoute$ in routeset $\mathcal{R}$
    **end if**
**until** every demand point in $\mathcal{Z}_{O,D}$ is marked as visited
**return** $\mathcal{R}$
```



(a) $\delta = 0.1$, two routes were added.

(b) $\delta = 0.5$, four routes were added.

(c) $\delta = 0.9$, sixteen routes were added.

Figure 5.4 Routesets between Hilversum (H) and Weesp (W) with different values for δ.

Outline

Recall that incidents are aggregated to demand points $i \in \mathcal{I}$, and the route alternative visits various waypoints in order, denoted by the sequence $r = (O, \ell_1, \ell_2, \ldots, D)$. These transitional waypoints are not limited to the origin, the decision points and the destination. All changes of travel speeds occur at a waypoint. This is, the speed between two neighboring waypoints is assumed to be constant.

The coverage value of route r can be approximated by adding two terms: (1) the transitional coverage while being on route,* and (2) the coverage when being at the destination.

The resulting coverage value of route r is given by the expression:

$$\Xi_r = \sum_{\substack{\ell \in r \\ \ell \neq O}} \sum_{i \in \mathcal{I}} \int_{\tilde{t}_{\ell-1}}^{\tilde{t}_\ell} f(t_{x(\theta),i}) e^{-\gamma\theta} d\theta + \sum_{i \in \mathcal{I}} \int_{\tilde{t}_D}^{\infty} f(t_{D,i}) e^{-\gamma\theta} d\theta. \tag{5.2}$$

Here, the preceding waypoint to ℓ is denoted by $\ell - 1$. The travel time from O to v is $\tilde{t}_v$. Variable $x(\tau)$ is the position of the ambulance at time τ, that is a linear interpolation in time and space between ℓ and $\ell - 1$. Discount parameter γ models the fact that uncertainty increases as time goes on. Value function $f(\tau)$ gives the marginal contribution of the relocating ambulance to the coverage value as a function of the driving time τ. The integral provides a fair comparison between various routes, because the coverage at the destination weighs heavier if the ambulance arrives sooner.

The coverage value for the maximum expected coverage location problem

Recall that the marginal coverage function is denoted by $f(\tau)$ for travel time τ. Function f depends on the location of the other ambulances. For MEXCLP we can use the known result

$$f(\tau) = \mathbb{1}_{\{\tau \leq R\}} d_i (1-q) q^{k_i - 1}, \tag{5.3}$$

where $\mathbb{1}_E$ denotes the indicator function on the event E, q is the average ambulance busy fraction, d_i is the demand at i, and the relocating ambulance is the k_i-th ambulance that can reach i within time threshold R.

*For the first term, that is the contribution to the transitional coverage while being on route, we integrate over the travel time θ during the route. We split the integral over the entire route into $\ell - 1$ integrals, one for each road segment. Next, for each point $x(\tau)$ on the route r we calculate the marginal contribution to each demand point i as defined by function $f(\cdot)$, where $t_{x(\theta),i}$ denotes the travel time from the position on the route $x(\theta)$ at time θ to i.

Inspired by literature [14, 63], we fix the locations of all other ambulances $\mathcal{A}^-$ at their current position for the ease of calculations. That is, if they are on a main road we teleport them to the next decision point, and when they are in a residential area we teleport them to the closest demand point. Using the preprocessed travel time matrix, we can rapidly calculate $k_i = 1 + \sum_{a \in \mathcal{A}^-} \mathbb{1}_{\{t_{a,i} \leq R\}}$ the number of ambulances that can reach i within time R, in the case that the relocating ambulance can be on-time at i ($i \in \mathcal{I}$). We compute the contribution if the relocating ambulance is at most R time units away from i, which is independent of the path chosen:

$$C_i = d_i(1-q)q^{k_i-1}.$$

From hereon, we evaluate each route alternative $r \in \mathcal{R}$ in the routeset. For waypoints $\ell \in r$ on the main road network we calculate the coverage from the next decision point the ambulance visits, where we make a correction by subtracting the driving time towards this decision point from the value function's argument. For all other waypoints, which are usually waypoints in residential areas, we calculate the coverage as if the ambulance is at the closest demand point. This is a good approximation because the demand point aggregation is assumed sufficiently dense. Note that by this method the coverage calculation from each waypoint is always calculated from an element in $\mathcal{W}$.

For each pair of demand point i and the route segment preceding ℓ, we compute by linear interpolation the number of seconds that the ambulance is within R time units from i, while driving on this route segment. We assume that the travel speed does not change between two waypoints, which allows us to perform a linear interpolation inside the route segment. Hence, when substitution Equation (5.3) in the first term of Equation (5.2), the summation over all road segments of the route r gives the following *contribution* $\xi_{r,i}$ to i:

$$\begin{aligned}
\xi_{r,i} &= \sum_{\substack{\ell \in r \\ \ell \neq O}} \int_{\tilde{t}_{\ell-1}}^{\tilde{t}_\ell} f(t_{x(\theta),i}) e^{-\gamma\theta} d\theta = \sum_{\substack{\ell \in r \\ \ell \neq O}} \int_{\tilde{t}_{\ell-1}}^{\tilde{t}_\ell} \mathbb{1}_{\{t_{x(\theta),i} \leq R\}} C_i e^{-\gamma\theta} d\theta \\
&= C_i \sum_{\substack{\ell \in r \\ \ell \neq O}} \int_{\tilde{t}_{\ell-1}}^{\tilde{t}_\ell} \mathbb{1}_{\{t_{x(\theta),i} \leq R\}} e^{-\gamma\theta} d\theta = C_i \sum_{\substack{\ell \in r \\ \ell \neq O}} \int_{lb(\ell-1,\ell,i)}^{\tilde{t}_\ell} e^{-\gamma\theta} d\theta \\
&= \frac{C_i}{\gamma} \sum_{\substack{\ell \in r \\ \ell \neq O}} \left(e^{-\gamma\, lb(\ell-1,\ell,i)} - e^{-\gamma \tilde{t}_\ell} \right).
\end{aligned}$$

Here the lower bound $lb(\ell-1, \ell, i)$ is the time on the road segment between $\ell - 1$ and ℓ when the ambulance driving on this segment becomes within the R time units driving of i. The time measure starts at the route's origin. If i cannot be reached in R time units from the entire segments, we say $lb(\ell-1,\ell,i)=t_\ell$, which results in a zero contribution of this segment. Recall that the travel speed does not change on a road segment that lays between two waypoints. We get the following expression for lb:

$$lb(\ell-1,\ell,i)=\begin{cases}\tilde{t}_\ell & R<t_{\ell,i},\\ \tilde{t}_{\ell-1} & t_{\ell-1,i}\leq R \text{ and } t_{\ell,i}\leq R,\\ \tilde{t}_{\ell-1}+\frac{R-t_{\ell-1,i}}{t_{\ell,i}-t_{\ell-1,i}}(\tilde{t}_\ell-\tilde{t}_{\ell-1}) & R<t_{\ell-1,i} \text{ and } t_{\ell,i}\leq R.\end{cases}$$

Here, we use the assumption that the ambulance has a constant speed between two waypoints. For the moment of arrival at the destination we get:

$$\int_{\tilde{t}_D}^{\infty} f(t_{D,i})e^{-\gamma\theta}d\theta = \mathbb{1}_{\{t_{D,i}\leq R\}}\frac{C_i}{\gamma}e^{-\gamma\tilde{t}_D}.$$

This gives us the simplified *path contribution* for the MEXCLP coverage function:

$$\begin{aligned}\Xi_r &= \sum_{\substack{\ell\in r\\ \ell\neq O}}\sum_{i\in\mathcal{I}}\left(\frac{C_i}{\gamma}\left(e^{-\gamma\, lb(\ell-1,\ell,i)}-e^{-\gamma\tilde{t}_\ell}\right)\right)+\sum_{i\in\mathcal{I}}\mathbb{1}_{\{t_{D,i}\leq R\}}\frac{C_i}{\gamma}e^{-\gamma\tilde{t}_D}\\ &= \frac{1}{\gamma}\sum_{i\in\mathcal{I}}C_i\left(\left(\sum_{\substack{\ell\in r\\ \ell\neq O}}\left(e^{-\gamma\, lb(\ell-1,\ell,i)}-e^{-\gamma\tilde{t}_\ell}\right)\right)+\mathbb{1}_{\{t_{D,i}\leq R\}}e^{-\gamma\tilde{t}_D}\right).\end{aligned}$$

5.4 Results

In this section we show simulation results for the ambulance regions Gooi & Vechtstreek, Amsterdam-Waterland, and Utrecht. We compare the *fastest route* to the *best route generated by the MEXCLP dynamic routing policy*. As key performance indicators we use the fraction of late arrivals and the mean response time. In both policies we use DMEXCLP for the calculation of the OD-pair.

Setup

We use a so-called *trace-driven simulation* strategy for the months September and October 2015. In these months there are no major holidays. In a trace-driven simulation we simulate incidents at exact same time and place as they occurred in reality. The only difference is the way we relocate the ambulances. More information about simulation is provided in Chapter 7.

A hexagonal equidistant demand point grid is used, such that there is a 1 kilometer distance between the grid points. Demand patterns are calculated from one year historical data. We use $\delta=0.5$, $q=0.3$ and $\gamma=1/(\lambda M)$ the expected inter-arrival time for a particular ambulance. Here, M is the number of ambulances in the region. The Open Source Routing Machine (OSRM) [O4]

	Fastest route		Dynamic routing	
	Fraction of late arrivals (%)	Mean response time (min:sec)	Fraction of late arrivals (%)	Mean response time (min:sec)
Gooi & Vechtstreek	15.03	11:23	15.87	11:28
Amsterdam-Waterland	0.69	7:50	0.67	7:38
Utrecht	7.03	9:32	7.15	9:28

Table 5.1 Simulation results for the three regions.

is used for navigation—this software is constructed such that the speed does not change between two waypoints.

From OpenStreetMap (OSM) [O5], we create the set of decision points on the major road network $\mathcal{Z}$. These decisions points are identified in OSM as the nodes that lay on a way with highway tag highway, trunk or primary (and their resp. links) that either intersect with another road type or are included in at least three ways. In the Netherlands this results in 49,887 decision points. The choice of including primary but not secondary roads is made because it is generally not hard to turn roads of type secondary or lower. We assume that ambulances pass the following decision point when being dispatched to an incident. For each region, we truncate the set $\mathcal{Z}$ to demand points that are either in the region, or are less than 30 km away from any point in the region. The choice is based such that all demand points of the region are unreachable in 12 minutes driving when not exceeding 150 km/h. This significantly reduces the set $\mathcal{Z}$.

Table 5.1 shows the simulation results for the *fastest route* and *dynamic routing* policies. In the remainder of this section we consider each of the three ambulance regions separately.

5.4.1 Gooi & Vechtstreek

The EMS region Gooi & Vechtstreek is the smallest in the Netherlands, both in size and number of calls. It is a rural area with a population just over 250,000. Most people live in the north and the east of the region. The south-west mainly consists of lakes and forests. Yearly, there are over 18,000 incidents. There are three base locations, situated in Hilversum, Blaricum and Weesp. Figure 5.5 shows a map of the region with its base locations and call volume distribution.

Table 5.1 shows that there is a small increase in the mean response time. To get a better understanding of where and when we get the most improvement, we look at Figure 5.6. Nodes are colored green if the dynamic routing method outperforms the fastest route policy, that is where dynamic routing has less late arrivals. A red node indicates that dynamic routing method performs worse. Deeper analysis gave the insight that the high percentage of late arrivals is due to a shortage of ambulances in the evening, especially in the weekends. At that time there are barely enough ambulances available to handle the incoming calls.

Figures 5.6 and 5.8 show that the most improvement is in Wijdemeren, the municipality in the south-west corner of the region. This comes at the cost of more late arrivals in the municipality Gooise Meren, which is located in the north. Dynamic routing thus moves the location of late arrivals to more rural areas, since we now take the highway through Gooise Meren less and instead drive through Wijdemeren.

Because the region is understaffed in the evening, we are interested if our method performs better or worse in the evening compared to the rest of the day. Figure 5.7 shows the late arrivals in the evening compared to the rest of the day. Here the evening is from 16:00 to midnight.

We observe that dynamic routing performs significantly worse in the evening compared to the rest of the day. Sometimes there is a shortage in the number of available ambulances during the evening, which might be the reason that dynamic routing performs worse. The results might improve if we make the discount parameter γ and the busy fraction q time dependent.

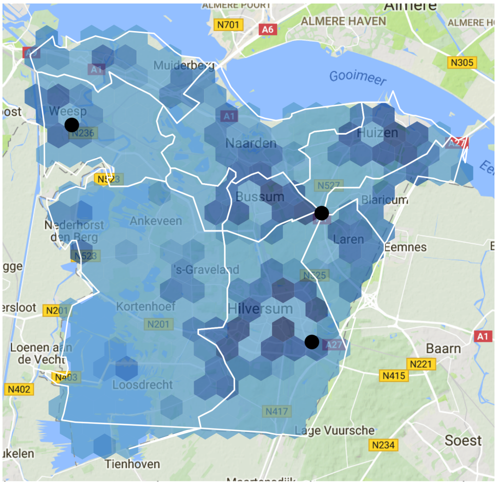


Figure 5.5 The blue area is the EMS region Gooi & Vechtstreek. The three base locations are indicated by black dots. Darker shades of blue correspond to a higher call volume.

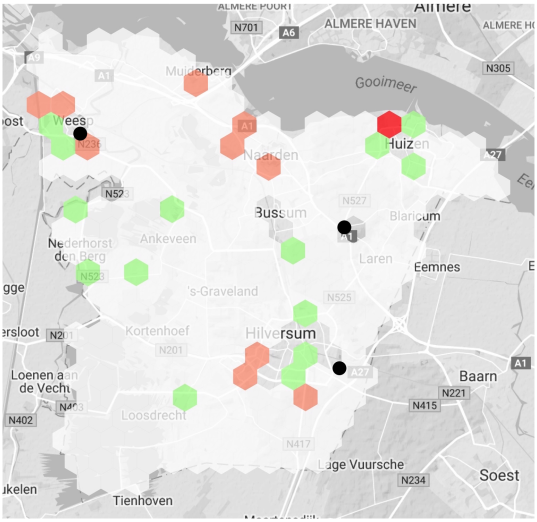
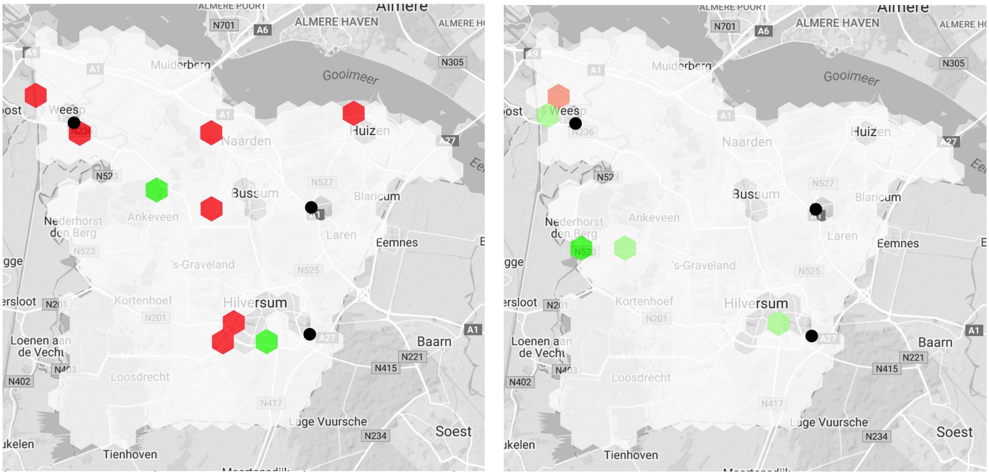


Figure 5.6 Comparison of the number of late arrivals in Gooi & Vechtstreek. Green areas show improvement with dynamic routing, in contrast to reds. The black dots indicate the base locations.



(a) Evening.

(b) Rest of the day.

Figure 5.7 Comparison of the number of late arrivals for different times of the day for workdays. Green areas are improved with dynamic routing, in contrast to reds.

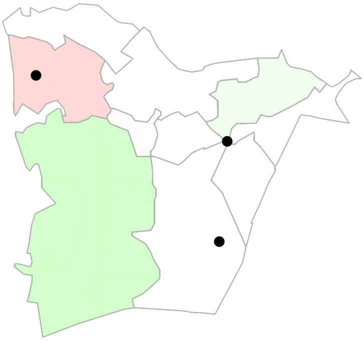


Figure 5.8 The relative improvement in late arrivals for Gooi & Vechtstreek. Green municipalities show improvement with dynamic routing, in contrast to reds. The black dots indicate the base locations.

If we focus on the per-municipality statistics, we see a shift in the number of late arrivals. Since there is a relatively low call volume in Wijdemeren, any extra on-time arrival results in a larger relative improvement compared to the more densely populated areas that have a high call volume. Figure 5.8 shows this relative improvement.

There is a relative improvement in Wijdemeren and Huizen, while dynamic routing performs worse in Weesp. Note that both Wijdemeren and Huizen do not have a base location. Wijdemeren normally has the lowest percentage on time arrivals. Dynamic routing redistributes the late arrivals so the percentage late arrivals of each municipality gets closer together.

5.4.2 Amsterdam-Waterland

Amsterdam-Waterland has a higher call volume than any other ambulance region in the Netherlands, as it counts 121,000 incidents a year. Fraction 68% of its 1.30 million inhabitants live in the city Amsterdam. This region is densely populated compared to Gooi & Vechtstreek. Figure 5.9a shows the region, its base locations and distribution of demand.

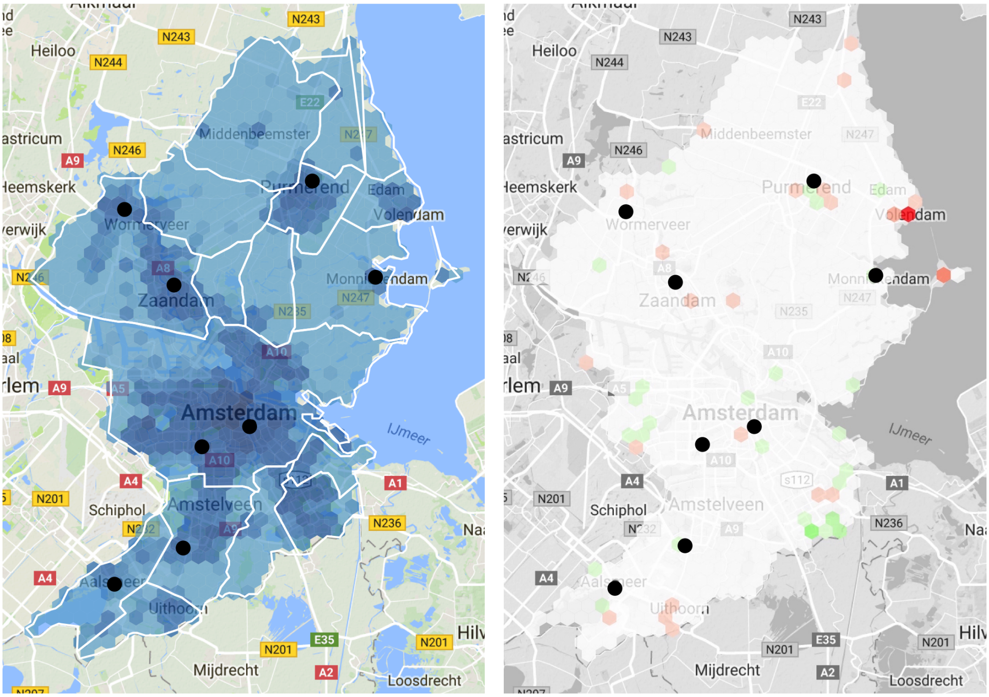
Figure 5.9b shows the difference in the number of late arrivals for both policies, where a green node indicates more on-time arrivals for the dynamic routing policy.

Recall that Table 5.1 shows that the late arrivals in the ambulance region stay about the same, but the mean response time decreases when we use dynamic routing. This is mainly because dynamic routing has the largest improvement in Amsterdam Zuid-Oost (south-east Amsterdam), shown in Figure 5.9b. This comes at the expense of the semi-rural areas outside of the city Amsterdam that get more late arrivals, especially at Volendam. Observe that Amsterdam Zuid-Oost does not have a base location for ambulances. Thus dynamic routing sends an ambulance over Amsterdam Zuid-Oost to increase coverage over that part of the region. Nowadays, there is a base located in Amsterdam Zuid-Oost.

This illustrates that dynamic routing can be used to cover an area where one would want a base location, and might even be used to search for appropriate base locations.

5.4.3 Utrecht

Utrecht is a densely populated area with approximately 1.27 million inhabitants. It is amongst the largest EMS regions in the Netherlands. The ambulance provider handles over 90,000 incidents each year. The region and its base locations are shown in Figure 5.10. We compare both methods similar to the analysis for Gooi & Vechtstreek and Amsterdam-Waterland.



(a) Darker shades of blue correspond to a higher call volume.

(b) The number of late arrivals. Green areas show improvement with dynamic routing, in contrast to reds.

Figure 5.9 Comparisons for the EMS region Amsterdam-Waterland. The base locations are indicated by black dots.

Recall that Table 5.1 shows a slight increase in the number of late arrivals for the dynamic routing policy. In Figure 5.11 we see that the most decrease happens in the cities Amersfoort and Veenendaal. There is a small decrease in the mean response time as well. This can be because the ambulances respond quicker to incidents farther away from base locations when we use dynamic routing. The mean response time for this ambulance region is slightly faster.

The most improvement is gained in semi-rural areas with no base location. Especially in Lopik in the south-west and Eemnes in the north-east we see a large increase of on-time arrivals. This is because dynamic routing relocates ambulances through these regions. However, there is a lower performance in the other corners of Utrecht. In both in the north-west and the south-east dynamic routing is outperformed by the fastest route policy. Both these regions have base locations, as opposed to Lopik and Eemnes. Thus because of dynamic routing, the ambulances arrive later at the base locations in these corners of the region, which results in more late arrivals. Hence, we see a redistribution of the late arrivals over the region, where the areas with a lower percentage on time arrivals improve.

Since we have the most improvement in more thinly populated areas, we are interested in the relative improvement of the region. Figure 5.12 shows the relative improvement for each municipality in Utrecht.

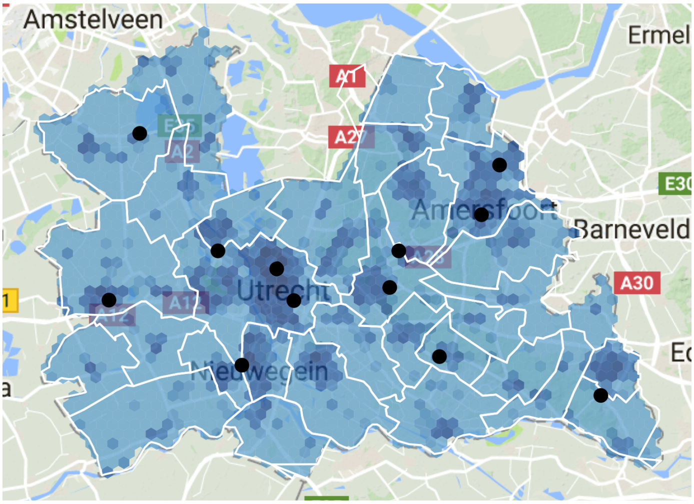


Figure 5.10 The blue area is the EMS region Utrecht. The base locations are indicated by black dots. Darker shades of blue correspond to a higher call volume.

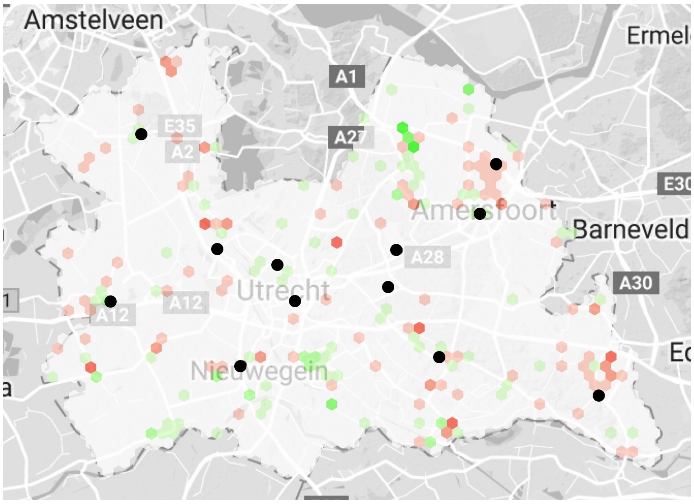


Figure 5.11 Comparison of the number of late arrivals in Utrecht. Green areas show improvement with dynamic routing, in contrast to reds. The base locations are indicated by black dots.

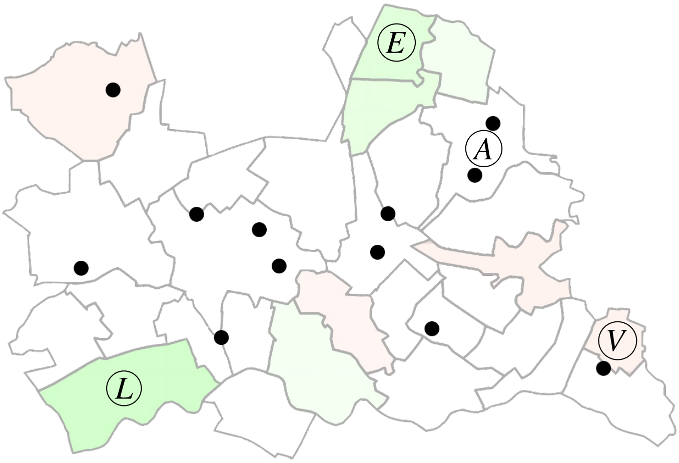


Figure 5.12 Comparison of the number of late arrivals in Utrecht. Green municipalities show improvement with dynamic routing, in contrast to reds. The base locations are indicated by black dots. Marked are Lopik (L), Eemnes (E), Amersfoort (A) and Veenendaal (V).

5.5 Conclusion

Classically ambulances are relocated to a base location using the fastest route. In this chapter we proposed a so-called dynamic routing policy that looks for the best relocation route instead. Simulation results increase the fairness of the ambulance region, while keeping the fraction of late arrivals over the entire ambulance region stable.

Simulations provide interesting additional insights.

First, in the ambulance regions Gooi & Vechtstreek and Utrecht dynamic routing gives more on-time arrivals in rural areas, at a relatively small cost in the larger cities. The large relative improvement in the urban areas and areas with no base locations shows that dynamic routing ensures a more even distribution of the ambulances.

Second, Amsterdam-Waterland shows that dynamic routing can be used to compensate for suboptimal locations of ambulance bases. Dynamic routing contributes the most in Amsterdam Zuid-Oost in regards to the number of late ambulance arrivals. This can be explained by a densely populated area without any base location. We note, however, that at the time of writing a new base location in Zuid-Oost is installed.

Third, an interesting topic for further research is to investigate the effect of the discount parameter γ which reduces the ambulance's contribution on the overall coverage value as time passes. Possibly, there is too much emphasis on the beginning of the route, which is the result of a high choice for the discount parameter. This can lead to ambulances arriving later at their destination, and potentially to unnecessary late arrivals.

Last, it is possible to restrict dynamic routing when certain restrictions are satisfied, i.e., we can only consider dynamic routing during certain times of the day. Further research is needed on the effects of such limitations.

Acknowledgements

Our thanks go to RAVU, Ambulance Amsterdam, and GGD Gooi & Vechtstreek for making the call center records available.

6

PILOT STUDY FOR DYNAMIC AMBULANCE MANAGEMENT

A promising means to reduce late arrivals is to *proactively relocate ambulances* to have 'good coverage' of the available ambulances in real time. This chapter evaluates two dynamic relocation policies that are adjusted for operational use by an ambulance service provider in the Netherlands and implemented in a software tool for real-time decision support. These policies were evaluated in the dispatch center of GGD Flevoland for a period of twelve weeks. This chapter describes the relocation methods, evaluates the pilot, provides statistics for the efficiency improvements, and discusses the experiences of ambulance dispatchers and management.

This chapter is based on the following publication:

[A6] M. van Buuren, C. J. Jagtenberg, T. C. van Barneveld, R. D. van der Mei, and S. Bhulai. "Ambulance Dispatch Center Pilots Proactive Relocation Policies to Enhance Effectiveness". *To appear in Interfaces* (2018)

6.1 Introduction

Ambulance service providers, ASPs, worldwide deal with incentives to work more efficiently. They can obtain efficiency via changes to medical equipment, staff training, and the logistics domain. In this chapter, we focus on the latter. The goal is to allocate the *right resources at the right time at the right place*, such that the probability of meeting the response time targets, within given budget constraints, is maximized.

The traditional ambulance service provisioning paradigm is *static* and *reactive*. That is, each ambulance has a fixed *base location*, from which it is dispatched in response to an incoming emergency call. When the ambulance becomes available again, it is sent back to either its home location or to service another emergency. This classic static and reactive approach to ambulance service provisioning is simple, but often highly inefficient, particularly in situations where multiple emergencies occur simultaneously, and potentially leads to coverage problems, the late arrival of ambulances, and ultimately to the loss of lives.

A promising and powerful means to boost efficiency is enforcing *proactive relocations*, i.e., to proactively relocate ambulances to locations at which they can provide a better coverage to the ambulance region compared to a simple policy such as assigning every ambulance a fixed base location to which it always returns after completing a service. In practice, one has to carefully balance the trade-off between coverage improvement and additional cost: over time, a relocation leads to additional fuel costs, and wear and tear on the ambulances. Moreover, ambulance personnel are often reluctant to making seemingly unnecessary relocations, unless they believe the relocations are absolutely necessary. That is, practitioners accept the enforcement of proactively relocations only if they improve efficiency and limit the number of relocations.

Motivated by these factors, we developed several algorithms to optimize proactive relocations; that is, we developed methods that generate suggestions to the agents (dispatchers) in the ambulance dispatch centers about *when* to relocate, *which ambulance* to relocate, and *where* to relocate that ambulance. In practice, relocation suggestions are made by simply displaying an arrow on the dispatcher's monitor; the arrow indicates which ambulance to relocate to which base location. Relocations suggestions are simply made by showing an arrow that appears on a monitor in front of a dispatcher. We emphasize that these arrows only give *suggestions* for relocations: the dispatching agent makes the final decision on whether to enforce a relocation.

To assess the practical usefulness and performance of our algorithms, we have performed a real-life pilot for a period of twelve weeks. In doing so, we adapted these dynamic relocation policies so that they comply with local regulations and ease integration to the dispatchers' daily practice. In this study we partnered with GGD Flevoland, an ambulance service provider in the Netherlands, and CityGIS Homeland Security, a company that provided us with real-time data streams and navigation software. Based on the success of this pilot, GGD Flevoland and other dispatch centers adopted our policies for ongoing use and permanent implementation.

We organized the remainder of this chapter as follows. In Section 6.2, we review the relevant literature. Outlines for two relocation policies are described in Section 6.3. Subsequently, in Section 6.4, we show how we adjust these two dynamic relocation models for use in practice. In Section 6.5, we discuss the results of the pilot, which we ran in a real EMS dispatch center to evaluate performance statistics and practitioner experiences. We provide concluding remarks and give recommendations in Section 6.6.

6.2 Literature

Static location models such as the ones presented in Section 3.2, do not explicitly consider the state of the system following events, such as a change in the availability of an ambulance (when an ambulance has been dispatched or completed servicing a patient), the arrival of an ambulance at the scene of an emergency, or the departure of an ambulance for a hospital. In contrast, relocation executed in real time is the topic of many papers in the literature on dynamic models.

In the literature on dynamic models, one can distinguish *offline* and *online* models. Offline models, which can be solved *a priori*, generate a look-up-table solution. Such a table provides a redeployment strategy for each possible system state. If the state of the system is described by the number of available ambulances, i.e., ambulances not busy with patient-related matters, such a table is called a *compliance table*. This table indicates the ideal locations for each possible number of available ambulances. Examples can be found in [13, 50], and [129].

Other offline dynamic models, not related to compliance tables, include the approximate dynamic programming (ADP) approaches proposed in [86] and in [87]. In these papers, an approximate policy iteration is run offline to search for a good value function approximation. Once such a value function is obtained, the computation of a redeployment decision is fast and can be executed in real time. The computation of relocation and dispatching decisions by ADP is also the subject in [121]. Examples of other methods include stochastic programming [91] and simulation-based optimization [25].

In online models, no precomputation is executed. Based on the system state, a relocation decision is computed in real time, without using the results of an a priori computation. The first online relocation model, based on the double standard model of [49], is proposed in [48]. A tabu search heuristic and parallel computing are used to solve the relocation problem. The notion of *preparedness* is used by [5] in the computation of redeployment decisions. Finally, our work is based on two online relocation models in [63] and [14]. These two policies have in common that they consider all redeployment options, compute a heuristic value for the benefit of each movement, and eventually propose the move with the best value. The policies differ, however, in how they define the value for a specific movement, and how they handle the statuses of ambulances.

Few of the dynamic relocation policies described above have been implemented in practice. To our knowledge only one company, Optima Corporation (recently acquired by Intermedix [60]), had implemented reposition models prior to the study we describe in this chapter. It developed a commercial package, Optima Live; however, because this package is commercial all its details are not available to us. One feature that has been published is that Optima Live uses the real-time multiple-view generalized-cover repositioning model, from [85]. It provides real-time ambulance relocation suggestions to dispatchers that maximize service quality and minimize relocation costs. Other work that the Optima Corporation supports is presented in [111] and [142]; the latter includes more details on the software.

6.3 Models

In our real-life pilot, we tested two relocation policy models in practice: (1) the *Dynamic Maximum Expected Coverage Location Problem* (DMEXCLP), and (2) the *Penalty Heuristic* (PH). This section describes the two relocation policies, and compares them.

6.3.1 The dynamic maximum expected coverage location problem

The DMEXCLP policy, proposed in [63], mandates that when a vehicle becomes idle after completing service for a patient, that vehicle goes to the base of choice within the region. Its sole objective is to maximize the number of incidents that are addressed within the time threshold R. We first describe which aspects of the current state of the ambulance system should be used as input for the policy, and then we explain how to compute the relocation decision based on this input.

At a *decision moment*, the current state of the ambulance system may be observed. Equally to Chapters 3, 4, and 5, we denote the set of base locations by $\mathcal{J}$, and the set of demand points by $\mathcal{I}$. The DMEXCLP policy disregards all information about ambulances that are busy, and it focuses purely on the set of available ambulances. As we mentioned above, it uses the *destination* of the ambulances, rather than the actual location. For ambulances that are idle at a base, the destination equals the current location. This information is captured by the variables n_j; that is the number of available ambulances that have destination j $(j \in \mathcal{J})$.

We next describe how the DMEXCLP algorithm computes the recommended relocation based on the previously described information. In some sense, we can regard this policy as a dynamic version of the maximum expected covering location problem—hence, its name. MEXCLP was designed to calculate an optimal static distribution of ambulances over base locations, by calculating the *coverage* of the region using an integer linear programming (ILP) formulation. The DMEXCLP policy reuses this definition of coverage, but it computes it for relocation purposes (without resorting to ILP solvers).

The DMEXCLP policy is a dynamic version of the MEXCLP model that is discussed in Section 3.2.2 and 5.3.2. The busy fraction q is predetermined and the same for all vehicles. Consider a demand point $i \in \mathcal{I}$, which is within the range of $k_i = \sum_{j \in \mathcal{M}_i} x_j$ ambulances (see the definition of $\mathcal{M}_i$ at page 58). Recall that the probability of at least one of these k_i ambulances is available at any point in time is then given by $1 - q^{k_i}$. If we let d_i be the demand at demand point i, the expected covered demand of this vertex is $E_i = E_i(k_i) = d_i(1 - q^{k_i})$. The MEXCLP policy positions the ambulances such that the total expected covered demand is the ambulance region is maximized.

The DMEXCLP policy proposes to send the ambulance that just became idle, to the base, such that this allocation results in the greatest coverage according to the MEXCLP model. This is equivalent to choosing the base that gives the largest *marginal* coverage over all demand. This marginal coverage for demand point i can be interpreted as the added value of having a k_i-th ambulance nearby, and is given by $C_i = E_i(k_i) - E_i(k_i - 1) = d_i(1-q)q^{k_i - 1}$. The base that gives the largest marginal coverage over the entire region, and hence the destination that DMEXCLP proposes, can be expressed as follows:

$$\text{argmax}_{j \in \mathcal{J}} \sum_{i \in \mathcal{I}} d_i (1-q) q^{\tilde{k}(i,j,n_{j_1},\dots,n_{j_{|\mathcal{J}|}})-1} \cdot \mathbb{1}_{\{t_{j,i} \leq R-PTD\}}, \tag{6.1}$$

$$\text{where } \tilde{k}(i,j,n_{j_1},\dots,n_{j_{|\mathcal{J}|}}) = \sum_{j' \in \mathcal{J}} n_{j'} \cdot \mathbb{1}_{\{t_{j',i} \leq T\}} + \mathbb{1}_{\{t_{j,i} \leq R-PTD}. \tag{6.2}$$

Recall from Section 3.3 that PTD denotes the pretrip delay. The expression for $\tilde{k}$ in Equation (6.2) simply counts the number of available ambulances that have a destination within the range of demand point i, assuming that the ambulance that is up for relocation will be sent to j. That is, it counts the number of ambulances that in the near future may respond timely to an incident at demand point i. Since the number of base locations is typically small, the maximization in Equations (6.1) and (6.2) can be computed by brute force (i.e., we iterate over all possible base locations and select the best location).

6.3.2 The penalty heuristic

The PH policy, proposed in [14], consists of two steps performed in sequential order. In Step 1, we compute the desired ambulance configuration (i.e., the distribution of the available ambulances over the different base locations). In the computation of this configuration, it uses the unpreparedness of each demand point, a reachability measure based on the response time of the closest available ambulance—that unpreparedness score. In Step 2, we calculate the actual movements of ambulances needed to reach the *desired configuration* (determined in Step 1), starting at the current configuration. The set of movements may include the use of *chain relocations* (i.e., where multiple relocations are executed simultaneously in order to achieve the desired configuration in minimal time).

Based on the observed information, the destinations of available ambulances, and the location and elapsed service time of ambulances at hospitals, an ambulance configuration minimizing the *unpreparedness* is suggested. Unpreparedness is a measure of the (in)ability to quickly respond to incoming emergency calls, based on the configuration of ambulances. We refer to [14] for a formal definition of this concept. We prefer to talk about unpreparedness instead of coverage due to the objective criterion of interest: we define a *penalty function* by assigning a specific penalty to each realized response time. Note that this induces a generalization of the coverage concept: one can incorporate the commonly used performance criterion of coverage by defining a 0-1 function. Other performance criteria, such as response time or lateness minimization, or maximization of survival probabilities, can also be incorporated.

To compute the unpreparedness level of the region, we consider for each demand point i the ambulance $a_i^{fastest}$ that can be onsite as quickly as possible ($i \in \mathcal{I}$). This ambulance could be idle, but it is also possible that none of the ambulances can respond to such an incident in a timely manner. In that case, an ambulance currently busy with the transfer of a patient at the hospital may be asked to wrap up its task and depart for the emergency scene as quickly as possible. We are only allowed to preempt if the hospital transfer time has already lasted for a substantial amount of time, e.g., ten minutes.

Let τ_i denote the expected travel time to demand point i of the ambulance that can arrive the fastest. This can either be an idle ambulance, or an ambulance that is at a hospital. Note that in the latter case we must add the remainder of a ten minute allowed transfer time to the ambulance's driving time:

$$\tau_i = t_{a_i^{fastest},i} + \max\{0,\ 10 \text{ minutes} - \text{passed transfer time}\}.$$

	DMEXCLP	PH
Uses destinations of idle vehicles.	X	X
Uses time until busy vehicle becomes idle.	-	X
Focuses solely on one response time target.	X	optional
Uses multiple coverage.	X	-
Allows relocation when vehicle becomes idle.	X	X
Allows relocation when vehicle becomes busy.	optional addition	X
Relocates multiple vehicles per decision moment.	-	X
Computes solution brute force in real time.	X	X

Table 6.1 Summary of properties for both relocation methods, and whether the property is included (X), not-included (-) or optional.

The unpreparedness is defined as the weighted sum of these τ_i. That is, $\sum_{i\in\mathcal{I}} d_i f(\tau_i)$, where d_i denotes the demand probability of point i, and $f(\tau_i)$ denotes the penalty value that corresponds to a minimal response time τ_i for demand point i.

There are two decision moments: (1) when an ambulance has just been dispatched, and (2) when an ambulance becomes available after servicing a patient. At decision moments of the first type, the ambulance configuration, which is the resulting configuration if each idle ambulance is at its destination, may be changed at *at most* one pair of base locations. That is, one base location is selected as *origin* and one as *destination*. An ambulance leaves the origin and one arrives at the destination. Using brute force, we compute the unpreparedness among all allowed configurations. For a decision moment of the second type, the origin is given. This concludes the first part of the policy.

In the second step, we compute the optimal move to obtain the desired ambulance configurations, which is based on the current location of the ambulances, not the destinations. Quickly attaining this configuration is important. Therefore, we solve a *Linear Bottleneck Assignment Problem* [96]. In this problem, one aims to find an assignment of ambulances to base locations that minimizes the maximum travel time to attain the desired configuration. Note that relocating multiple vehicles is allowed if this reduces the time until compliance. We refer to [14] for an illustration.

6.3.3 Comparison of the relocation methods

In this section, we compare the characteristics of the DMEXCLP and the PH policies that we discussed in the two preceding sections. In Table 6.1, we compare properties of the relocation methods. The main difference is that the PH provides an optimal homogeneous distribution of ambulances over the ambulance region, while the DMEXCLP focuses on multiple coverage. As a result, rural areas tend to get more ambulances with the PH, and large cities get fewer ambulances. In contrast, DMEXCLP keeps ambulances near the cities, and only provides coverage to rural areas when a sufficient number of available ambulances is available.

An additional difference is that the PH considers ambulances that are in a hospital. This slightly favors rural areas, because hospitals are often located in cities. An ambulance can be sent out of a city when another becomes idle at a hospital on short notice.

We end this section with remarks about the model assumptions of the DMEXCLP and PH policies.

Remark 1: Discretization of the area in demand points
Both the DMEXCLP and the PH policies assume that the service area is discretized and partitioned into, for example, N subareas (which are represented by the *demand points*). Thus, the next incident will occur at exactly one of these demand points; the probability that the next incident will occur at demand point i is denoted by a vector of probabilities p_i $(i = 1,\dots,N)$ that sum up to 1. Thus, the demand location is modeled by a vector $(p_1,\dots,p_N)$, which can be estimated or forecast based on historical data. For both policies, we aggregated to four-digit postal code numbers, each with an average of 4000 inhabitants, and used the normalized number of inhabitants as the demand.

Remark 2: Base locations and travel times
Both policies require information about the locations of the existing base locations, and the expected driving time between each base location and demand point. Based on discussions with ASP management, we decided to only use the base locations, and excluded relocation destinations that dispatchers used only occasionally (e.g., parking lots). Driving times between base locations and demand points were all precalculated and available in a database. We used navigation software to calculate the driving times from ambulances that were available, but not located at a base location.

Remark 3: Teleportation assumption
At a decision moment, both relocations policies use the locations of available ambulances in the coverage calculation. Some of these ambulances are typically waiting at a base location, while others are driving toward a base location. Instead of keeping track of their true locations, we assume that they are instantaneously moved to their *destination* (also referred to as the *teleportation assumption*). This choice has two important advantages. First, in a real-time system, keeping track of destinations is typically easier because these destinations change less frequently than the current location of ambulances, which reduces calculation times when driving times are stored in a memory cache. Second, there is a strategic advantage: for a moving ambulance, its current location is only relevant for a very short time, while our relocation

decision should be beneficial to the system for a longer time. Hence, using their destinations can in some sense be regarded as taking a snapshot of the future. In Chapter 5 we discussed an alternative for this teleportation assumption called dynamic routing.

6.4 Model adjustments for practice

The two relocation models described above cannot directly be used in practice. During the implementation phase we encountered a number of practical constraints that required adjustments in order to be applicable in an operational dispatch center. In this section we describe the adaptations that were made to make the models ready for use.

6.4.1 Relocation chains

We implemented a post processor for both models, such that a long relocation distance is ‘cut’ into multiple simultaneous ambulance movements, forming a *relocation chain*. Relocation chains provide a means to quickly reach at a desired configuration of an ambulance, given the current ambulance configuration. For example, consider the situation where a relocation policy determines that an ambulance must be relocated from A to C, which takes 30 minutes. If base location B, located half way through this route, also contains an ambulance, simultaneously relocating one ambulance from A to B, and another one from B to C is a better option. Using this chain, the relocation duration is decreased from 30 to 15 minutes.

In discussions with dispatchers and management, we determined that a relocation chain may contain *at most two* simultaneous ambulance movements, and that we would use such a chain only when we could ensure a relocation duration gain of at least ten minutes. When multiple relocation chains are possible, we use the one with the minimal relocation duration.

6.4.2 Adjustment series 1: Adjusting policies to the region’s workflow

In the implementation and system integration stage of the pilot, we had to make four adjustments to the original models to simplify the dispatchers’ daily work.

First, we extended the DMEXCLP policy with chain relocations similar to the PH policy, which is a fairly straightforward process.

Second, we realized that the theoretical definition of relocation and the definition used in practice were different. In the theoretical policies, each ambulance that finishes a call receives a relocation instruction back to a base location. In practice, each ambulance has a default location to which it returns when it becomes available. Because entering a relocation requires the dispatcher time and effort, and because the dispatcher will be more willing to accept the relocation module's recommendation when it is in accordance with the ASP's historical procedure (i.e., the procedure the ASP has used for many years), we changed the original policies to respect that all ambulances have historically defined default behaviors. Each shift has a default base location assigned. We assume that all available ambulances that are not actively performing a relocation must move to their default base location. For example, the D1 shift has as default base location, Dronten, and its hours are defined as 07:30 am to 16:30 pm. When an ambulance becomes available or when its shift starts, we assume that the vehicle moves to this base location. After 16:30 pm, we label this ambulance *in overwork*. Only when a dispatcher enters a relocation for an ambulance with the D1 shift into the system, the destination of that ambulance changes based on the information the dispatcher has entered.

Third, if the ambulance region is in a rural area, some shifts include a sleeping stage. During a night shift, the emergency medical technicians (EMTs) are allowed to sleep and may only be contacted when they are assigned to an incident. Hence, EMTs who are sleeping cannot be relocated, and assigning another ambulance to a base location at which they are sleeping is also prohibited. In the models, these base locations with sleeping ambulance teams are not included in the set of destinations. Base locations that are located in cities include some night shifts during which the emergency personnel must stay awake.

Fourth, we added decision moments at which a new relocation could be proposed. The DMEXCLP policy had only one decision moment: when an ambulance became available again, it should be optimally included in the fleet (i.e., optimal with respect to the objective function of the chosen policy). Discussions with the ASP management motivated us to include five additional decision moments, which we list below, to these policies. These decision points are used in both policies, but only at the moments at which relocation decisions are made.

1. Start of shift
   EMTs contact the dispatch center when they start their shifts, and a dispatcher assigns the employees to ambulances and also specifies shift codes. In practice, ambulances, EMTs, and shifts are coupled as units, and can therefore be considered as interchangeable for the purposes of our work. When the dispatcher has completed entering a new shift in the system, the relocation module is activated. In the DMEXCLP model, the ambulance to which a shift code has just been assigned will be the origin of the relocation proposal.

2. End of shift
   A shift can end in one of two ways. The most common way is when the scheduled end time of the shift is reached. In the case of overtime

(i.e., an EMT has exceeded his (her) scheduled number of working-hours), the relocation module excludes all vehicles on which the EMTs are working overtime, because they cannot be dispatched to a new emergency. Alternatively, a dispatcher can manually end a shift.

3. Ambulance dispatch
   When an ambulance is coupled to an incident, the relocation module proposes a new relocation.

4. Ambulance availability
   When an ambulance is coupled to an incident, the relocation module updates its relocation proposal.

5. Relocation entry into the system
   If an ambulance receives relocation instructions, the relocation module is updated with the new instructions. If the dispatcher follows the relocation proposal, as he (she) usually does, the system assumes that the optimal configuration has been achieved and does not provide any additional recommendations. Therefore, the relocation arrows on the dispatcher's monitor disappear. If a dispatcher makes a different relocation decision, the relocation module considers the relocation entered and generates a counter proposal (i.e., another relocation).

6. Sleep interval beginning
   When a sleep interval starts, all EMTs who are allowed to sleep go to their assigned night bases, and other EMTs who do not have permission to sleep are requested to leave all base locations where colleagues will go to bed. We model this by sending all these ambulances back to their default bases. Consequently, a new relocation recommendation is calculated to optimally redistribute the ambulances.

DMEXCLP-specific adjustment
We used the parameter value $q = 0.3$ for the busy fraction, which was a realistic value for the pilot region. Base locations that host sleeping shifts are excluded from $j \in \mathcal{J}$ in the argmax-argument of Equation (6.1), such that no relocation is recommended to a base at which people are sleeping. Ambulances that are out-of-region are not considered in calculating variable $n_i, (i \in \mathcal{I})$. At other relocation moments, we remove an ambulancefrom the system state, and calculate its relocation recommendation to be *when it would become available* at its current location. We repeat this procedure for each available ambulance, and suggest to the dispatcher the one that provides the highest contribution to the coverage. Relocation chains are formed similarly to the PH.

PH-specific adjustments

In the PH policy, bases that have at least one ambulance are not considered as destinations. In the adjusted version, we do allow a base to have a second ambulance when all bases are filled. This is computed by first teleporting available ambulances to their destination. Then, we remove base locations that reached their maximal capacity from the set of destinations to choose from. Then we look for the minimum number of ambulances at each base that remained, and remove this number from each base location that has the capacity. This way, there is at least one base with zero ambulances. We only consider ambulances on base locations with the maximum capacity as an origin for the relocation, such that a maximum ambulance spread is obtained. Finally, the PH value function on the resulting state space is calculated.

6.4.3 Adjustment series 2: Out-of-region

After the first six weeks of policy evaluation, we discussed updates that would improve the performance and could be implemented within a week. One update addressed ambulances that are outside the ambulance region.

In some cases, EMTs on an ambulance must drive a patient outside their ambulance region and for multiple hours; for example, a patient that needs basic live support (BLS) might require transportation to a hospital that has a particular medical specialization. When the ambulance becomes available again, the previous adjustment includes the current destination of the ambulance. Using coordinates to determine when the ambulance reenters its own region is difficult. Therefore, we use the navigation software to determine if a base location is within a 20-minute drive from any base location in its own region. If we do not find such a base location, we label the ambulance as *outside the region*. When the ambulance is within a 20-minute drive from any base location within its region, the ambulance is marked as *inside the region* and the relocation module is updated to show that this ambulance is available.

An out-of-region ambulance is never considered to be the closest ambulance to respond to an incident. Furthermore, such an ambulance is not counted as driving to a base location.

6.4.4 Technical details

We wrote the relocation module using the C++11-framework TIFAR, see Chapter 7. CityGIS Homeland Security provided the navigation software and the communications interface for the system state designed we for this chapter, which includes the location and status of each ambulance. This navigation software is the standard for emergency services in the Netherlands and includes all roads and travel speeds that EMS personnel use. The National Institute for Public Health and the Environment (RIVM), which also uses the navigational software, provided a look-up table for travel times between every pair of postal

Stage	Week no.	Policy
	1, . . . , 35	Implementation and system integration at the dispatch center.
1	36, 37, 38	Evaluating *Adjusted DMEXCLP 1*.
	39	Changing the policy.
2	40, 41, 42	Evaluating *Adjusted PH 1*.
	43	Fixed out-of-region ambulances.
3	44, 45, 46	Evaluating *Adjusted DMEXCLP 2*.
	47	Changing the policy.
4	48, 49, 50	Evaluating *Adjusted PH 2*

Table 6.2 Overview in what week of the year 2015 each pilot stage or switching week took place.

codes. Statistics Netherlands (CBS) provided demographic data for each postal code during the year 2013. The initial start-up of the program takes between ten to twenty seconds because of cache creation; the program usually computes run-time relocation recommendations almost instantaneously—but sometimes may take up to a few seconds if other systems also require processing time from shared resources.

6.5 Evaluation

We evaluated both policies for six weeks in the dispatch center of Flevoland, an ambulance region in the Netherlands. Table 6.2 lists the various stages. In the first stage, we tested the Adjusted DMEXCLP 1 policy for three weeks, spent a week switching policies, and then evaluated the Adjusted Penalty Heuristic 1 for three weeks. During the next week, we implemented and tested the Adjustment Series 2 for both policies. During the second half of the pilot, we evaluated three weeks of Adjusted DMEXCLP 2, one week of switching policies and one week of Adjusted Penalty Heuristic 2. In our study, we omitted data for the three weeks during which we switched policies (39, 43, and 47).

During the pilot, dispatchers were required to follow our relocation proposals, unless they had information not available to the system; examples include when an ambulance will be required for a BLS call in the near future or when a shift will ends. In 2015, no policy or operational changes were made, other than the use of the relocation decision support software.

The pilot evaluation of the dispatching policies included both quantitative and qualitative aspects, as we discuss below.

6.5.1 Quantitative analysis

We start the quantitative analysis by analyzing long-term patterns. Table 6.3 shows the results over 2015, which GGD Flevoland provided, and the preceding

Year	2015	2014	2013	2012	2011	2010
Call volume	24,136	23,337	22,459	22,427	21,521	20,884
Response time $\leq$ 15 minutes	95%	94%	94%	93%	93%	92%
High-urgency volume	13,427	11,006	9863	10,707	10,245	9534

Table 6.3 The *fraction of late arrivals* for high-urgency emergencies and the *call volume* (number of calls) over the years for ambulance service provider GGD Flevoland.

five years (see [D1]). As the table shows, the total call volume increases three to four percent per year, which is approximately the national average; the year 2013 was the only exception.

The primary performance indicator for Dutch ambulance care is the fraction of late arrivals for high-urgency calls. Measured over the calendar year, Dutch ambulance law requires that for high-urgency calls each ASP must meet a response time requirement of a fraction of 95% of the calls within 15 minutes; the timer starts when an agent at the regional EMS dispatch center answers the telephone, and stops when an ambulance arrives at the incident. Various improvements by GGD Flevoland have provided a steady increase in performance over the past years.

In the year 2014, only seven out of the 24 ambulance regions met this requirement; thus, a national average of 93% of the high-urgencies calls was met on-time (see [D1]). In 2015, the year of this pilot, GGD Flevoland met the 95% on-time criterion for the first time in its history.

GGD Flevoland provided us with a database that includes details of call records; this enabled us to calculate the performance indicators that we list in Table 6.4. Analyzing the four stages yielded the following insights.

First, in all stages, we met 96.0% to 97.3% of the high-urgency calls on-time, significantly exceeding the 95% requirement. Thus, the results we obtained during the pilot period compensated for the lower score of 94.4% achieved during the first months of the year; as we stated above, this was the first year that GGD Flevoland met the legal response time requirement.

Second, in the first stage, the dispatch center followed all of our relocation proposals, which resulted in 480 relocations in a three-week period. Historically, approximately 420 relocations are normal for this ambulance region. Based on feedback we received and follow-up discussions with the EMS management, we determined that we would have to assign a lower bound on the contributed value of a relocation (i.e., the improvement on the objective function of the chosen policy). We omitted relocations that provide a lower contributed value. Because EMTs went on strike during the final three stages, thus leading to a data transfer delay of several months, we could only directly adjust the bound after the first stage. The results show that we significantly reduced the number of relocations.

Third, we see that performance in the third and fourth stages was worse than the first two. We can explain this by the increase in high-urgency call volumes and the decrease in the number of relocations.

Finally, the number of ambulances remained almost constant over the last years six years. An additional ambulance was provided in the year 2013 only, resulting in a total of fourteen ambulances during the day. With an increase in call volume, a decrease in the number of relocations, an equal number of ambulances, and an increased fraction of on-time arrivals, we can conclude that our work resulted in more efficient relocations.

6.5.2 Qualitative analysis: End-user experiences

We observed that the dispatchers adopted our relocation proposals as much as they could. To accommodate legislation limitations or an approaching end of shift, or for another valid reason they were allowed to ignore our relocation proposals. If a dispatcher decided to enforce a relocation, it always matched our relocation proposal. In the feedback that dispatchers gave us, they mentioned that the relocation proposals often coincided with their own insight. In some instances, where our policies were counter to the dispatchers' intuition; however, when they applied our policies, they agreed these policies were better than their former ways of working (e.g., by intuition). Based on the new insights from our work, they have changed their daily routines. Other than the situations we discuss below, we are not aware of instances in which dispatchers strongly disagreed with our proposal.

At the start of the pilot, some dispatchers did not like the concept of a relocation tool, which they believed would tell them how to perform their work. Their opinions changed during their weeks of use, and the dispatchers realized that the tool was supporting them—not controlling their work. Dispatchers always

Stage	1	2	3	4
Response time $\leq$ 15 minutes	97.2%	97.3%	96.0%	96.8%
High-urgency volume	596	619	668	682
Number of relocations	480	353	360	328

Table 6.4 The three *key performance indicators* for each of the four pilot stages: the fraction of late arrivals at high-urgency calls, the number of high-urgency calls in three weeks, and the number of relocations.

had the final say in each relocation decision. At the completion of the pilot, they rated their overall user experiences as very good.

In their previous method of working, the dispatchers used a small offline relocation look-up table, which told them where the first five, of the 13 available ambulances during the day, had to be positioned. This left many degrees of freedom for the dispatcher. Our relocation tool put in place a uniform policy that depends less on human decisions, ensures that the relocation decision is not dispatcher dependent, provides good coverage, and improves communications among the ambulance team. Although during the first stage of the pilot, the EMTs told that there were too many relocations, in the later stages, because of the low number of relocations, they sometimes asked if the pilot had been terminated early.

In situations with many concurrent incidents, dispatchers know that the priority is communicating medical information to the healthcare professionals; because of this necessity to communicate, relocations have a high added value in ensuring optimal coverage. Another advantage of a relocation tool is that it reduces the time required to provide a relocation when the dispatcher is time constrained and under stress.

Using our software, the dispatchers could see the location of all available ambulances projected on one map. They were not given unnecessary information, which could cloud their ability to make decisions. It gave them a good overview of ongoing incidents and the status of the ambulances; in some situations, dispatchers were not able to determine the location and status of an ambulance prior to looking on our screen.

Ambulance care is a field where the logistic requirements change constantly. The causes include demands from local governments, agreements between neighboring ambulance regions, and new ambulance-management insights. Using a relocation tool provides an opportunity for management to modify the way that dispatchers work.

The DAM policies mentioned in this chapter leave room for improvement. Working-hour legislation dictates that the employees on 24-hour and 16-hour shifts may be working 13 hours and 9 hours, respectively. During the remainder of the time on these shifts, they must relax at their home base locations. Our implementation does not address these issues; hence, the dispatcher must ignore some relocation proposals. Only a few locations have shifts of this length.

The implementation does not ensure that an ambulance is back at its home location when its shift ends. Dispatchers using our software must always keep this in mind. A fairly straightforward solution that the dispatchers use is to instead send another ambulance that has sufficient remaining shift time and is also available at the same base location as the ambulance that the software suggests. We noticed that overtime by the ambulance teams did not increase during the pilot.

6.6 Conclusion

In this study, we put two dynamic ambulance management-policies into practice at an EMS dispatch center in Flevoland, an ambulance region in the Netherlands. We observed that the effectiveness of relocations improved when using a dynamic relocation policy, compared to previous years in which relocation algorithms where not used. One advantage we perceived is less latency, that is, the number of service calls for which the response time exceeds the threshold set, for a similar demand volume and number of relocations. The EMS region met the response time requirement of 95% within for the first time in its history in the year 2015. The results indicate that both DAM policies perform comparably.

Other advantages are (1) all dispatchers work in a consistent way, (2) relocation decisions can be made faster during busy times at the dispatch center, (3) new policies can be introduced more rapidly, (4) dispatchers have a better overview over their available vehicles, and (5) the management also prefers the use of scientifically proven policies instead of dispatcher intuition improves efficiency and enables management to provide better oversight.

Overall, the advantages strongly outweigh the disadvantages. Our implementation does not address the start and ends of shifts, or working-hour legislation; each would be an appropriate topic for further research. Methods to include the shift changes are proposed in [124]. Recently, a relocation method was developed that combines the features of the PH and DMEXCLP policies [15].

As a result of this study, multiple ASPs have adopted our policies for ongoing use and permanent implementation.

Acknowledgements

We thank the management and all dispatchers of GGD Flevoland for their enthusiastic participation in this research. We are also indebted to CityGIS for their live data streams with vehicle locations and statuses, and for use of their navigation services. We thank the Police Department of Flevoland for its support when implementing the tooling and providing access to their systems, and RIVM for providing us with the look-up table with driving times between every two postal codes in the Netherlands.

7

TIFAR SIMULATION AND COMPUTATION FRAMEWORK

In this chapter we introduce and discuss the *testing interface for ambulance research* (TIFAR) framework. The main goal of TIFAR is to ease the implementation of software programs—the so-called TIFAR refinements—that address operations research problems in the context of ambulance care. Simulation and decision support tools for researchers, EMS managers and dispatchers can be implemented in TIFAR. This chapter discusses the basic structure of TIFAR, and outlines four TIFAR refinements. Together these refinements show that TIFAR is an extremely powerful tool, which is able to address a wide range of EMS challenges.

This chapter is an extended version the following publication:

[A7] M. van Buuren, R. D. van der Mei, K. I. Aardal, and H. N. Post. "Evaluating Dynamic Dispatch Strategies for Emergency Medical Services: TIFAR Simulation Tool". *Proceedings of the 2012 Winter Simulation Conference*. Dec. 2012, 46:1–46:11

7.1 Introduction

In EMS research, we typically want to predict the effect of a certain policy change. For example, we are interested in the impact on the waiting times if we adjust the staffing policy in a dispatch center, or we want to know the difference in the number of late arrivals if we change the relocation policy.

All research questions addressed in this thesis have similarities: (1) they all apply to the logistics domain of ambulance research, and (2) they all need software (components to be programmed) to find an answer. Furthermore, our mathematical problems have many components in common, such as the ambulances and base locations. We transform each of these components into a structure that is called a *class* in computer science. The *TIFAR software framework* is a collection of classes that relate to the ambulance context, and which can relatively easily be interconnected and manually adjusted to form a fully functional computer program that fits the model and gives the required (graphical) model output. Such a computer program is called a *refinement* of the TIFAR framework.

We often resort to simulation models, as analytical models become too complex to solve for large instances. An advantage of simulation models is that they are open to a wide variety of model assumptions, and thereby, are highly flexible. The simulation models of the dispatch center (Chapter 2) and dynamic routing ambulance policy (Chapter 5) are simulation models programmed as TIFAR refinements. The structure of the operational software used in the real-life dispatch center (Chapter 6) has only a few differences with the latter model. Instead of the next 'system state' being calculated by a simulation engine, the new system state is taken from real-time third-party data resources that are available within the dispatch center. Additionally, where a relocation proposal is executed in the simulation software, the operational version only displays these proposals in the interface.

The results for most facility location and ambulance allocation models of Chapter 4 are also calculated by a specific TIFAR refinement. In order to solve the mixed integer programs, TIFAR is linked to the Coin-OR computational infrastructure [O2].

This chapter gives an overview of the TIFAR software framework. That is, a description of its classes and the relationship between them in each of the four refinements. We limit ourselves to the *basic idea* and the *basic program structure* for each refinement; a full software manual would be too extensive and falls outside the scope of this thesis.

In Section 7.2 we provide a literature overview of EMS simulation. Thereafter, we specify the inner workings of the TIFAR framework. Section 7.3 provides the classes that the refinements have in common. Next, we specify each refinement in order, starting in Section 7.4 with the road domain simulation, in Section 7.5 with the dispatch center simulation, in Section 7.6 with the facility location and allocation refinement, and in Section 7.7 with the operational branch that provides real-time advices to dispatchers. Section 7.8 concludes the chapter.

7.2 Literature

In this section we give a brief overview of the available literature on simulation models for EMS systems. An extensive review on simulation models in EMS is available in the literature [2]. A recent literature review on EMS models includes simulation models [7].

Most models, like the first EMS simulation model by Savas in 1969 [119], were designed as a DES tool for the road domain of the ambulance services, i.e., distribution or availability of ambulances. Other early work on ambulance simulation models can be found in [61, 133]. These models can be used to explore facility locations [6, 10, 52, 58, 88], ambulance allocation [21, 59], determining the number of resources needed for optimal services [44, 128], and dispatch policies of ambulances [12]. Also, simulation models are used to validate relocation policies [2, 54, 87, 105, 129, 143]. Hospital selection policies are studied in [138]. Although most models measure the performance based on a response time threshold or the average response time [2], there are simulation models that focus about the probability that the patient survives, and make use of survival curves in the objective [5, 12, 67, 77, 130]. The agent-based simulation technique can be used to model (parts of the) ambulance service processes [4, 6, 8, 137].

Two simulation packages are worth mentioning: BartSim and SIREN. BartSim is a simulation package developed in [55] for St. Johns Ambulance Service in Auckland, New Zealand, to assist in policy making. EMS vehicles have a computer-aided dispatch system that logs all call data such as travel times, treatment time and transfer time. This simulation engine is the first of its kind that uses real data for modeling calls. In this way, data does not have to be recorded manually as in previous EMS studies; Henderson and Mason state that that for the survey in [131] 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. The travel speed of the EMS vehicles is time-dependent in BartSim. The BartSim simulator works as a discrete-event simulator. SIREN is the successor of BartSim and it simulates EMS movements as well. This software package was used for both the Auckland and Melbourne areas. Currently, this simulation package is integrated into commercial packages. Some information about SIREN can be found in [55] and [84]. SIREN also includes stochastic travel times and non-homogeneous call generation, yielding an improvement of up to 9% on the previous strategy used by Melbourne. The simulation package can also dispatch more than one vehicle to the same call.

Simulation models are used as a decision and management support tool for the logistic part of the ambulance services. Henderson and Mason discuss simulation and data visualization in EMS by means of an example of a simulation model of the ambulance service in New Zealand [55]. The simulation model of Peleg and Plisking [94] use a graphical interface and draw time-dependent polygons. When each available ambulance gets repositioned into a polygon, the coverage gets maximized.

Many ambulance simulation models are tailer-made for the ambulance system of a particular country or region. Studies are available for Austria [68], Brazil [122], China [143], England [88], France [2], Italy [6], Japan [58], and Thailand [45], among others. Some simulation models differentiate between multiple ambulance types [68, 127, 130]. The helicopter ambulance services simulation is studied by [53].

Most EMS simulation models have an explicit focus on a particular question. Therefore, Pinto et al. [97] propose a generic ambulance simulator that can be adjusted for a wide variety of research questions that encountered in the literature. Various languages and tools are suitable for the implementation of ambulance simulation models, such as C++ [85, 88], Java [53], AnyLogic [6], Arena [2, 68, 97], and eM-Plant [127].

7.3 Model

The model description consists of two parts: (1) general information on the assumptions underlying TIFAR, and (2) descriptions of the classes that are frequently used.

7.3.1 Assumptions

In this section we provide assumptions and discuss choices made during the development of TIFAR. If required by the underlying model, a refinement can deviate from an assumption. In that case, it is explicitly specified in the text.

TIFAR and its refinements are programmed in ANSI C++11. This makes it easily transferable to different platforms and easy to perform maintenance.

Visual and speed simulation modes

TIFAR can run in two modes: *visual mode* and *speed simulation mode*. The only two ways in which these two modes differ are the moments on which they set the next time stamp where the (new) state of the system gets calculated, and how they perform output.

The *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 in real-time, or with a speed factor.

The *speed simulation mode* is a discrete event simulator that holds an ordered list with the end times of all currently ongoing events. For example, at the moment when an EMS vehicle departs, we can calculate the moment when the vehicle arrives. This gives a new end time. An end time enters the list when a call enters the system, when a vehicle departs from a location or when a vehicle arrives at a location. The next time at which the state of the system must be calculated then equals the first one in this ordered list. When a predetermined end time has passed, speed simulation mode terminates the program and displays statistics in the terminal. Speed simulation mode is faster than the visual mode, because the renewal loop only gets called at necessary time stamps. For both modes there is a separate main loop included in the program.

Call generation
The incident generator loop generates calls with the following properties:

1. The model time when the incident occurs, called the *start time*.

2. The location on the map where the incident occurs, called the *incident location*.

3. The *treatment time*, i.e., the time the EMS personnel must spend at the incident location.

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 or loss.

5. The *urgency* of the call, being A1, A2, or B.

6. Whether the call is declarable or EHGV. Recall that EHGV means that the patient does not require transportation to the hospital, whereas a declarable a call needs transportation of the patient.

There are two ways to generate calls: one can choose (1) to give each demand point its own interarrival time distribution, or (2) to have one distribution for the moment when a call in the ambulance region occurs and a separate distribution to determine the incident location of a call within the ambulance region. TIFAR makes use of the latter. Calls are generated according to an inhomogeneous Poisson process with known rates.

Spatial aggregation

Both for incident generation and coverage calculations there is a need for spatial aggregation. A refinement can choose to have a different aggregation for each of these purposes.

There are a couple of possible choices for generating the set of demand points:

1. Using RD-coordinates
   The Netherlands has its own Cartesian coordinate system called the *rijksdriehoekscoördinaten*, where the unit is 1 meter. When knowing the border of the ambulance region in RD-coordinates, one can generate an equidistant grid over the region, using a given granularity. Historic incidents or population can be mapped to the nearest demand points to get a spatial historic incident distribution. One can randomly select one of these demand point as the incident location. The main advantage is that an incident can happen at every location within the region, even on water. One still has to keep in mind that this location must be mapped upon the road network which might lead to major granularity errors. The main disadvantage is that rural areas may have relatively many demand points, which results in longer calculation times.

2. Using postal codes
   Each address with a mail box in the Netherlands has one postal code assigned. Such a postal code consists of four digits and two letters, for example 1011 AA. There are buildings with the same postal codes, i.e., a part of the same street can share a postal code. However, by design the combination of a postal code and house number forms a unique combination. The four digits are forming the neighborhood, while the two letters specify the location within this neighborhood. People involved in route planning in the Netherlands make the distinction between these so-called 6PP- and 4PP-postal codes. A rule of thumb states that the cumulative amount of mail for all houses with the same 6PP postal code is the quantity that a postman can hold in his hands. One can map a postal code onto RD-coordinates. The main advantage of this way of generating incident locations is that the population density is included, as one can estimate that the number of people located on each postal code is almost equal. One must however 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 have one postal code and a high potential that an EMS vehicle should be called. The disadvantage is that forests, beaches, water and highways do not have postal codes because there are no mail boxes. Still, incidents can happen at these places.

3. Using bootstrapping
   Using EMS data from the past, one knows where an incident has happened, and thus where an incident can happen again. In the Netherlands, all EMS data is stored. Incidents in the model can be generated using a bootstrap procedure from this data. The main advantage is that places where many incidents happen in reality, will be represented very accurately. The disadvantage is that there are a lot of places where no incident has happened before, but where new incidents might occur. Furthermore, new neighborhoods are not included in this way of incident generation. The use of historical EMS data can also involve privacy concerns, though if one can afford to lose some precision, the location can be made anonymous by mapping it onto the 4PP postal code.

4. Using trace-driven simulation
   When using the EMS's call record database, we know where and when every incident has happened. Using a simulation tool, we can simulate incidents with exactly the same characteristics and see how an alternative dispatch strategy performs. Comparing these statistics with the actual performance measured by EMS yields an excellent evaluation tool. An advantage is that it provides a good comparison between the simulated and actual data. However, we have the same privacy issues as with the bootstrapping procedure.

TIFAR mostly uses the postal codes method for the coverage calculation aggregation, and the trace-driven simulation for the incident generation.

Dispatching policy
When a call occurs, we have to assign an EMS vehicle to the call. TIFAR has a queue that contains all calls. We assign EMS vehicles to calls *in order of priority*: first we assign vehicles to calls with urgency A1 on a first-come-first-served basis, and if all calls with urgency A1 are served we start assigning vehicles to calls with urgency A2 on the same basis. The dispatch center always assigns the nearest (in time) available EMS vehicle to a call. Note that a similar approach has been taken in [106].

For ease of the model, it is assumed that each call is handled by exactly one EMS vehicle. Note that this assumption can easily be relaxed by having two calls simultaneously appearing at the same location.

Once an EMS vehicle has been 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 incident location than the already assigned vehicle. This assumption is based on the fact that in practice, it rarely occurs that an EMS vehicle gets assigned to another call.

When driving to an incident with urgency A1, we assume the vehicle has auditory and visual signals, and when driving to a call with urgency A2 we assume that the vehicle drives without them. Sometimes an EMS vehicle drives with these signals to a call with urgency A2, but since this rarely occurs we have not implemented this in TIFAR. When driving to a hospital, we assume that the EMS vehicle goes without auditory and visual signals and we adjust the speed accordingly. The speed at which a vehicle moves depends on the type of road it is on (e.g., highway, road in a city, road in the countryside) and whether or not auditory and visual signals are used. In reality, an EMS vehicle may drive with signals to a hospital in the case of a life-threatening situation.

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 instead of the nearest one, see [D3]. When an EMS vehicle departs from a hospital, it will head to the nearest base unless a relocation rule decides otherwise. The exception to this rule is a trace-driven simulation; then the patient is brought to the destination that is specified in the data set.

Code correctness

There are many techniques to validate the correctness of the source code [118, 141]. It is a good practice to use multiple validation techniques.

We have used the *animation* technique on many occasions. In the graphical user interface we follow the ambulances and the incidents as they are served. Many irregularities can be observed using this technique.

The operational refinement has been tested by manually entering events and checking if the relocation rules are performed correctly. This is called the *event validity* technique. Because test cases have to be developed and manually entered and checked, it is labor intensive.

Also, in the development and the shadow stages of the pilot study (see Chapter 6), the *face validity* technique was used. In this technique, specialists (dispatchers) monitor the model's outcome and report situations that are hard to explain. These situations have been investigated; sometimes this led to a bug fix, and sometimes the system differed from their original way of working, was perceived to the dispatchers as counterintuitive, but was actually correct.

For all simulation models the *traces* technique has been used. In this technique, the developer uses the debugger to closely follow each step of a given call, starting from the call initiation until the call is completed and written into the database. This procedure had been repeated for multiple calls. In the dispatch center simulation refinement, this technique was the one most used, in combination with the *trace analysis* technique which was applied to the output databases. The latter technique checks the input data and the random variables drawn (with fixed seed) against the output data. The facility location and allocation refinement also used a type of traces. For a couple of demand points, the calculation of the parameters was done manually and compared to the software. The outcomes were visually checked using the interface and compared to the real data.

For the road domain simulation refinement, we have compared the output of the visual mode against that of the speed simulation mode, which provided identical outputs. This is the so-called *comparison* technique. Also, the output of the simulation model was compared to the real data sets in order to find errors, e.g., to try to explain why strange outliers from the simulation model occurred. In trace-driven mode, the number of calls and ambulances at each given time can be compared to the real schedule to see if shift changes and incident handling always lead to an end—the so-called *historical data validation* technique.

In addition to the source code validation, various checks were performed on the *input data* sets that were provided by the ambulance service providers: the orders of the date stamps must be correct, and the fields must not be empty. Faulty rows were corrected if possible, or removed otherwise.

7.3.2 Classes

In this *class description* we provide the responsibilities and characteristic variables of the classes that are used in more than one refinement. The next sections that contain descriptions of the refinements use these model classes.

Timer

The *timer* is responsible for time-related operations. It knows the real time passed, and the current time in the model. Furthermore, the timer holds the event list for the discrete event simulators. In a discrete time model, the timer also progresses the model time to the next time step at which the new system state gets calculated. If there are no events left, or if the simulation end time is reached, the timer activates the processes that close down the program.

MySQL and PostgreSQL

TIFAR has connectors for easy access to database systems from other classes: it supports *MySQL* (and its open source fork MariaDB [O3]), and *PostgreSQL* [O6].

The MySQL connector can run in three modes: *input*, *output* and *operational*. Input is mainly used during the program initiation to get the base locations, hospital locations, schedules, incident information, and demand point information. Output is used to write completed calls to the database. For the ease of data analyses, the structure for the calls table of both the input and output tables are similar. At initiation, the output database gets the unique simulation identifier that is used for later data-analysis; each completed call in the output database is accompanied by this simulation identifier. Also at the program's initiation, the output database writes the simulator's parameters into the designated tabels, e.g., the call arrival rate during the dispatch center refinement. The third mode is only used in an operational dispatch center, and is used to read the current system state real-time from a common database.

The Postgres database connector is used by our navigation software, see the route class, to get information from datasets provided by OpenStreetMap [O5] and the Netherlands' cadastral Basisadministratie Adressen en Gebouwen (BAG) [O1].

Log entry and logbook

TIFAR provides an easy to use logging system. At each significant change to the system state a class is supposed to create a new *log entry* in the *logbook*. Each log entry also has both the real- and model times, and is written in the standard out. The logbook is used for debugging purposes only.

Schedule

The *schedule* holds the shifts for a unit. It administrates if this unit is on- or off-shift. The schedule holds a list of shift intervals. A shift interval stores at which timestamp in the week the shift starts and at timestamp in the week the shift ends. A timestamp in the week is a combination of a day name and a time. The schedule assumes a weekly repeating pattern. In the case the schedule changes from one year to the next, a shift interval has a valid from and to dates. Only between those dates the shift interval gets evaluated. An example of a shift interval is: from 2015-01-01 00:00:00 to 2015-12-31 23:59:59 the schedule must output status *on-shift* between Monday 23:00 and Tuesday 07:00. If no shift interval yields an on-shift, the schedule assumes the unit is off-shift. The schedule also holds the base where the unit starts and ends its shift. For the operational refinement the schedule is extended with sleeping times, see Section 7.7.

Vertex

A *vertex* is a data structure that represents a geographical point of interest. It holds a set of coordinates, and has many properties attached. Hospitals and base locations are represented by a vertex. Demand points are also vertices in the system, and incidents always happen on a vertex. Demand point related data, such as the neighborhood parameters in the heuristic from the facility location model, are stored as vertex properties.

Environment

The *environment* is a container that holds vertices. It can efficiently perform mass selections on the vertices, such as returning all base locations or all demand points.

Call

A *call* object acts as a data container for a particular incident. At creation, it gets all properties that we cannot influence: the vertex with the incident location, whether the ride is declarable or EHGV, and if the ride declarable the destination of the patient. Other properties include the time of arrival in the dispatch center and the durations of the CTT, CHT, TMT, and TFT (defined on page 55). The urgency of a call is also an input parameter.

During the simulation data gets written into the call. The first data is the status of the call; its value tells if it is still a so-called *future call* that waits to get released into the call takers queue, the call is being handled by the dispatch center, the call is coupled to an ambulance, or if it is completed. During execution the call object receives values for the location from where the ambulance departed, the driving time to the incident, the driving time to the destination, and what ambulance served the call. When a call gets released from an ambulance, it gets the status *completed* which marks that it can be cleaned from memory.

For the dispatch center model refinement the call class is extended, as will be discussed in Section 7.5.

Calls

The *calls* class is not to be confused with the call class. The plural form that we now discuss is a container that holds multiple *call*s and is able to perform *group operation* on them, e.g., it can return all calls that are waiting at the dispatch center, all ongoing calls, or all calls that have been completed.

A special feature of the calls class is that it can ‘clean’ itself, that is, writing all completed calls into the output database and deleting them from memory.

Incident generator

The *incident generator* creates new calls and puts them in the queue of the dispatch center. The incident generator can operate in two modes. In the first mode it generates new incidents according to a *Poisson process*. All properties of the new call are drawn from probability distributions. The second mode, called *trace-driven* simulation, does not generate new calls at random. Instead, it takes the call and its properties from a historic data set that is provided by an ASP.

When the current model time corresponds with the starting time of a future call, the incident generator sets the status of this call to queued at the dispatch center. This mechanism allows a new call to be dispatched at the right time.

Route

The *route* class is more powerful than the name suggests at first glance. It provides an API to navigation software of multiple third-parties and even contains a resolver that finds the coordinates that belong to a text-written input string. The navigation software the API supports is Transdev's routebeheersysteem (RBS, previous version), CityGIS navigator, and Open Street Map Routing Machine (OSRM). For the helicopters also the RIVM lifeliner model is implemented.

The resolver gets an input string and tries to find coordinates best-effort making use by matching the input string to two known datasets: the BAG for addresses and a list with hectometer poles for incidents that happened on a highway. For example "1098XG 123" and "Amsterdam Science Park 123" both resolve into the coordinate that corresponds to the front door of CWI. And "A10 L 1210" gets resolved to the left side (outer ring) of the A10 highway where the exit to the Middenweg starts.

When getting travel information from the route class, one has to provide values to the arguments *from*, *to*, *fastest*, *routing model* and *what*. The *from*-part can be an ambulance object, a vertex, a list with ambulances, a list with vertices, a set of coordinates, or an input string that needs resolving. The *to*-part has similar options. The *fastest* argument has a boolean value, that only when set to true, ensures that the designated data fields are holding routing information for the ambulance that can be present the fastest. The routing model specifies if a roadside navigation model or helicopter model is used. The *what*-part specifies if we only want the travel time, or also the distance and trace. Only getting the driving time is a relatively cheap operation.

Based on the arguments, selected variables get set. If ambulances are used in the from-part in the combination with the fastest, the route object knows what the closest ambulance is. Similar, when a list of vertices, each holding a hospital, is given in the to-part, the route knows what hospital can be reached the fastest. In the case the trace is requested, the route object knows the exact position of the unit at every given moment in time.

Because routing is such a all-round core functionality, the route class can be accessed from anywhere in the program.

Ambulance

An *ambulance* object (1) holds the properties of an ambulances, and (2) mimics the status behavior of the ambulance. Properties include, but are not limited to, the identification number of the ambulance, type of ambulance, its schedule, its status, the call it is currently serving (if any), the route it is currently driving (if not on a vertex) and the vertex it is currently on (if not having a route).

The ambulance can receive orders from another class. Giving it an unattended call acts as a dispatch. The ambulance is responsible for all status updates, that is having a chute, moving to the incident, treating the patient at the destination, and if applicable transporting and transferring the patient to its destination. At each status update the ambulance writes statistics in the call. At becoming available, the ambulance calls back to the dispatch center and asks to what base location it should return. The dispatch center returns a swift answer based on a default behavior that does not require extensive calculation, such as head to the nearest base or head to its default base. Also, at the moment of becoming available, the ambulances registers itself in the relocation module as becoming available. If the relocation policy provides a better destination, the relocation module directly overwrites the base to what the ambulance must return to.

A call gets released by an ambulance when it returned back at a base location, or when it was dispatched to another call.

When the schedule of an available ambulance says that an ambulance is off-shift, this ambulance returns to its default base (as defined in the schedule) and gets status off-shift. Similar, when the schedule becomes on-shift, the ambulance gets status *at base* and is considered available for dispatch.

Also common questions can be asked to an ambulance, such as if it is available for dispatch, or relocatable to a given base location.

Fleet

The *fleet* is a container that holds ambulance objects. Its purpose is to do mass operations on ambulances, or select a subset of ambulances based on a criterium. For example, upon request the fleet can return all available ambulances, or all ambulances that are currently driving to a particular base.

Dispatch center

The *dispatch center* has two functions: (1) *dispatching calls* that the incident generator provided, and (2) *providing call-back answers* from ambulances.

The dispatch center first gets all calls of the highest urgency, and dispatches them first come first served. Always the nearest (in time) available ambulance gets dispatched to a call. After all calls of the highest urgency are dispatched, the calls of the next urgency class get dispatched on the same first come first served base. We repeat this procedure until all calls are dispatched or no ambulances are available.

Ambulances can ask the dispatch center to what base they should return, or in the case no destination is known to what hospital the patient should be brought.

In the dispatch center refinement the dispatch center also acts as a container for the agents, and coordinates the telephone calls of the dispatch center.

Relocation module

The *relocation module* is the core of the dynamic ambulance management. Although it originally was part of the dispatch center class, it became its own class because of the extent of the source code.

At every decision moment an ambulance, or other class, registers the event in relocation module. Examples of relocation events are the program initiation, the manual entry of a relocation (in the operational mode), or updating the status of an ambulance: going on- or off-shift, getting inside or outside the region, being dispatch or becoming available.

The relocation policy decides if a relocation proposal must be calculated. If so, the program starts with the so-called *forced relocations*, that is for example sending ambulances to their base in the case they may sleep. Forced relocations are relocations inspired by legislation regardless of the DAM policy.

After the relocation is calculated, the programs sees if relocation chains can speed up the relocation. In the simulation mode, all relocation movements are followed. In the operational refinement, the resulting relocation proposals are only displayed in the interface.

Interface

The *interface* displays the current system state on a graphical user interface (GUI). It shows the map of the ambulance region, its bases and current position of each ambulance. Figures 7.1–7.3 (at end of the current chapter) illustrate the capabilities of the interface.

Next to an ambulance base the number of ambulances at that base and at close-by hospitals is shown. The color of the ambulance identification number indicates its status: green means available, red ambulances are responding to a call, yellow are at an incident, orange ambulances are driving to a patient's destination and purple indicates that the vehicle is at destination. Usually, off-shift ambulances are not displayed, but there is an option to show them in blue. Ambulances in overwork turn dark red, regardless of their status.

An ambulance identification number can have a text appended. ‘VWS’ means they are currently performing a relocation, and ‘Zzz’ indicates that an ambulance is currently sleeping. If an ambulance nears it end of shift, a text indicates how long the shift remains.

A relocation proposal is displayed with black arrows. In the case an arrow becomes too small to notice, a black circle appears around the base. The upper left corner contains the relocation advices in plain text to so one can differentiate between ambulances that are near to each other.

Green arrows indicate relocations being followed. Each available ambulance that drives to a base is accompanied by a green arrow. Forced relocations are indicated by orange arrows.

Routes that are currently being traveled are highlighted in blue when ambulances travel with regular speed, or red when ambulances are driving with optical and auditory signals.

TIFAR also monitors keyboard input, e.g., pressing the escape key will terminate the program. Keyboard inputs are commonly used to toggle displayed elements and texts.

Initiator
The *initiator* uses an input database to populate the entire program. At first, the demand points and their properties get filled. Next, the hospitals and base locations follow.

The input database only has the shift intervals. While loading the shifts into ambulances, the initiator tries to find an ambulance with a matching default base location, which has no other shift interval overlapping in its schedule with the one we are trying to add. If such an ambulance is found, we add the shift interval to this ambulance. If not, we create a new ambulance on this base location and assign this shift interval. This procedure ensures that limited ambulances are added to the simulation engine.

If the program runs in a trace-driven simulation mode, the historic incidents are loaded from the input database into the calls, such that gets simulated. All these incidents get the status of a future call. The incident generator pushes the call forward to the dispatch center at the right time.

Main
As is common in software engineering, the *main* class has the task to create all classes, make the simulation engine run, and correctly shut down the program when the simulation has finished. The main class also makes the links between

various classes, e.g., it tells the interface where it can find all ambulances and vertices, such that they can be displayed.

7.4 Refinement 1: Road domain simulation

The *road domain simulation* tool is the first refinement of TIFAR. The main goal of the roadside simulation refinement is to determine what the effect of policy change is on the ambulance response times. This refinement is especially suited to evaluate the effect of a change in the relocation or the dispatch policy.

Algorithm 4 outlines the structure of the road domain simulation model. The main function starts by creating the timer, the logbook, the environment, the fleet, the calls, the database connectors, the incident generator, the dispatch center, and the relocation module. Because logging and accessing the time are such basic functionalities, all classes need access to it. Although not explicitely specified in the algorithm, all classes also have full access to the functionalities of the route class.

Next, the environment gets populated by reading the demand points and their properties from the input database. Amongst the properties is the demand and the display name of the demand point. Also base locations and hospitals are loaded from the input database. Using the shift table from the input database the ambulances and their schedules are created and put into the fleet object.

If we are performing a trace-driven simulation, historical incidents that occurred during our period of interest are loaded from the input database, each being transformed into a call object. The call objects get *future call* as their status.

Based on the exact goal of the simulation, parameters are written into the output database. A parameter is the unique identification number of the simulation, accompanied by a variable name, a variable value and an optional variable unit. Examples are the region number, number of ambulances, or the arrival rate. In the data analysis phase that follows the simulation, the parameters can be used to get the simulation identifier that belongs to certain settings.

After the initiation the simulation loop is executed. Only at $t = 0$ we register the relocation trigger called *program initiation*, such that the ambulances get distributed in line with the current relocation policy.

First, the simulation engine checks for each ambulance if it should should go on- or off shift. TIFAR makes a distinction between the schedule being off shift and an ambulance having status off shift. When the schedule reaches the end time, it goes off shift. An ambulance can only go off shift when it is at its own default base. When a schedule goes off shift, the ambulance returns to its

Algorithm 4 Roadside simulation

```
create new timer
create new empty logbook
create new empty environment
create new empty fleet
create new empty calls
create new database connectors
create new incident generator
create new dispatch center
create new relocation module
bind all classes to the timer and logbook
bind all other classes where needed
load the settings for this simulation
load all bases, hospitals, demand points
load all shifts
load all historic incident into calls            ▷ only for trace-driven simulation
write the simulation parameters into the output database
start the interface                               ▷ only in visual mode
register the program initiation at the relocation module
repeat
    renew the fleet
        for each ambulance do
            check if the ambulance goes on-shift or off-shift
                if so, register the relocation trigger at the relocation module
            check if the ambulance goes inside or outside the region
                if so, register the relocation trigger at the relocation module
            check if the ambulance starts or ends a sleeping interval
                if so, send the ambulance to the base and mark it as sleeping, or
                ... mark it as awake
            check if the status of the ambulance should be updated
                if so, update the status and release the call when returning at base
                register a relocation trigger if applicable for the change of status
        end for
    renew the accident generator
        generate new calls if the list if there are no future calls   ▷ not when trace-driven
        Set all calls that enter the dispatch center at the current model time to queued
        ... at call taker
        clean the completed calls.
    renew the dispatch center
        Dispatch all calls queued at the call taker
    renew the relocation module
        empty the cache holding all relocation arrows
        first, perform all forced relocations. Put the corresponding ‘orange’ arrows
        ... in the cache.
        for each registered trigger do
            calculate the relocation proposal
            calculate the relocation chain
            put all ‘black’ relocation arrows in the cache
            perform the resulting relocations
        end for
    renew the interface
        update the ambulances in the interface
        update the texts displayed at the vertices, including the bases and hospitals
        update the texts at the calls
        update the arrows, i.e., relocation proposals, relocation arrows and
        ... forced relocations
    progress the timer to the next event time
until the simulation end time is reached or no events are left or received a manual
... closedown instruction
write all finished calls in the output database
write concluding remarks in the standard out
close the program
```

own base at the first moment it is available. If the schedule of an ambulance is off shift, it cannot be dispatched to a new call.

The next check for the ambulances is if they are inside or outside the region. Ambulances that are marked as outside the region are not taken into consideration by the relocation module while calculating a relocation proposal. We define an ambulance out-of-region when it is over 20 minutes driving from every base of the ambulance region. When an available ambulance drives inside or outside the region, a new relocation advise will be computed. The program also checks if an ambulance starts or end a sleeping shift. If the right of sleeping starts, the ambulance is marked as sleeping. Sleeping ambulances get a forced relocation to their sleeping post; all other ambulances cannot be relocated to this post when a sleeping ambulance is present and it gets emptied. When a sleeping ends, the ambulance is marked as not sleeping and a relocation trigger gets registered.

Next, the program checks if the status of the ambulance must be updated. This happens when the end of a time interval has been reached. At a status update the ambulance class registers a trigger and it updates its call, both only if applicable. The status update can depend on the call's properties, e.g., the call being declarable or EHGV. Also, the ambulance can contact the dispatch center to ask for a new destination. In the update step the ambulance can log timestamps and durations into the call object, which later find their way to the output database. A status update ends by writing the time of the next status update into the event list. Directly after a status update, the ambulance looks back if it must go off shift now, or if it can directly progress to the next status (that is, the current time period takes zero seconds).

After all ambulances received their correct status for the current model time, the calls are getting updated. In the case incidents are not taken from a historical

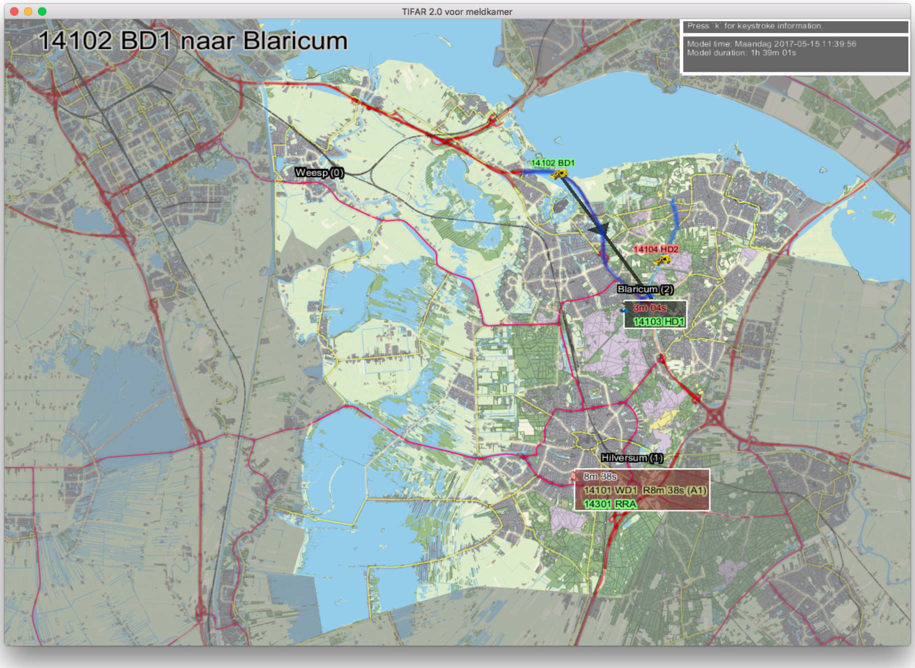


Figure 7.1 A screen shot of the road simulation domain for the region Gooi & Vechtstreek.*

dataset, new calls are generated if the number of future calls gets under a given threshold. Subsequently, each future call that should arrive at the dispatch center gets the status *queued at the call taker*. Next, the completed calls are written to the output database and removed from the calls object.

In the next step the dispatch center considers all calls that are queued at the call taker, in order of urgency. If two calls have the same urgency they are handed first come first served. In any case, the available ambulance that can be at the incident the fastest gets dispatched. At dispatch, a relocation trigger gets registered at the relocation module. Note: the dispatch center model (Chapter 2 and the next section) makes a clear distinction between urgency and priority. The current model makes only use of the output variables of the dispatch center

* Ambulance 14104 with shift HD2 responds to an A2 call in the western part of the town Huizen. Currently, the response time of this call is 3:04 minutes and counting. As a consequence of this dispatch, the available ambulance 14102 BD1 that was originally driving to base Weesp, has been requested to return to base Blaricum. In Hilversum a rapid responder ambulance is present. Also, the 14101 WD1 is in Hilversum at location of an A1 incident. Its response time was 8:38—an on-time response.

process, which are the urgency of the call and the CTT. In other words, this model does not distinct being queued at the call taker and being queued at the dispatcher.

Next, the relocation module calculates and performs the relocations. First it performs the so-called forced relocations. These are empty ambulance movements that must happen and on which the dispatch center has limited influence, e.g., the end of a shift, the labor hour legislation (in Dutch: arbeidstijdenwet [ATW]), and for the right of sleeping during the night. The relocation module calculates a relocation proposal on the resulting state space. That is, for each registered trigger it contacts the relocation module of choice (defined by a setting) and asks if a new relocation must be calculated for the trigger. If so, it provides the relocation method with the ambulance activating the trigger and the current system state (the environment and the current fleet). The relocation method returns a relocation proposal that exists of a single ambulance movement: what ambulance must be moved to what post. The chain forming method takes this output and calculates if the movement can be faster by performing multiple ambulances movement simultaneous. The resulting relocation proposal and the forced relocations are stored in a cache that is emptied every time a new relocation proposal gets calculated. The relocation module ends by performing all proposed ambulance movements, that is, these ambulances get the instruction to relocate to the specified destination base.

In the case the program runs in visual mode, the graphical user interface gets updated; see Figure 7.1. The ambulances are displayed at their new locations, the incidents are drawn, the capacity of each base location gets updated, and all relocation arrows are drawn.

The simulation loop ends by progressing the model time. In visual mode, there are two candidates for the next model time: (1) the next event time in the list, and (2) a time based on the simulation speed setting of the graphical mode. The minimum of the two is taken as the next even time, and in the first case the event time is removed from the event list. This method ensures that each event time in the event list gets evaluated, and the simulation goes not too fast for human eyes to follow. In the discrete event simulator mode that has no interface, the next event time is always taken from the event list.

7.5 Refinement 2: Dispatch center simulation

The *dispatch center simulation* refinement is the only one of the four discussed that does not use the fleet and ambulance classes. Because there are no relocations generated, the relocation module is also absent. Of the refinements discussed, this is the only one without a graphical user interface.

7.5.1 Call

The call class gets extended to hold service time information for each block: we make a distinction between the total service time and service time left, both being equal at construction. If there is no contact with the dispatcher, it holds a zero service time for that block. The call also has an extra status that tells if it is queued or attended by an agent.

Furthermore, the call stores the incoming stream and the priority that depends on this incoming stream.

7.5.2 Dispatch center agent

The dispatch center has an extra class, the so-called *dispatch center agent*. Each agent has a unique identifier and a type: nullifier, call taker, dispatcher or generalist. The nullifier is a programming technique that is used to progress tasks that require zero seconds of service, and which put no load on an agent. For example, this is the case when a block does not require contact with the dispatch center. Directly after progressing the task, the nullifier becomes available again for a next service.

Similar to how an ambulance progresses the status of a call, the dispatcher also sets the new status of a call. Every time that an agent ends a phone conversation, it logs the call identifier, waiting time, call duration, agent identifier, priority and call status in the output database such that making a detailed analysis is possible.

Note that the class definition does not make a distinction between the types of dispatch center agent.

7.5.3 Dispatch center

For the current refinement the dispatch center gets extended. To suit its additional purpose of being the dispatch center agents' container class it holds an ordered list. This ordered list first has the nullifier, then all call takers, then all dispatchers, and finally the generalists.

In the process of answering waiting calls, it iterates over the list and attaches a task to the first agent that may take its call. The dispatch center also progresses the status of a call.

7.5.4 Incident generator

The incident generator is adapted to populate the additional data fields of a call. The inter arrival times between the incidents are drawn from an exponential distribution. The incoming stream and the service times during the follow-up contact blocks are also drawn from random distributions.

7.5.5 Algorithm

The initiation of the dispatch center refinement is similar to that of the road simulation: the required objects are created and interlinked where needed.

The settings provide the number of call takers, dispatchers and generalists. Besides the nullifier, the dispatch center gets populated with the right number of agents of all classes.

The parameters are written inside the output database. In this refinement a key parameter is the arrival rate.

The simulation loop starts with the generation of new calls in the case the number of future calls are below a preset threshold. Subsequently, the status of the calls that must enter the queue at the call taker at the current model time gets their corresponding status. Additionally, each of these calls is marked as queued.

The dispatch center considers the queued calls in order of priority and arrival time. Because the first available agent in the ordered list serves a call, the generalist only receives work if there are no call takers and dispatchers available. A higher priority call can interrupt an ongoing service. In that case the remaining service time of the interrupted call gets updated, and this interrupted call gets queued again. If an agent continues with this call later on, it only serves for the duration of the remaining service time. If a service is completed, the call and agent get disconnected, after which the telephone contact gets written into the output database.

Next, the dispatch center checks for each call if the status must be progressed at the current model time. The methodology is similar to how an ambulance object progresses the status of the road domain simulation refinement. An additional feature of the dispatch center refinement is that an extra check can be included such that a status update can only happen when the remaining service time at the dispatch center equals zero.

The simulation ends when the required number of calls has been cleaned, i.e., written into the output database.

Algortihm 5 provides details on the dispatch center simulation.

7.6 Refinement 3: Facility location and allocation

The third refinement is used for the calculation of the Q-PLSCP and the AQ-HPLSCP heuristic. The structure differs from the previous two refinements, because there is no simulation involved. Therefore, this refinement does not make use of the timer, the dispatch center, the fleet, the ambulance, the calls and the incident generator classes. Instead, it extends the vertex class such that

Algorithm 5 Dispatch center simulation

```
create new timer
create new empty logbook
create new empty environment
create new empty calls
create new database connectors
create new incident generator
create new dispatch center
bind all classes to the timer and logbook
bind all other classes where needed
load the settings for this simulation
create the required number for each agent skill into the dispatch center
write the simulation parameters into the output database
repeat
    renew the accident generator
        generate new calls if the list if there are no future calls
        set all calls that enter the dispatch center at the current model time to
        ... queued at call taker
    renew the dispatch center
        clean the completed calls
        sort all calls waiting for an agent in order of priority and time of arrival.
        for each waiting call do
            if the end time of the call its current status has been reached then
                if an agent is currently servicing the call, disconnect this agent
                if the last status was a call being served at the call center then
                    update the end time of the current service as the current time plus
                    ... the length of the current interval
                    update the status of the call
                    write the new end of service time in the event list
                else
                    for each agent do
                        if the agent is available then
                            Set the agent as serving this call
                            if another call is interrupted, update its remaining
                            ... service time and set its status to queued
                            update the status of the call
                            update the end time of this service
                            write this end of service time in the event list
                        end if
                    end for
                end if
            end if
            In the case a call was interrupted, restart the for-loop
        end for
    progress the timer to the next event time
until there are no events left
write concluding remarks in the standard out
close the program
```

all demand point and base dependent variables used in the heuristic can be properly stored. Also a new connector class with COIN-OR's CBC solver is introduced.

7.6.1 Vertex

For the facility location and allocation refinement, the *vertex* class gets extended. It gets new variables for the arrival frequency, the arrival rate, the round-trip time, the mean service time in the case it is a base, the mean service time for the demand point, the reliability threshold, the density value, the minimal number of required ambulances and the number of ambulances allocated to the base. For the post-processor also additional variables are required: the lower and upper bound on the ambulances busy fraction, and in the case of a base the busy fraction of the ambulances stationed over there. Also the getters and the setters are provided. The arrival frequency, the density and the reliability threshold for each demand point are set during the initiation of the program.

7.6.2 Optimizer

The *optimizer* is the class that connects the TIFAR classes with the Coin-OR CBC framework [O2]. It takes the current environment and fleet and transforms it into the data arrays that make up the constraint matrix and the objective function's coefficients of the IP formulation, amongst others. We refer to the CBC documentation for details on the implementation of an IP program. When the problem is solved, it stores the optimal objective value in a way that it can be easily be accessed from other classes. It also stores the number of ambulances allocated to each base, by writing these number in the designated (new) variable within each vertex.

7.6.3 Algorithm

The initiation is quite similar to the previous two refinements: all required objects are created, the environment gets populated and the necessary bindings are formed. The fleet puts 1000 (or any other large enough number of) ambulances on a base location, such that the variables for the round trip time can be populated, and an improvement on the initial solution can be found. Details are given in Algorithm 6.

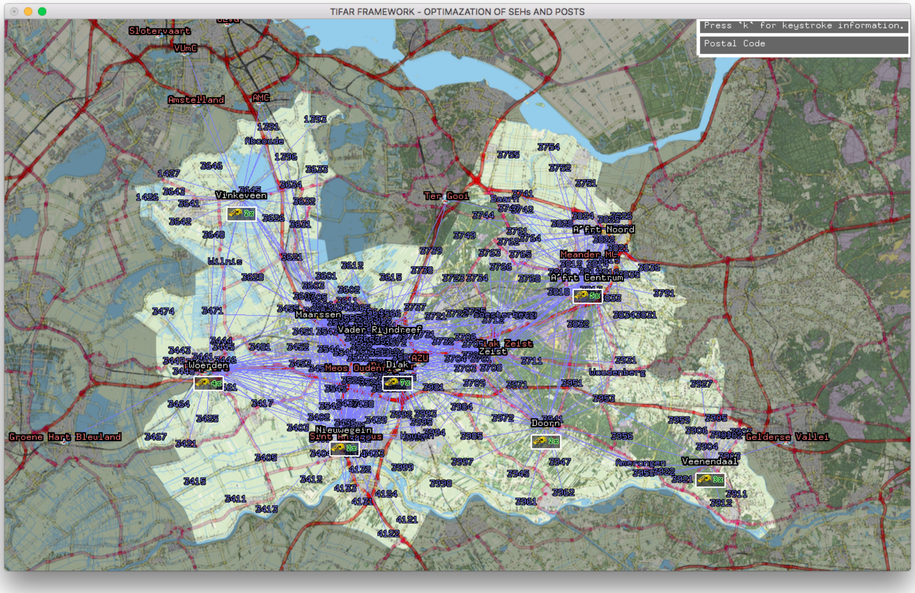
The iterations go as follows; we refer to *the mean service time paragraph* on page 102 for the details. First we calculate for each demand point i the exclusive mean service time $\tilde{\beta}_i$ to the mean service time of an ambulance serving demand point i. Next, we calculate for each base j the reasonable approximation of the mean service time $\hat{\beta}_j$. Next, we can calculate a good approximation for the mean service time β_i of the neighborhood around each demand point i.

Algorithm 6 Facility location and allocation

```
create new empty logbook
create new empty environment
create new database connectors
create new graphical user interface
bind all classes to the logbook
bind all other classes where needed
load the settings for this facility location
load the demand points with the arrival frequency and density values
write the simulation parameters into the output database
populate the fleet: put 1000 on the first base loaded in memory
calculate the arrival rate for each demand point based on a given density value
repeat
    update the mean service time from the nearest ambulance to nearest base ($\tilde{\beta}_i$)
    update the mean service time for each base ($\hat{\beta}_j$)
    update the mean service time for each demand point ($\beta_i$)
    calculate the minimal number of required ambulances for each demand point ($b_i$)
    populate the optimizer with:
        · the demand points, the base locations, and the maximal base capacities
        · the minimal number of required ambulances for each demand point
        · the driving time matrix
    Call the solver of the optimizer
    For each base, get the number of allocated ambulances from the solver and store
    ... this number in the base vertex.
    Run the post-processor
until the objective function is not improved
write the solution from the algorithm in the standard output
start the graphical user interface
update the interface: the demand points, the vertices and show the number of ambulances
... at each base
wait until the graphical user interface is closed
write concluding remarks in the standard out
close the program
```

Knowing the arrival rate, the mean service time and the reliability threshold for each demand point, we can calculate the minimal number of required ambulances b_i. Now, all input parameters for the IP formulation are known and provided to the optimizer. That is, each base location and its maximal capacity, each demand point and its minimal required number of ambulances b_i, and the driving time matrix with driving from each base location to each demand point. The optimizer transforms it into the IP formulation, and solves it. The resulting ambulance allocation gets stored in the vertices that represent the base locations.

The post-processor mentioned in Algorithm 1 is implemented in the environment class, and it raises the number of ambulances until the workload condition is satisfied.



TIFAR FRAMEWORK - OPTIMAZATION OF SEHs AND POSTS

Press `k` for keystroke information.

Postal Code

Figure 7.2 A screen shot of the allocation by the DAQ-HPLSCP with fixed bases, ALS-calls only, for the ambulance region Utrecht.*

With the new allocation, the values of $\tilde{\beta}_i$ change. Hence, we do a new iteration until there is no improvement in the number of ambulances.

If there is no better solution found, the loop ends. The best solution found is written to the standard output. At this time, the graphical user interface is started and it displays the number of ambulances allocated to each base, alongside debug information that can be toggled by keyboard strokes.

When the interface is closed, some final remarks are written inside the standard output and the program will be closed. This refinement has the option to disable the graphical user interface.

7.7 Refinement 4: Operational modus for practice

The fourth and last refinement has many similarities with the road domain simulation refinement. The main purpose of the current refinement is to provide the dispatch center agents with real-time relocation advices. The only

* The number of ambulances that results from the allocation model is shown at each base. The names of bases are in white; some are left empty. In blue, the centroid of each four position postal code—the aggregation level of this study—is mentioned. Thin blue lines show what base location can reach what demand point within the response time limit.

major changes are found in the way the state space is progressed and the way the the relocation triggers are registered. The new state of the ambulances is taken from a new class, the operational API that takes the current system state from real-time third-party operational database: specialized software from our partner CityGIS writes the current state space of all ambulances of interest in this database, which the operational API reads. We cannot give ambulances relocation instructions, because these are still manually entered by the dispatchers. Instead, we only show the relocation arrows. The timer takes the wall clock time as the current model time, but still has a (less used) event list such that schedules can go off-shift in-time.

7.7.1 Operational API

The *operational API* gets the state space data from the shared operational database and updates TIFAR's fleet and active relocation proposals accordingly. This database contains four interesting data sets: (1) The set of ambulances and their current shift name, (2) the set of ambulances that are available and their current coordinates, (3) the set of ambulances that are currently at a hospital, their coordinates, and the time they arrived at the hospital, and (4) the set of ambulances that have active relocation instructions with the coordinates of their current location and destinations.

Note that this operational API does not make a distinction between ambulances that are off shift or are serving an incident, because all these ambulances are not present in any of the datasets. Ambulances become visible again once they arrive at a hospital, or start a shift. In the interface we only show the available ambulances and the ambulances that have the *at destination* status.

7.7.2 Algorithm

The initiation goes similar to all previous refinements; the timer, container classes and database connectors are started and where necessary bound. The operational refinement does not load the ambulances, as this will be done through the operational API. The program initiation trigger is registered at the relocation module.

The program loop starts by updating the state space from the operation API. It first loads all available ambulances that have no relocation instruction, and puts them at their respective coordinates. For this ambulance, it requests the name of the shift and sets the ambulance's shift accordingly. If the coordinates of the ambulance are within a predetermined euclidean distance from its default base we set the ambulance status to *at base*, if not, we set the ambulance status to *driving to base* with the destination to its default base. A similar technique is applied for ambulances that have a relocation instruction; instead of using the default base we use the base that is defined by the destination coordinates that is taken from the operational database. The so-called hospital ambulances are set to status *at destination* at their current location. Also, their duration at the destination is calculated and set, as this is a required variable for the penalty heuristic. All ambulances that are not present in these three datasets are put off shift.

In the ambulance renewal the same checks are performed as in the road simulation refinement. That is, the booleans are set for inside or outside the region, going off shift, and sleeping. The status update loop is disabled in the current refinement.

The relocation module is also similar to the road simulation. The only change is that the ambulances do not receive any relocation instruction, as these are manually entered by the agent and received by use through the operational database.

Every time the status or the shift name of an ambulance is changed, the operational API looks if for the transition a trigger must be registered. Subsequently it registeres the necessary triggers in the relocation module.

The operational interface does not differ from the one of the road simulation refinement. Because ambulances that serve an incident are put off shift, only the available ambulances are displayed in the interface.

Algorithm 7 describes the operational modus. A photograph of the interface in the dispatch center is displayed in Figure 7.3.

Algorithm 7 Operational modus for practice

```
create new timer
create new empty logbook
create new empty environment
create new empty fleet
create new database connectors
create new dispatch center          ▷ only the call back functionalities are used.
create new operational API
create new relocation module
bind all classes to the timer and logbook
bind all other classes where needed
load the settings for this simulation
write the model parameters into the output database
start the interface
register the program initiation at the relocation module
```

```
repeat
    renew the fleet
        load changes from the API
            for each ambulance do
                check if the ambulance is already seen by the ambulance
                if not, create the ambulance
                check if the name of the shift is still correct
                if not, change the current shift and register a relocation trigger
                update the coordinates
                update the destination, in the case of a relocation
                update the time at the destination, in the case of status at hospital
                update the status. Non-available ambulances (except at hospital)
                ... are put off-shift
                register a relocation trigger if applicable for the change of status
            end for
        for each ambulance do
            check if the ambulance goes off-shift because the end-of-shift time
                if so, register the relocation trigger at the relocation module
            check if the ambulance goes inside or outside the region
                if so, register the relocation trigger at the relocation module
            check if the ambulance starts or ends a sleeping interval
                if so, mark it as sleeping, or mark it as awake
        end for
    renew the relocation module
        empty the cache holding all relocation arrows
        first, put 'orange' arrows in the cache for all forced relocations
        for each registered trigger do
            calculate the relocation proposal
            calculate the relocation chain
            put all 'black' relocation arrows in the cache
        end for
    renew the interface
        update the ambulances in the interface
        update the texts displayed at the vertices, including the bases and hospitals
        update the texts at the calls
        update the arrows, i.e., relocation proposals, relocation arrows and forced
        ... relocations.
    progress the timer to the next event time
until the simulation end time is reached or no events are left or received a manual
... closedown instruction
write all finished calls in the output database
write concluding remarks in the standard out, and close the program
```

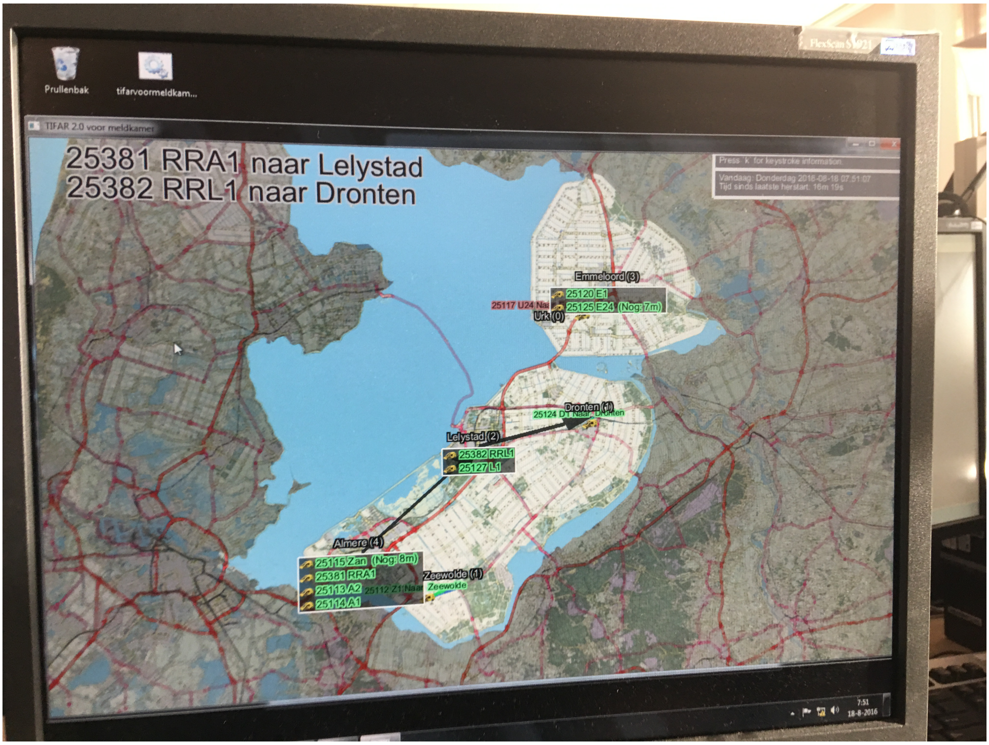


Figure 7.3 Photograph of a screen at the dispatch center in Lelystad.*

7.8 Conclusion

TIFAR is a powerful software framework that enables us to create so-called refinements. Researchers, managers and dispatch center agents can use these refinements as decision support software. After providing high-level descriptions of TIFAR's classes and assumptions, the current chapter showed the basic structure of four refinements that are used during the thesis: (1) The road romain simulator, (2) the dispatch center model, (3) the facility location and allocation model, and (4) the operational branch for the pilot study.

TIFAR allows highly detailed simulations and calculations, leading to small granularity errors. The graphical user interface can display effects of decisions

* This display shows that the photo is taken one year after the pilot study, on 2016-08-18 07:51:07AM. The ambulance 25112 drives to base Zeewolde, as is indicated by a small green arrow and its text. Each base mentions a number beside its name; this is the number of available ambulances driving to or stationed at this base. Remaining ambulance shift time is mentioned between brackets. The 25117 U24 is displayed in red because it has to return to Emmeloord because its shift ends; an orange arrow is present but barely visible. The black arrows indicate a relocation chain by two ambulances from Almere via Lelystad to Dronten. The text in the upper left corner emphasizes the names of the vehicles.

in real-time. It is easy to implement relocation rules. It is also easy to adapt models, relocation methods and other components, which make this a powerful tool. The structure of the output database allows for a broad range of statistical analysis.

Although not discussed in this thesis, many other refinements are created in the TIFAR framework during the REPRO project. In the initial data cleaning process, the bots that clean the data sets are TIFAR refinements that use the data-interface and resolver (see the route class). One of these bots enriches each address field (departure location, incident, destination) with the coordinates. Another TIFAR bot writes a travel time matrix from every base location and demand point, that the route class loads as a pre-calculated travel time matrix, which, in turn, enormously speeding up the start-up time of any TIFAR refinement. Other refinements are made to display routes, reachability, or other static images.

Many extensions on the TIFAR framework are possible in future developments. Other fields of application can be included, such as firefighters, policy, roadside assistance, and spare part distributions, amongst others. The current version of TIFAR is primarily tailered to the Dutch ambulance system design and rijksdriehoeks (RD) coordinate system. It would be a welcome extension to include the characteristics that can be found in other countries.

Acknowledgements

Thanks to Henk Post, Karen Aardal, Simon Visser, Maya Ronday, Suzanne van der Leeuw and Rob Bosman for their valuable input during the early stage of the development of the road simulation model. Also, thanks to Joost Bosman, Martijn Onderwater, Arjen de Rijke and Chris Wesseling for the countless software engineering related discussions that resulted in increased efficiency and stability of both the TIFAR source code and the database design.

BIBLIOGRAPHY

[1] M. Abkowitz, M. Lepofsky, and P. Cheng. “Selecting Criteria for Designating Hazardous Materials Highway Routes”. *Transportation Research Record* 1333 (July 1992), pp. 30–35.

[2] L. Aboueljinane, E. Sahin, Z. Jemaï, and J. C. Marty. “A Simulation Study to Improve the Performance of an Emergency Medical Service: Application to the French Val-de-Marne Department”. *Simulation Modelling Practice and Theory* 47 (Sept. 2014), pp. 46–59.

[3] H. C. Abrams, P. H. Moyer, and K. S. Dyer. “A Model of Survival from Out-of-Hospital Cardiac Arrest using the Boston EMS Arrest Registry”. *Resuscitation* 82.8 (Aug. 2011), pp. 999–1003.

[4] A. Anagnostou, A. Nouman, and S. J. E. Taylor. “Distributed Hybrid Agent-based Discrete Event Emergency Medical Services Simulation”. *Proceedings of the 2013 Winter Simulations Conference*. Dec. 2013, pp. 1625–1636.

[5] T. Andersson and P. Värbrand. “Decision Support Tools for Ambulance Dispatch and Relocation”. *Journal of the Operational Research Society. Special Issue on OR in Health* 58.2 (Feb. 2007), pp. 195–201.

[6] R. Aringhieri, G. Carello, and D. Morale. *Ambulance Location through Optimization and Simulation: The Case of Milano Urban Area*. Tech. Rep. University of Turin, Nov. 2007.

[7] R. Aringhieri, M. E. Bruni, S. Khodaparasti, and J. T. van Essen. “Emergency Medical Services and Beyond: Addressing new Challenges through a Wide Literature Review”. *Computers & Operations Research* 78 (Feb. 2017), pp. 349–368.

[8] R. Aringhieri. “An Integrated DE and AB Simulation Model for EMS Management”. *Proceedings of the Workshop on Health Care Management*. Feb. 2010, pp. 1–6.

[9] J. A. Azevedo, M. E. O. Santos Costa, J. J. E. Silvestre Madeira, and E. Q. Vieira Martins. “An Algorithm for the Ranking of Shortest Paths”. *European Journal of Operational Research* 69.1 (Aug. 1993), pp. 97–106.

[10] M. H. Azizan, C. S. Lim, W. A. Lutfi, H. W. M., T. L. Go, and S. S. Teoh. “Simulation of Emergency Medical Services Delivery Performance Based on Real Map”. *International Journal of Engineering and Technology* 5.3 (June 2013), pp. 2620–2627.

[11] M. O. Ball and F. L. Lin. “A Reliability Model Applied to Emergency Service Vehicle Location”. *Operations Research* 41.1 (Feb. 1993), pp. 18–36.

[12] D. Bandara, M. E. Mayorga, and L. A. McLay. “Priority dispatching strategies for EMS systems”. *Journal of the Operational Research Society* 65.4 (Sept. 2013), pp. 572–587.

[13] T. C. van Barneveld. “The Minimum Expected Penalty Relocation Problem for the Computation of Compliance Tables for Ambulance Vehicles”. *INFORMS Journal on Computing* 28.2 (May 2016), pp. 370–384.

[14] T. C. van Barneveld, S. Bhulai, and R. D. van der Mei. “The Effect of Ambulance Relocations on the Performance of Ambulance Service Providers”. *European Journal of Operational Research* 252.1 (July 2016), pp. 257–269.

[15] T. C. van Barneveld, C. J. Jagtenberg, S. Bhulai, and R. D. van der Mei. “Real-Time Ambulance Relocation: Assessing Real-Time Redeployment Strategies for Ambulance Relocation”. *To apprear in Socio-Economic Planning Sciences* (Nov. 2017).

[16] R. Batta, J. M. Dolan, and N. N. Krishnamurthy. “The Maximal Expected Covering Location Problem: Revisited”. *Transportation Science* 23.4 (Nov. 1989), pp. 277–287.

[17] V. Bélanger, A. Ruiz, and P. Soriano. *Recent Advances in Emergency Medical Services Management*. Manuscript. CIRRELT, 2015.

[18] M. E. Ben-Akiva, M. J. Bergman, A. J. Daly, and R. Ramaswamy. “Modelling inter-urban route choice behaviour”. *Proceedings of the Ninth International Symposium on Transportation and Traffic Theory*. Eds. J. Volmuller and R. Hamerslag. VNU Science Press, Utrecht, July 1984. Chapter 15, pp. 299–330.

[19] P. L. van den Berg and J. T. van Essen. “Comparison of Static Ambulance Location Models”. *To appear in International Journal of Logistics Systems and Management* (2018).

[20] P. L. van den Berg, J. T. van Essen, and E. J. Harderwijk. “Comparison of Static Ambulance Location Models”. *3rd IEEE International Conference on Logistics and Operations Management*. May 2016, pp. 1–10.

[21] G. N. Berlin and J. C. Liebman. “Mathematical Analysis of Emergency Ambulance Location”. *Socio-Economic Planning Sciences* 8.6 (Dec. 1974), pp. 323–328.

[22] O. Berman and D. Krass. “Chapter 11: Facility Location Problems with Stochastic Demands and Congestion”. *Facility Location: Applications and Theory*. Eds. H. Hamacher and Z. Drezner. Springer, 2002, pp. 329–371.

[23] S. Bhulai. “Dynamic Routing Policies for Multiskill Call Centers”. *Probability in the Engineering and Informational Sciences* 23.1 (Jan. 2009), pp. 101–119.

[24] G. Bianchi and R. L. Church. “A Hybrid Fleet Model for Emergency Medical Service System Design”. *Social Science & Medicine* 26.1 (Jan. 1988), pp. 163–171.

[25] R. Bjarnason, P. Tadepalli, A. Fern, and C. Niedner. “Simulation-based Optimization of Resource Placement and Emergency Response”. *Proceedings of the Twenty-First Innovative Applications of Artificial Intelligence Conference* (July 2009), pp. 47–53.

[26] M. Bliemer and P. Bovy. “Impact of Route Choice Set on Route Choice Probabilities”. *Transportation Research Record* 2076 (Dec. 2008), pp. 10–19.

[27] V. A. Bolotin. “Telephone Circuit Holding Time Distributions”. *The Fundamental Role of Teletraffic in the Evolution of Telecommunications Networks*. Eds. L. Labetoulle and J. W. Roberts. Vol. 1. Elsevier, Mar. 1994, pp. 125–134.

[28] F. Borrás and J. T. Pastor. “The Ex-post Evaluation of the Minimum Local Reliability Level: An Enhanced Probabilistic Location Set Covering Model”. *Annals of Operations Research* 111.1 (Mar. 2002), pp. 51–74.

[29] N. Bos, M. Krol, C. Veenvliet, and A. M. Plass. *Ambulance Care in Europe*. Tech. Rep. Nivel, 2015.

[30] P. H. Bovy and S. Fiorenzo-Catalano. “Stochastic Route Choice Set Generation: Behavioral and Probabilistic Foundations”. *Transportmetrica* 3.3 (Jan. 2007), pp. 173–189.

[31] L. Brotcorne, G. Laporte, and F. Semet. “Ambulance Location and Relocation Models”. *European Journal of Operational Research* 147.3 (June 2003), pp. 451–463.

[32] R. Church and C. ReVelle. “The Maximal Covering Location Problem”. *Papers of the Regional Science Association* 32.1 (Dec. 1974), pp. 101–118.

[33] L. A. Cobb, H. Alvarez, and M. K. Kopass. “A Rapid Response System for Out-Of-Hospital Cardiac Emergencies”. *Medical Clinics of North America* 60.2 (Mar. 1976), pp. 283–290.

[34] C. F. Daganzo and Y. Sheffi. “On Stochastic Models of Traffic Assignment”. *Transportation Science* 11.3 (Aug. 1977), pp. 253–274.

[35] M. S. Daskin. “A Maximum Expected Covering Location Model: Formulation, Properties and Heuristic Solution”. *Transportation Science* 17.1 (Feb. 1983), pp. 48–70.

[36] R. B. Dial. “A Probabilistic Multipath Traffic Assignment Model which Obviates Path Enumeration”. *Transportation Research* 5.2 (June 1971), pp. 83–111.

[37] R. P. Dwars. “Capacity Planning of Emergency Call Centers”. M.Sc. Thesis. VU University Amsterdam, 2013.

[38] M. J. El Sayed. “Measuring Quality in Emergency Medical Services: a Review of Clinical Performance Indicators”. *Emergency Medicine International* (Aug. 2012), pp. 1–7.

[39] E. Erkut, A. Ingolfsson, and S. Budge. “Maximum Availability/Reliability Models for Selecting Ambulance Station and Vehicle Locations: A Critique”. Self published at Researchgate. 2008.

[40] E. Erkut, A. Ingolfsson, and G. Erdoğan. “Ambulance Location for Maximum Survival”. *Naval Research Logistics* 55.1 (Feb. 2008), pp. 42–58.

[41] E. Erkut and V. Verter. “Modeling of Transport Risk for Hazardous Materials”. *Operations Research* 46.5 (Oct. 1998), pp. 625–642.

[42] C. Fisk. “Some Developments in Equilibrium Traffic Assignment”. *Transportation Research Part B: Methodological* 14.3 (Sept. 1980), pp. 243–255.

[43] J. Fitch. “Response Times: Myths, Measurement & Management.” *JEMS: a Journal of Emergency Medical Services* 30.9 (Sept. 2005), pp. 47–56.

[44] J. A. Fitzsimmons. “An Emergency Medical System Simulation Model”. *Proceedings of the 1971 Winter Simulation Conference*. Dec. 1971, pp. 18–25.

[45] O. Fujiwara, T. Makjamroen, and K. K. Gupta. “Ambulance deployment analysis: A case study of Bangkok”. *European Journal of Operational Research* 31.1 (July 1987), pp. 9–18.

[46] R. D. Galvão and C. ReVelle. “A Lagrangean Heuristic for the Maximal Covering Location Problem”. *European Journal of Operational Research* 88.1 (Jan. 1996), pp. 114–123.

[47] N. Gans, G. M. Koole, and A. Mandelbaum. “Telephone Call Centers: Tutorial, Review, and Research Prospects”. *Manufacturing & Service Operations Management* 5.2 (Apr. 2003), pp. 79–141.

[48] M. Gendreau, G. Laporte, and F. Semet. “A Dynamic Model and Parallel Tabu Search Heuristic for Real-time Ambulance Relocation”. *Parallel Computing* 27.12 (Nov. 2001), pp. 1641–1653.

[49] M. Gendreau, G. Laporte, and F. Semet. “Solving an Ambulance Location Model by Tabu Search”. *Location Science* 5.2 (Aug. 1997), pp. 75–88.

[50] M. Gendreau, G. Laporte, and F. Semet. “The Maximal Expected Coverage Relocation Problem for Emergency Vehicles”. *Journal of the Operational Research Society* 57.1 (May 2005), pp. 22–28.

[51] J. B. Goldberg. “Operations Research Models for the Deployment of Emergency Services Vehicles”. *EMS Management Journal* 1.1 (Mar. 2004), pp. 20–39.

[52] J. B. Goldberg, R. Dietrich, J. M. Chen, M. Mitwasi, T. Valenzuela, and E. Criss. “A Simulation Model for Evaluating a Set of Emergency Vehicle Base Locations: Development, Validation, and Usage”. *Socio-Economic Planning Sciences* 24.2 (Jan. 1990), pp. 125–141.

[53] E. Gunes and R. Szechtman. “A Simulation Model of a Helicopter Ambulance Service”. *Proceedings of the 2005 Winter Simulation Conference*. Winter Simulation Conference. 2005, pp. 951–957.

[54] S. I. Harewood. “Emergency Ambulance Deployment in Barbados: a Multi-objective Approach”. *Journal of the Operational Research Society* 53.2 (Feb. 2002), pp. 185–192.

[55] S. G. Henderson and A. J. Mason. “Ambulance Service Planning: Simulation and Data Visualisation”. *Operations Research and Health Care: International Series in Operations Research & Management Science*. Eds. M. L. Brandeau, F. Sainfort, and W. P. Pierskalla. Vol. 70. Springer Nature, 2005, pp. 77–102.

[56] K. Hogan and C. ReVelle. “Concepts and Applications of Backup Coverage”. *Management Science* 32.11 (Nov. 1986), pp. 1434–1444.

[57] M. Hoogeveen. *Ambulance Care in Europe*. Report. Ambulancezorg Nederland, Jan. 2010.

[58] K. Inakawa and T. F. A. Suzuki. “The Effect of Ambulance Station Locations and Number of Ambulances to the Quality of the Emergency Service”. *Ninth International Symposium on Operations Research and Its Applications*. Aug. 2010, pp. 340–347.

[59] A. Ingolfsson, E. Erkut, and S. Budge. “Simulation of Single Start Station for Edmonton EMS”. *Journal of the Operational Research Society* 54.7 (June 2003), pp. 736–746.

[60] Intermedix. *Intermedix Acquires Optima Corporation, Continues Global Expansion*. Ed. L. Osborne. Apr. 16, 2014. URL: `https://www.intermedix.com/news/intermedix-acquires-optima-corporation-continues-global-expansion`.

[61] W. H. Iskander. “Simulation Modeling For Emergency Medical Service Systems”. *Proceedings of the 1989 Winter Simulation Conference*. Dec. 1989.

[62] A. A. Jagers and E. A. van Doorn. “On the Continued Erlang Loss Function”. *Operations Research Letters* 5.1 (June 1986), pp. 43–46.

[63] C. J. Jagtenberg, S. Bhulai, and R. D. van der Mei. “An Efficient Heuristic for Real-time Ambulance Redeployment”. *Operations Research for Health Care* 4 (Mar. 2015), pp. 27–35.

[64] C. J. Jagtenberg. “Efficiency and Fairness in Ambulance Planning”. Ph.D. Thesis. VU University Amsterdam, 2016.

[65] J. P. Jarvis. “Approximating the Equilibrium Behavior of Multi-Server Loss Systems”. *Management Science* 31.2 (Feb. 1985), pp. 235–239.

[66] R. B. O. Kerkkamp. “Facility Location Models in Emergency Medical Service: Robustness and Approximations”. M.Sc. Thesis. Delft University of Technology, May 28, 2014.

[67] V. A. Knight, P. R. Harper, and L. Smith. “Ambulance Allocation for Maximal Survival with Heterogeneous Outcome Measures”. *Omega* 40.6 (Dec. 2012), pp. 918–926.

[68] O. Koch and H. Weigl. “Traffic and Road Planning Simulation: Modeling Ambulance Service of the Austrian Red Cross”. *Proceedings of the 2013 Winter Simulations Conference*. Dec. 2013, pp. 1701–1706.

[69] K. L. Koenig. “Quo Vadis: Scoop and Run, Stay and Treat, or Treat and Street?” *Academic Emergency Medicine* 2.6 (June 1995), pp. 477–479.

[70] G. J. Kommer. *Origins of the Dutch Key Performance Indicators*. Personal communication. Amsterdam, Oct. 12, 2017.

[71] G. M. Koole and A. Mandelbaum. “Queueing Models of Call Centers: An Introduction”. *Annals of Operations Research* 113.1-4 (July 2002), pp. 41–59.

[72] G. M. Koole and A. Pot. “An Overview of Routing and Staffing Algorithms in Multi-skill Customer Contact Centers”. Self published at Researchgate. 2006.

[73] E. Kozan and N. Mesken. “A Simulation Model for Emergency Centres”. *Proceedings of the International Congress on Modelling and Simulation. Advances and Applications for Management and Decision Making*. Dec. 2005, pp. 2602–2608.

[74] M. P. Larsen, M. S. Eisenberg, R. O. Cummins, and A. P. Hallstrom. “Predicting Survival from Out-of-Hospital Cardiac Arrest: A Graphic Model”. *Annals of Emergency Medicine* 22.11 (Nov. 1993), pp. 1652–1658.

[75] R. C. Larson. “A Hypercube Queuing Model for Facility Location and Redistricting in Urban Emergency Services”. *Computers & Operations Research* 1.1 (Mar. 1974), pp. 67–95.

[76] R. C. Larson. “Approximating the Performance of Urban Emergency Service Systems”. *Operations Research* 23.5 (Oct. 1975), pp. 845–868.

[77] S. Lee. “The Role of Preparedness in Ambulance Dispatching”. *Journal of the Operational Research Society* 62.10 (Nov. 2010), pp. 1888–1897.

[78] X. Li, Z. Zhao, X. Zhu, and T. Wyatt. “Covering Models and Optimization Techniques for Emergency Response Facility Location and Planning: A Review”. *Mathematical Methods of Operations Research* 74.3 (July 2011), pp. 281–310.

[79] G. de Luca. “Time Delay to Treatment and Mortality in Primary Angioplasty for Acute Myocardial Infarction: Every Minute of Delay Counts”. *Circulation* 109.10 (Mar. 2004), pp. 1223–1225.

[80] M. B. Mandell. “Covering Models for Two-tiered Emergency Medical Services Systems”. *Location Science* 6.1–4 (May 1998), pp. 355–368.

[81] V. Marianov and C. ReVelle. “Chapter 10: Siting Emergency Services”. *Facility Location*. Ed. Z. Drezner. Springer Nature, 1995, pp. 199–223.

[82] V. Marianov and C. ReVelle. “The Queueing Maximal availability location problem: A Model for the Siting of Emergency Vehicles”. *European Journal of Operational Research* 93.1 (Aug. 1996), pp. 110–120.

[83] V. Marianov and C. Revelle. “The Queuing Probabilistic Location Set Covering Problem and some Extensions”. *Socio-Economic Planning Sciences* 28.3 (Jan. 1994), pp. 167–178.

[84] A. J. Mason. “Siren: An Ambulance Simulation System”. *Mathematical Models for Optimizing Transportation Services*. University of Auckland, Apr. 2005.

[85] A. J. Mason. “Simulation and Real-Time Optimised Relocation for Improving Ambulance Operations”. *Handbook of Healthcare Operations Management*. Ed. B. T. Denton. Springer Nature, Jan. 2013, pp. 289–317.

[86] M. S. Maxwell, S. G. Henderson, and H. Topaloglu. “Ambulance Redeployment: An Approximate Dynamic Programming Approach”. *Proceedings of the 2009 Winter Simulation Conference (WSC)*. Dec. 2009, pp. 1850–1860.

[87] M. S. Maxwell, M. Restrepo, S. G. Henderson, and H. Topaloglu. “Approximate Dynamic Programming for Ambulance Redeployment”. *INFORMS Journal on Computing* 22.2 (May 2010), pp. 266–281.

[88] R. McCormack and G. Coates. “A Simulation Model to Enable the Optimization of Ambulance Fleet Allocation and Base Station Location for Increased Patient Survival”. *European Journal of Operational Research* 247.1 (Nov. 2015), pp. 294–309.

[89] L. A. McLay and M. E. Mayorga. “Evaluating Emergency Medical Service Performance Measures”. *Health Care Management Science* 13.2 (Aug. 2009), pp. 124–136.

[90] L. Moore. “Measuring Quality and Effectiveness of Prehospital EMS”. *Official Journal of the National Association of EMS Physicians and the National Association of State EMS Directors* 3.4 (Nov. 1999), pp. 325–331.

[91] J. Naoum-Sawaya and S. Elhedhli. “A Stochastic Optimization Model for Real-Time Ambulance Redeployment”. *Computers & Operations Research* 40.8 (Aug. 2013), pp. 1972–1978.

[92] C. D. Newgard, R. H. Schmicker, J. R. Hedges, J. P. Trickett, D. P. Davis, E. M. Bulger, T. P. Aufderheide, J. P. Minei, J. S. Hata, K. D. Gubler, T. B. Brown, J.-D. Yelle, B. Bardarson, and G. Nichol. “Emergency Medical Services Intervals and Survival in Trauma: Assessment of the “Golden Hour” in a North American Prospective Cohort”. *Annals of Emergency Medicine* 55.3 (Mar. 2010), pp. 235–246.

[93] O. A. Nielsen. “A Stochastic Transit Assignment Model Considering Differences in Passengers Utility Functions”. *Transportation Research Part B: Methodological* 34.5 (June 2000), pp. 377–402.

[94] K. Peleg and J. S. Pliskin. “A Geographic Information System Simulation Model of EMS: Reducing Ambulance Response Time”. *The American Journal of Emergency Medicine* 22.3 (May 2004), pp. 164–170.

[95] R. W. Petri. “The Effect of Prehospital Transport Time on the Mortality from Traumatic Injury”. *Prehospital and Disaster Medicine* 10.1 (Dec. 1995).

[96] U. Pferschy. “Solution Methods and Computational Investigations for the Linear Bottleneck Assignment Problem”. *Computing* 59.3 (Sept. 1997), pp. 237–258.

[97] L. R. Pinto, P. M. S. Silva, and T. P. Young. “A Generic Method to Develop Simulation Models for Ambulance Systems”. *Simulation Modelling Practice and Theory* 51 (Feb. 2015), pp. 170–183.

[98] W. B. Powell and Y. Sheffi. “The Convergence of Equilibrium Algorithms with Predetermined Step Sizes”. *Transportation Science* 16.1 (Feb. 1982), pp. 45–55.

[99] J. N. Prashker and B. Shlomo. “Route Choice Models Used in the Stochastic User Equilibrium Problem: A Review”. *Transport Reviews* (4 July 2004), pp. 437–463.

[100] C. G. Prato and S. Bekhor. “Modeling Route Choice Behavior: How Relevant is the Composition of Choice Set?” *Transportation Research Record* 2003 (Jan. 2007), pp. 64–73.

[101] L. Price. “Treating the Clock and not the Patient: Ambulance Response Times and Risk”. *Quality & Safety in Health Care* (May 5, 2006), pp. 127–130.

[102] J. Puts. “Emergency Call Center: Finding a Balance Between Costs and Quality of Service when Dealing with Emergency Calls”. M.Sc. Thesis. VU University Amsterdam, 2011.

[103] H. K. Rajagopalan, F. E. Vergara, C. Saydam, and J. Xiao. “Developing Effective Meta-heuristics for a Probabilistic Location model via Experimental Design”. *European Journal of Operational Research* 177.1 (Feb. 2007), pp. 83–101.

[104] M. S. Ramming. “Network Knowledge and Route Choice”. Ph.D. Thesis. MIT, Feb. 2002.

[105] J. F. Repede and J. J. Bernardo. “Developing and Validating a Decision Support System for Locating Emergency Medical Vehicles in Louisville, Kentucky”. *European Journal of Operational Research* 75.3 (June 1994), pp. 567–581.

[106] M. Restrepo. “Computational Methods for Static Allocation and Real-time Redeployment of Ambulances”. Ph.D. Thesis. Cornell University, 2008.

[107] M. Reuter-Oppermann, P. L. van den Berg, and J. L. Vile. “Logistics for Emergency Medical Service systems”. *Health Systems* 6.3 (Feb. 2017), pp. 187–208.

[108] C. ReVelle and K. Hogan. “A Reliability-Constrained Siting Model with Local Estimates of Busy Fractions”. *Environment and Planning B: Planning and Design* 15.2 (June 1988), pp. 143–152.

[109] C. ReVelle and K. Hogan. “The Maximum Availability Location Problem”. *Transportation Science* 23.3 (Aug. 1989), pp. 192–200.

[110] C. ReVelle and K. Hogan. “The Maximum Reliability Location Problem and α-reliable p-center Problem: Derivatives of the Probabilistic Location Set Covering Problem”. *Annals of Operations Research* 18.1 (Dec. 1989), pp. 155–173.

[111] D. P. Richards. “A Study of Optimised Ambulance Redeployment Strategies”. *The Operation Research Society of New Zealand Annual Conference*. Dec. 2006, pp. 123–132.

[112] N. Rieser-Schüssler, M. Balmer, and K. W. Axhausen. “Route choice sets for very high-resolution data”. *Transportmetrica A: Transport Science* 9.9 (Oct. 2013), pp. 825–845.

[113] E. Ross. “Simulation Analysis of Toronto Emergency Medical Service’s Communications Centre”. B.Sc. Thesis. University of Toronto, 2007.

[114] D. Roubos and S. Bhulai. “Approximate Dynamic Programming Techniques for Skill-based Routing in Call Centers”. *Probability in the Engineering and Informational Sciences* 26.4 (July 2012), pp. 581–591.

[115] F. F. Saccomanno and A. Y.-W. Chan. “Economic Evaluation of Routing Strategies for Hazardous Road Shipments.” *Transportation Research Record* 1020 (1985), pp. 12–18.

[116] J. S. Sampalis, A. Lavoie, J. I. Williams, D. S. Mulder, and M. Kalina. “Impact of On-site care, Prehospital Time, and Level of In-Hospital Care on Survival in Severely Injured Patients”. *The Journal of Trauma* 34.2 (Feb. 1993), pp. 252–261.

[117] R. Sánchez-Mangas, A. García-Ferrrer, A. De Juan, and A. M. Arroyo. “The Probability of Death in Road Traffic Accidents. How Important is a Quick Medical Response?” *Accident Analysis & Prevention* 42.4 (July 2010), pp. 1048–1056.

[118] R. G. Sargent. “Verification and Validation of Simulation Models”. *Proceedings of the 2011 Winter Simulation Conference*. Dec. 2011, pp. 183–198.

[119] E. S. Savas. “Simulation and Cost-Effectiveness Analysis of New York’s Emergency Ambulance Service”. *Management Science* 15.12 (Aug. 1969), pp. 608–627.

[120] D. Schilling, D. J. Elzinga, J. Cohon, R. Church, and C. ReVelle. “The Team/Fleet Models for Simultaneous Facility and Equipment Siting”. *Transportation Science* 13.2 (May 1979), pp. 163–175.

[121] V. Schmid. “Solving the Dynamic Ambulance Relocation and Dispatching Problem using Approximate Dynamic Programming”. *European Journal of Operational Research* 219.3 (June 2012), pp. 611–621.

[122] P. M. S. Silva and L. R. Pinto. “Emergency Medical Systems Analysis by Simulation and Optimization”. *Proceedings of the 2010 Winter Simulation Conference*. Dec. 2010, pp. 2422–2432.

[123] R. A. Sivakumar, R. Batta, and M. H. Karwan. “A Multiple Route Conditional Risk Model For Transporting Hazardous Materials”. *Information Systems and Operational Research* 33.1 (Feb. 1995), pp. 20–33.

[124] A. E. Sloof. “Optimization of Ambulance Care: Working shifts and Dispatching”. M.Sc. Thesis. Leiden University, Aug. 12, 2016.

[125] D. R. Smith and W. Whitt. “Resource Sharing for Efficiency in Traffic Systems”. *Bell System Technical Journal* 60.1 (Jan. 1981), pp. 39–55.

[126] P. Sorensen and R. Church. “Integrating Expected Coverage and Local Reliability for Emergency Medical Services Location Problems”. *Socio-Economic Planning Sciences* 44.1 (Mar. 2010), pp. 8–18.

[127] S. Su and C.-L. Shih. “Modeling an Emergency Medical Services System using Computer Simulation”. *International Journal of Medical Informatics* 72.1-3 (Dec. 2003), pp. 57–72.

[128] S. Su and C.-L. Shih. “Resource Reallocation in an Emergency Medical Service System using Computer Simulation”. *The American Journal of Emergency Medicine* 20.7 (Nov. 2002), pp. 627–634.

[129] K. Sudtachat, M. E. Mayorga, and L. A. McLay. “A Nested-Compliance Table Policy for Emergency Medical Service Systems under Relocation”. *Omega* 58 (Jan. 2016), pp. 154–168.

[130] K. Sudtachat, M. E. Mayorga, and L. A. McLay. “Recommendations for Dispatching Emergency Vehicles under Multitiered Response via Simulation”. *International Transactions in Operational Research* 21.4 (Mar. 2014), pp. 581–617.

[131] C. Swoveland, D. Uyeno, I. Vertinsky, and R. Vickson. “Ambulance Location: A Probabilistic Enumeration Approach”. *Management Science* 20.4 (Dec. 1973), pp. 686–698.

[132] C. Toregas, R. Swain, C. ReVelle, and L. Bergman. “The Location of Emergency Service Facilities”. *Operations Research* 19.6 (Oct. 1971), pp. 1363–1373.

[133] P. Trudeau, J.-M. Rousseau, J. A. Ferland, and J. Choquette. “An Operations Research Approach for the Planning and Operation of an Ambulance Service”. *Information Systems and Operational Research* 27.1 (Jan. 1989), pp. 95–113.

[134] T. D. Valenzuela, D. J. Roe, S. Cretin, D. W. Spaite, and M. P. Larsen. "Estimating Effectiveness of Cardiac Arrest Interventions: a Logistic Regression Survival Model". *Circulation* 96.10 (Nov. 1997), pp. 3308–3313.

[135] R. B. Vukmir. "Survival from Prehospital Cardiac Arrest is Critically Dependent upon Response Time". *Resuscitation* 69.2 (May 2006), pp. 229–234.

[136] R. A. Waalewijn, R. de Vos, J. G. Tijssen, and R. W. Koster. "Survival Models for Out-of-hospital Cardiopulmonary Resuscitation from the Perspectives of the Bystander, the First Responder, and the Paramedic". *Resuscitation* 51.2 (Nov. 2001), pp. 113–122.

[137] Y. Wang, K. L. Luangkesorn, and L. Shuman. "Modeling Emergency Medical Response to a Mass Casualty Incident using Agent Based Simulation". *Socio-Economic Planning Sciences* 46.4 (Dec. 2012), pp. 281–290.

[138] R. L. Wears and C. N. Winton. "Simulation Modeling of Prehospital Trauma Care". *Proceedings of the 1993 Winter Simulation Conference*. Dec. 1993, pp. 1216–1224.

[139] R. D. White, B. R. Asplin, T. F. Bugliosi, and D. G. Hankins. "High Discharge Survival Rate After Out-of-Hospital Ventricular Fibrillation With Rapid Defibrillation by Police and Paramedics". *Annals of Emergency Medicine* 28.5 (Nov. 1996), pp. 480–485.

[140] World Health Organization. *Emergency Medical Services Systems in the European Union*. Rapport. World Health Organization, 2008.

[141] X. Xiang, R. Kennedy, G. Madey, and Cabaniss. "Verification and Validation of Agent-based Simulation Models". *Agent-Directed Simulation Conference*. Apr. 2005, pp. 47–55.

[142] L. Zhang. "Simulation Optimisation and Markov Models for Dynamic Ambulance Redeployment". Ph.D. Thesis. University of Auckland, 2012.

[143] L. Zhen, K. Wang, H. Hu, and D. Chang. "A Simulation Optimization Framework for Ambulance Deployment and Relocation Problems". *Computers & Industrial Engineering* 72 (June 2014), pp. 12–23.

[144] B. Zuzáková. "Optimal Emergency Medical Service System Design". M.Sc. Thesis. VU University Amsterdam, May 21, 2012.

IN DUTCH

[D1] I. Boers. *Ambulances In-zicht 2014*. Vereniging Ambulancezorg Nederland, 2015.

[D2] I. Boers, P. Duijf, E. Grummels, G. Leerkes, and H. van der Werff. *Ambulances In-zicht 2012*. Vereniging Ambulancezorg Nederland, May 2013.

[D3] A. Groen. *Preklinische Traumazorg*. Dec. 2009.

[D4] A. Klink. “Onderzoeksrapport ‘Ambulance A1 Spoedritten: Wat is de Relatie tussen Responstijden en Gezondheidswinst?’ [Kamerstuk CZ-EKZ-2876149]”. *Letters of Government* (Oct. 6, 2008).

[D5] G. J. Kommer and L. Zwakhals. *Modellen Referentiekader Ambulancezorg: Ontwikkeling van Modellen voor Spreiding en Capaciteit*. Rapport. RIVM, May 23, 2013. 171 pp.

[D6] Ministerie van Volksgezondheid, Welzijn en Sport. *Tijdelijke Wet Ambulancezorg*. Den Haag, Apr. 26, 2012. URL: `http://wetten.overheid.nl/BWBR0031557`.

[D7] E. I. Schippers. “Tijdelijke bepalingen over de ambulancezorg (Tijdelijke wet ambulancezorg) [Kamerstuk 32854, nr. 16]”. *Letters of Government* (June 27, 2013).

ONLINE RESOURCES

[O1] Basisadministaties Adressen en Gebouwen. *(Accessed: 2018-01-21)*. `http://www.kadaster.nl/bag` (In Dutch).

[O2] Coin-OR Foundation. *(Accessed: 2016-03-31)*. `http://www.coin-or.org`.

[O3] MariaDB. *(Accessed: 2018-01-22)*. `http://www.mariadb.com`.

[O4] Open Source Routing Machine. *(Accessed: 2018-01-21)*. `http://project-osrm.org`.

[O5] OpenStreetMap. *(Accessed: 2018-01-21)*. `http://www.openstreetmap.org`.

[O6] PostgreSQL. *(Accessed: 2018-01-22)*. `http://www.postgresql.org`.

PUBLICATIONS BY THE AUTHOR

[A1] M. van Buuren, G. J. Kommer, R. D. van der Mei, and S. Bhulai. "A Simulation Model for Emergency Medical Services Call Centers". *Proceedings of the 2015 Winter Simulation Conference*. Dec. 2015, pp. 844–855.

[A2] M. van Buuren, G. J. Kommer, R. D. van der Mei, and S. Bhulai. "EMS Call Center Models With and Without Function Differentiation: A Comparison". *Operations Research for Health Care* 12 (Mar. 2017), pp. 16–28.

[A3] M. van Buuren, R. D. van der Mei, and S. Bhulai. "Demand-point Constrained EMS Vehicle Allocation Problems for Regions with Both Urban and Rural Areas". *To appear in Operations Research for Health Care* (2018).

[A4] M. van Buuren, R. D. van der Mei, and S. Bhulai. "Ambulance Allocation in a Mixed Region with Guaranteed Performance at Every Urban and Rural Area". *Submitted* (2018).

[A5] M. van Buuren, C. W. P. Huibers, and R. D. van der Mei. "On-route Coverage by Available Ambulances Through Route Optimization". *Submitted* (2018).

[A6] M. van Buuren, C. J. Jagtenberg, T. C. van Barneveld, R. D. van der Mei, and S. Bhulai. "Ambulance Dispatch Center Pilots Proactive Relocation Policies to Enhance Effectiveness". *To appear in Interfaces* (2018).

[A7] M. van Buuren, R. D. van der Mei, K. I. Aardal, and H. N. Post. "Evaluating Dynamic Dispatch Strategies for Emergency Medical Services: TIFAR Simulation Tool". *Proceedings of the 2012 Winter Simulation Conference*. Dec. 2012, 46:1–46:11.

ABBREVIATIONS

Throughout the thesis a consistent set of abbreviations is used, as listed below.

Variable	**Description**
Ambulance Terminology	
ASP	*Ambulance service provider* An organization that is responsible for ambulance care in a specific geographical region.
CPR	*Cardiopulmonary resuscitation* The name of a particular reanimation protocol.
DAM	*Dynamic ambulance management* The process of distributing available ambulances over the regions to achieve a higher coverage.
ED	*Emergency department* The hospital department where ambulances usually bring their patients.
EMT	*Emergency medical technician* Someone who works on an ambulance vehicle.
EMS	*Emergency medical services* Ambulance services in general.
GP	*General practitioner* A family doctor.
TI	*Telephone information* Metadata from a ASP's telephone recording system.
Vehicle Types	
ALS	*Advanced life support* Focuses on patients that are in a life-threatening situation.
BLS	*Basic life support* Focuses on stable patients that cannot move with regular taxis.
RR	*Rapid responder* An ALS vehicle that can usually move faster through traffic but has no transportation capacity.

Variable	Description
Modeling Terminology	
ADP	*Approximate dynamic programming* A technique to solve large discrete time stochastic control problems.
DES	*Discrete event simulation* A simulation technique in which time jumps from one event to the next event, without evaluating the system state between the events.
DSS	*Decision support software* Supporting software that helps professionals in decision making in complex situations.
KPI	*Key performance indicator* A mean performance measure of a system.
MDP	*Markov decision problem* A mathematical framework for modeling and solving stochastic discrete time processes.
OR	*Operations research* The field of mathematics that focusses on optimization of processes.
TIFAR	*Testing interface for ambulance research* A software framework dedicated to simulation and optimization problems in ambulance care, which is widely used in this thesis.
Dispatch Center Terminology	
C / CT	*Call taker related* A variable or queue decorator that reflects a property of a call taker.
D / Disp	*Dispatcher related* A variable or queue decorator that reflects a property of a dispatcher.
FD	*Function differentiation* A policy for a dispatch center staffed with call center agents of types call taker and dispatcher, but not with generalists.
SG	*Solely generalists* A policy for a dispatch center staffed with generalists, but not with call takers and dispatchers.

Variable	Description
Selected Models	
Model Classes	
RCLP	*Regional coverage location problem* The class of optimization problems that aggregate an ambulance region's performance into one single number.
MR-MA	*Minimal reliability and maximum availability* The class of optimization problems where the performance is evaluated for each demand point individually.
min-rel	*Minimal reliability* A model class that minimizes the required number of ambulances such that each demand point is covered.
max-av	*Maximal availability* A model class that maximizes the covered demand with a limited number of ambulances.
Well-Known Models	
MEXCLP	*Maximum Expected Coverage Location Problem* [35] A model that allocates a limited number of ambulances such that the expected demand is maximized.
MCLP	*Maximal Coverage Location Problem* [32] A single coverage facility location problem that maximizes the demand covered with limited capacity.
BACOP	*Backup Coverage Problem* [56] A double coverage facility location problem. An extension also weighs single covered demand.
DSM	*Double Standard Model* [49] A double coverage facility location problem that forces a given fraction of the demand covered by one ambulance.
TEAM	*Tandem Equipment Allocation Model* [120] A single coverage model with two types of vehicles. A demand point is covered if both vehicle types can reach it in time.
FLEET	*Facility-Location, Equipment-Emplacement Technique* [120] The TEAM model with a limit on the number of bases.
TTM	*Two-tiered Model* [80] The TEAM model with a generalized objective function.

Variable	Description
MOFLEET	*Multiple-cover, One-unit FLEET Problem* [24] The FLEET model with a MEXCLP objective function.
TIMEXCLP	*The MEXCLP with Time Variation* [105] A time-dependent MEXCLP model.
SCLP	*Set Covering Location Problem* [132] A single coverage facility location problem that minimizes the required number of ambulances to cover all demand.
MALP	*Maximum Availability Location Problem* [109] A binomial based max-av model.
Rep-P	*Reliability Perspective* [11] A binary-valued min-rel model.
PLSCP	*Probability Location Set Coverage Problem* [110] A binomial based min-rel model.
Queuing Related Models	
Q-PLSCP	*Queuing Probability Location Set coverage Problem* [83] A queuing based min-rel model.
Q-MALP	*Queuing Maximum Availability Location Problem* [82] A queuing based min-rel model.
AQ-MIPLSCP	*Adjusted Queuing Mixed Integer Probability Location Set Coverage Problem* An adjusted queuing mixed-integer min-rel model, that is proposed in Section 4.2.
Q-. . .	*Queueing prefix* A queuing approach to an MR-MA model.
AQ-. . .	*Adjusted queuing prefix* An adjusted queuing approach to a MR-MA model, as is proposed in this thesis.
FAQ-. . .	*Frequency adjusted queueing prefix* AQ with the demand point's frequency as density function.
DAQ-. . .	*Density adjusted queueing prefix* AQ with the demand point's population density as density function.

Variable	Description
Specific for the Netherlands	
EHGV	*Eerste hulp geen vervoer* A treatment provided at the incident location, but no patient transport required.
RAV	*Regionale ambulancevoorziening* Collaboration of ambulance service providers within one region.
RD	*Rijksdriehoekcoördinaat* The coordinate system in use in the Netherlands government organizations.
VWS	*Voorwaardescheppende rit* A relocation instruction to an ambulance.
BAG	*Basisadministraties adressen en gebouwen* A cadastral data set containing extensive (geographical) information on every building in the Netherlands. [O1]
Urgencies	
A1	An urgent call with an acute threat to the patient's life.
A2	A call where the patient's life is not under direct threat, but there might be serious injuries.
B	An ordered transportation: the patient must be transported within a given predetermined time interval.

VARIABLE LISTING

Throughout the thesis a consistent set of variables is used, as listed below.

Variable	**Description**
Chapter 2	
$n_{dispatcher}$	Number of dispatchers in the system.
$n_{calltaker}$	Number of call takers in the system.
$n_{generalists}$	Number of generalists in the system.
n	Total number of call center agents in the system.
α_1, r_1	Fraction of the calls that are picked up by the call taker (or generalist) within at most r_1 time units.
$\alpha_2(u), r_2(u)$	Fraction $u \in \mathcal{U}$ of the honored calls that have a call center time at most $r_2(u)$ time units.
M_1	Number of priority queues at the call taker of the dispatch center.
M_2	Number of priority queues at the dispatcher of the dispatch center.
$Q_1^{CT}, \ldots, Q_{M_1}^{CT}$	Priority queues of the call taker, decreasing in priority.
$Q_1^{D}, \ldots, Q_{M_2}^{D}$	Priority queues of the dispatcher, decreasing in priority.
ph	The probability that a call is honored.
pa	Probability that an applicant needs additional instructions.
B	Number of blocks in the block sequence.
$qf_b^m(u)$	Probability that a feedback contact occurs with urgency u, priority queue m and block b.
$pf_b(u)$	Probability that no feedback occurs in block b for ugency u.
$pe_b(u)$	Probability that another block follows after block b for urgency u. Alternatively, an end of call occurs.
$pj_b^{b'}(u)$	Probability that block b' directly follows after block b for urgency u (jumping probability).
λ_i	Arrival rate at incoming stream i.
f_k	Arrival rate for applicant class k.
Chapter 3	
q	Busy fraction of an ambulance, a system-wide constant variable.
d_i	Demand at demand point i, that is a proxy for the fraction of the total workload that is aggregated i.
E_i	Expected covered demand at demand point i.
α_i	Probability that an ambulance is available to reach a patient at i within R time units must be at least α_i (minimal required reliability level).

Variable	Description
Chapter 3 (cont'd)	
$\mathcal{H}$	Set of hospitals.
$\mathcal{I}$	Set of demand points.
$\mathcal{J}$	Set of (potential) base locations.
$\mathcal{V}$	Set that contains all hospitals, demand points and potential base locations.
$\mathcal{U}$	Set of urgencies.
t_{v_1,v_2}	The travel time from v_1 to v_2 $(v_1, v_2 \in V)$.
$r_{i,j}$	Response time from demand point i from base j.
R	Response time threshold.
Z	Total number of ambulances in the system (only for min-rel).
x_j	Number of ambulances assigned to base j.
y_i	Number of ambulances that are positioned at bases in neighborhood $\mathcal{M}_i$ (definition $\mathcal{M}_i$ follows).
f_i	Arrival frequency at demand point i.
β_i	Average service time at demand point i.
n_i	Number of ambulances that serve demand at i, which is solution dependent.
ρ_i	Workload that is generated at demand point i.
$b_i(\rho)$	Minimal required number of ambulances at workload ρ at demand point i.
$\mathcal{A}$	Set of all ambulances.
$\mathcal{A}_i$	Set of ambulances that may serve demand at i.
$q^{low}(\rho, n)$	Lower bound on the busy fraction for an ambulance at workload ρ and n serving ambulances.
$q^{*,low}(\rho)$	Lower bound on the busy fraction for an ambulance at workload ρ in our solution.
${}^{a}q^{dum}$	Dummy busy fraction of an ambulance: a virtual ambulance busy fraction that after solving the program is within some bounds to the actual value.
$q^{upp}(\rho, n)$	Upper bound on the busy fraction for an ambulance at workload ρ and n serving ambulances.
$q^{*max}(\rho)$	Upper bound on the busy fraction for an ambulance at workload ρ in our solution.
O_i	Workload that ambulances stationed at $\mathcal{M}_i$ can serve outside neighborhood $\mathcal{N}_i$ (overcapacity).
ρ_i^{upp}	Workload at the time when b_i ambulances work that hard that exactly the allowed reliability level α_i is met.
$\rho_i^{*,upp}$	Workload at the time when n_i ambulances work that hard that exactly the allowed reliability level α_i is met.
$\mathfrak{p}_{\mathcal{N}_i}$	The population of neighborhood $\mathcal{N}_i$.
$\mathfrak{a}_{\mathcal{N}_i}$	The area of neighborhood $\mathcal{N}_i$ in square meters.
ψ_i	A density assigned to i.
$\hat{\mathcal{I}}_i$	An ordered list of all demand points, using densities ψ_i.

Variable	Description
The following variables $\mathcal{N}_i$, $\mathcal{N}_j$, $\mathcal{M}_i$, b_i and λ_i may be augmented by Bin, Erl, Q, AQ, FAQ, DAQ. The decoration specifies the calculation method of the variable's value.	
$\mathcal{N}_\imath$	(Set of demand points in the) neighborhood of $\imath \in \mathcal{I} \cup \mathcal{J}$, which depend on a given response time threshold.
$\mathcal{M}_i$	Set of base locations in the neighborhood of i, which depend on a given response time threshold.
b_i	The number of ambulances that must at least be assigned to neighborhood $\mathcal{M}_i$.
$\lambda_i(\psi)$	Arrival rate at demand point i given density measure ψ.
Chapter 4	
The notation is a continuation of Chapter 3.	
$\bar{y}_{ai}$	Binary variable equal to one if and only if ambulance a provides coverage to demand point i.
$\bar{x}_{aj}$	Binary variable equal to one if and only if ambulance a is allocated at base location j.
g_a	Binary variable equal to one if and only if ambulance a is allocated to a base location.
C_j	Capacity for base location j: the maximum number of ambulances that can be allocated to j.
$b_i(\rho)$	The minimal required number of ambulances at workload ρ at demand point i.
b_i	Short notation for $b_i(\rho_i)$.
$\bar{\beta}_{h,i,j}$	The round trip time (without the driving to base time) for a trip from j to i to h.
$\tilde{\beta}_i$	An approximation for the mean service time at i.
$\hat{\beta}_j$	An approximation for the mean service time of an ambulance stationed at j.
Chapter 5	
q	Busy fraction of an ambulance. A system-wide constant variable.
d_i	The demand of demand point i, that is a proxy for the fraction of the total workload that is aggregated to i.
R	The response time threshold.
O	Origin of a route, i.e., its starting point.
D	Destination of a route, i.e., where it ends.
r	A route.
$\mathcal{R}$	A routeset: a set that contains multiple routes.
$\mathcal{L}$	Set of waypoints.
$\mathcal{Z}$	Set of decision points on the road network.
$\mathcal{Z}_{O,D}$	A subset of $\mathcal{Z}$ that can be regarded as in between O and D.

Variable	Description
Chapter 5 (cont'd)	
$\mathcal{W}$	The set containing all demand points, base locations and waypoints.
Δ	Maximum allowed snapping distance threshold between two waypoints at snapping.
S	Maximum allowed snapping distance threshold between the pivot and the waypoint at snapping.
δ	The maximum allowed fraction of overlap between two routes, in order to add an route candidate into a routeset.
$\tilde{t}_\ell$	Cumulative travel time from the origin to waypoint ℓ.
τ	Variable of time, usually used in the context of integration.
γ	Time discount parameter.
Ξ_r	Coverage value of route r.
$\xi_{r,i}$	Contribution of the coverage value of route r to demand point i.
C_i	(Marginal) contribution if demand point i receives coverage by the moving ambulance.
Chapter 6	
E_i	Expected covered demand at demand point i.
n_j	The number of ambulances positioned at base $j \in \mathcal{J}$.
n_i	The number of idle ambulances with destination $i \in \mathcal{I}$.
t_i	The response time to $i \in \mathcal{I}$ of a given ambulance.
$f(\tau)$	The penalty value corresponding to response time τ.
$\mathcal{A}^-$	The set of available ambulances that are currently not considered for relocation.
a_i	An ambulance that is stationed at a hospital, and can be fasted on-site at demand point i.
Chapter 7	
The notation is a continuation of Chapter 4.	

SUMMARY

Mathematical models can contribute substantially to improve the service quality of ambulance care. This Ph.D. thesis entails various new models, which are not only interesting from a scientific point of view, but which have also widely found their way into practice.

The *dispatch center model*, as presented in Chapter 2, is a simulation model that describes the processes of the dispatch center agents in detail. Although the literature on general dispatch centers is diverse and extensive, it appears that specific models for ambulance dispatch centers are scarce. Characteristic for dispatch centers is that there are very strict requirements on the response times and a large diversity of tasks. Unique in our approach is that we have included various feedback moments, respect the priorities of the incoming call type, and that we can simulate the multiple dispatch centers staffing policies. More specifically, our model allows for both the function differentiation and the only generalist policies. In (complete) *function differentiation* there is a group of specialized centralists called the *call takers* who answer incoming emergency calls, and who perform the *triage* procedure. Another group of specialists are responsible for the coordination of the fleet and the communication to third-parties such as hospitals; these are the so-called *dispatchers*. Generalists are flexible because they can do both the call taking and dispatching, but they are more expensive. The mixed model allows for a function differentiation policy that is supplemented with generalists. The dispatch center model, in all its forms, predicts the waiting and sojourn times, given the number of operators of each specialism. Based on the model results a cost–benefit analysis can be carried out. Using meta-data of telephone conversations by the dispatch center in Utrecht, we have been able to make good parameter estimations. We conclude that depending on the workload of the dispatch center, the optimal staffing policy changes: for low arrival rate, generalists are the most cost-efficient, but as this rate increases, function differentiation pays off. On top of the function differentiation policy, it is beneficial to have one generalist who can assist when the workload peaks, either in the call taking or the dispatching.

Chapters 3 and 4 present the so-called *adjusted queuing* framework. Primarily, ambulance service providers are assessed on the fraction of in-time ambulance arrivals for (potentially) life-threatening situations. For example, a fraction of 94% of the high urgency calls of an ambulance region has a response time of at most 15 minutes. In this calculation, the annual performance of the entire

ambulance region is aggregated into a single number, i.e., into one point in space and time. The disadvantage of this approach is that local interests are barely represented; after all, it pays off to concentrate all ambulances around the urban areas that have the majority of the demand, and as a consequence, take the relatively small amount of late ambulance arrivals in the rural areas for granted. *Minimal-reliability* (min-rel) models approach the performance differently. They calculate the minimal number of ambulances required, such that a given response time performance can be achieved at any subarea in the region. The related *maximal availability* (max-av) models try to place a limited number of ambulances, such that as many potential patients as possible are served within the given performance target. Existing min-rel and max-av models have a number of drawbacks. Many models have a system-wide busy fraction that is the equal for every ambulance, which is unrealistic as ambulances in rural areas are more at the base locations than urban ambulances. In previously published models, the arrival rate, the mean service time and the minimal required reliability threshold may not differ too much between neighboring (demand) points in the ambulance region; if that happens, it results in a higher number of required ambulances. Our approach, on the other hand, allows the aforementioned three parameters to differ for every demand point, and on top of that, in our approach the busy fraction can have a different value for each ambulance. In Chapter 3 we explain why the earlier models give an overestimation, and we show how these models have to be adapted such that they can be applied to the mixed regions that have both urban and rural demand. This results in the so-called *adjusted queuing framework*.

Chapter 4 introduces two new min-rel models that use the adjusted queuing framework from the previous chapter. The first model formulates a *mixed integer program* that finds the absolute optimum for the minimum number of ambulances, such that the required performance is achieved at every point in the ambulance region. A drawback of this model is that it is only applicable for relatively small model instances, because of the complexity and the calculation times. The second model is a heuristic that can provide solutions for larger model instances. We evaluate these models on the basis of real ambulance regions that contain a mixture of both rural and urban subareas. Numerical calculations show that (and how) we can meet the given reliability requirements for each point with fewer ambulances than in the previous models. The results of Chapter 4 are in line with the numbers that are used in practice. Consequently, these are the first min-rel models that can be used in practice.

Chapter 5 focuses on an operational issue, the so-called *dynamic routing*. In addition to achieving the response time of as many emergency calls as possible within the 15 minutes threshold, ambulance services also want each municipality to score well: if a municipality scores very badly, ambulance service providers have to give a good explanation to the regional politicians.

During a relocation, an available ambulance is requested to move to a specific location for the benefit of providing an improved coverage. Various models for dynamic ambulance management (DAM) already exist to determine what vehicle should drive to what location. According to what route this ambulance has to drive to its destination such that it can provide an *as good as possible* coverage while driving, is still an open question in the literature. This might contribute to the performance of (parts of) the region, because the ambulance generates coverage during the driving to the relocation destination station, in particular over the areas it passes. By requiring an ambulance to drive a given route, it may take a little longer for the ambulance to arrive at its destination, but in the meantime it provides a better coverage of (subareas of) the region. In this chapter a method is developed to calculate good alternative routes for ambulances. Then we show to assign a coverage value to a route. By calculating alternative routes for each route to the destination and giving each route a coverage value, an efficient route to the destination can be given to the ambulance. We show that timeliness is distributed more fairly across the region, while the number of journeys that arrive on time does not change. Most local performance gain is achieved in places that are difficult to reach from a base location. Dynamic routing is much faster to implement and cheaper to realize as a solution than building a new base location.

Chapter 6 describes a *pilot study*, which we performed to evaluate DAM policies in practice. In collaboration with ambulance service provider GGD Flevoland, two policies that have been developed within the REPRO research project have been evaluated over a period of twelve weeks. For this study both policies have been adjusted to be able to apply the policies in practice. Furthermore, software has been created and made available for operational use in the dispatch center. During the pilot the fraction of late ambulance arrivals was decreased by a third, making it the first year that the ambulance region reached the national standard of responding in 95% of the high urgency calls within 15 minutes. We also observed qualitative advantages that we had not previously predicted: centralists take faster relocation decisions (especially when there is a high workload in the dispatch center), and the software enables the dispatchers to have a better overview of the available ambulances.

The *testing interface for ambulance research* software, TIFAR, is a software framework that focuses on calculating the implications of decisions in the logistic processes within ambulance care—though simulation and optimization. In Chapter 7 we give an overview of the structure of TIFAR. This can be used to evaluate the consequences of policy changes, such as moving base locations, adjustments in personnel planning, or a replacement of the relocation policy. TIFAR contains classes that can accurately address the behavior of ambulances and the dispatch center, and it has easy-to-use connectors to various databases that contain information about the ambulance region, including information about schedules and historical call record data. Through a link with the Coin-OR optimization software, mathematical models can also be implemented in TIFAR that calculate the optimal base locations. The results of Chapters 2, 4 and 5 were obtained using TIFAR's simulation engine or calculated using its classes and database infrastructure. Furthermore, this framework was used to implement the software of the pilot study as described in Chapter 6.

SAMENVATTING

Ambulancezorg is een toepassingsgebied waaraan wiskundige optimalisatiemodellen substantieel kunnen bijdragen. Dit proefschrift behandelt diverse nieuwe modellen, die niet alleen vanuit een wetenschappelijk oogpunt interessant zijn, maar die ook grotendeels hun weg gevonden hebben naar de praktijk.

Het *meldkamermodel*, dat in Hoofdstuk 2 behandeld wordt, is een simulatiemodel dat in detail de processen van de meldkamer-centralisten nabootst. Hoewel de literatuur over algemene meldkamers divers en omvangrijk is, blijkt dat specifieke modellen voor ambulancemeldkamers nagenoeg afwezig zijn. Meldkamers ambulancezorg (MKA's) onderscheiden zich door zeer strikte eisen aan de responstijden en een grote diversiteit aan taken. Uniek in onze benadering is dat de diverse feedback-momenten in kaart worden gebracht, de prioriteiten van de verschillende soorten telefoongesprekken gerespecteerd worden en vooral dat er meerdere manieren zijn waarin de simulatiemodellen voor MKA's ingericht kunnen worden, d.w.z. dat zowel functiedifferentiatie als (uitsluitend) generalisten wordt toegelaten. Bij (volledige) *functiedifferentiatie* is er een groep gespecialiseerde centralisten die de aanname doet, o.a. de *triage*, en een andere groep specialisten die verantwoordelijk zijn voor de coördinatie van het wagenpark en de communicatie met derde partijen zoals ziekenhuizen, de zogenaamde *uitgiftecentralisten*. Generalisten zijn flexibel omdat ze zowel de aanname als de uitgifte kunnen doen, maar zij zijn daarentegen wel duurder. Het tevens behandelde *mixed model* is de 'gulden middenweg' die een combinatie van aannamecentralisten, uitgiftecentralisten en generalisten toestaat. Het meldkamermodel, in al zijn vormen, voorspelt de wacht- en doorlooptijden gegeven het aantal centralisten van ieder specialisme. Op basis daarvan kan een kosten–baten analyse gedaan worden. Aan de hand van meta-data van telefoongesprekken door meldkamer Utrecht konden er goede schattingen gemaakt worden van de modelparameters. Er wordt geconcludeerd dat, afhankelijk van de werklast van de meldkamer, de optimale indeling verandert: bij lage aantallen hulpverzoeken zijn generalisten het meest kosten-efficiënt, maar naarmate dit aantal toeneemt, loont functiedifferentiatie. Bij functiedifferentiatie zal het vaak zin hebben om één generalist te houden die bij pieken van zowel aanname als uitgifte kan bijspringen.

Hoofdstukken 3 en 4 behandelen het zogenaamde *adjusted queueing* framework. Primair worden ambulancediensten beoordeeld op de fractie van ritten

waarbij de ambulance tijdig aanwezig. In deze berekening beschouwt men alleen de (potentieel) levensbedreigende situaties, dat zijn de ritten met A1-urgentie. Als behaalde prestatie wordt bijvoorbeeld vermeld dat 94% van de ritten met A1-urgentie van een verzorgingsgebied is binnen de 15 minuten ter plaatse. Hierin wordt in dimensie tijd het jaargemiddelde genomen, en in ruimte alles platgeslagen tot één punt, waardoor er één regionaal getal berekend kan worden dat de jaarprestaties van de hele regio omvat. Het nadeel van deze benadering is dat lokale belangen amper vertegenwoordigd zijn. Het loont immers om alle ambulances rondom de stedelijke gebieden met het overgrote deel van de zorgvraag te concentreren en de enkele rijtijdoverschrijding in de landelijke gebieden voor lief te nemen. *Minimal-reliability* (min-rel) modellen benaderen de prestaties anders. Zij rekenen het minimale aantal ambulances uit dat nodig is zodanig, dat op elk punt van de regio een gegeven tijdigheidspercentage gehaald kan worden. De *maximal availibility* (max-av) modellen zijn hieraan verwant; deze proberen een beperkt aantal ambulances zodanig te plaatsen dat zoveel mogelijk potentiële patiënten binnen dit gegeven tijdigheidspercentage bereikt kunnen worden. Bestaande min-rel en max-av modellen hebben een aantal nadelen. Veel modellen hebben een vaste bezettingsgraad die voor elke wagen gelijk is, wat onrealistisch is aangezien ambulances in stedelijke gebieden meer dan in landelijke gebieden op een post zijn. Bij andere voorgaande modellen mogen de aankomstintensiteit, de bedieningsduur en het toegestane percentage rijtijdsoverschrijdingen niet teveel afwijken tussen verschillende punten in de veiligheidsregio; als dat toch gebeurt, zal dat meer benodigde ambulances als gevolg hebben. Onze aanpak staat daarentegen wel toe dat de voorgenoemde drie parameters voor ieder punt ongelimiteerd verschillen, en dat de bezettingsgraad per ambulance ook niet gelijk hoeft te zijn. In Hoofdstuk 3 wordt uitgelegd waarom de eerdere modellen overschattingen geven, en wordt uiteengezet op welke wijze de modellen aangepast moeten worden, zodat ze toepasbaar worden op veelzijdige regio's. Dit resulteert in het adjusted queuing framework.

Hoofdstuk 4 introduceert twee nieuwe min-rel modellen die gebruik maken van het adjusted queuing framework uit het voorgaande hoofdstuk. Het eerste model formuleert een *mixed integer program* dat het absolute optimum vindt voor het minimaal aantal ambulances zodanig, dat op elk punt in de regio de vereiste prestaties behaald wordt. De keerzijde van dit model is dat het alleen toepasbaar is voor relatief kleine modelinstanties, vanwege de complexiteit en de daarmee gemoeide rekentijden. Het tweede model is een *heuristiek* die ook in staat is om voor grotere modelinstanties oplossingen te geven. Deze modellen worden geëvalueerd aan de hand van gegevens van echte ambulanceregio's die zowel landelijk als stedelijk gebied bevatten. Numerieke berekeningen laten zien dat voor elk punt aan de gegeven eisen met minder ambulances dan in de voorgaande modellen voldaan kan worden. De resultaten uit Hoofdstuk 4 komen overeen met de aantallen die in de praktijk in gebruik zijn, waardoor dit de eerste min-rel modellen zijn die bruikbaar zijn in de praktijk.

Hoofdstuk 5 richt zich op een operationeel vraagstuk, de *alternatieve routes*. Ambulancediensten willen, behalve de responstijd van zoveel mogelijk spoedritten binnen de 15 minuten te halen, ook dat elke gemeente afzonderlijk goed scoort: als er een gemeente zeer slecht scoort, dan moeten ze daar een goede uitleg voor geven. Bij voorwaardescheppende (VWS) ritten wordt een vrij inzetbare ambulance verzocht om naar een specifieke standplaats te bewegen ten behoeve van de dekking. Diverse modellen voor dynamisch ambulancemanagement (DAM) bestaan al om te bepalen welk voertuig naar welke standplaats moet rijden. Een open vraag is nog volgens welke route deze ambulance "het beste" naar z'n bestemming kan rijden. De routekeuze kan uitmaken voor de prestaties van de regio, omdat de ambulance tijdens het rijden naar de VWS-bestemmingspost dekking genereert over de gebieden waar hij langs komt. Door een ambulance te vragen om te rijden kan het wel zijn dat hij er langer over doet om op zijn bestemming aan te komen, maar als voordeel levert deze ambulance ondertussen een betere dekking over de regio op. Netto kan dat er dus voor zorgen dat er (lokaal) meer hulpverzoeken binnen de normtijden gehaald worden. In dit hoofdstuk wordt een methode ontwikkeld waarmee voor ambulances goede alternatieve routes kunnen worden berekend. Vervolgens wordt getoond hoe er aan een route een dekkingswaarde toegekend kan worden. Door voor elke route naar de bestemming alternatieve routes te berekenen en iedere route een dekkingswaarde te geven, kan een efficiënte route naar de bestemming aan de ambulance worden meegeven. De resulaten tonen aan dat de tijdigheid eerlijker over de regio verdeeld wordt, terwijl het aantal ritten dat op tijd komt niet verandert. De meeste winst wordt behaald op plaatsen die vanaf een standplaats slecht te bereiken zijn. Alternatief routeren is bovendien een stuk sneller en goedkoper te realiseren dan een standplaats bijplaatsen.

In Hoofdstuk 6 wordt de *pilotstudie* beschreven, waarin de in het onderzoeksproject ontwikkelde methoden in de praktijk getest worden. In samenwerking met ambulancedienst GGD Flevoland zijn gedurende twaalf weken twee DAM methoden getest, die binnen het REPRO-onderzoeksproject ontwikkeld zijn. Voor deze studie zijn de methoden aangepast om deze in de praktijk toe te kunnen passen. Bovendien is er software geschreven die de VWS-voorstellen uitrekent en deze inzichtelijk met pijlen op een kaart van de ambulanceregio aan de centralisten toont. Tijdens de pilot was er een reductie van een derde in rijtijd-overschrijdingen, waardoor dit het eerste jaar was dat de veiligheidsregio de landelijke norm haalde. Ook waren er kwalitatieve voordelen die we voorheen niet voorspeld hadden: centralisten zijn, wanneer het drukker is, sneller in staat een VWS-beslissing te nemen en worden door onze software in staat gesteld om een beter overzicht te houden van de vrij inzetbare ambulances.

De *testing interface for ambulance research* software, kortweg TIFAR, is een simulatie-framework dat zich richt op het simuleren en doorrekenen van logistieke vraagstukken in de ambulancezorg. In Hoofdstuk 7 wordt de structuur van TIFAR beschreven. Hiermee kan in kaart gebracht worden gebracht wat het gevolg is van het veranderen van de standplaatslocaties, aanpassingen aan dienstroosters of een vervanging van het VWS-beleid. TIFAR bevat klassen die het gedrag van ambulances en de meldkamer nauwkeurig kunnen nabootsen en eenvoudige koppelingen hebben met diverse databases die informatie over de regio bevat, waaronder informatie over de roosters en historische ritten. Door een koppeling met de Coin-OR optimalisatie software kunnen in TIFAR ook wiskundige modellen geïmplementeerd worden die de optimale standplaatslocaties berekenen. De resultaten van Hoofdstukken 2, 4 en 5 zijn in TIFAR gesimuleerd of met z'n klassen en database-infrastructuur berekend. Ook de pilotstudie van Hoofdstuk 6 is in dit framework geschreven.