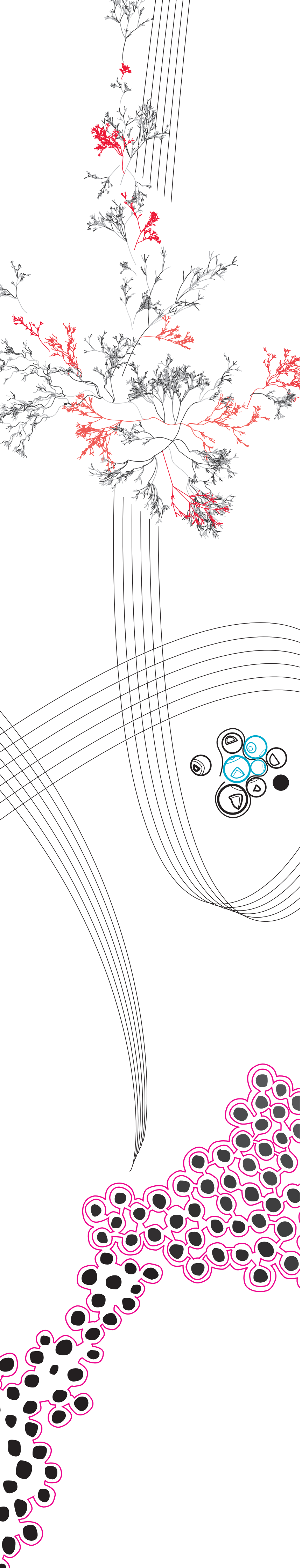


August 30, 2018

MASTER THESIS

OPTIMISING THE INVITATION STRATEGY FOR COLON CANCER SCREENING

Jasmijn Manders

Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS)
Applied Mathematics (AM)
Stochastic Operations Research (SOR)

Supervised by:
Prof. dr. Richard J. Boucherie
Dr. ir. Gréanne Leeftink
Eric L. Oukes

bevolkingsonderzoek oost

UNIVERSITY OF TWENTE.

Acknowledgements

This report is based on the research that I did for my final project of the master program Applied Mathematics at the chair of Stochastic Operations Research at the University of Twente. This research was performed in corporation with Bevolkingsonderzoek Oost in Deventer, where I looked into the invitation strategy for the colon cancer screening program.

I want to thank my supervisors Richard Boucherie and Gréanne Leeftink from the University of Twente. In the first place for giving me the great opportunity to finish my master with this interesting research. In the second place I want to thank them for giving me good advices for research directions and their valuable feedback. I also want to thank Eric Oukes for giving me the opportunity of executing this research at Bevolkingsonderzoek Oost. He helped me to understand the invitation process and helped me with making my model fit to reality. Henk Bouma and Wouter van Nieuwenhoven from Bevolkingsonderzoek Oost were always willing to answer my questions about the screening process, which I really appreciated. Together with the other employees of Bevolkingsonderzoek Oost they gave me a warm welcome. Finally I would like to thank my family and friends for supporting me any time I needed it. Especially, Nicky Schuermans who always gave me some useful tips both organisational and content focused when I came to a dead end.

Jasmijn Manders

Enschede, August 2018

Management Summary

This final project is in corporation with Bevolkingsonderzoek Oost. Each year $420,000$ clients in the Eastern part of the Netherlands have to be invited to participate in the colon cancer screening program. An invitation consists of a letter and a self-test. When the client decides to participate he sends the test sample to the laboratory. This test gives an indication whether cancer might be present or not. In case of a negative (desirable) result, the client will be invited again 2 years later. In case of a positive (undesirable) result the client is referred and should get an intake appointment within 3 weeks in a nearby colonoscopy centre (CC) for a follow-up examination. Average participation- and referral rates are 73% and 4.7% respectively.

Goals

Currently Bevolkingsonderzoek Oost uses two separate algorithms in the screening process. The first algorithm decides which clients can be invited, based on the available intake capacity of the nearby colonoscopy centres (CCs). The second algorithm schedules the intake appointments for clients with positive results without using any information of the first linking algorithm. This method is not as generic and stable as it is desired to be. Frequently parameters in the current algorithms need to be changed in order to react to events and to adapt the invitation process. To overcome this fire-fighting and to find an optimal invitation strategy, this research is started.

Method

The first step is to find an optimal matching between clients from postcode areas (PC4) to week numbers and CCs where travel time is minimized and the number of clients linked to the nearest CC is maximized. We develop a Mixed Integer Linear Program (MILP) to determine how many clients we can invite on the available capacity in a week and CC such that possible intake appointments can take place at these intake slots. We also minimize the number of clients that cannot be invited, called the rest group.

Participation- and referral rate are stochastic variables, because we do not know how many clients will participate and how many will have a positive result. Under possible realizations of these uncertain parameters, within their intervals $[70\%, 76\%]$ and $[4.3\%, 5.1\%]$, we still want to invite all clients and make sure that intake appointments can take place in a nearby CC within 3 weeks. We use robust optimisation to find a safe solution to the matching problem. When a client is linked to a CC and a week, we want to have a high probability that his intake appointment can take place in the determined CC and week.

In the first two parts of the research the time that a client needs to respond is not taken into account, this response time is uncertain. In part three we determine the optimal moment of sending the invitations to clients, such that the possible intake appointment can take place at the desired week and CC as determined previously by the MILP or the robust optimisation model. We develop a Stochastic Dynamic Program (SDP) for this, which we then decompose into smaller SDPs corresponding to a single CC each. We model the response time of clients with an exponential distribution. We find an optimal invitation strategy based on the number of outstanding invitations, the number of positive results and the number of already invited clients in each week of the year.

Results and Conclusion

The results of the MILP and the robust optimisation consist of invitation strategies which tell us how many clients from which postcode areas we should link to which week and CC. The main output parameters that tell us the quality of the solutions are the size of the rest group, the adherence (which postcode areas are linked to which CCs) and the percentage of clients that cannot be linked to the nearest CC.

The deterministic MILP gives a matching with other adherence numbers than currently used at Bevolkingsonderzoek Oost. The number of clients that cannot be linked to the nearest CC is decreased from 40% in the current situation to 17.3%. The adherence of the MILP model is therefore more optimal and the number of rescheduled intake appointments can be decreased. However, the available intake capacity over region East still does not align with the distribution of clients over region East. Our model gives a possible rest group of 3.3% instead of 7.8% with the current adherence.

With robust optimisation we are able to find a ”safe” solution to the matching problem and we immunize against the uncertainty in participation- and referral rate. We use the budgeted uncertainty approximation type, which rules out any large deviations from the cumulative number of needed intake appointments in a CC and week. By using the best suitable safety parameter we build in buffer capacity in the intake slots, by inviting slightly less clients than possible in the deterministic case. The safe solutions give us a rest group of 9% at the end of the year. The great advantage of the safe solution is that the probability that a client cannot have his intake appointment in the determined week and CC decreases to only 8% instead of the 22% when the MILP solution is considered.

In the time uncertainty model we neglect the pré-announcement period of 2 weeks. One of the reasons for this is that not sending the pré-announcement letters can save €37,500 each year. We only have results of the single CC SDP model for small instances with few clients and higher participation and referral rates than in practice. These preliminary results suggest a structured invitation strategy where invitations are send during the year when enough clients still need to be invited. Inviting will occur under the conditions that in the current week (1) the number of outstanding invitations is small and (2) the number of positive results is not to large. However, further research is needed to develop the SDP models for the use in practical instances.

Management Samenvatting

As this research is executed at the Dutch screening organisation Bevolkingsonderzoek Oost, we also include a management summary in Dutch.
Aangezien dit onderzoek is uitgevoerd bij het Nederlandse Bevolkingsonderzoek Oost, voegen we ook een management samenvatting in het Nederlands toe.

Dit afstudeerproject is in samenwerking met Bevolkingsonderzoek Oost. Ieder jaar moeten er 420.000 cliënten uit het oostelijke gedeelte van Nederland uitgenodigd worden om deel te nemen aan het bevolkingsonderzoek darmkanker. Een uitnodiging bestaat uit een brief en een zelftest. Als de cliënt besluit om deel te nemen, stuurt hij het monster naar het laboratorium. De test geeft een indicatie of kanker aanwezig kan zijn. In het geval van een negatieve (gunstige) uitslag zal de cliënt na 2 jaar opnieuw uitgenodigd worden. In het geval van een positieve (ongunstige) uitslag wordt de cliënt doorverwezen en moet hij binnen 3 weken een intake afspraak krijgen in een coloscopie centrum (CC) in de buurt voor vervolg onderzoek. De gemiddelde deelname- en verwijs percentages zijn respectievelijk 73% en $4,7\%$.

Doelen

Op dit moment gebruikt Bevolkingsonderzoek Oost twee afzonderlijk algoritmes binnen het uitnodigingsproces. Het eerste algoritme beslist op basis van de beschikbare capaciteit in omliggende coloscopie centra (CCs) welke cliënten uitgenodigd kunnen worden. Het tweede algoritme plant de intake afspraken voor cliënten met een positieve uitslag, waarbij geen enkele informatie van het eerste algoritme gebruikt wordt. Deze methode is niet zo algemeen een stabiel als gewenst wordt. In de huidige algoritmes moeten parameters regelmatig aangepast worden om te reageren op gebeurtenissen en om het uitnodigingsproces aan te passen. Dit onderzoek is gestart om een optimale uitnodigingsstrategie te vinden waarbij het huidige systeem van continu brandjes blussen voorkomen wordt.

Methode

De eerste stap in dit onderzoek is het vinden van een optimale toewijzing van cliënten uit postcode gebieden (PC4) aan week nummers en CCs waarbij de reistijd wordt geminimaliseerd en het aantal cliënten dat gekoppeld wordt aan het dichtstbijzijnde CC wordt gemaximaliseerd. We ontwikkelen een Mixed Integer Linear Program (MILP) om te bepalen hoeveel cliënten we kunnen uitnodigen op de beschikbare capaciteit in een week en CC, zo danig dat een mogelijke intake afspraak kan plaatsvinden op deze intakeslots. Daarbij minimaliseren we het aantal cliënten dat niet uitgenodigd kan worden, dit heet de restgroep.

De deelname- en verwijs percentages zijn stochastische variabelen, omdat we niet weten hoeveel cliënten zullen deelnemen en hoeveel van hen een positieve uitslag zullen hebben. Onder de mogelijke realisaties van deze onzekere parameters (ze variëren in de intervallen $[70\%;76\%]$ en $[4,3\%;5,1\%]$) willen we nog steeds alle cliënten uitnodigen, waarbij de intake afspraken binnen 3 weken in een dichtstbijzijnd CC kunnen plaatsvinden. We gebruiken robuuste optimalisatie om een "veilige" oplossing te vinden voor het toewijzingsprobleem. Wanneer een cliënt gekoppeld is aan een CC en een week, willen we dat de kans groot is dat zijn intake afspraak in die week en in dat CC kan plaatsvinden.

In de eerste twee delen van dit onderzoek wordt de tijd die een cliënt nodig heeft om te reageren op de uitnodiging niet mee genomen, deze reactie tijd is onzeker. In deel drie van dit onderzoek bepalen we het optimale moment om de uitnodigingen te versturen, zodat een mogelijk intake afspraak kan plaatsvinden in het gewenste CC en de gewenste week. Deze gewenste locatie en week zijn eerder bepaald door het MILP of het robuuste optimalisatie model. We ontwikkelen een Stochastic Dynamic Program (SDP) voor dit probleem. Vervolgens decomposeren we het SDP in kleinere deelmodellen, voor ieder CC één. We modelleren de reactietijd van cliënten met een exponentiële verdeling. We vinden een optimale uitnodigingsstrategie die gebaseerd is op het aantal openstaande uitnodigingen, het aantal positieve resultaten en het aantal al uitgenodigde cliënten in iedere week van het jaar.

Resultaten en Conclusie

De resultaten van het MILP en de robuuste optimalisatie bestaan uit uitnodigingsstrategieën die ons vertellen hoeveel cliënten we vanuit welke postcode gebieden moeten koppelen aan welke week en welk CC. De belangrijkste output parameters zijn de grootte van de restgroep, de adherentie (welke postcode gebieden worden gekoppeld aan welke CCs) en het percentage cliënten dat niet aan het dichtstbijzijnde CC gekoppeld kan worden. Deze output parameters zeggen iets over de kwaliteit van de oplossing.

Het deterministische MILP geeft een toewijzing met een andere adherentie dan de huidige adherentie van Bevolkingsonderzoek Oost. Het aantal cliënten dat niet aan het dichtstbijzijnde CC gekoppeld kan worden is gedaald van 40% in de huidige situatie naar 17.3%. De adherentie van het MILP is daarom beter en het aantal intake afspraken dat verzet wordt kan worden verminderd. Toch komt de beschikbare capaciteit in regio Oost nog niet helemaal overeen met de verdeling van cliënten over regio Oost. Er is in totaal genoeg capaciteit aanwezig, maar niet op de juiste plaatsen. Ons model geeft een mogelijk restgroep van 3.3% in plaats van 7.8% bij de huidige adherentie.

Met robuuste optimalisatie kunnen we een ”veilige” oplossing vinden voor het toewijzingsprobleem en beschermen we ons tegen onzekerheid in de deelname- en verwijspercentages. We gebruiken het benaderingstype Budgeted voor de onzekerheid, deze sluit grote afwijkingen van het cumulatieve aantal benodigde intake afspraken in een CC en een week uit. Door gebruik te maken van de veiligheidsparameter die het best aansluit bij de praktijk, bouwen we buffer capaciteit in de intakeslots in. We nodigen iets minder cliënten uit dan dat mogelijk is in het deterministische geval. De robuuste oplossing geeft een restgroep van 9% aan het eind van het jaar. Het grote voordeel van de robuuste oplossing is dat de kans dat de intake afspraak van een cliënt niet kan plaatsvinden in het bepaalde CC of de bepaalde week daalt naar slechts 8%, in plaats van de 22% wanneer gebruik wordt gemaakt van de MILP oplossing.

In het tijdonzekerheidsmodel negeren we de vooraankondigingsperiode van 2 weken. Een van de redenen hiervoor is dat het niet langer verzenden van vooraankondigingsbrieven €37,500 per jaar kan besparen. We hebben alleen oplossingen voor het SDP model voor één CC waarbij we kleine instanties met weinig cliënten en hogere deelname- en verwijs percentages dan in praktijk bekijken. Deze voorlopige resultaten suggereren een gestructureerde uitnodigingsstrategie waar uitnodigingen verzonden worden gedurende het jaar wanneer er nog genoeg cliënten uitgenodigd moeten worden. Dit gebeurd onder de voorwaarde dat in de huidige week (1) het aantal openstaande uitnodigingen klein is en (2) het aantal positieve uitslagen niet te groot is. Echter is er verder onderzoek nodig om de SDP modellen door te ontwikkelen zodat ze geschikt zijn voor praktijk instanties.

Contents

Chapter 1

Introduction

This research is executed in corporation with Bevolkingsonderzoek Oost. This organization is responsible for executing the Dutch screening programs in the Eastern part of the Netherlands in order to detect specific cancer types in an early stage. In particular, we look at the screening process for colon cancer, which is the most recently introduced screening program. We investigate whether this process can be optimised by using mathematical models for the invitation strategy of inviting clients to the screening program. At this moment the used invitation strategies work, but the experience of fire fighting is present. With this research we want to investigate whether we can make the invitation process more stable and continuous and have some more general invitation strategies.

Section 1.1 explains the process for colon cancer screening that is executed by Bevolkingsonderzoek Oost. This process is based on the requirements that are set by the Dutch government, [Rijksinstituut voor Volksgezondheid en Milieu, Ministerie van Volksgezondheid, Welzijn en Sport, 2017]. The goals of this research are given in Section 1.2. This section also explains the different parts of the research with the corresponding research questions. We give an overview of the related literature in Section 1.3. The outline of this thesis is given in Section 1.4.

1.1 Screening process

The goal of Bevolkingsonderzoek Oost is to invite at least 95% of the 420,000 clients in the provinces Gelderland and Overijssel to participate in the colon cancer screening program each year. The screening process exists of 4 steps:

1. **Selecting and inviting:** Men and women between the age of 55 and 75 are invited every two years. Each workday of the year invitations can be sent, which contain a test (FIT) and some information. 14 days before sending the actual invitation, the invitation is already created. A client that is invited for the first time, receives a pre-announcement 14 days before the actual invitation. For each next time, the invitation is sent on the same day as it was sent 2 years ago. The client can take the test and send it to the laboratory, when this is done the client is a participant. It is also possible to sign out. The part that is sent to the laboratory is called a feces sample. The participation rate is known, from historical data.
2. **Screening:** The laboratory investigates the feces sample and determines the result. This result is published in ScreenIT, which is the software that is used. A positive result means that there is a suspicion of colon cancer and further examination is needed.
3. **Inform and refer:** In case of a negative (favourable) result, the participant receives the result by mail and the process ends. In case of a positive (unfavourable) result, the participant receives an appointment in a colonoscopy centre (CC) for an intake. Multiple CCs are spread over the eastern region of the Netherlands and these CCs all offer some intake slots. An intake is scheduled within at most 15 workdays after sending the results to the participant. Preferably, a client is scheduled in a CC that is nearby the client. The referral rate ($\%$ positive results) is known. A participant can reschedule the intake, which currently occurs in 40% of the cases. Both the time and the location of this intake can be rescheduled. During the intake appointment it is determined whether the participant needs to undergo a colonoscopy. When this is not possible the process ends.
4. **Diagnostics:** After the intake a colonoscopy is scheduled by the CC within 15 workdays. This colonoscopy is a follow-up examination and can have a normal or abnormal result. In case of a normal result the process ends. In case of an abnormal result a medical trajectory starts which

includes the treatment of the participant (who is now called a patient) or surveillance. Bevolkingsonderzoek Oost is not involved in both these options.

The time line corresponding to this screening process is shown in Figure 1.1, where the red numbers indicate the mean length of each period. The ”unfavourable result wait period” is used to inform the general practitioner, so he can inform the participant. This time is also used to send the result letter.

At this moment the invitation for colon cancer screening is done using software called ScreenIT. This software contains all information about clients and colonoscopy centres and uses two algorithms. The first algorithm sends invitations (including the FIT) and the second schedules intake appointments after a positive FIT result. These two algorithms are not connected to each other. As mentioned in step 4 an colonoscopy should be scheduled after the intake appointment. However this is not the responsibility of Bevolkingsonderzoek Oost, but of the colonoscopy centre itself. Therefore, in this research the process ends when an intake appointment is scheduled.

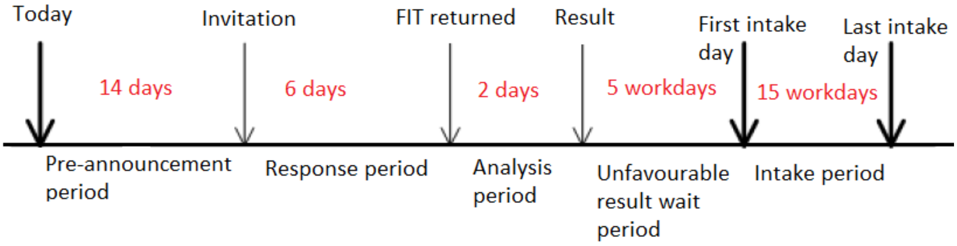


Figure 1.1: The time line of the screening process.

In the current situation within Bevolkingsonderzoek Oost, the two algorithms work. All clients can be invited and intake appointments are scheduled for clients that need an intake appointment. However, the process is not as generic and stable as it is desired to be. Frequently, parameters in the current algorithms need to be changed in order to react to events and adapt the invitation process. This fire fighting is not desired and does not give an optimal solution for all clients. To overcome these issues, this research is started to find optimal invitation strategies that are generic and advantageous for the clients.

1.2 Goals and research questions

We divide the research in three different parts. The invitation process contains uncertainty in participation of clients, uncertainty in test results of clients, and the throughput times within the process are not always known in advance. The three different parts take these uncertainties in steps into account.

First, we look at the screening process in a deterministic case. This deterministic case does not include any time dependency and fixed participation and referral rates are assumed. We want to find an optimal matching of clients from postcode areas to week numbers and colonoscopy centres (CC). We determine which clients will be linked to which CC and also in which week their possible intake appointment can take place. The corresponding research question of this first part of the research is:

- *Which matching between clients from postcode areas to week numbers and colonoscopy centres is optimal?*

A Mixed Integer Linear Program (MILP) is developed in order to find such a matching. For this MILP, we need input data in order to test the model. The following research questions describe the needed data.

- *Which colonoscopy centres are available and what is their intake capacity per week?*
- *Which aggregation level (council/PC4/PC5/PC6) for the postcode areas is desired?*
- *How many clients should be invited per postcode area and invitation round?*
- *What are the travel resistances between postcode areas and colonoscopy centres?*
- *What is the distribution of the participation rate and referral rate?*

In the second part of this research we take the uncertainty in participation and referral rates into account. Beforehand, it is not known how many and which clients will participate in the screening program and we also do not know which and how many client will have a positive result. These two uncertain rates influence the screening process, because you want to make sure that clients with a positive result are able to have their intake appointment within a reasonable time and CC. We use robust optimisation to find a safe solution to the matching problem. This means that under all possible realizations of the uncertain rates, we still want to have an optimal solution. The following research questions are answered in this second part of the research.

- *How should the uncertainty set of participation and referral rate be specified?*
- *What is the effect on the matching between clients, weeks and colonoscopy centres of making the model stochastic and solving with robust optimisation?*

Finally, in the third part, we take the time uncertainty of the screening process into account. The main time uncertainty comes from the respond time of clients. You do not know in advance how long a client will take to respond on his invitation. Therefore we define the following research question:

- *Which probability distribution can be used to describe the time uncertainty?*

The goal of this research is to determine the moment of sending the invitation in such a way that the possible intake can take place at the desired week in the desired CC as determined previously by the MILP or robust optimisation model. This gives us the last research question.

- *Which strategy for determining the moment of invitation is optimal for having the intake appointment at the predetermined moments and colonoscopy centres?*

For this third step of the research we develop a Stochastic Dynamic Program (SDP) which has actions that should be chosen optimally. The actions represent the number of invitations that are send to clients in certain postcode areas at certain moments in time.

1.3 Literature review

The screening process for colon cancer is relatively new in the Netherlands. Until now, no literature on optimising screening processes is found. Except for Toes-Zoutendijk et al. [2017], they analyse the first year of executing the Dutch colon cancer screening program. Toes-Zoutendijk et al. [2017] monitor whether the requirements set by the Dutch government are met, and they analyse the performances of the screening program in the first year. Thanks to this research some small adjustments in tolerance levels of medical tests could be made. In this way the screening program performance is already optimised in terms of detecting early stages of colon cancer in the correct way. Some countries outside the Netherlands also have a colon cancer screening program. The country with the most literature and research on colon cancer screening is Norway. Unfortunately, also here most research is from medical perspectives. Bretthauer et al. [2002] executed a study to evaluate the effect of screening for colon cancer on mortality rates. This study was a once only screening model with a randomized test group, which were split in two groups: colonoscopy and faecal occult blood test (FOBT)/colonoscopy. Bretthauer et al. [2002] concluded that the FOBT is suitable for future countrywide screening. This test also is used in the Dutch colon cancer screening process. To the best of our knowledge, literature on how to organize the screening process in a mathematical way is not present.

The screening process as it is executed by Bevolkingsonderzoek Oost (BVO Oost) does have a few similarities with airline revenue management in selling seats. BVO Oost invites multiple clients on one available intake slot, because not all clients that are invited participate and many of them do not need an intake appointment because of a negative result. Airlines sell more tickets than seats available in a plane, because cancellations and no-shows occur in these businesses. This overbooking is done in order to maximize revenue, however the number of cancellations is uncertain. We can see the clients that do not need an intake appointment as cancellations on the intake slot that was used to invite these clients. In the screening program these cancellations are uncertain because we do not know how many clients will participate and how many will have a positive result. Bertsimas and Popescu [2003] develop dynamic policies for satisfying uncertain demand with limited capacity in network environments. By allowing oversales decisions in the linear programming formulation, cancellations and no-shows can be handled. Bertsimas and Popescu [2003] develop a control policy for accepting or denying reservations over the booking horizon by approximate dynamic programming. Taking cancellations into account can be done in a static way by virtually increasing capacity by the expected number of cancellations. This is similar to inviting multiple clients on one intake slot in the case of BVO Oost. However, Bertsimas and Popescu [2003] also gives a method for incorporating these cancellations in the dynamic programming model. An other approach is to use simulation in order to find an optimal seat-allocation in the airline industry including cancellations and overbooking. Gosavi et al. [2007] give two efficient optimisation techniques in combination with simulation to solve this problem.

We have the idea to use robust optimisation in order to find safe solutions for the invitation strategy of colon cancer screening. Robust optimisation is a relative new technique for handling data uncertainty. Robust optimisation is easier than stochastic optimisation. However, robust optimization gives more conservative solutions because worst case values are incorporated. The basic theory of robust optimisation is given in Ben-Tal et al. [2009]. We used this book to understand the first principals in robust optimisation. Gabrel et al. [2014] gives an overview of the developments in robust optimisation and gives a wide range of application areas. In particular the areas inventory and logistics and finance make often use of robust optimisation models. For example, Aouam et al. [2018] uses robust optimization techniques to find an optimal decision strategy and production planning where orders are allowed to be rejected. The demand of products is uncertain and the products have to be produced in batches. An application of robust optimisation in finance is given in Chen and sha Zhou [2018]. Investing in a stock comes with many uncertainties, such as returns and covariances. Chen and sha Zhou [2018] develop a robust method for constructing an optimal portfolio that invests in multiple stocks on the financial market.

Robust optimisation in healthcare has less examples, but more and more robust optimisation models are developed in the healthcare sector. Addis et al. [2016] plans patients from a waiting list into operating room blocks over a rolling horizon. Goal is to minimize the waiting time and tardiness of the patients. First an ILP is developed, but to take uncertainty of extensions of surgeries into account a robust formulation of this ILP is proposed. The type of robust counterpart that is used is the budgeted type, where the number of longer surgery times is bounded.

An other way of approaching uncertainty is by using stochastic programming. Beraldi et al. [2004] use a stochastic programming model with probabilistic constraints to solve where emergency service sites should be located and how many emergency vehicles should be available. The uncertainty in Beraldi et al. [2004] comes from the randomness in occurrence and location of emergencies. The location issues are comparable with our research at BVO Oost, because we want to minimize travel time for clients that need to go to a CC, whereas emergency vehicles want to have short access times to the place of the emergency. The main difference is that within our research the locations of clients and CC is certain and only needed capacity for these locations is uncertain. In Beraldi et al. [2004] both location and frequency of emergencies is uncertain, which results in a more complex setting.

To come back at inviting multiple clients on one intake slot in the screening program and selling more airline seats than available, we want to combine cancellations and healthcare. In home care problems cancellations occur relatively often because of elderly people who decease. In home care multiple aspects should be planned. Cappanera et al. [2017] look at caregiver-to-patient assignments, scheduling of patient requests and caregiver routing at the same time. The uncertainty of patient requests and cancellations are taken into account by cardinality constraint robustness, also called budgeted uncertainty. The colon cancer screening process also deals with cancellations because only a small part of the invited clients on an intake slot will finally need an intake appointment.

In addition Cappanera et al. [2017] make use of a decomposition method to reduce the solution space of their scheduling problem. The idea is to fix some of the decisions on beforehand which are likely to be optimal. The care plan of a patient is in practice variable, but according to a deterministic approach the optimal care plan is fixed in order to decompose the complete scheduling problem. We can use this idea of decomposition by prefixing certain decisions in our stochastic dynamic programming model for inviting the clients to the screening process at the optimal moment in time.

1.4 Thesis outline

Chapter 2 gives an elaborate description of the screening process that we consider in this research. The mathematical model (MILP) that we develop for the deterministic case is given in Chapter 3. This chapter contains all sets, parameters and variables that are used and the objective function and constraints are explained. Chapter 4 gives a description of which data is used in this research and how the data is modified in order to fit the model. The results of finding an optimal matching for the deterministic MILP are given in Chapter 5. The second part of the research starts in Chapter 6. This chapter describes which parameters and constraints of the MILP become uncertain and explains how a robust counterpart can be developed. Solving this robust counterpart gives the desired safe solution to the matching problem of clients from postcode areas to week numbers and CCs. These robust results are given in Chapter 7, which also contains a comparison between the robust and deterministic results. In Chapter 8 we look at the time uncertainty in the screening process. We first describe approximating the throughput times followed by a Stochastic Dynamic Program (SDP) that is developed. We start with a general SDP for the entire region East, followed by a decomposition approach to multiple smaller SDPs for each CC one. Chapter 9 contains the results for the single CC SDP. In Chapter 10 we discuss the research and give some suggestions for further research. Finally, Chapter 11 gives the conclusion of this research and some recommendations for Bevolkingsonderzoek Oost.

Chapter 2

Context analysis

This chapter gives a more elaborate description of the screening process. First some background information about colon cancer and the general screening program is given in Section 2.1. Section 2.2 contains a detailed description of how the screening program is executed at this moment. Finally, some quality criteria are set for the screening program which are given in Section 2.3. The information in this chapter is based on Rijksinstituut voor Volksgezondheid en Milieu, Ministerie van Volksgezondheid, Welzijn en Sport [2017] and Topicus Zorg [2017] and conversations with employees of Bevolkingsonderzoek Oost.

2.1 Background information

The colon cancer screening considered in this research is part of the Dutch screening program, together with breast cancer and cervical cancer screening. A screening program is executed as an initiative of the care system, not as a reaction to care demand. A screening program is dedicated to a pre-specified target group in which the people do not have symptoms (yet) and the screening has a systematic nature. The screening should be carried out throughout the whole country in the same way and with high quality standards. Therefore the following organizational structure is used. The minister of Volksgezondheid, Welzijn en Sport (VWS) holds the Centrum voor Bevolkingsonderzoek of the Rijksinstituut voor Volksgezondheid en Milieu (RIVM-CvB) responsible for the national organization. Five screening organizations are responsible for the coordination within a part of the Netherlands. The five parts are North, South, Mid-West, South-West and East and are shown in Figure 2.1. Each of these screening organizations have their own office and a sixth office coordinates the data over the whole of Netherlands. This data is contained in the software called ScreenIT, which is also used to send invitations and schedule appointments. This research focuses on the invitation strategy within the Eastern part of the Netherlands (green area in Figure 2.1), so the research is executed in co-operation with the screening organization East.

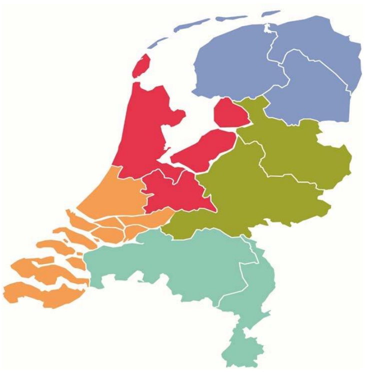


Figure 2.1: The five screening organizations divided over the Netherlands. The Eastern part is coloured green.

The types of colon cancer that can be detected by the screening are cancer in the colon or in the rectum. Also polyps, called adenomas, can be detected during the colonoscopy in further examination. The first test that is done is called FIT (or iFOBT) and is done by the participant himself. The feces sample that is taken is send to the laboratory to be analysed. This test does not indicate whether cancer or adenomas are present, but it indicates only the possibility. The test measures the hemoglobin in the feces, which indicates traces of blood in the feces. A positive (undesirable) result of this test corresponds with a value of 47 $\mu g\ Hb/g$ feces or higher. This value indicates the suspicion of cancer and further examination through an intake and colonoscopy are needed, which take place in a colonoscopy centre (CC).

The colon cancer screening invites men and women between the age of 55 and 75. A client is invited every 2 years, because the test that is used is only a single test at a specific moment. This is therefore not fully reliable. The first time the test is executed the sensitivity is about 65%. This is the probability that the test gives a positive result when the disease is present. By participating every 2 years this sensitivity number increases to 80%-90% with the following tests. The polyps grow very slow, it can take 10 years until cancer is really present. Detecting such polyps and removing them can prevent the participant from developing cancer. In this way the colon cancer screening can prevent yearly 2400 deaths in the Netherlands.

Throughout this research some terms and abbreviations are used, these terms are explained in Table 2.1.

Table 2.1: A list of used terms and their explanation.

Term	Description
FIT (iFOBT)	Fecaal Immunochemische Test, the first test that a client takes himself.
ScreenIT	The software that is currently used in the screening process.
BVO	Bevolkingsonderzoek, the term for the Dutch screening program.
VWS	Volksgezondheid, Welzijn en Sport, ministry of the Dutch government for public health and sports.
RIVM	Rijksinstituut voor Volksgezondheid en Milieu, Dutch national institute for public health and environment.
CvB	Centrum voor Bevolkingsonderzoek, institute for the screening program.
CC	Colonoscopy Centre, where the further examination as intake and colonoscopy takes place.
Screening organization	The organization that is responsible for the coordination of the screening within its own part of the Netherlands (region).
Clients	Men and women in the Netherlands between age 55 and 75 (the target group). These people need to get an invitation for the screening program. The people that have received an invitation but did not respond (yet) are also called clients.
Participant	A person that has decided to participate in the screening program and has sent the feces sample to the laboratory.
Patient	A participant that has an abnormal result in the colonoscopy. In this case a observation or treatment trajectory is started.
Intake slot	A time frame in a colonoscopy centre where a intake appointment can be scheduled.
Adherence	The name for linking regions to colonoscopy centres. For each region it is set how many percent of the inhabitants will go to a certain colonoscopy centre.
Postcode area	An area where clients live that corresponds to a specific postcode.
Region	A part of the Netherlands that corresponds to a single screening organization; East, North, South, Mid-West or South-West.
Laboratory	The laboratory analyses the feces sample (FIT) of a participant and determines the result.
Feces sample	The FIT that is done by the participant and that needs to be analysed.
PostNL	The postal service in the Netherlands that delivers the feces samples to the laboratories.
Day	A day of the year.
Workday	A day that is not a weekend day or public holiday.

2.2 Screening process

The screening process is already described in Section 1.1. Figure 2.2 contains the flowchart of the screening process. The two main steps are to determine which clients receive an invitation at which moment and secondly to schedule the needed intake appointments for clients with positive (unfavourable) results. These two steps each come with an algorithm in the software ScreenIT. The invitation algorithm is described in Section 2.2.1 and Section 2.2.3 describes the algorithm for scheduling the intake appointments.

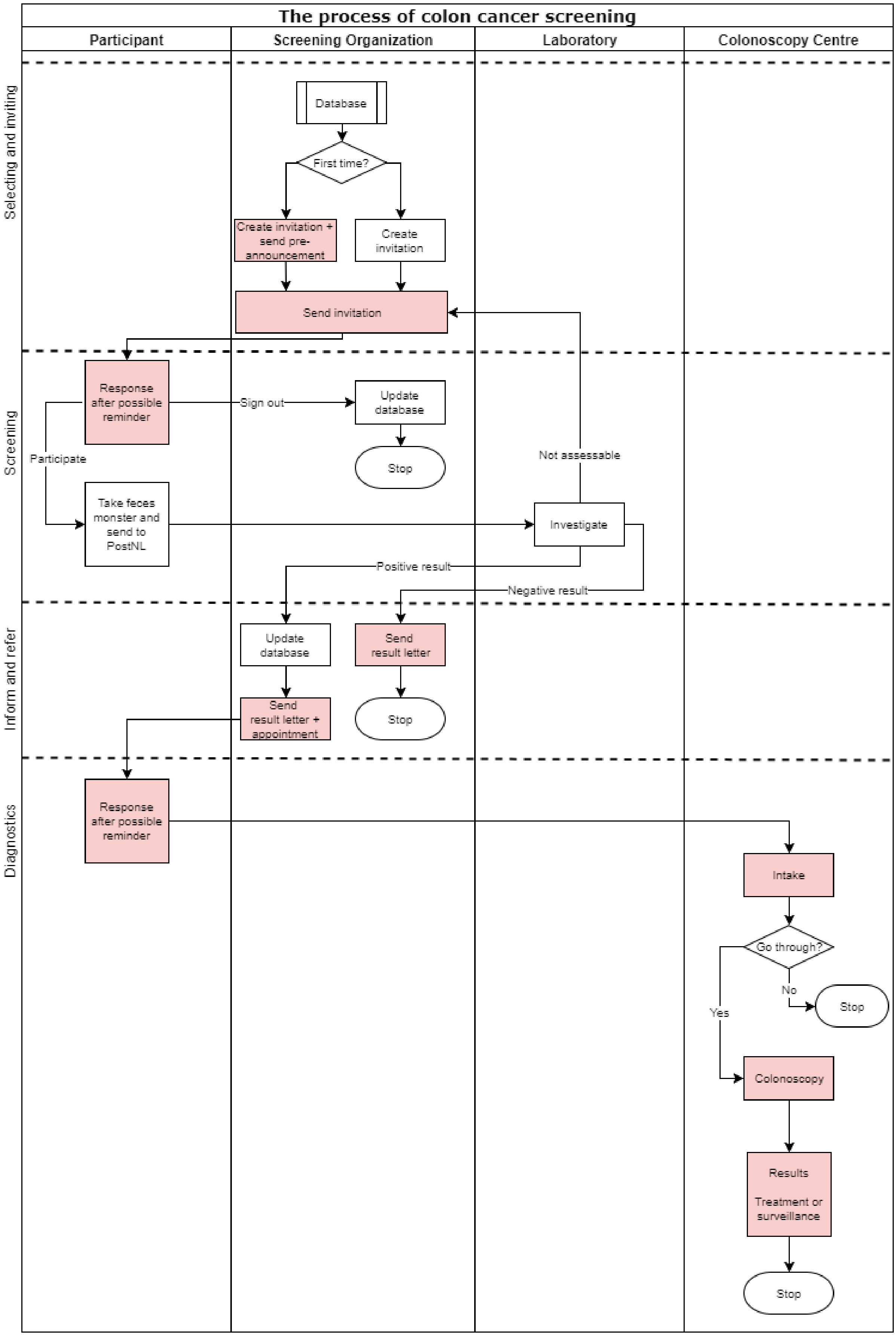


Figure 2.2: The flowchart of the screening process. The pink parts indicate contact with the client/participant.

2.2.1 Invitation algorithm

The invitation algorithm is used to determine how many and which clients receive an invitation. The idea behind this algorithm is that a client is only invited when on short term capacity is available for an intake appointment in case of a positive FIT result. Currently BVO Oost looks at groups of clients that all live in the same council.

For each year it is known how many people should be invited in each council and these numbers are communicated to all CCs. With this information and the agreements with insurance companies the CCs announce in October their capacity for the next year. This is done in number of intake appointments they can handle in total coming year. It is assumed that this total number is divided equally over all 52 weeks in the year. However, this capacity can change over the year, for instance when a doctor leaves and they retract capacity halfway the year. The exact capacity is supposed to be known seven weeks ahead of time and the possible intake slots are then denoted specifically on day and time. These slots are used to invite participants, although it is possible that these slots are rescheduled by the CC. When an intake is planned for a specific participant the slot becomes fixed. The seven week period is based on the fact that time between pre-announcement (create invitation) and intake is on average seven weeks, as shown in Figure 1.1.

The invitations are send to clients, based on the number of intake slots that are available in week seven from now. Each intake slot corresponds to approximately 30 invitations that can be send. This is because the current participation rate is 75% and the referral rate is 4.5%, so $1/(0.75 \cdot 0.045) \approx 30$. In order to decide which clients are invited on this invitation capacity, the so-called adherence is used.

Adherence is the name for linking councils to CCs. For each council it is set how many percent of the people will go to a certain CC. It can be seen as a matrix $A = [a_{ij}]$, where each row (i) corresponds with a council and each column (j) corresponds to a CC. The element a_{ij} is then the percentage of people from council i that are linked to CC j. For each row, the sum over all columns should be equal to 100%. This is because all people in a council need to be linked to a CC, otherwise they will not be invited. It is desired that councils are linked to CCs that are close by (within 40 km), because a possible intake appointment should be scheduled nearby. The adherence can also been seen from a CC point of view. In this case the CC j indicates what percentage (b_{ji}) of their participants come from council i. This is a matrix $B = [b_{ji}]$, where the sum over all columns i should be equal to 100%.

The adherence from a CC point of view is used to distribute the invitation capacity of a CC over the different councils. From one intake slot in CC j, 30 invitations can be send of which b_{ji} percent come from council i. When these numbers are determined for each council by adding up the invitation capacities from all CCs, the total number of possible invitations per council is known. This invitation capacity is used in the following order of clients for each council separately:

1. The participants that received an invitation exactly 2 years ago will receive a new invitation. The needed invitation capacity for these participants is subtracted from the total invitation capacity for this council.
2. The clients that should have been invited last year will receive an invitation. These clients are the clients in the rest group of last year. Also the invitation capacity that these clients need is subtracted.
3. The clients of the current year will be invited. These clients participate for the first time. All the residual invitation capacity is used in this step.

In the second and third step an age-mix is used to invite clients, because clients of different ages have different probabilities of referring. The age-mix is used to level the total referral rate approximately at the average referral rate. When for each council these three steps are done, it is checked whether all capacity for each council is used. In the case of a council that has no more clients, but intake capacity is still reserved for this council, the intake capacity will be distributed over other councils. Again, the above described order is used. Concluding, the invitations are send based on the announced capacity. However, it is desirable that the invitations are spread evenly over the year, as clients can then be invited every two years on the same day.

The capacity that is looked at is only the capacity for the intake appointments. The capacity that is needed to perform the colonoscopy itself and all other appointments (e.g., results) is not included. However, the CC guarantees that these follow-up appointments are possible. The colonoscopy should be scheduled within 15 work days of the intake appointment. This implies that three weeks ahead of a holiday, the CCs do not want to do any intake appointments. Therefore, a one week holiday period is actually four weeks long. An intake appointment takes half an hour, while the colonoscopy takes almost a whole day. Therefore, it might be better to use the colonoscopy capacity in the algorithms. On the other hand, all participants with a positive result need an intake appointment, but not all of them need a colonoscopy. Some participants get the advice after the intake appointment not to undergo a colonoscopy due to other medical issues.

CCs do not known the capacity of each other, as the system should be a free market process. This system leads to some challenges, because the current invitation algorithm reacts on the amount of intake slots indicated by the CCs. Due to the free market process this number of available intake slots can change. For instance seven weeks before Easter clients are invited based on the slots that are announced, but a week before Easter the CC decides to close during Easter. In this case the invited participants that need an intake cannot be scheduled any more. However, it seems that this problem is caused by communication issues. Another problem is that, due to this free market, each year the capacity of intake slots can vary. A negative consequence of this free market process is when in a year many clients where invited in January because many slots where available, all these clients should be invited at the same date in 2 years again. However, it need not be the case that at that moment also enough slots are available.

2.2.2 Laboratory

The clients that decide to participate take their own feces sample (FIT) and send the test to PostNL, which distributes all feces samples of the Netherlands over the four laboratories. Each laboratory is able to process $\geq$ 1600 feces samples per day ($\geq$ 445,000 / year), except for weekend days, Mondays and public holidays. It is required that the time between receiving the feces sample and sending the result to ScreenIT is maximal 48 hours. On average each laboratory can handle minimal 800 feces samples as overcapacity and the laboratory can store minimal 1600 feces samples for 24 hours before processing them. This capacity, which is needed to process the feces samples of the whole of Netherlands, is not considered to be restrictive in this research.

2.2.3 Intake appointment algorithm

When a positive (unfavourable) result is reported, an intake appointment is scheduled for a specific participant at a specific time and intake location (CC). This scheduling is done each day for all participants that need an intake appointment simultaneously. The algorithm does not use the adherence from the inviting algorithm any more. All possible intake slots are considered and for each participant a slot is chosen that is as soon as possible and as close-by as possible.

The first step of the algorithm is to find at least twice as much intake slots as participants that need an intake appointment. These intake slots are considered in the whole of the Netherlands and between 5 workdays from know and 20 workdays from now (the intake period). When not enough free intake slots are available, this period is extended with one day until enough free intake slots are available. With these free intake slots the algorithm schedules each participant that needs an intake appointment at a free intake slot, this is the initial solution. This solution is optimised by a Tabu search algorithm where both distance between address and CC and time till appointment are minimized. This is done by giving each combination of participant and intake slot a score. This score is given in equation (2.1).

$$\text{Score} = (\text{distance score} \cdot \text{distance factor})^2 + (\text{time score} \cdot \text{time factor})^2 \tag{2.1}$$

Both distance (km measured in a straight line, also known as euclidean distance) and time (hours) are converted to numbers between zero and hundred, where a score of 0 is optimal. For example, the maximum distance is 50 km, so a scheduled appointment where the participant should travel 20 km has a distance score of 40. The factors can be used to make distance or time more important, currently both factors are set to fifty.
Initially the Tabu search algorithm is executed for 30 minutes. When the algorithm finds intake slots for participants in a CC that is more than 40 km from the home address, a second run is done with some other settings. In this extra iteration the distance is more import and some extra days can be added to the intake period. At most 5 extra iterations are done. After the required iterations all intake appointment are scheduled, because all the participants need an appointment after all. The working of a Tabu search algorithm is explained below.

Tabu search

A Tabu search starts with an initial solution. Then multiple iterations are done to find a better solution until some stopping criterion is met. In the case of the intake appointment algorithm the stopping criterion is that the running time is 30 minutes. Finding a better solution is done by looking at the neighbourhood of the current best solution, although not all elements in this neighbourhood are considered. Only elements that are not in the Tabu-list are considered. The Tabu-list is a list of solutions that are not advantageous. This are the solutions that already have been evaluated, but where not better than the current best solution. In this way the algorithm tries to avoid solutions that will send the algorithm into a direction that is not optimal. In each iteration the solutions that are in the neighbourhood of the current best solution and that are not on the Tabu-list are evaluated. From these possible solutions the best solution is selected, which is then compared to the current best solution. If the new solution is better, it will be stored as best solution. Whereas the previous best solution is put on the Tabu-list. This process is repeated until the stopping criterion is met. At that moment the final solution of the algorithm is the best solution found until then.

2.3 Quality criteria

For all steps in the process, quality criteria are set. These are listed in Table 2.2.

Table 2.2: Quality criteria

Description	Requirement
The part of the people that is invited for the first time, that receives a pre-announcement.	$\geq 95\%$
The part of the people in the target group that receives an invitation.	$\geq 95\%$
The part of the participants that had a negative FIT result in the previous round that receives a next invitation within 22-26 months after the previous invitation.	$\geq 95\%$
The part of the participants with a positive FIT result that has an intake appointment within a radius of 40 km of the home address.	$\geq 95\%$
The time between creating the invitation (and sending the pre-announcement) and sending the invitation.	14 days
The part of participants with an assessable FIT where the result letter is send within 5 workdays of registration of the FIT result in ScreenIT.	$\geq 95\%$
The time between sending the result letter and the date of the intake appointment	$\leq$ 15 workdays.
The time between the intake appointment and the colonoscopy	$\leq$ 15 workdays.

The previous sections described the current situation at Bevolkingsonderzoek Oost (BVO Oost). All next chapters are based on our research according to the goals as explained in Section 1.2. We first start with the deterministic case where participation and referral rate are assumed to be fixed and we want to find an optimal matching between clients from postcode areas to week numbers and CCs.

Chapter 3

Deterministic model

In order to match the clients, week numbers and colonoscopy centres in the deterministic case a Mixed Integer Linear Program (MILP) is developed. The used sets, parameters and variables are declared in respectively Sections 3.1, 3.2 and 3.3. In Section 3.4 the constraints of the model are given and explained and Section 3.5 contains the objective function. The output parameters are described in Section 3.6.

3.1 Sets

This section describes which sets are used in the developed MILP. The clients live at postcode areas which are denoted by set

$$\mathcal{P} \quad = \quad \{1,\ldots,P \mid \text{postcode areas}\}\,, \quad p \in \mathcal{P}.$$

These clients should be linked to a colonoscopy centre. The colonoscopy centres are denoted by set

$$\mathcal{C} \quad = \quad \{1,\ldots,C \mid \text{colonoscopy centres}\}\,, \quad c \in \mathcal{C}.$$

The planning horizon is one year which is divided in time periods corresponding to weeks, which is denoted by set

$$\mathcal{T} \quad = \quad \{1,\ldots,T \mid \text{week numbers}\}\,, \quad t \in \mathcal{T}.$$

In practice the invitation is send to a client 7 weeks before a possible intake appointment should be planned. In this model this seven week shift is ignored, which gives only one time set. The clients are matched to a CC in a certain week, where this week represents the possible intake appointment week. The actual invitation week would be seven weeks in front of matched week. Another property of the time set, is that is is considered to be cyclic. This means that different years are not indicated. When week 52 is ended, the next week is week 1 again.

3.2 Parameters

This section describes the used parameters in this model. The first parameters are the participation and referral rate, these parameters indicate which fraction of the clients do participate in the screening and which fraction of the participants get a positive (unfavourable) result. In practice these values are stochastic variables, but in this deterministic model they take a fixed value. The following parameters are used to indicate the participation and referral rate.

$$\begin{aligned} PR \in (0,1] \quad &= \quad \text{participation rate} \\ RR \in (0,1] \quad &= \quad \text{referral rate} \end{aligned}$$

Second, we introduce the parameters that indicate the travel time between postcode areas and CCs. The first parameter contains for each combination of postcode area and CC the time that is needed to travel between them. It might be the case that a CC is nearby in travel time, but clients do not want to visit that CC, because of cultural preferences. For example, cities Arnhem and Nijmegen have small travel time to each other, but people from Arnhem do not want to go to Nijmegen and vice versa. To incorporate these preferences in the model the second parameter is used.

$$\begin{aligned} D_{p,c} &= \text{travel time [min] between postcode area } p \text{ and colonoscopy centre } c \\ K_{p,c} &= \begin{cases} 1 & \text{if combination } (p,c) \text{ is not desired} \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

Undesired combinations of postcode areas and CC are penalized by adding extra value to the travel time. The previous parameter values vary in different intervals, therefore $K_{p,c}$ should be weighted extra in determining which CC is nearest by. The weighting factor is denoted with α_k. Travel time varies in the interval 0 to 150 minutes, whereas the undesired combinations take only values 0 or 1. Therefore we decided that α_k should have value 100 in order to have a good balance between the two. The combined quantity for travel time and undesired combinations is called travel resistance and is indicated in the following parameter.

$$\hat{D}_{p,c} = D_{p,c} + \alpha_k \cdot K_{p,c} = \text{travel resistance between postcode area } p \text{ and colonoscopy centre } c$$

The following parameter indicates which CC is nearest by from postcode area p. Nearest by means in this case the CC that has the lowest travel resistance from postcode are p. It may be possible that from a postcode area two CCs are both nearest by. In this case the row of matrix $F_{p,c}$ corresponding to this postcode area contains two ones.

$$F_{p,c} = \begin{cases} 1 & \text{if colonoscopy centre } c \text{ is nearest by for postcode area } p, \text{ based on travel resistance} \\ 0 & \text{otherwise} \end{cases}$$

Each CC has a number of intake slots available in each week where clients can be invited. This is given in the following parameter.

$$I_c^t = \text{number of available intake slots in colonoscopy centre } c \text{ in week } t$$

Also the number of clients that should be invited is known and given by the following parameters. The first one indicates the number of clients living in each postcode area. The second parameter is the sum over all postcode areas, so the total number of clients that should be invited within region East.

$$\begin{aligned} N_p &= \text{number of clients living in postcode area } p \\ TN &= \sum_p N_p = \text{the total number of clients} \end{aligned}$$

A part of those clients is already invited earlier (2 years ago), these clients are the subsequent round clients. The subsequent round clients should be invited within 22 to 26 months after their previous invitation. Therefore, it should be known in which week they were invited 2 years ago. This is indicated by the following parameter.

$$E_p^t = \text{number of subsequent round clients in postcode area } p \text{ that should be invited around week } t$$

3.3 Variables

This section describes the variables that are used in the model for inviting clients from postcode areas to CC and weeks. Our main decision variable $x_{p,c}^t$ is restricted to take only integer values.

$$x_{p,c}^t = \text{number of clients from postcode area } p \text{ that are linked to colonoscopy centre } c \text{ in week } t$$

To guarantee that a feasible solution exist, despite not enough capacity is available, two dummy variables are introduced. These variables can both take only integer values.

$$\begin{aligned} d_p &= \text{number of clients from postcode area } p \text{ that cannot be invited} \\ e_p^t &= \text{number of subsequent round clients that cannot be invited within } \pm 2 \text{ months around week } t \end{aligned}$$

The last variable is used to distribute the workload in a CC equally over the year. This variable is restricted to lie in the interval $[0,1]$ because it represents the occupancy rate.

$$m_c = \text{the maximum workload at colonoscopy centre } c$$

3.4 Constraints

This section describes the constraints which a solution to the MILP should fulfil. Section 3.4.1 describes the general constraints of the MILP, whereas Section 3.4.2 describes the bounding constraints.

3.4.1 General constraints

Constraint (3.1) ensures that for each postcode area all clients are considered. The clients that are matched by $x_{p,c}^{t}$ are invited. When no more capacity is available, the remainder of the clients of postcode area p is denoted by the dummy variable d_p. These two variables sum to the total number of clients at postcode area p.

$$\sum_{c,t} x_{p,c}^{t} + d_p = N_p \qquad \forall p \tag{3.1}$$

The limited capacity constraint is denoted in (3.2). The fraction on the right hand side is the amount of invitations that can be send on I_c^t intake slots, because $PR \cdot RR$ is the expected fraction of clients that will get a positive FIT result and need an intake appointment. The number of invitations at colonoscopy centre c in week t cannot exceed this number.

$$\sum_{p} x_{p,c}^{t} \leq \left\lfloor \frac{I_c^t}{PR \cdot RR} \right\rfloor \qquad \forall c, t \tag{3.2}$$

To ensure that subsequent round clients are invited within ± 2 months of their previous invitation week t, constraints (3.3), (3.4) and (3.5) are introduced. The ± 2 months is translated into the interval $[t-8, t+8]$, because 2 month correspond to approximately 8 weeks. Depending on the week of the year of the original invitation week t one of the three constraints should be satisfied. If t is between week 9 and 44, the entire interval lays in the current year. In this case constraint (3.3) should hold. The total number of invitations from postcode area p within the time interval $[t-8, t+8]$ should equal minimally the number of subsequent round clients from postcode area p with previous invitation week t. When this is not possible the dummy variable e_p^t will be used. The CC that is used for this invitations is not important, therefore we sum over all possible CCs.

$$\sum_{c} \sum_{t'=t-8}^{t'=t+8} x_{p,c}^{t'} + e_p^t \geq E_p^t \quad \forall p, 9 \leq t \leq 44 \tag{3.3}$$

If t is at the begin of the year ($t \leq 8$) a part of the interval $[t-8, t+8]$ lies in the previous year. Therefore the summation over t' is different. It takes all weeks in the current year and some weeks at the end of previous year. This in shown in constraint (3.4). The different years are not indicated because we use a cyclic set of week numbers. When week 52 is ended, the next week is week 1 again.

$$\sum_{c} \sum_{t'=1}^{t'=t+8} x_{p,c}^{t'} + \sum_{c} \sum_{t'=44+t}^{t'=52} x_{p,c}^{t'} + e_p^t \geq E_p^t \quad \forall p, t \leq 8 \tag{3.4}$$

If t is at the end of the year ($t > 44$) a part of the interval $[t-8, t+8]$ lies in the next year. Therefore the summation over t' is different. It takes all weeks in the current year and some weeks at the beginning of next year. This in shown in constraint (3.5).

$$\sum_{c} \sum_{t'=t-8}^{t'=52} x_{p,c}^{t'} + \sum_{c} \sum_{t'=1}^{t'=t-44} x_{p,c}^{t'} + e_p^t \geq E_p^t \quad \forall p, t > 44 \tag{3.5}$$

CCs want that the intake appointments are scheduled evenly distributed over the year. To achieve this constraint (3.6) is used. This constraint states that the fraction of used capacity in a certain week at a certain CC should be smaller than the maximum workload in that CC. This should hold for all weeks. By minimizing the variable m_c, the maximum occupancy rate in CC c will be minimized.

$$\frac{\sum_p x_{p,c}^t}{\left\lfloor \frac{I_c^t}{PR \cdot RR} \right\rfloor} \leq m_c \quad \forall c, t \tag{3.6}$$

This implies that in busy weeks, where a lot of intake slots are used, more buffer capacity is available than in a week with less available intake slots. In other words, in all weeks the occupancy rate will be equal.

3.4.2 Bounding constraints

The last constraints are bounding constraints for the variables. These constraints indicate which values the different variables can take, which decrease the size of the solution space. First note that no variables can take negative values, so they are bounded by zero. Constraint (3.7) indicates that you cannot invited more clients then the number of clients that lives at postcode area p. This also holds for the dummy variable, which is shown in constraint (3.8).

$$0 \leq x^t_{p,c} \leq N_p \qquad \forall p, c, t \tag{3.7}$$

$$0 \leq d_p \leq N_p \qquad \forall p \tag{3.8}$$

Constraint (3.9) indicates that the number of subsequent round clients that are not invited in the specified interval around week t cannot exceed the total number of subsequent round clients of that week. This should hold for all postcode areas.

$$0 \leq e^t_p \leq E^t_p \qquad \forall p, t \tag{3.9}$$

Constraint (3.10) shows that the maximum workload in a CC cannot be greater than 1 for each week. These workloads are measured in fraction of used intake capacity.

$$0 \leq m_c \leq 1 \qquad \forall c, t \tag{3.10}$$

Finally, we have variables that can only take integer values, so this gives the integrality constraints shown in equation (3.11).

$$x^t_{p,c}, \in \mathbb{Z} \quad d_p, \in \mathbb{Z} \quad e^t_p \in \mathbb{Z} \qquad \forall p, c, t \tag{3.11}$$

3.5 Objective function

The last part of the Mixed Integer Linear Program is to formulate an objective function in order to find an optimal solution. This section explains which objective function is used and why. Equation (3.12) shows the objective function, it minimizes 5 different factors which are all weighted with a scaling factor α.

$$\text{minimize} \quad \alpha_s \cdot \sum_{p,t} e^t_p + \alpha_r \cdot \sum_p d_p - \alpha_n \cdot \sum_{p,c,t} x^t_{p,c} \cdot F_{p,c} + \alpha_d \cdot \sum_{p,c,t} x^t_{p,c} \cdot \hat{D}_{p,c} + \alpha_o \cdot \sum_c m_c \tag{3.12}$$

The five factors are explained below:

Subsequent round clients: $\alpha_s \cdot \sum_{p,t} e^t_p$
It is desired that all clients that have participated before are invited in an interval of ± 2 months around their previous invitation week. Therefore, the number of clients for which this is not possible (e^t_p) is minimized.

Rest group: $\alpha_r \cdot \sum_p d_p$
The goal is to invite all clients of the target group. Therefore, the number of clients that cannot be invited (d_p) is minimized.

Nearest CC: $-\alpha_n \cdot \sum_{p,c,t} x^t_{p,c} \cdot F_{p,c}$
When a client should have an intake appointment, this client wants to go to the CC that is nearest by for this client. In order to minimize the number of rescheduled intake appointments, the number of clients that is linked to the nearest CC is maximized. This is equivalent with minimizing the number of clients linked to the nearest CC multiplied with -1. This is a greedy rule, but guarantees a limited number of rescheduled intake appointments. It is assumed that a client who is not linked to his nearest by CC, will more likely reschedule his appointment. Remember that nearest by is defined as the CC with the lowest travel resistance.

Total travel resistance: $\alpha_d \cdot \sum_{p,c,t} x^t_{p,c} \cdot \hat{D}_{p,c}$
Next to the nearest CC factor, the total travel resistance of all clients is minimized. Which implies that clients who are not linked to the nearest by CC, will be linked to a CC that is as close by and most desired as possible. By adding this factor some clients might not reschedule their intake appointment, because they are willing to travel a bit longer than to the nearest CC.

Workload: $\alpha_o \cdot \sum_c m_c$
Finally the maximum workload in each CC is minimized. This ensures that each week of the year has the same occupancy rate. This is a positive effect for the workload in the CCs, because they want an evenly levelled workload over the year.

To determine the values of the scaling factors we first need to know which objective function part is most important. The importance factors are shown in Table 3.1 and are determined by looking at the three different stakeholders perspectives. The most important stakeholder is the RIVM, who determines the content and targets of the screening process. At least 95% of all subsequent clients should be invited within the pre specified interval around previous invitation week t. Also at least 95% of the clients in the target group should be invited. The RIVM also states that subsequent clients have priority, because they are already part of the screening process. These rules set by the RIVM imply that the *subsequent round clients* factor is more important than the *rest group* factor in the objective function. The second stakeholder in the screening process are the clients, who are less important than the RIVM. The clients want to have an intake appointment in a CC that is nearby and desired. The corresponding objective function factors are *nearest CC* and *total travel resistance*. The first is more important than the second, because the number of rescheduled intake appointments should be minimized. The third and least important stakeholder are the CCs. They want to have an equally spread workload over the year, which is achieved by the *workload* factor in the objective function. To indicate the importance difference between the different stakeholders, the RIVM has importance factor of order thousand, the clients of order hundred and the CCs of order ten. The exact values of the importance factors are determined in consultation with BVO Oost.

However, the importance factor is not equal to the scaling factor. The different objective function parts take values that lie in different intervals, so they cannot be compared equally to each other. In order to solve this, the different parts need to be normalized. For this we need to have the size class of each objective function part. The size class indicates how large the values are that the objective function part can take. The parts *subsequent round clients*, *rest group* and *nearest CC* all correspond to number of clients and are therefore of the same size class 1. The travel resistances varies between 0 and 150, so one client corresponds to a travel resistance of about 100. The workload in the CCs is measured in percentages (occupancy rate). We state that one client in the objective function is comparable to 1% occupancy rate. This implies that the size class of *workload* is 0.01.

The actual scaling factors α are determined by dividing the importance factor by the size class. This means that the importance factors are all scaled to the same size class of order 1. The size classes and final scaling factors are listed in Table 3.1.

Table 3.1: Determination of the scaling factors in the objective function.

Objective function part	Importance factor	Size class	Scaling factor
Subsequent round clients	2000	1	$\alpha_s = 2000$
Rest group	1000	1	$\alpha_r = 1000$
Nearest CC	500	1	$\alpha_n = 500$
Total travel resistance	100	100	$\alpha_d = 1$
Workload	10	0.01	$\alpha_o = 1000$

3.6 Output parameters

This section gives the output parameters that can be calculated after finding an optimal solution to the MILP. These output parameters say something about the properties and quality of the optimal solution.

- Total number of invitations sent.

$$TNI = \sum_{p,c,t} x^t_{p,c}$$

- The rest group is the number of clients that is not invited.

$$RG = TN - TNI = \sum_p N_p - \sum_{p,c,t} x^t_{p,c} = \sum_p d_p$$

- The rest group percentage is the fraction of clients that is not invited.

$$RGP = \frac{RG}{TN} \cdot 100\%$$

This output parameter indicates whether the goals set by the RIVM are met.

- Number of invitations in week t.

$$NI^t = \sum_{p,c} x^t_{p,c} \qquad \forall t$$

  This output parameter is used to compare the found invitation strategy with the current strategy of BVO.

- Average travel resistance.

$$TATT = \frac{\sum_{p,c,t} x^t_{p,c} \cdot \hat{D}_{p,c}}{TNI}$$

  This output parameter indicates whether the travel resistances are acceptable for the clients.

- The percentage of clients that is linked to a CC further away than the nearest CC.

$$PNN = \frac{TNI - \sum_{p,c,t} x^t_{p,c} \cdot F_{p,c}}{TNI} \cdot 100\%$$

  This output parameter indicates both the amount of satisfied clients and whether the intake slots are divided in the right way over region East.

- The percentage of subsequent round clients that is not invited within 22 to 26 months after the previous invitation.

$$PSC = \frac{\sum_{p,t} e^t_p}{\sum_{p,t} E^t_p} \cdot 100\%$$

  This output parameter indicates whether the goals set by the RIVM are met.

- Adherence, the fraction of clients from postcode area p that are linked to colonoscopy centre c.

$$AH_{p,c} = \frac{\sum_t x^t_{p,c}}{\sum_{c,t} x^t_{p,c}} \cdot 100\% \qquad \forall c, p$$

  This output parameter indicates which postcode areas are linked to which CCs and will be used to compare the results with the current situation at BVO.

- Occupancy in colonoscopy centre c.

$$O_c = \frac{NI_c}{\sum_t \left\lfloor \frac{I^t_c}{PR \cdot RR} \right\rfloor} \cdot 100\% \qquad \forall c$$

  This output parameter indicates how busy it is at the CC. A CC with a low occupancy rate probably offers to many intake slots for its region.

- Maximum workload at CC c, m_c
  This output parameter indicates how busy it is at the CC. When this output parameter has value 1, all the available intake slots of this CC are used. A lower value of this output parameter implies a lower occupancy in the CC.

Chapter 4

Input data

This chapter explains which input data is used during this research. The used data can be divided over 5 categories namely geographical information, number of clients, colonoscopy centres, travel times and single value parameters. For each of these categories the following sections explain which data is used and how the data is modified in order to fit in the developed model.

4.1 Geographical information

The region East that is considered in this research consists of the Dutch provinces Gelderland and Overijssel. These provinces contain 79 councils and these councils are each divided in postcode areas. A postcode consists normally of 4 digits and 2 letters, for example 1234AB. Region East consists of 790 postcode areas that have the same 4 digits, which is called PC4. When a letter is added to a PC4, multiple PC5 areas arise. Region East contains 6352 postcode areas of type PC5. The total number of PC6 areas in region East is 94,300. This data is obtained from the CBS (Centraal Bureau voor de Statistiek) of the Dutch government with a public licence [Nationaal Georegister, 2018, CBS, 2017].

The developed model of Chapter 3 can be used for both PC4 and PC5 areas. In theory the model can even be used for PC6 areas or individual locations of clients, but due to dimensional issues this is not possible in practice. When individual clients are considered the set P of postcode areas transforms into the set P of persons and the number of inhabitants at p equals 1. In other words, the decision variables become binary variables instead of integers. All other variables, parameters and constraints can be trivially converted to individuals in a similar way.

Both PC4 and PC5 aggregation levels are tested in the model. However, the running time of finding a solution takes in the case of PC4 areas about 3 minutes and for PC5 areas more than 2 hours. The results of both aggregation levels of postcodes areas are similar. Therefore we decided that PC5 areas do not give more insight than PC4 in comparison to the extra effort that has to be taken (running time). Also, the data that is needed in this case is less confidential. Therefore, in the remainder of this research we use PC4 areas.

4.2 Number of clients

The number of inhabitants in each PC4 area is given by CBS [2017], for each age category, that consists of 5 consecutive ages, for example 55 years till 59 years. However, the number of clients for each specific age is required. Multiple datasets are generated by distributing the number of inhabitants in a PC4 area in an age group randomly over the various ages. For each inhabitant a random age is drawn, by using a probability distribution that is based on the age population within the councils of region East.

4.2.1 Age probability distribution

The probability of having a specific age is calculated with help of the number of inhabitants in a council with certain age. These numbers are obtained from the CBS, [CBS StatLine, 2017]. The probability of having age y is the fraction of people that have age y in the age group over 5 ages containing age y. These numbers are known for each council. After evaluating these numbers, it can be said that there is no substantial difference between the different councils. This is because the standard deviation is small enough. Therefore, the same age distribution for the entire region East will be used. The probability of having a specific age in an age group is given in Table 4.1. These probabilities differ significantly from probability 1/5. This is proved by testing all ages in an age group against each other by a paired t-test. Each pair of ages $a1$ and $a2$ have a statistical significant different mean at a confidence level of 95%. The sample size in this test was 79, the number of different councils. Therefore, the shown probabilities in Table 4.1 are used for drawing a random age for each inhabitant.

Table 4.1: The probability distribution of the different ages.

Age group	Age	probability
55-59	55	0.21
	56	0.20
	57	0.20
	58	0.20
	59	0.19
60-64	60	0.21
	61	0.20
	62	0.20
	63	0.19
	64	0.20
65-69	65	0.20
	66	0.19
	67	0.20
	68	0.20
	69	0.21
70-75	70	0.26
	71	0.19
	72	0.19
	73	0.18
	74	0.18
75-79	75	0.22
	76	0.22

4.2.2 Subsequent round clients

It should also be known how many subsequent round clients should be invited in which week. For this it is assumed that all clients with age 57 or higher are subsequent round clients. This is indeed the case when the screening process is no longer in a start up mode and no rest groups are present. For each postcode area it is known how many subsequent round clients live there. To determine the week in which the subsequent round clients were invited, the historic invitation strategy of BVO Oost is used. This indicates in which week how many invitation were send in 2017. These numbers are converted to percentages of the total number of invitations in that year. The subsequent round clients for the MILP are iteratively assigned to a week in the following way. The subsequent round clients from the first postcode area are assigned to week 1. The subsequent round clients from the next postcode area are assigned to week 2. When week 52 is reached, we go back to week 1. This is repeated as long as the number of subsequent round clients in a week does not exceed the total number of subsequent round clients that is allowed in a week. This allowed number of subsequent round clients in a week is determined by the percentages of the historic invitation strategy and the total number of subsequent round clients in this dataset. Figure 4.1 shows an example of the number of subsequent round clients that should be invited in each week of the year.

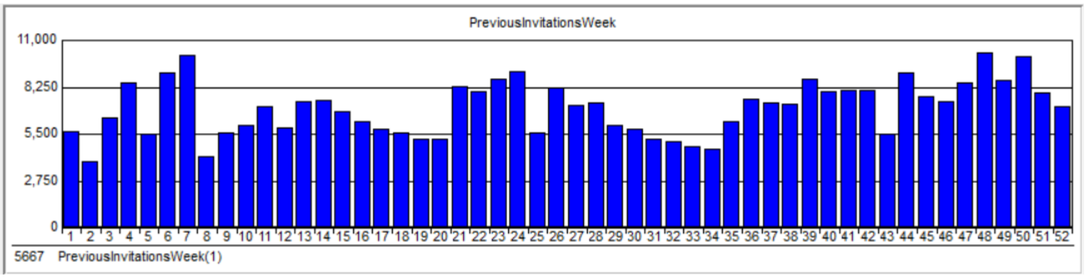


Figure 4.1: The number of subsequent round clients that should be invited in each specific week of the coming year.

4.3 Colonoscopy Centres

The information about CCs and their capacity is obtained from Bevolkingsonderzoek Oost. They have made arrangements with the CCs about the total number of intake slots that a CC centre has available for a year. In total 22 CCs are available of which 4 are not located within region East. This is because clients who live on the boarder of region East might prefer to go to a close-by CC that is in for example region North. Also clients from region North might want to visit a CC in region East because that is closer by than a CC in their region. In order to handle these boundary conditions we reserve capacity for clients from another region in some of the CCs in region East. We also add a few CCs from outside region East to our model which have a small number of intake slots available for clients from region East. The total number of available intakes slots per year are listed in Table 4.2. The CC denoted with letters A-R are situated in region East, the four CC that start with the letter Z are situated outside region East.

Table 4.2: The total number of intake slots that is available in each CC.

CC	Yearly Capacity
A	1100
B	600
C	600
D	1100
E	1000
F	335
G	725
H	900
I	700
J	1800
K	480
L	600
M	580
N	600
O	165
P	585
Q	1161
R	396
ZA	46
ZB	20
ZC	180
ZD	40

4.3.1 Holidays

The yearly capacity should be divided over the year for each CC. By analysing historical data of the number of available intake slots in each week it becomes visible that CCs have less intake slots available in holiday weeks. Figure 4.2 shows that the holiday weeks are around week 1, 6, 17-18, 28-33, 41 and 52 in 2016. These holiday weeks are known for each CC, because they are situated in a holiday region in the Netherlands.

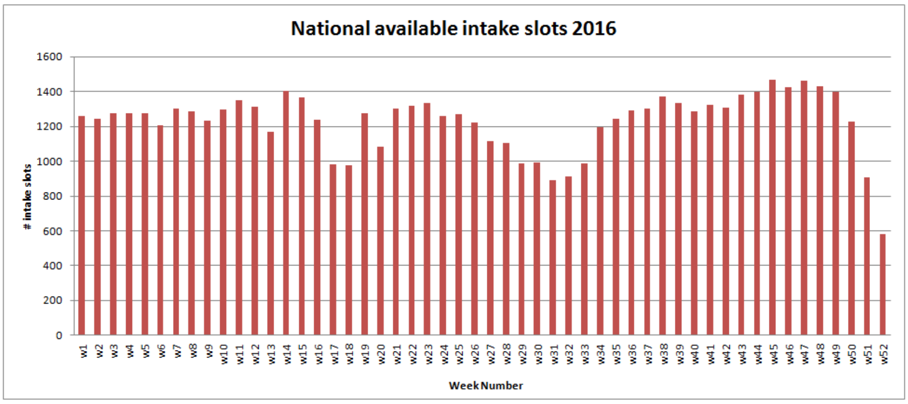


Figure 4.2: The available intake slots per week in the whole of the Netherlands in 2016.

In our model, initially each week of the year gets 1/52 of the yearly number of intake slots. For the holiday weeks this number is now reduced by 50%. The slots that arise again are equally distributed over the non holiday weeks starting at the beginning of the year. This procedure of distributing the intake slots results in a division of the intake slots over the year, an example is shown in Figure 4.3.

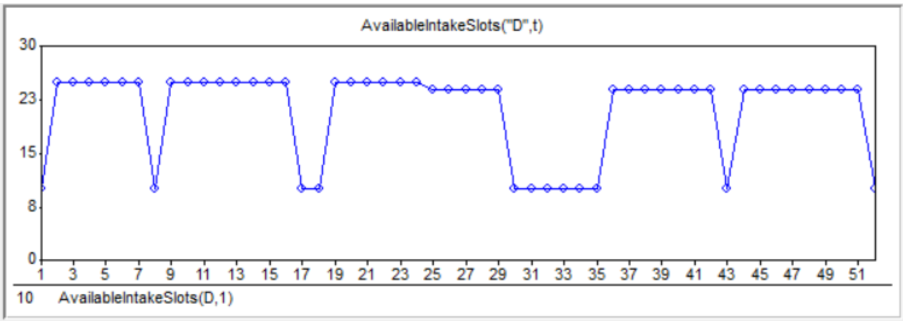


Figure 4.3: An example of the intake slot division over the year.

4.4 Travel Resistance

Currently, BVO Oost uses euclidean distances between postcode areas of clients and CCs. This research uses travel resistance, because this better represents the effort that a client should take to reach the CC better. The travel resistance consists of two factors, namely travel time and undesired postcode-CC combinations. In this way we want to accomplish that clients are only send to a CC of their choice. Both factors are explained below in more detail.

4.4.1 Travel time

Clients do not want to travel long to a CC. Travel times represent in the best way which CC is closer by than another CC. This is because, in the case of Euclidean distances it can occur that a CC is nearby, but reaching the destination is difficult because a river, lake or forest blocks the direct way. These problems do not occur when travel times are used. The travel time between the postcode areas (PC4) and CC is derived from the impedance table of Object Vision [Object Vision, 2011]. This is allowed under the CC-SA-BY-NL-30, Creative Commons licence. This table contains the travel time by road in minutes between all pairs of PC4 areas in the Netherlands. This travel time is determined by the length and type of the road and the number of crossroads. As start and end point the central point of addresses is used. This table gives the travel time between a PC4 area and a CC (which stands in a PC4 area).

4.4.2 Unwanted geographical combinations

It might occur that a postcode area has a small travel time to a CC, but the client does not want to visit that CC for cultural reasons. An example is that regions Arnhem and Nijmegen have a small travel time between each other, however the cultural distance between them is very large. People who live in

Arnhem will not go to Nijmegen and vice versa. In such cases we want the model to avoid matching these areas to each other. For this we need to know for each postcode area - CC combination whether it is desired or not. This data can be retrieved from historical rescheduling movements. However, the current situation does not use PC4 area and therefore the rescheduling movements are not known. Another reason for not using rescheduling movements, is that the screening process is only executed for a couple of years. This means that not enough data is available to make statistically correct conclusions. In conclusion, for this research the unwanted geographical combinations are disregarded. The parameter that indicates which postcode area - CC combinations are not desired is set to the default value of zero. The travel resistance is therefore the same as the travel time.

4.5 Single value parameters

Single value parameters are the parameters that do not depend on any of the sets that are used in the model. For the participation rate and the referral rate the values 73% and 4.7% are used respectively in the deterministic MILP. These rates are based on historical data and forecasts by BVO.

During the start of the colon cancer screening program in 2014 the participation rate was about 70%. This rate increased over the years, because the screening program became more known by the public. In 2017 the average participation rate was 75%. The forecasts of BVO Oost say that in the coming years the participation rate will lie around 73%. Although this is difficult to predict, because clients over the whole of the Netherlands will react differently. Taking these different values for the participation rate into account, it is decided to use the following values for the participation rate:

$$PR \in [70\%, 76\%], \quad \text{with mean } 73\% \text{ in the deterministic case}$$

In 2017 in total 1,366,486 test where analysed in the laboratories, of which 70,397 gave a positive (unfavourable) result. Therefore the referral rate is 5.1% on average.

At this moment the screening organization uses a participation rate of 70% and a referral rate of 4.5% in the software ScreenIT, but these rates do not match reality. Instead, these rates are chosen to be able to invite all clients. With higher rates this will be much more difficult, because less clients can be invited per available intake slot. The difference in the participation rate (70% VS 75%) has less impact than the difference in the referral rate (4.5% VS 5.1%). An other reason for using a lower referral rate is that the referral rate will decrease if more subsequent round clients participate. This will happen in the future, because from 2019 onwards only clients of one age (55 years) will be first round clients in a screening year. Approximately 11% of the clients in the target group will then be invited for the first time. The other 89% of the clients already has participated before and will have a lower referral rate. According to BVO Oost the referral rate for first round clients is 5.9% and for subsequent round clients the referral rate is 4.6%. Taking a weighted average of these referral rate results in a predicted referral rate of $0.11 \times 5.9 + 0.89 \times 4.6 = 4.7\%$. BVO Oost has decreased this to 4.5% because a small part of the clients that get a positive result do not show up for an intake appointment, but this is not based on real numbers. Therefore this research uses the average referral rate of 4.7% in the deterministic case. To determine the interval in which the referral rate can vary a binomial distribution is used.

For each week and each CC on average 380 invitation are send. This results in approximately $380 \times 0.73 \approx 280$ test to be analysed that correspond to a week in a CC. Each of these tests has a probability of 0.047 to be positive. Therefore, the number of positive results in that case is binomially distributed with $n = 280$ and $p = 0.047$. By repeating a drawing from the binomial distribution 100 times, a 95%-confidence interval for the number of positive results is obtained. The borders of this confidence interval correspond to a referral rate of 4.3% and 5.1%. These values are taken to be the minimum and maximum values in between which the referral rate is likely to vary. Therefore the following values for the referral rate are used in this research:

$$RR \in [4.3\%, 5.1\%], \quad \text{with mean } 4.7\% \text{ in the deterministic case}$$

4.6 Data sets

The geographical information, travel times, and participation and referral rate are considered to be fixed, but the age of the clients are randomly assigned. Therefore, eight different data sets are generated and these datasets are used to test the model. Four different random seeds are used to generate the number of clients of each age. Each of these divisions result then in two different data sets, namely one that only includes clients that have an even age and one that that only includes clients with odd ages.

This is because each client needs to get invited every two years, so each year half of the target group should be invited. For this, the age of 76 is included because someone who is currently of age 76, was of age 75 last year, and therefore belongs to the target group. The eight different data sets slightly differ in the number of clients that live in each postcode area.

Chapter 5

Results deterministic model

The developed model of Chapter 3 is programmed in AIMMS, also the input data from Chapter 4 is loaded into AIMMS. A solution will be found by the software Gurobi. Gurobi uses LP-relaxation and branch and bound techniques to solve an MILP. This chapter contains the results of the deterministic model. An optimal solution is found for different scenarios. The different solutions and corresponding output parameters (from Section 3.6) are compared. The results are also compared with the current situation at BVO Oost.

5.1 Scenarios

The deterministic MILP model is solved for five different scenarios. These scenarios differ in the values that are used for the participation and referral rate. Each scenario is indicated with a letter as shown in Table 5.1. Scenarios A uses a participation rate of 73% and a referral rate of 4.7%, the nominal values. The other scenarios use the extreme values in the intervals for participation and referral rate. By modifying the participation rate and referral rate we analyse the effect of these rates to the output parameters. Each scenario consists of eight experiments. These experiments correspond to the eight datasets as described in Section 4.6. Table 5.2 gives an overview of the different datasets/experiments. By running the scenario for each dataset, a 95% confidence intervals of the output parameters can be calculated for each scenario.

Table 5.1: The different scenarios that are evaluated.

Scenario	Participation rate	Referral rate
A	73%	4.7%
B	76%	5.1%
C	70%	5.1%
D	76%	4.3%
E	70%	4.3%

Table 5.2: The different experiment numbers that are evaluated.

Experiment	Random seed	Ages
1	122113	odd
2	122113	even
3	935274	odd
4	935274	even
5	194027	odd
6	194027	even
7	762588	odd
8	762588	even

5.2 Scenario results

Table 5.3 shows the results of the five scenarios. The first three columns show which scenario it is and which participation and referral rate it uses. In each scenario the number of clients varies over the experiments. The 95% confidence interval of the number of clients is $[411, 925\ ;\ 413, 494]$, which is the size of the target group for one single year. In each data set the clients are differently distributed over the postcode areas. The total number of clients is of the same size as it is in reality. The other columns correspond to the output parameters and contain the 95% confidence interval for the output parameter in the scenario.

Table 5.3: The 95% confidence intervals of each scenario.

Scenario	PR	RR	Rest group %	Average Travel Time [min]	% not in nearest CC	% not within 22-26 months
A	73%	4.7%	[3.1 , 3.5]	[16.2 , 16.4]	[17.2 , 17.4]	[0 , 0]
B	76%	5.1%	[14.3 , 14.6]	[15.7 , 15.9]	[18.5 , 19.1]	[3.6 , 3.8]
C	70%	5.1%	[6.8 , 7.2]	[16.2 , 16.3]	[17.3 , 18.0]	[0 , 0]
D	76%	4.3%	[0 , 0]	[16.4 , 16.5]	[18.2 , 18.5]	[0 , 0]
E	70%	4.3%	[0 , 0]	[15.7 , 15.7]	[15.5 , 15.7]	[0 , 0]

This table shows that in the case of a participation rate of 73% and a referral rate of 4.7%, it is not possible to invite all clients, as about 3.3% of the clients cannot be invited. This aligns with the current situation at BVO. The rest group obviously becomes smaller when the participation rate and referral rate are lowered. The effect of a 0.8% decrease in the referral rate is larger than a decrease of 6% in the participation rate. This makes sense because the relative effect in small percentages is larger.

The results also show that the average travel time increases when the rest group decreases. This is caused by the fact that the rest group will exist of clients that have long travel times to CCs because these clients are ”the most expensive” to invite with respect to the used objective function. An average travel time of about 16 minutes is acceptable, but an average does not say everything. Therefore we look at the $\%$ of clients that is not invited in the nearest CC. Depending on the scenario, 15.5% to 19% of the clients is not invited in the CC that is nearest by. This indicates that the capacity of intake slots is not distributed over region East in a way that aligns with the distribution of clients. This is discussed in more detail in Section 5.2.1. The percentage of clients that is not linked to the nearest CC increases when the participation and referral rate increase. This can be explained by the fact that higher rates imply that less people can be invited on an intake slot. Therefore less people can be linked to a CC, and also less people can be linked to the nearest CC. The clients that are not linked to the nearest CC have an extra travel time, or are likely to reschedule their appointment. In the last case the waiting time until this intake appointment will increase. However, in the nominal scenario A the rescheduling percentage is expected to lie around 17%, which is much better than it is in the current situation at BVO.

The last output parameter in Table 5.3 is the percentage of subsequent round clients that are not invited within 22 to 26 months after their previous invitation. In almost all scenarios this percentage is 0, which means that it is possible to invite all subsequent round clients within their preferred time interval. Scenario B is the exception with a percentage of clients not invited in the right time interval of 3.7%. This is caused by the fact that scenario B has both the highest participation rate and the highest referral rate. Therefore the amount of invitations at an intake slots is the smallest of all scenarios, which makes it impossible to invite the subsequent round clients in the preferred time interval. The subsequent round clients can also be part of the rest group, but this is not preferable. As long as it is possible the rest group will only contain clients that are invited for the first time.

5.2.1 CC Capacity distribution

In scenario E the participation rate and referral rate are the lowest, which means that there is enough capacity available. This results in an invitation strategy that has space for modifications as the number of invitations in each week and CC is not equal to the maximum available capacity. Therefore some CC have a lower occupancy rate than 100%. Table 5.4 shows the average occupancy rates of the CCs in scenario E. CCs 'E', 'I', 'J' and 'N' clearly have a lower occupancy rate than 100%. This indicates that these CCs offer more intake slots than needed in the region. CCs with an occupancy rate of 100% are likely to offer less intake slots than needed.

The CC indicated with letters 'E' and 'I' have the lowest occupancy rates according to the results of the model. In practice these CCs have lower occupancy rates than other CCs as well. This indicates that the model represents reality. The CCs with an occupancy rate lower than 100% are the CCs for which the distribution workload constraint is used. These CCs have a levelled workload that is the same (up to some rounding errors, due to integer numbers in available capacity) in each week. This workload equals the occupancy rate as shown in Table 5.4.

Table 5.4: The average occupancy rates for each CC in scenario E.

CC	Occupancy [%]
A	93.6
B	100
C	94.3
D	100
E	72.1
F	100
G	100
H	100
I	42.5
J	86.8
K	100
L	100
M	100
N	81.7
O	100
P	100
Q	91.4
R	100
ZA	100
ZB	100
ZC	100
ZD	100

The amount of needed intake slots in a CC can be calculated before using the MILP. This is done by determining for each CC the number of clients that lives closest by to that CC. This gives the needed number of intake slots for each CC when all clients should be linked to the nearest CC. These numbers are shown in Table 5.5. The difference in the last column of this table indicates whether enough intake slots are offered by the CCs. A positive difference means that to many slots are offered, a negative difference means that the CC offers to little intake slots for its region. It is clear that the distribution of intake slots does not align with distribution of clients of the region East. The total amount of offered intake slots is larger than the needed number of intake slots, however not all clients can be linked to the nearest by CC.

When Tables 5.4 and 5.5 are compared, it seems that they do not align with each other. For example, CC 'A' offers 1100 slots, where only 604 slots are needed to fulfil the demand of its region. This would correspond to an occupancy rate of 55% when all clients are linked to the nearest CC. However, this is not possible. Therefore, the extra slots that CC 'A' offers are used to invite clients from areas around CC 'A'. These clients could not be invited in their nearest CC, but CC 'A' is the next best option. In this way the occupancy rate of CC 'A' increases to 93.6%, which is much larger than the expected 55%.
CCs that offer to little intake slots for their region (negative difference) all have an occupancy rate of 100%. This is what you would expect, because all the slots that are available will be used for the clients that have such a CC as nearest CC. Clients that cannot be invited on those negative difference CCs any more, will be linked to other CCs. Which results in higher occupancy rates in the CCs with enough capacity.

Table 5.5: The amount of intake slots actual needed for each CC in scenario E.

CC	Offered Intake slots	Needed Intake slots	Difference
A	1100	604	+496
B	600	611	-11
C	600	552	+48
D	1100	1157	-57
E	1000	575	+425
F	335	585	-247
G	725	338	+387
H	900	691	+209
I	700	314	+386
J	1800	1309	+491
K	480	361	+119
L	600	849	-249
M	580	707	-127
N	600	394	+206
O	165	749	-584
P	585	689	-104
Q	1161	689	+472
R	396	872	-467
ZA	46	64	-18
ZB	20	34	-14
ZC	180	249	-69
ZD	40	83	-43
Total	13,713	12,473	+1240

CC 'E' is an example of a CC that offers more intake slots than needed and therefore has an occupancy rate lower than 100%. Figure 5.1 shows the number of clients invited in each week in blue and the available invitation capacity in yellow. It is clear that the blue bars are approximately 72% of the yellow bars, witch corresponds to an occupancy rate of 72%. This occupancy rate is constant over the year, because one of the constraints in the MILP makes sure this happens. The unused capacity can be used by the CC as buffer capacity, when unexpectedly the amount of intake appointments increases. When more clients are invited the probability that more intake slots are needed is larger than when less clients are invited. This is because of the fact that each client has the same probability of needing an intake appointment (getting a positive result). When more clients are invited, you can expect more extra intake appointments. Therefore, the buffer capacity in busy weeks is larger than the buffer capacity in quite weeks, which corresponds with the same occupancy rate in all weeks.

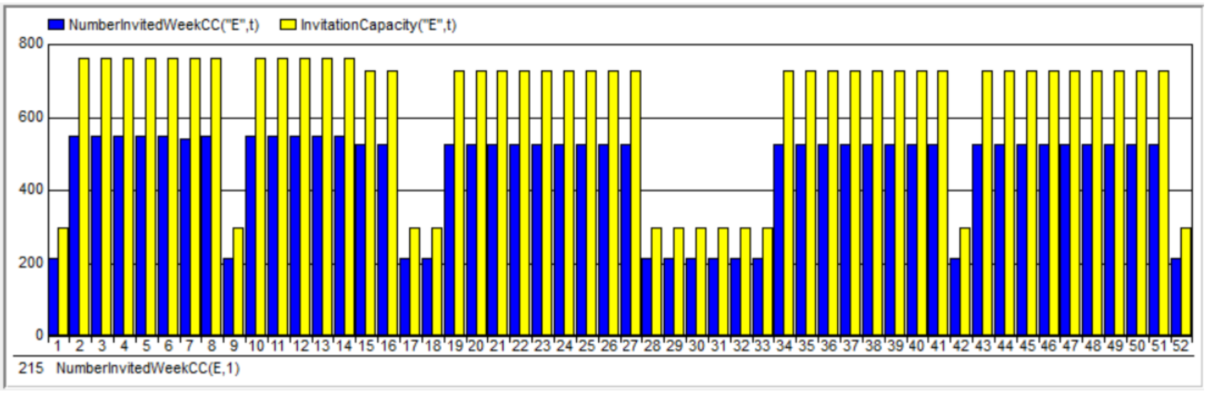


Figure 5.1: The available and used invitation capacity of CC 'E' in scenario E.

The nice distribution of workload as shown in Figure 5.1 need not always be possible. Figure 5.2 also shows the available (yellow) and used capacity (blue) of CC 'E' in scenario E, but then in a different experiment. In this experiment the distribution of clients over the postcode areas and the distribution of subsequent round clients over the year is different. Subsequent round clients are far more important than distributing workload, therefore, the workload distribution in Figure 5.2 is not entirely constant. The maximum workload is not equal to the mean workload. When the workload would have been constant, as the CCs want it to be, some subsequent round clients would not be invited within their 22 to 26 months interval after their previous invitations. This is not allowed and therefore the workload distribution is disturbed slightly.

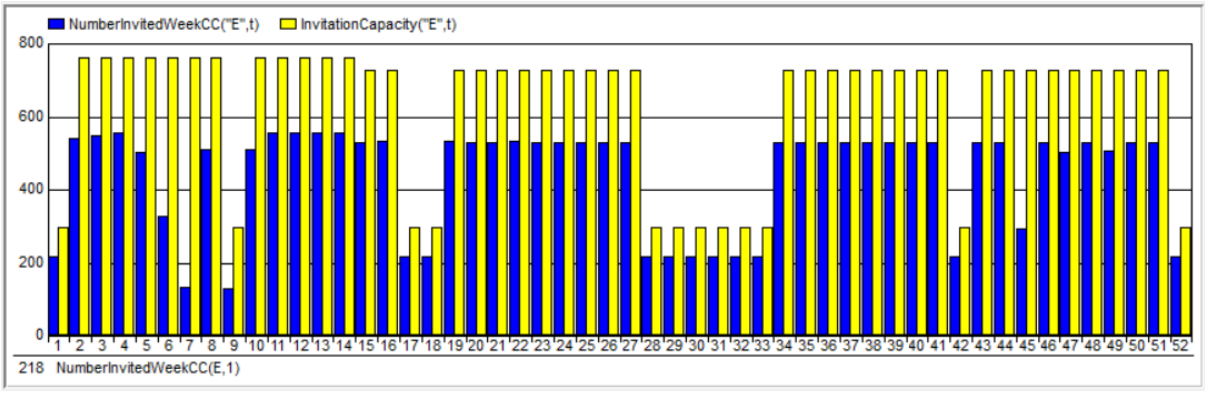


Figure 5.2: The available and used invitation capacity of CC 'E' in scenario E, where subsequent round clients are more important.

5.3 Invitation strategy

The total amount of invitations in region East, according to the developed model and used datasets is shown in Figure 5.3. The holiday weeks are clearly visible by the lower number of invitations in those weeks. Within region East multiple holiday regions are present which explains the varying number of invitations in some holiday weeks.

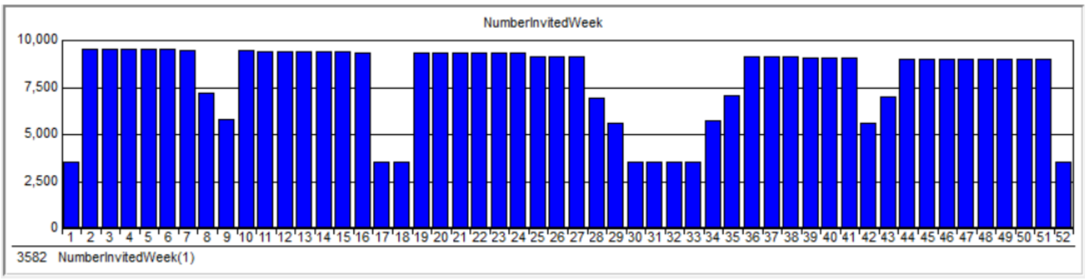


Figure 5.3: The total number of invitations in each week of the year.

The total number of invitations consists of two parts, the subsequent round clients and the first round clients. Figure 5.4 shows for all weeks of the year the number of subsequent round clients that should be invited in that week. These clients are not equally divided over the year. However, the model ensures that an equal distribution of clients happens in the new invitation year. This is possible by using two methods. First we use the interval of ± 2 months around the predefined week for subsequent clients, which gives a more evenly distributed number of subsequent round clients over the year. In the second place, the first round clients are invited at weeks where capacity is left. The new invitation strategy is shown in Figure 5.3, where also the first round clients are included. We see that by using both methods, we get an equal distribution of invitations over the year in total.

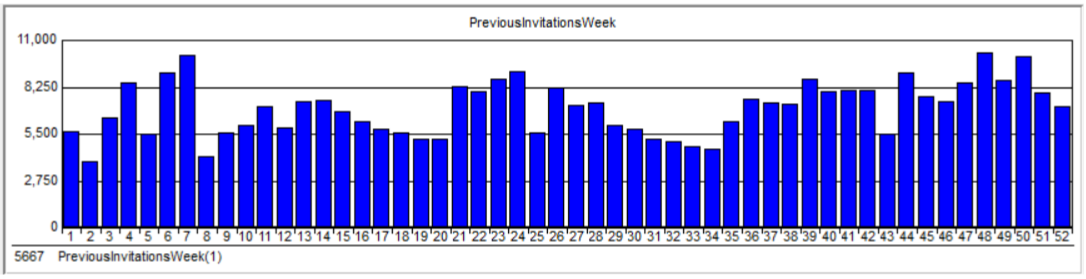


Figure 5.4: The number of subsequent round clients that should be invited in each week.

Figure 5.5 shows the number of invitations that is sent by the screening organisation East over the year in practice. The shown number of invitations in week t is the number of invitations that is actually sent in week $t-7$. An invitation in week t results on average in an intake appointment in week $t+7$. The MILP neglected this 7 weeks shift, so by shifting the BVO Oost invitations (which is done in Figure 5.5) the current situation at BVO Oost and the results of the developed model can be compared.

For this purpose we compare Figure 5.3 and Figure 5.5. The shape of both graphs is more or less the same. Both have lower invitation numbers in holiday weeks. The main difference is that the invitation strategy of BVO Oost is more variable over the year, whereas Figure 5.3 is more stable. The total number of invitations in the BVO Oost case is 369,661. The scenario with rates 73% and 4.7% has in total 412,710 invitations in one year. The MILP invites more clients, because the MILP evaluates the entire region East and the entire year in one go. This results in a more optimal adherence for the different postcode areas and CCs and a better distribution of invitations over the year. With the current strategy of BVO Oost clients can only be invited when their postcode area (council) is linked manually to a CC with capacity. When this link is not present but a CC in the region of the council does have capacity, the clients will not be invited. This situation cannot occur in the MILP, because the adherence is an output parameter, not an input parameter as within BVO.

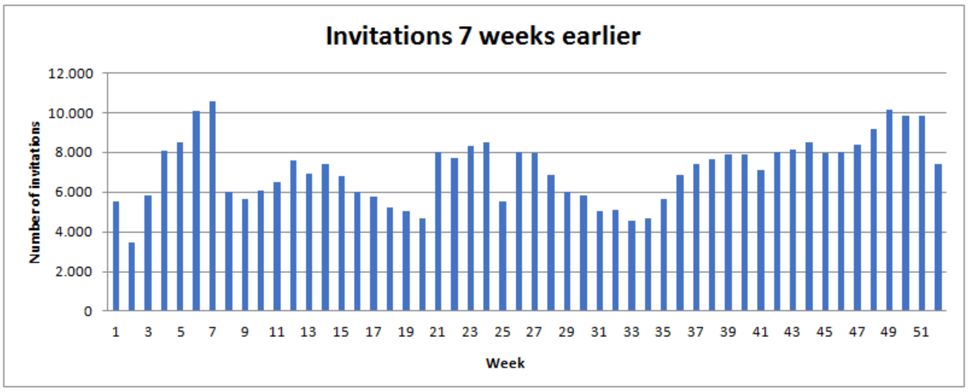
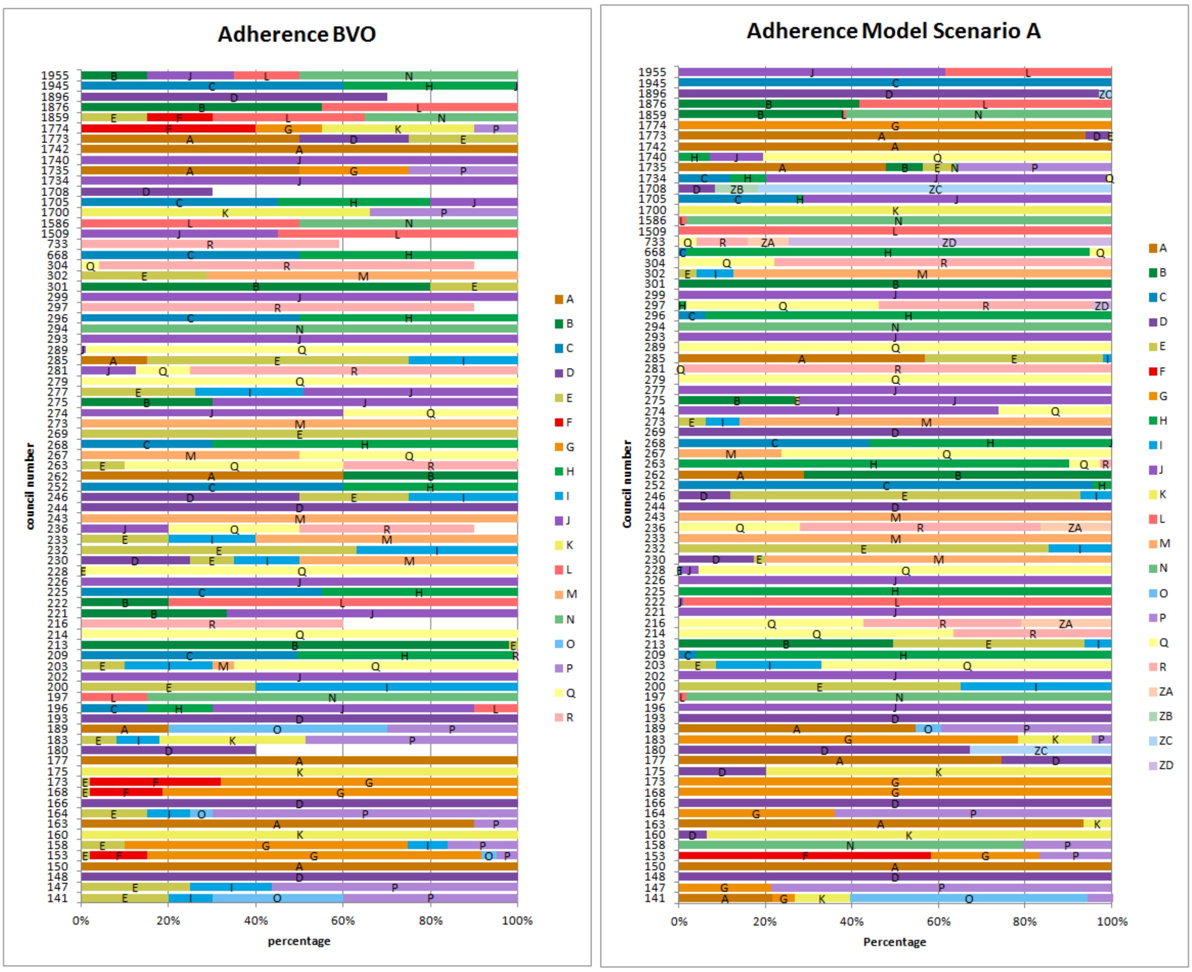


Figure 5.5: The historical invitations of BVO Oost with a shift of seven weeks.

5.4 Adherence

In order to compare the adherence that comes out of the mathematical model with the adherence input of the current situation used by BVO, the adherence numbers of the model need to be modified. BVO Oost uses namely adherence from postcode areas that correspond to entire councils, whereas the mathematical model has PC4 areas, these areas are much smaller. The adherence numbers of the PC4 areas are aggregated into the councils, each council has its own number. Figure 5.6a and 5.6b show the adherence of BVO Oost and the model respectively. The CCs are indicated with colours and corresponding letters. The vertical axis consists of all council numbers. The coloured bars behind a council number indicate how many percent of the clients of that council are invited on the capacity of the CC that corresponds with the colour/letter. For example, look at council number 1945. According to the BVO Oost adherence (Figure 5.6a) 60% of the clients in council 1945 are invited to CC 'C' and 40% to CC 'H'. Whereas the output of the MILP shows that 100% of the clients from council 1945 are invited to CC 'C'. The adherence of the various experiments in the MILP within a scenario does not differ significantly. Also the adherence of the various scenarios is comparable. Therefore only the adherence table of scenario A experiment 1 is shown.

Figure 5.7 visualizes the adherences of BVO Oost (Figure 5.7a) and the model (Figure 5.7b) in another way. The councils are now plotted on the map of region East. Each area corresponds with a council. The CCs are indicated on their correct location on the map by stars and the corresponding letter of the CC. The CCs have again each their own colour, similar as in Figure 5.6. The council areas have got the colour of the CC where the clients are invited to. When clients from a council are send to multiple CCs the area has got multiple colours. In order to keep the map clear, only CCs where at least 20% of the clients from that council are sent to are incorporated. The adherence of BVO Oost and the model will be compared with the help of Figure 5.6 and Figure 5.7.



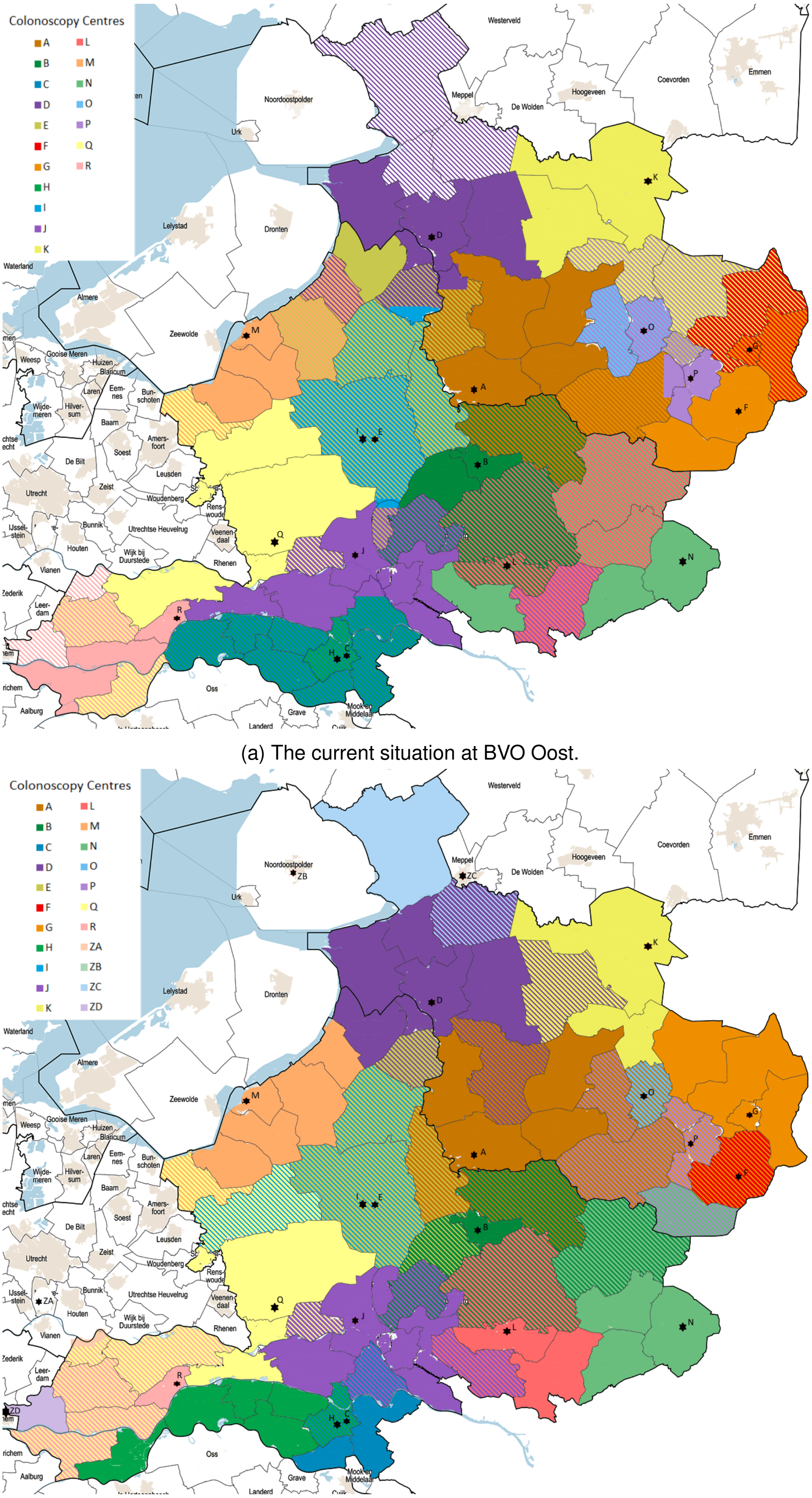
(a) The adherence that is used by BVO.

(b) The adherence that arises from the model.

Figure 5.6: The adherence numbers for each council number, the letters indicate the CCs.

Some of the councils in Figure 5.6a do not have a total of 100%, these councils are the councils that use intake slots of CCs outside region East. It can be seen that these councils have an adherence with a CC that starts with the letter ’Z’ in the model (Figure 5.6b). Recall that the CCs that start with ’Z’ are located outside region East. Generally, the CC that are used for a council are the same in both the BVO Oost case and the model case.

Both adherence tables differ, the main cause is that BVO Oost uses Euclidean distances and their own intuition, whereas the MILP uses travel times. Therefore, a CC that might be nearby in Euclidean distance might not appear in the model that uses travel times and vice versa. Two other differences can be seen between Figure 5.6a and 5.6b namely some councils use more CCs in the BVO Oost case than in the model and some councils use less CCs in the BVO Oost case than in the model. In the case that a council uses more CCs in the model (e.g. council 302), this difference can be explained by the fact that the adherence in Figure 5.6b is based on much smaller areas. These smaller areas each have a more specific distance to each CC and therefore a CC that is not optimal for a large council can be close by for a small part of the council (PC4). This specific CC will then be used by the PC4 area and later during the aggregation this CC appears as one of the CCs that is used in the council. On the other hand some councils use less CC according to the model (e.g. council 1700). This is caused by the fact that the model maximizes the number of clients that is linked to the nearest CC. When intake slots are available, a whole PC4 area will be invited to the nearest CC, which results in large adherence numbers and less different CCs per council. This effect is even more clear when we look at Figure 5.7. Many councils only have one colour in Figure 5.7b (MILP), while they have two colours in Figure 5.7a (BVO Oost).



(a) The current situation at BVO Oost.

(b) The adherence that arises from the model.

Figure 5.7: The adherence numbers plotted on a map. The base map comes from [Imergis, 2018].

5.5 Compare BVO Oost and deterministic model

In the previous sections we already gave some comparisons between the results of the deterministic MILP model and the current situation at BVO Oost. However, this was only based on visual differences in the adherence maps. To compare the solutions of the MILP model with the current invitation strategy of BVO Oost on a quantitative level, we implemented the adherence numbers of BVO Oost into the MILP model in AIMMS. In other words, the MILP model finds the optimal invitations strategy which satisfies the adherence numbers as used by BVO Oost. These adherence numbers are on the aggregation level of councils, we therefore adjusted the MILP to councils by aggregating all PC4 areas in a council to one council area.

Table 5.6 shows the results of three levels of optimisation. The first level is the current situation with BVO Oost adherence when the invitations are optimally divided over the year. The second level gives the results when the developed MILP is used on aggregation level councils and the entire invitation strategy (time and location-CC linking) is optimised. The third level consists of the developed MILP used on the PC4 areas aggregation, which are the results as described in the previous sections.

Table 5.6: The output parameters for different levels of optimisation in scenario A with $PR = 73\%$ and $RR = 4.7\%$.

	BVO Oost adherence	MILP	MILP
	councils	councils	PC4 areas
Objective value	-72,294,930	-141,503,213	-143,536,905
Rest group [%]	7.8	3.6	3.6
Average Travel Time [min]	18.7	16.7	16.3
Not in nearest CC [%]	39.4	18.2	17.3
Not within 22-26 months [%]	0.6	0	0

We see in Table 5.6 that the solutions improve over the different levels of optimisation (from left to right). The BVO Oost adherence gives the worst (highest because we minimize) objective function. When we use the developed MILP to find the optimal linking between PC4 areas and CCs and inviting the clients at the right moments in time, we get an objective value much smaller. The improvement in this step from BVO Oost adherence to MILP solution is:

$$\text{Improvement} = \frac{-141,503,213 - -72,294,930}{-72,294,930} \cdot 100\% = 96\%$$

This improvement is equivalent with saying that the invitation strategy found by the MILP is almost twice as good as the current invitation strategy at BVO Oost. Using PC4 areas as aggregation level instead of councils gives an extra improvement of 1.4% relative to the MILP councils level. This improvement seems small, however the real improvement of using PC4 areas instead of councils is the level of detail in the solution. By using PC4 areas, the invitation strategy is much more specific for the clients. The clients are more treated like individuals, because they are grouped in smaller areas. The travel times from PC4 areas to CCs are more realistic than travel times from council to CCs, because councils are larger. By using PC4 areas it is also more clear which clients should be invited when, because we simple have less clients in a PC4 area than in a council. This level of detail for PC4 areas also explains the lower values for average travel time and percentage of clients not in nearest CC than on councils aggregation level. Due to more precise travel times, clients can be better linked to CC in the PC4 aggregation level.

The above mentioned improvements in the objective value over the three levels of optimisation are also visible in the values for the output parameters, as shown in Table 5.6. The improvement in rest group percentage is due to the fact that with the BVO Oost adherence, not all available capacity is used. Clients cannot be invited to CC where capacity is left, because the councils of these clients are not linked to that CC. In the MILP councils, all available capacity is used, because adherence numbers are determined afterwards. The CCs where not all available capacity is used by the BVO Oost adherence are; 'D', 'E', 'F', 'G', 'H', 'N', 'ZC' and 'ZD'. The MILP finds an optimal linking between councils and CC, which implies a lower average travel time and percentage of clients not linked to the nearest CC than the BVO Oost adherence. In the BVO Oost adherence case almost 40% of the clients is not linked to the nearest CC, which aligns with the 40% rescheduled intake appointments which is observed. By using the adherence that comes out of the MILP model the percentage of clients not linked to the nearest CC is decreased to about 18%. We therefore expect the number of rescheduled intake appointments to decrease in this same amount. The decrease in subsequent round clients that is not planned within the predefined interval from BVO Oost adherence to MILP councils, is caused by the use of all available capacity in the MILP.

Chapter 6

Robust Optimisation

We now have a deterministic solution to the invitation strategy problem of colon cancer screening, however uncertainty plays a roll in the process. Neglecting this uncertainty in participation and referral rates can lead to good solutions, but problems occur when more clients need an intake appointment. Optimising the invitation strategy for the case that all clients need an intake appointment clearly gives no problems of waiting time for clients. However, it leads to infeasibility, because CCs do not have enough capacity for all clients. To find a trade-off between these solutions, we propose to use robust optimisation techniques. Based on the uncertainty in the participation and referral rate, the MILP of Chapter 3 will be modified. The resulting uncertain MILP is used to find a robust (”safe”) solution under uncertainty. In other words, we want to find the best possible solution that still satisfies the constraints when the uncertain parameters have a slightly different values. Four things will change when developing a robust optimisation model:

- The parameters for participation rate (PR) and referral rate (RR) become stochastic variables denoted with ρ_p. How this is done is explained in Section 6.1. All other parameters stay the same.
- The objective function stays the same, just like most of the constraints. Only constraints that contain PR and RR are modified to obtain a robust optimisation model. Section 6.2 describes in which way these uncertain constraints are modified.
- An uncertainty set $\mathcal{U}$ is added that describes which values the uncertain parameter ρ_p can take. This uncertainty set is needed to modify the uncertain constraints into chance constraints. Section 6.3 describes this uncertainty set.
- With the help of chance constraint approximation the robust counter part of the uncertain MILP is found. Solving the robust counterpart results in a robust solution to the uncertain MILP. This is described in Section 6.4.

6.1 Uncertain parameters

The uncertain parameters in this research are the participation rate and referral rate. Section 4.5 already explained which values they can take, namely:

$$\begin{aligned} PR_p &\in [70\%, 76\%] \quad \forall p, \quad \text{with mean } 73\% \text{ in the deterministic case} \\ RR_p &\in [4.3\%, 5.1\%] \quad \forall p, \quad \text{with mean } 4.7\% \text{ in the deterministic case} \end{aligned}$$

In the robust optimisation model, these two parameters become stochastic variables, which depend on the postcode area. In each postcode area a group of clients will be invited. The uncertainty is which part of these clients will participate and which part of the participants gets a positive (unfavourable) result. Therefore in each postcode area a random sample is drawn from these stochastic variables. These two stochastic variables are multiplied with each other in the model of Chapter 3. To handle this in the robust optimisation model, a new stochastic variable is introduced which is defined as follows:

$$\rho_p = PR_p \cdot RR_p$$

ρ_p is the random variable for the fraction of clients at postcode area p that need an intake appointment, because first PR_p of the clients participate and subsequently RR_p of these clients have a positive test result.

The distributions of the random variables PR_p and RR_p are not known, therefore the distribution of ρ_p is neither known. We only know that the values of ρ_p vary in the interval

$$\begin{aligned} &\rho_p \in [0.0301, 0.03876]. \\ \iff\ &\rho_p = 0.03443 + \xi_p \cdot 0.00433 = \bar{\rho} + \xi_p \cdot \hat{\rho} \quad \text{where,} \quad \xi_p \in [-1, 1] \end{aligned} \tag{6.1}$$

These numbers come from multiplying the lower bounds of PR_p and RR_p with each other, as well as the upper-bounds. This is done in fractional notation instead of percentages. $\bar{\rho}$ and $\hat{\rho}$ are the nominal value and basic shift value of ρ_p respectively. ξ_p indicates the amount of perturbation within postcode area p.

6.2 Uncertain constraints

When the uncertain parameters PR and RR are replaced by the new random variable ρ_p, the model as described in Chapter 3 is transformed in an uncertain MILP.

$$\text{minimize} \quad \alpha_s \cdot \sum_{p,t} e_p^t + \alpha_r \cdot \sum_p d_p - \alpha_n \cdot \sum_{p,c,t} x_{p,c}^t \cdot F_{p,c} + \alpha_d \cdot \sum_{p,c,t} x_{p,c}^t \cdot \hat{D}_{p,c} + \alpha_o \cdot \sum_c m_c \tag{6.2}$$

such that

$$\sum_{c,t} x_{p,c}^t + d_p = N_p \qquad \forall p \tag{6.3}$$

$$\sum_p \rho_p \cdot x_{p,c}^t \leq I_c^t \qquad \forall c, t \tag{6.4}$$

$$\frac{\sum_p \rho_p \cdot x_{p,c}^t}{I_c^t} \leq m_c \qquad \forall c, t \tag{6.5}$$

$$\sum_c \sum_{t'=t-8}^{t'=t+8} x_{p,c}^{t'} + e_p^t \geq E_p^t \qquad \forall p, 9 \leq t \leq 44 \tag{6.6}$$

$$\sum_c \sum_{t'=1}^{t'=t+8} x_{p,c}^{t'} + \sum_c \sum_{t'=44+t}^{t'=52} x_{p,c}^{t'} + e_p^t \geq E_p^t \qquad \forall p, t \leq 8 \tag{6.7}$$

$$\sum_c \sum_{t'=t-8}^{t'=52} x_{p,c}^{t'} + \sum_c \sum_{t'=1}^{t'=t-44} x_{p,c}^{t'} + e_p^t \geq E_p^t \qquad \forall p, t > 44 \tag{6.8}$$

$$0 \leq x_{p,c}^t \leq N_p \qquad \forall p, c, t \tag{6.9}$$

$$0 \leq d_p \leq N_p \qquad \forall p \tag{6.10}$$

$$0 \leq e_p^t \leq E_p^t \qquad \forall p, t \tag{6.11}$$

$$0 \leq m_c \leq 1 \qquad \forall c, t \tag{6.12}$$

$$x_{p,c}^t, \in \mathbb{Z} \quad d_p, \in \mathbb{Z} \quad e_p^t \in \mathbb{Z} \qquad \forall p, c, t \tag{6.13}$$

$$\rho_p \in [0.0301, 0.03876] \qquad \forall p \tag{6.14}$$

The brackets that indicate rounding to integer numbers disappeared, because this is no longer possible in an uncertain MILP. Besides that, rounding to integer numbers is not necessary, as I_c^t still is an integer number. The restriction that $\sum_p \rho_p \cdot x_{p,c}^t$ should be less than I_c^t is enough to make sure that it fits, because $\sum_p \rho_p \cdot x_{p,c}^t$ rounded to the nearest integer will still be smaller than I_c^t.

Two of the constraints of the above uncertain MILP model contain the uncertain parameter ρ_p. Namely, constraints (6.4) and (6.5), which make sure that the limited capacity is not violated and that the workload is spread evenly over the year. Both these constraints should be modified in order to transform the uncertain MILP into a model that has a robust solution. The two constraints are treated in different ways. Constraint (6.4) will be transformed into a chance constraint in order to find a robust solution that is allowed to violate the constraint under a small tolerance ϵ. Once the solution is robust for the limited capacity constraint it is no longer required that also the workload distributing constraint, (6.5), is made robust. It is enough to distribute the workload evenly over the year in the nominal case. This is because of the fact that the two constraints (6.4) and (6.5) have a large resemblance. When workload is distributed evenly in the nominal case, each week has the same starting point. It is not known in advance which week will be more busy than an other week, so you better treat each week similar. In this way each week has in mean the same occupancy rate, but due to fluctuations in the uncertain parameter the realizations might differ per week. The robust limited capacity constraint already makes sure that the probability that the occupancy rate becomes larger than 100% is small (tolerance ϵ). Section 6.2.1 describes what is needed to transform constraint (6.4) into a chance constraint.

The new form of constraint (6.5) is given in (6.15). $\bar{\rho}$ is the nominal value of the random variable ρ_p, and therefore the workload is distributed evenly over the year in the nominal case.

$$\frac{\sum_p \bar{\rho} \cdot x^t_{p,c}}{I^t_c} \leq m_c \quad \forall c, t \tag{6.15}$$

6.2.1 Limited capacity constraint

The limited capacity constraint is given in (6.4), to transform this constraint into a chance constraint we introduce a matrix notation for this constraint. In this way we can use the techniques from Ben-Tal et al. [2009] to formulate the chance constraint and find the robust counterpart.

Matrix notation

In order to write constraint (6.4) in matrix form we use the following notation.

P = Number of elements in postcode areas (p) set
C = Number of elements in Colonoscopy Centres (c) set
T = Number of elements in Week numbers (t) set
$\mathbf{1^P}$ = vector of length P with all ones
$\mathbf{0^P}$ = vector of length P with all zeros
$\mathbf{z^{t_i}_{c_j}}$ = $\begin{bmatrix} x^{t_i}_{p_1,c_j} & x^{t_i}_{p_2,c_j} & \cdots & x^{t_i}_{p_P,c_j} \end{bmatrix}$ = vector of length P with values $x^{t_i}_{p,c_j}$ for fixed t_i, c_j and $p = p_1, \ldots, p_P$
$\mathbf{z^{t_i}}$ = $\begin{bmatrix} \mathbf{z^{t_i}_{c_1}} & \mathbf{z^{t_i}_{c_2}} & \cdots & \mathbf{z^{t_i}_{c_C}} \end{bmatrix}$ = vector of length C with vectors $\mathbf{z^{t_i}_c}$ for t_i fixed and $c = c_1, \ldots, c_C$
$\mathbf{I^{t_i}}$ = $\begin{bmatrix} I^{t_i}_{c_1} & I^{t_i}_{c_2} & \cdots & I^{t_i}_{c_C} \end{bmatrix}$ = vector of length C with values $I^{t_i}_c$ for fixed t_i and $c = c_1, \ldots, c_C$
$\boldsymbol{\rho}$ = $\begin{bmatrix} \rho_{p_1} & \rho_{p_2} & \cdots & \rho_{p_P} \end{bmatrix}$ = vector of length P with values ρ_p

The decision vector $\mathbf{y}$ of length $P \cdot C \cdot T$ that contains the variables of constraint (6.4) is defined as follows:

$$\mathbf{y}' = \begin{bmatrix} \mathbf{z^{t_1}} & \mathbf{z^{t_2}} & \cdots & \mathbf{z^{t_T}} \end{bmatrix}$$

The right-hand-side vector $\mathbf{b}$ that belongs to constraint (6.4) has length $C \cdot T$ because the constraint should hold for all combinations of CCs and week numbers and is defined as follows:

$$\mathbf{b}' = \begin{bmatrix} \mathbf{I^{t_1}} & \mathbf{I^{t_2}} & \cdots & \mathbf{I^{t_T}} \end{bmatrix}$$

When we look at an individual constraint of (6.4) for CC c in week t, the coefficients are denoted in the following vector.

$$\mathbf{a_{c,t}} = [\underbrace{\mathbf{0^P} \quad \cdots \quad \mathbf{0^P}}_{(t-1)\cdot C+(c-1)} \quad \boldsymbol{\rho} \cdot \mathbf{1^P} \quad \underbrace{\mathbf{0^P} \quad \cdots \quad \mathbf{0^P}}_{(T-t)\cdot C+(C-c)}]$$

By putting all constraints in (6.4) together we get the coefficients matrix A with $P \cdot C \cdot T$ columns and $C \cdot T$ rows.

$$A = \begin{bmatrix} \mathbf{a_{c_1,t_1}} \\ \mathbf{a_{c_2,t_1}} \\ \vdots \\ \mathbf{a_{c_C,t_1}} \\ \mathbf{a_{c_1,t_2}} \\ \vdots \\ \mathbf{a_{c_C,t_T}} \end{bmatrix}$$

Concluding, constraint (6.4) is equivalent with (6.16) in matrix notation.

$$\begin{aligned} \sum_p \rho_p \cdot x^t_{p,c} \quad &\leq \quad I^t_c \qquad \forall c, t \\ &\Longleftrightarrow \\ A \cdot \mathbf{y} \quad &\leq \quad \mathbf{b} \end{aligned} \tag{6.16}$$

For one individual CC - week number combination (c, t) the constraint looks as follows:

$$\begin{aligned} \sum_p \rho_p \cdot x^t_{p,c} &\leq I^t_c \\ &\Longleftrightarrow \\ \mathbf{a_{c,t}} \cdot \mathbf{y} &\leq I^t_c \end{aligned} \tag{6.17}$$

With these matrix notations the limited capacity can be transformed into a chance constraint. This transformation is presented in Section 6.4. But first we define the uncertainty set in Section 6.3.

6.3 Uncertainty set

The uncertainty set that comes with a robust optimisation model contains all possible values that the uncertain parameters can take. As described in Ben-Tal et al. [2009, Sec. 1.2.1] the uncertainty set $\mathcal{U}$ of the entire system can be written as the Cartesian product of the uncertainty sets $\mathcal{U}_i$ that correspond to a single constraint i. This means that we treat each uncertain constraint individually, by formulating its robust counterpart on its own. Taking the Cartesian product of the uncertainty sets is equivalent with enumerating all individual robust counterparts. In other words we can add $\forall c, t$ behind the individual robust counterpart constraint. Whereas only constraint (6.4) contains the uncertain parameter, we have $C \cdot T$ uncertain constraints each with their own uncertainty set. The individual uncertainty set $\mathcal{U}_{c,t}$ that corresponds to constraint (6.17), where CC c and week t are considered, is defined as follows:

$$\begin{aligned} \mathcal{U}_{c,t} &= \left\{ \mathbf{a_{c,t}} = \mathbf{a^{(0)}_{c,t}} + \sum_{p=1}^{P} \xi_p \cdot \mathbf{a^{(p)}_{c,t}} \mid \boldsymbol{\xi} \in \mathcal{Z} \subset \mathbb{R}^P \right\} && (6.18) \\ \text{where,} \quad \mathbf{a^{(0)}_{c,t}} &= [\underbrace{\mathbf{0^P} \quad \cdots \quad \mathbf{0^P}}_{(t-1)\cdot C+(c-1)} \quad \bar{\rho} \cdot \mathbf{1^P} \quad \underbrace{\mathbf{0^P} \quad \cdots \quad \mathbf{0^P}}_{(T-t)\cdot C+(C-c)}] && (6.19) \\ \mathbf{a^{(p)}_{c,t}} &= [\underbrace{\mathbf{0^P} \quad \cdots \quad \mathbf{0^P}}_{(t-1)\cdot C+(c-1)} \quad \hat{\rho} \cdot \mathbf{e^P_p} \quad \underbrace{\mathbf{0^P} \quad \cdots \quad \mathbf{0^P}}_{(T-t)\cdot C+(C-c)}] && (6.20) \\ \mathcal{Z} &= \text{Perturbation set} \end{aligned}$$

The uncertainty set for the entire system is then denoted as:

$$\mathcal{U} = \bigotimes_{c \in C, t \in T} \mathcal{U}_{c,t} = \mathcal{U}_{c_1,t_1} \times \ldots \times \mathcal{U}_{c_C,t_1} \times \mathcal{U}_{c_1,t_2} \times \ldots \times \mathcal{U}_{c_C,t_2} \times \ldots \times \mathcal{U}_{c_1,t_T} \times \ldots \times \mathcal{U}_{c_C,t_T}$$

The values in vector $\mathbf{a_{c,t}}$ have nominal values denoted with $^{(0)}$ and deviate from these nominal values which is denoted with the basic shift value $^{(p)}$. The values of ξ_p indicate how much the value for ρ_p deviates from the nominal value. The nominal and basic shift values are $\bar{\rho}$ and $\hat{\rho}$ in (6.19) and (6.20) respectively, as also indicated in (6.1). The deviations ξ_p are captured in the perturbation set $\mathcal{Z}$ which is a closed and convex set [Ben-Tal et al., 2009]. Different configurations are possible for this uncertainty set. Depending in which way the chance constraint will be approximated one of these configurations will be chosen. Four of the most common representations for the perturbation set are listed below.

Interval uncertainty, where $\mathcal{Z}$ is a box:

$$\mathcal{Z} = \{\boldsymbol{\xi} \in \mathbb{R}^P \mid \|\boldsymbol{\xi}\|_\infty \leq 1\} \tag{6.21}$$

This is also called worst case, because for each postcode area p the shift can take the largest possible value.

Ellipsoidal uncertainty, where $\mathcal{Z}$ is a ball:

$$\mathcal{Z} = \{\boldsymbol{\xi} \in \mathbb{R}^P \mid \|\boldsymbol{\xi}\|_2 \leq \Omega\} \tag{6.22}$$

In this case the perturbations lie within a ball with radius Ω. For some postcode areas the perturbation will be larger and for other postcode areas the perturbation is small. In this way the number of intake appointments needed in different postcode areas can compensate each other.

Interval/ellipsoidal uncertainty, where $\mathcal{Z}$ is the intersection of a box and a ball:

$$\mathcal{Z} = \{\boldsymbol{\xi} \in \mathbb{R}^P \mid \|\boldsymbol{\xi}\|_\infty \leq 1, \|\boldsymbol{\xi}\|_2 \leq \Omega\} \tag{6.23}$$

This configuration for the perturbation set is a combination of the box and the ball.

Budgeted uncertainty, where $\mathcal{Z}$ is the intersection of a box and the 1-norm:

$$\mathcal{Z} = \{\boldsymbol{\xi} \in \mathbb{R}^P \mid \|\boldsymbol{\xi}\|_\infty \leq 1, \|\boldsymbol{\xi}\|_1 \leq \gamma\} \tag{6.24}$$

With budgeted uncertainty all perturbations cannot be larger than 1 and the total amount of perturbations is bounded by γ. This represents the fact that it is really unlikely that all uncertain parameters take their worst case value.

6.4 Chance constraint approximation

Based on the notation in the previous sections, the uncertain Mixed Integer Linear Program that is considered in this research is given by equations (6.2), (6.3), (6.6) – (6.13), (6.15) and (6.16) with $A \in \mathcal{U}$. We want to accomplish that under the realizations of the uncertain parameter ρ_p, the limited capacity constraint (6.5) (i.e. (6.16)) is still satisfied. However, requiring satisfying it in all cases is to hard, because it is not realistic that all uncertain parameters will take the worst value. Therefore, the limited capacity constraint (6.16) is transformed into a chance constraint. This chance constraint is shown in equation (6.25), which is equivalent with (6.26) when the matrix notation is considered. The idea is that the constraint is allowed to be violated, but the probability that the constraint is violated under the distribution of the parameter realizations should be smaller than the tolerance ϵ.

$$\mathbb{P}\left(\textstyle\sum_p \rho_p \cdot x_{p,c}^t > I_c^t\right) \leq \epsilon \quad \forall c, t \tag{6.25}$$

$$\Longleftrightarrow$$

$$\mathbb{P}_{\xi\sim\mathcal{P}}\left(\mathbf{a_{c,t}^{(0)}} \cdot \mathbf{y} + \textstyle\sum_{p=1}^P \xi_p \cdot \mathbf{a_{c,t}^{(p)}} \cdot \mathbf{y} > I_c^t\right) \leq \epsilon \quad \forall c, t \tag{6.26}$$

The ξ_p are independent random variables with $\mathbb{E}[\xi_p] = 0$ and $|\xi_p| \leq 1$. However, the complete distribution of the perturbation vector ξ is not known. Moreover, even if the distribution is known, it is hard to find solutions for the chance constraints. Therefore, the chance constraint will be replaced with its computationally tractable safe approximation, as explained in Ben-Tal et al. [2009]. By this replacement the robust counterpart of the problem is formulated. The robust counterpart is a deterministic program that gives a robust solution for the uncertain program. The robust counterpart can be found in different ways, by using different configurations of the perturbation set $\mathcal{Z}$. The radius Ω or the budget γ are the so called safety parameters and depend on the desired tolerance level ϵ. Ben-Tal et al. [2009] gives the standard form of the robust counterpart in all four possible configurations of the perturbation set, including the needed value for the safety parameter. The following sections give the robust counterpart of the uncertain MILP of this research in each of the four possible uncertainty sets.

In all these sections the uncertain constraint (6.27) is replaced by its robust counterpart corresponding to the chosen uncertainty set approximation.

$$\mathbf{a_{c,t}^{(0)}} \cdot \mathbf{y} + \sum_{p=1}^{P} \xi_p \cdot \mathbf{a_{c,t}^{(p)}} \cdot \mathbf{y} \leq I_c^t \qquad \forall c, t \qquad \text{where,} \quad \xi \in \mathcal{Z} \tag{6.27}$$

By replacing constraint (6.16) (i.e. (6.27)) in { (6.2), (6.3), (6.6) – (6.13), (6.15) and (6.16) } with its robust counterpart, we obtain the total robust counterpart of the entire model. This total robust counterpart can be used to find a robust (”safe”) solution to the problem of finding an invitation strategy in the colon cancer screening.

6.4.1 Interval uncertainty

By using Ben-Tal et al. [2009, Ex. 1.3.2], the robust counterpart of the uncertain linear constraint (6.27) with uncertainty set Box (6.21) is equivalent to:

$$\mathbf{a_{c,t}^{(0)}} \cdot \mathbf{y} + \textstyle\sum_{p=1}^P \mathbf{a_{c,t}^{(p)}} \cdot \mathbf{y} \leq I_c^t \qquad \forall c, t \tag{6.28}$$

$$\Longleftrightarrow$$

$$\textstyle\sum_p (\bar{\rho}_p + \hat{\rho}_p) \cdot x_{p,c}^t \leq I_c^t \qquad \forall c, t \tag{6.29}$$

This is similar with the deterministic constraint with the worst case value of $\bar{\rho}_p + \hat{\rho}_p$ for all ρ_p as is expected with a worst case box approximation. This robust constraint has a total immunization, which means that the chance constraint is violated with probability $\epsilon = 0$.
The robust counterpart constraint (6.29) together with constraints (6.3), (6.6) – (6.13), (6.15) and objective function (6.2) give the complete robust counterpart of the uncertain MILP when interval uncertainty is assumed.

6.4.2 Ellipsoidal uncertainty

When ellipsoidal uncertainty is considered, the safety parameter, that is the radius of the ball, should have value $\Omega = \sqrt{2\ln(1/\epsilon)}$, according to Ben-Tal et al. [2009, Prop. 2.3.1]. By using Ben-Tal et al. [2009, Cor. 2.3.2], the robust counterpart of the uncertain linear constraint (6.27) with uncertainty set Ball (6.22) is equivalent to the following conic quadratic constraint.

$$\mathbf{a_{c,t}^{(0)}} \cdot \mathbf{y} + \Omega \cdot \sqrt{\textstyle\sum_{p=1}^{P} \left(\mathbf{a_{c,t}^{(p)}} \cdot \mathbf{y}\right)^2} \leq I_c^t \qquad \forall c,t \tag{6.30}$$

$$\Longleftrightarrow$$

$$\textstyle\sum_p \bar{\rho}_p \cdot x_{p,c}^t + \Omega \cdot \sqrt{\textstyle\sum_{p=1}^{P} \left(\hat{\rho}_p \cdot x_{p,c}^t\right)^2} \leq I_c^t \qquad \forall c,t \tag{6.31}$$

Before this constraint can be solved, it needs to be slightly adapted, because solvers cannot handle square roots. We introduce the non-negative auxiliary variable $z_{c,t}$, which represents the square root part of the constraint.

$$\sum_p \bar{\rho}_p \cdot x_{p,c}^t + \Omega \cdot \left(z_c^t\right)^2 \quad \leq \quad I_c^t \qquad \forall c,t \tag{6.32}$$

$$\sum_{p=1}^{P} \left(\hat{\rho}_p \cdot x_{p,c}^t\right)^2 \quad \leq \quad \left(z_c^t\right)^2 \qquad \forall c,t \tag{6.33}$$

$$z_c^t \quad \geq \quad 0 \qquad \forall c,t \tag{6.34}$$

Constraints (6.32) – (6.34) together is equivalent with (6.31), because it represents constraint (6.31) according to Ben-Tal et al. [2009, Def. 1.3.1]. A solution of (6.32) – (6.34) can be translated to a solution of (6.31) by discarding the values of z_c^t. You might say that constraint (6.33) needs to be an equality constraint, but in that case it is no longer a second order conic constraint. When constraint (6.33) is tight, (6.32) – (6.34) is one-to-one equivalent with (6.31), $z_c^t = \sqrt{\sum_{p=1}^{P} \left(\hat{\rho}_p \cdot x_{p,c}^t\right)^2}$. However, when z_c^t becomes larger than the square root and (6.32) is still satisfied, also (6.31) will be satisfied. When it is better to have a lower value for z_c^t in order to have a better solution for $x_{p,c}^t$ this can still happen in (6.32) – (6.34). The actual value of z_c^t does not matter. Therefore, (6.32) – (6.34) represent constraint (6.31).

The robust counterpart constraints (6.32) – (6.34) together with constraints (6.3), (6.6) – (6.13), (6.15) and objective function (6.2) give the complete robust counterpart of the uncertain MILP, when ellipsoidal uncertainty is assumed.

6.4.3 Ball-Box uncertainty

When Ball-Box uncertainty is considered, the safety parameter should have value $\Omega = \sqrt{2\ln(1/\epsilon)}$. By using Ben-Tal et al. [2009, Prop. 2.3.3], the robust counterpart of the uncertain linear constraint (6.27) with uncertainty set Ball-Box (6.23) is equivalent to the following system of conic quadratic constraints.

$$\begin{aligned} h_{p,c}^t + g_{p,c}^t &= \mathbf{a_{c,t}^{(p)}} \cdot \mathbf{y} && \forall p,c,t \\ \sum_p |h_{p,c}^t| + \Omega \cdot \sqrt{\sum_p \left(g_{p,c}^t\right)^2} &\leq I_c^t - \mathbf{a_{c,t}^{(0)}} \cdot \mathbf{y} && \forall c,t \end{aligned} \tag{6.35}$$

$$\Longleftrightarrow$$

$$\begin{aligned} h_{p,c}^t + g_{p,c}^t &= \hat{\rho} \cdot x_{p,c}^t && \forall p,c,t \\ \sum_p |h_{p,c}^t| + \Omega \cdot \sqrt{\sum_p \left(g_{p,c}^t\right)^2} &\leq I_c^t - \sum_p \bar{\rho} \cdot x_{p,c}^t && \forall c,t \end{aligned} \tag{6.36}$$

$h_{p,c}^t$ and $g_{p,c}^t$ are additional variables, but do not have any meaning in the solution. Before this constraint can be solved, it needs to be rewritten slightly, because solvers cannot handle square roots and absolute values. We introduce the non-negative auxiliary variable $z_{c,t}$, which represents the square root part of the constraint similar to what is done in Section 6.4.2. Also the non-negative auxiliary variable $u_{p,c}^t$ is introduced, which represents the absolute value of $h_{p,c}^t$.

$$\begin{aligned} h_{p,c}^t + g_{p,c}^t &= \hat{\rho} \cdot x_{p,c}^t && \forall p,c,t \\ \sum_p u_{p,c}^t + \Omega \cdot z_c^t &\leq I_c^t - \sum_p \bar{\rho} \cdot x_{p,c}^t && \forall c,t \\ \sum_p \left(g_{p,c}^t\right)^2 &\leq \left(z_c^t\right)^2 && \forall c,t \\ -u_{p,c}^t &\leq h_{p,c}^t \leq u_{p,c}^t && \forall p,c,t \\ u_{p,c}^t \geq 0, &\quad z_c^t \geq 0 && \forall p,c,t \end{aligned} \tag{6.37}$$

The system of constrains (6.37) represents the system of constraints (6.36), according to Ben-Tal et al. [2009, Def. 1.3.1]. Why this is correct is already explained in Section 6.4.2, only in this case also the absolute values need to be represented. This is done by introducing $u^t_{p,c}$ which represents $|h^t_{p,c}|$. By representing the absolute value, the same reasoning as with representing the square root can be followed. When the constraint $-u^t_{p,c} \leq h^t_{p,c} \leq u^t_{p,c}$ is tight at one of the sides, we have $u^t_{p,c} = |h^t_{p,c}|$. Otherwise, the value of $u^t_{p,c}$ is allowed to be larger as long as the second constraint in (6.37) holds. The actual value of $u^t_{p,c}$ and z^t_c do not matter, therefore constraints (6.37) represent (6.36).

The robust counterpart constraints (6.37) together with constraints (6.3), (6.6) – (6.13), (6.15) and objective function (6.2) give the complete robust counterpart of the uncertain MILP, when Ball-Box uncertainty is assumed.

6.4.4 Budgeted uncertainty

When budgeted uncertainty is considered, the safety parameter should have value $\gamma = \sqrt{2\ln(1/\epsilon)} \cdot \sqrt{P}$, where P has value 790 which is the total number of PC4 areas in region East. By using Ben-Tal et al. [2009, Prop. 2.3.4], the robust counterpart of the uncertain linear constraint (6.27) with uncertainty set Budgeted (6.24) is equivalent to the following system of linear constraints.

$$\begin{aligned} h^t_{p,c} + g^t_{p,c} &= \mathbf{a^{(p)}_{c,t}} \cdot \mathbf{y} && \forall p, c, t \\ \sum_p |h^t_{p,c}| + \gamma \cdot \max_p\{|g^t_{p,c}|\} &\leq I^t_c - \mathbf{a^{(0)}_{c,t}} \cdot \mathbf{y} && \forall c, t \end{aligned} \tag{6.38}$$

$$\Longleftrightarrow$$

$$\begin{aligned} h^t_{p,c} + g^t_{p,c} &= \hat{\rho} \cdot x^t_{p,c} && \forall p, c, t \\ \sum_p |h^t_{p,c}| + \gamma \cdot \max_p\{|g^t_{p,c}|\} &\leq I^t_c - \sum_p \bar{\rho} \cdot x^t_{p,c} && \forall c, t \end{aligned} \tag{6.39}$$

$h^t_{p,c}$ and $g^t_{p,c}$ are additional variables, but do not have any meaning in the solution. Before this constraint can be solved, it needs to be rewritten into a linear system of constraints, because solvers cannot handle absolute values and maximum functions. The non-negative auxiliary variable $u^t_{p,c}$ is introduced, which represents the absolute value of $h^t_{p,c}$. This is already described in the Section 6.4.3. In order to get rid of the maximum function we introduce the auxiliary variable w^t_c.

$$\begin{aligned} \sum_p u^t_{p,c} + \gamma \cdot w^t_c + \sum_p \bar{\rho} \cdot x^t_{p,c} &\leq I^t_c && \forall c, t \\ h^t_{p,c} + g^t_{p,c} &= \hat{\rho} \cdot x^t_{p,c} && \forall p, c, t \\ -u^t_{p,c} \leq h^t_{p,c} &\leq u^t_{p,c} && \forall p, c, t \\ w^t_c &\geq g^t_{p,c} && \forall p, c, t \\ w^t_c &\geq -g^t_{p,c} && \forall p, c, t \\ u^t_{p,c} \geq 0, \quad w^t_c &\geq 0 && \forall p, c, t \end{aligned} \tag{6.40}$$

The system of constrains (6.40) represents the system of constraints (6.39), according to Ben-Tal et al. [2009, Def. 1.3.1]. How the absolute value is represented is already explained in the previous section. In this case also the maximum function needs to be represented, which is done by introducing variable w^t_c. Again the same reasoning as with the square root and absolute value in the previous sections holds. The actual values of $u^t_{p,c}$ and w^t_c do not matter, therefore constraints (6.40) represent (6.39).

The robust counterpart constraints (6.40) together with constraints (6.3), (6.6) – (6.13), (6.15) and objective function (6.2) give the complete robust counterpart of the uncertain MILP, when budgeted uncertainty is assumed.

Chapter 7

Results robust model

The previous chapter described four different ways of chance constraint approximation. Each of these possibilities gives a deterministic robust counterpart of the uncertain model. Solving such a robust counterpart results in a ”safe” solution for the matching problem of inviting clients from postcode areas to nearby CCs in specific weeks. In this thesis only one of the possible robust counterparts will be solved and different tolerance levels ϵ will be used. Section 7.1 describes the choice of the chance constraint approximation type that is made. The different experiments that correspond to different tolerance levels are given in Section 7.2. Unfortunately, we came across some computational problems while solving the robust counterparts. These problems and our solutions to overcome these problems are described in Section 7.3. Section 7.4 presents the results of the robust optimisation model, which will be compared to the results of the deterministic MILP.

7.1 Choice of approximation type

The four possible chance constraint approximations are box (interval uncertainty), ball (ellipsoidal uncertainty), ball-box and budgeted uncertainty. Each of these possibilities comes with its own perturbation set and robust counterpart as described in Section 6.4. All approximation types come with a perturbation set which do not arises naturally form practice, but we should keep in mind that they mathematically arise when you want to immunize against uncertainty.

The number of variables in each of the approximation types differs, as shown in Table 7.1. This Table also shows the resulting model type of the different approximation types. The first line of the Table corresponds to the deterministic MILP model as described in Chapter 3. Both box and budgeted approximation types have the advantages that the resulting robust counterpart is again a Mixed Integer Linear Program (MILP), which is the same type as the original deterministic model. An MILP is easier to solve than an Quadratic Constraint Program (QCP), which is the resulting robust counterpart when the ball or ball-box approximation is used. The number of variables of the box robust counterpart is exactly the same as the number of variables in the original MILP, this is a big advantage, because more variables implies a longer computation time for finding a solution.

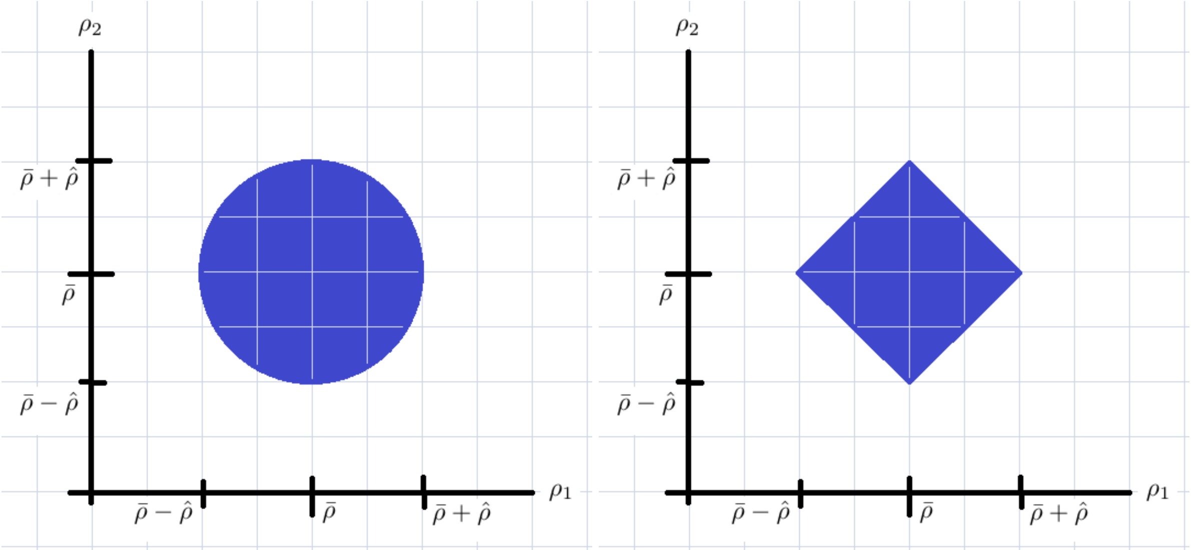
Table 7.1: The number of variables needed for each approximation type.

Approximation type	Model type	Number of variables	Number of variables in specific region East case
Deterministic	MILP	$P \cdot C \cdot T + P + P \cdot T + C$	945,652
Box	MILP	$P \cdot C \cdot T + P + P \cdot T + C$	945,652
Ball	QCP	$P \cdot C \cdot T + P + P \cdot T + C + C \cdot T$	946,796
Ball-Box	QCP	$4 \cdot P \cdot C \cdot T + P + P \cdot T + C + C \cdot T$	3,658,076
Budgeted	MILP	$4 \cdot P \cdot C \cdot T + P + P \cdot T + C + C \cdot T$	3,658,076

We can also look at the size of the perturbation set, which depends on the dimension of perturbation vector ξ. The dimension of ξ is equal to $P = 790$ in this research. As explained by Ben-Tal et al. [2009, pp. 33], the size of the perturbation set of the ball is much smaller than the perturbation set of the box, when the dimension becomes larger. A smaller perturbation set implies a less conservative robust counterpart, despite the same tolerance level is used. When the dimension of the perturbation vector is small (≤ 237), according to Ben-Tal et al. [2009], the box uncertainty set is smaller than the uncertainty set of ball. In order to reduce the uncertainty set of ball, the ball-box approximation can be used instead of the ball. By intersecting the perturbation set of the ball and the box the perturbation set of ball-box will not be greater than the perturbation set of the box. As the dimension of the perturbation vector ξ is larger than 237 in this research, namely 790, the approximation type ball-box is unnecessary. Therefore, we do not use approximation type ball-box.

When the approximation type box is used, you have a 100% immunization against uncertainty. This means that for all invited clients in CC c and week t that need an intake appointment, an intake slot is definitely available in that CC and week. The box robust counterpart is similar with using the worst case values of the uncertain parameter in the original MILP. This is already done in scenario B as described in Chapter 5. However, a total immunization is not desired, because it is very unlikely that all uncertain parameters will take their worst case value at the same time. When the highest possible participation and referral rate are used, very few clients will be invited on the available intake slots. In the first place this implies that not all clients can be invited in the year. In the second place it is very likely that the CCs will have low occupancy rates, because in practice less clients need an intake appointment than planned by the used rates. Both these consequences are not desired and therefore we do not use the approximation type box.

As approximation types box and ball-box will not be used as explained above, we need to choose between ball and budgeted. The theoretical differences are already explained in Table 7.1. The budgeted uncertainty set is larger and therefore results in a more conservative solution than with using ball approximation. However, the great advantage is that the robust counterpart of budgeted uncertainty is an MILP. We want to determine which approximation type is most useful from a practical view. The ball uncertainty set that contains all possible values for ρ_p can be seen as a ball around the nominal value $\bar{\rho}$. The budgeted uncertainty set is similar to a rhombus (diamond shape). The budgeted uncertainty set rules out large deviations in the cumulative value of ρ, so over all postcode areas. These perturbation sets are visualized in Figure 7.1 for the case of two postcode areas. The blue areas contain all values that the uncertain parameters ρ_1 and ρ_2 can take.



(a) The uncertainty set of ball approximation.

(b) The uncertainty set of budgeted approximation.

Figure 7.1: A visualization of the uncertainty sets with two postcode areas.

In practice most frequently budgeted uncertainty is used, for example as in Bertsimas and Thiele [2006] and Bertsimas and Thiele [2004]. They use budgeted uncertainty in robust optimisation of a supply chain with uncertain demands. The budgeted approximation type rules out the large deviations in the cumulative demand. Each time period the stock can be increased by setting new orders and each time period demand should be satisfied from the stock. They have a constraint that stock level should be at least as large as the uncertain demand. This is comparable to the cases that are considered in our research, where the uncertain responses of positive results should fit within the limited capacity. The total capacity in a week in a CC should be smaller than the uncertain amount of positive results in that week and CC. The budgeted approximation type is therefore suitable to rule out the large deviations in the cumulative (over the postcode areas) amount of positive test results. Also in healthcare applications the budgeted approximation type in robust optimisation is used, for example in Addis et al. [2016] and Cappanera et al. [2017]. These examples of using budgeted uncertainty in healthcare convince us that budgeted uncertainty is best used in the practice of our screening process also. Especially, the similarities in ”cancellations” of our screening process with scheduling home care in Cappanera et al. [2017] as explained in Section 1.3 give us strong reasons to choose budgeted uncertainty as approximation type.

Combining the pros and cons from theoretical view between the four different possible approximation types and looking from a practical view, we decided to use the budgeted uncertainty approximation type in this research. Budgeted uncertainty seems best suitable because it represents the practical uncertainty better, the ρ_p values are added over all postcode areas and the total number of intake appointments will therewith be limited. The budgeted uncertainty can be used well to control the level of conservatism of the robust solution. Also, from theoretical considerations, an MILP is easier to solve.

7.2 Scenarios

In generating robust optimisation results of the matching problem of clients from postcode areas to week numbers and CCs we examine different scenarios. These scenarios consist of different levels of tolerance and different dimensions of the perturbation vector all under the budgeted approximation type. The scenarios that are evaluated are indicated with a ✓in Table 7.2.

To find the best tolerance and the effect of the tolerance level on the solution, multiple values of ϵ will be used. When ϵ becomes larger, the safety parameter $\gamma = \sqrt{2\ln(1/\epsilon)} \cdot \sqrt{P}$ becomes smaller, which means that less buffer capacity for the invitations is present. Therefore, we expect that with a larger tolerance level, more clients can be invited. By the different scenarios we find out what tolerance level is acceptable. A tolerance level ϵ means that the probability that the limited capacity constraint is violated is at most $\epsilon\%$. In other words, the invited clients in a CC and in a week will be able to have their possible intake appointment in that CC and week in at least $(1-\epsilon)\%$ of the cases.

The dimension P of the uncertainty vector can also be changed. In theory all $P = 790$ postcode areas can have an uncertain value of ρ_p, however not all postcode areas will be used in practice to invite clients in one week in one CC. For one single limited capacity constraint, corresponding to CC c and week t, only clients from a few postcode areas will be invited. Only these postcode areas will have an uncertain value for ρ_p and all other postcode areas do not play a roll. We therefore can decrease the dimension of the uncertainty vector to L, where L is much smaller than P. According to the MILP results we use on average 4 postcode areas in one week for one CC. We therefore propose to consider values 4,5,6,7 and 8 for L. The safety parameter $\gamma = \sqrt{2\ln(1/\epsilon)} \cdot \sqrt{L}$ becomes smaller when L decreases. This results in less buffer capacity in a constraint and we expect that more clients can be invited when we use smaller L. In Table 7.3 the values for γ are shown. For example in the case of $\epsilon = 10\%$ and $L = 8$, we have $\gamma = 6.070$. This value of γ means that the sum of all perturbations to $\bar{\rho}$ can be at most 6.070. You can see this as that not more than 6 of the postcode areas can have the maximum value for ρ_p, or for example at most 12 postcode areas can have a ρ_p value of $\bar{\rho} + 0.5\hat{\rho}$. The safety parameter γ tells you against how many uncertainty the solution of the robust counterpart will protect. As you can see, in the case of $L = 790$ the safety parameter γ takes values that are really large. A value of $\gamma = 50$ means that at most 50 postcode areas can have the worst case value for ρ. If only 4 postcode areas are used to invite clients from, you will end up with a robust counterpart which will give a solution for a worst case scenario. All postcode areas are then assumed to have the worst case value for ρ. This effect made us decide to look also at values L that are much smaller than 790.

Table 7.2: The different scenarios that are evaluated in the robust optimisation model.

L ϵ	4	5	6	7	8	790
5%	×	×	×	✓	✓	✓
10%	×	✓	✓	✓	✓	✓
20%	✓	✓	✓	✓	✓	✓
75%	–	–	–	–	–	✓

Table 7.3: The safety parameter value γ for the different scenarios.

L ϵ	4	5	6	7	8	790
5%	–	–	–	6.476	6.923	68.799
10%	–	4.799	5.257	5.678	6.070	60.317
20%	3.588	4.012	4.395	4.747	5.075	50.427
75%	–	–	–	–	–	21.320

For the safety parameter γ should hold $0 \leq \gamma \leq L$. Therefore some of the combinations of ϵ and L are not possible as indicated with $\times$ in Table 7.2. Scenarios indicated with $-$ are not considered because we think they will not give valuable information. A tolerance level of $\epsilon = 75\%$ is not desirable in practice.

All the scenarios we execute only in one dataset / experiment, unlike we did with finding solutions for the MILP. We use only experiment 5 of Table 5.2 of Chapter 5, because this is the most average one. We only use one experiment because we want to see the mutual effect of ϵ and L. Due to time constraints it was not possible to execute the robust scenarios also on other datasets.

7.3 Computational problems

We programmed the robust counterpart for the budgeted uncertainty as given in Section 6.4.4 in AIMMS. However solving this robust counterpart by Gurobi leads to computational problems, so we are not able to find a robust solution to the matching problem of clients to CCs in this direct way. For these observations we used the 4.54.1 64 bits version of AIMMS and the 7.5.1 64 bits version of Gurobi on a HP Elitebook 8570w laptop with Intel Core i7 processor (8GB RAM) and 64 bits Windows 8.1.

The main problem that we have is that due to all the auxiliary variables that are introduced by creating the robust counterpart, we have to many variables. This amount of variables (3658076, as shown in Table 7.1) is to large to be handled by AIMMS which results in memory issues. Gurobi is not able to solve an MILP with this number of variables. Using a desktop computer with 16GB RAM also does not work and we therefore have to think of another way of finding solutions to the robust optimisation problem.

One possible option to find solutions to the robust optimisation problem is to reduce the number of variables. This can be done by only allowing a postcode area to be linked to a limited number of CCs. For instance it might be possible to say that only CCs that are within a travel time of 1 hour (approximately 40 km in euclidean distance) are allowed to be used. However, this reduction will cause some problems because if not enough capacity is available in the neighbourhood of a postcode area, these clients cannot be invited. Therefore, we cannot make this reduction in variables if we want to get realistic solutions.

We want to find a solution to the robust counterpart, where all variables are present, so we do not want to approximate the solution space by reducing it. The large number of variables cannot be handled by an MILP solver, but it is possible to solve Linear Programs (LP) with this large number of variables. We therefore propose to solve the robust counterpart as formulated in Section 6.4.4 with the LP solver of Gurobi. For this we discard the integrality constraints of the variables. The idea behind this approach is that the difference between inviting 24 clients from a postcode area to a week and CC instead of 23.5 or 24.4 clients will not be very large. These 24 clients are the average number of clients that are invited from a postcode area p, to a week t and CC c, so the average over all $x^t_{p,c}$. The value 24 is relatively large and rounding errors are expected to have only a small impact on the solution in this case. Therefore rounding the LP relaxation solution of the robust counterpart to an integer solution in a clever way might give good results. Together with the fact that we do not want to reduce the solution space this reasoning made us decide to solve the robust counterpart of inviting the clients from postcode areas to week numbers and CC under budgeted uncertainty as described in Section 6.4.4 with an LP relaxation.

After we found the optimal LP solution to the robust counterpart, we use Algorithm 1 to obtain a feasible integer solution, indicated with $\tilde{x}^t_{p,c}$. We first round down all decision variables $x^t_{p,c}$ to the nearest integer. This new integer solution is feasible, however this solution might invite less clients then possible and is therefore to pessimistic. We want to have an integer solution where the sum over all invited clients in a specific week at a specific CC equals the sum over all invited clients in the LP solution. We therefore iteratively increase the integer values $\tilde{x}^t_{p,c}$ by one client. By prioritizing the postcode areas that have subsequent round clients that need an intake around the current week, we try to reach an integer solution where as many subsequent round clients are invited in the predefined interval as possible. We first increase the number of invited clients from postcode areas where the fractional part of the LP solution is the highest. We can only increase the number of invited clients when not all clients from a postcode area all already invited. We continue increasing the number of invitations for a week and CC until the total number of integer invited clients equals the sum of invited clients in the LP solution. Once we have the integer values for the decision variable $\tilde{x}^t_{p,c}$ we can easily find the values for the other variables by following the constraints.

```
$x^t_{p,c}$, $d_p$, $e^t_p$ and $m_c$ are given from LP-relaxation ;
Set $\tilde{x}^t_{p,c} := \lfloor x^t_{p,c} \rfloor \ \forall p, c, t$ ;
for $(c, t)$ do
    ToAdapt := round $\left(\sum_p x^t_{p,c} - \sum_p \tilde{x}^t_{p,c}\right)$;
    MaxFrac := $\max_{p | \sum_{c,t} \tilde{x}^t_{p,c} < N_p} \{\sum_p x^t_{p,c} - \sum_p \tilde{x}^t_{p,c}\}$;
    while ToAdapt $> 0 \wedge$ MaxFrac $> 0$ do
        for $p \mid \sum_p x^t_{p,c} - \sum_p \tilde{x}^t_{p,c} =$ MaxFrac $\wedge \sum_{c,t} \tilde{x}^t_{p,c} < N_p$ do
            prioritize $p$ where subsequent round clients need an invitation around $t$;
            if ToAdapt $> 0$ then
                $\tilde{x}^t_{p,c} := \tilde{x}^t_{p,c} + 1$;
                ToAdapt := ToAdapt $- 1$;
            end
            if ToAdapt $= 0$ then
                stop while loop;
            end
        end
        MaxFrac := $\max_{p | \sum_{c,t} \tilde{x}^t_{p,c} < N_p} \{\sum_p x^t_{p,c} - \sum_p \tilde{x}^t_{p,c}\}$;
    end
end
$\tilde{d}_p := N_p - \sum_{c,t} \tilde{x}^t_{p,c} \quad \forall p$;
$\tilde{e}^t_p :=$ max$\{0, E^t_p - \sum_c \sum_{t'=t-8}^{t'=t+8} \tilde{x}^{t'}_{p,c}\} \quad \forall p, 9 \leq t \leq 44$ ;
$\tilde{e}^t_p :=$ max$\{0, E^t_p - \sum_c \sum_{t'=1}^{t'=t+8} \tilde{x}^{t'}_{p,c} + \sum_c \sum_{t'=44+t}^{t'=52} \tilde{x}^{t'}_{p,c}\} \quad \forall p, t < 9$ ;
$\tilde{e}^t_p :=$ max$\{0, E^t_p - \sum_c \sum_{t'=t-8}^{t'=52} \tilde{x}^{t'}_{p,c} + \sum_c \sum_{t'=1}^{t'=t-44} \tilde{x}^{t'}_{p,c}\} \quad \forall p, t > 44$ ;
$\tilde{m}_c :=$ max$_t\{\sum_p \bar{\rho} \cdot \tilde{x}^t_{p,c} / I^t_c\}$;
```

Algorithm 1: The pseudo code to obtain a feasible integer solution from the LP-relaxation.

In order to compare the obtained integer solution with the optimal LP solution we calculate the optimality gab between the objective values with the following formula:

$$\text{optimality gab} = \frac{\text{objective integer} - \text{objective LP}}{\text{objective LP}} \cdot 100\% \tag{7.1}$$

A negative optimality gab value indicates that the integer objective value is worse (higher because we minimize) than the LP objective value, this will always be the case, because requiring an integer solution reduces the solution space. The value of the optimality gab indicates how good the obtained integer solution is in comparison with the LP relaxation. The objective value of the LP relaxation is the best objective value that any integer solution can ever take, i.e. the optimal integer solution can theoretical have the same objective value but will never have a lower objective value. An optimality gab value closer to zero indicates a better integer solution. For example, when the optimality gab is -2% it means that the obtained integer solution is at most 2% away from the optimal integer solution.

7.4 Results adapted robust counterpart

In this section we give the results of solving the robust counterpart according to the strategy as described in the previous section. Section 7.4.1 contains the results of all scenarios and we explain the effects that ϵ and L have on the robust solution. In Section 7.4.2 we compare the results of the robust optimisation model to the results that we have from the deterministic MILP model. Finally, in Section 7.4.3, we give some estimations on the actual probability that the limited capacity constraint will be violated when the robust solution will be used to invite clients.

7.4.1 Scenario results

The results of solving the robust counterpart using an LP-relaxation and then obtaining an integer solution are given in this section for all different scenarios as given in Section 7.2. The results of the scenarios with the theoretical dimension of the perturbation vector of $L = 790$ are shown in Table 7.4. Tables 7.5, 7.6 and 7.7 contain the values of the output parameters for the scenarios with lower dimensions of the perturbation vector and tolerance levels $\epsilon = 5\%$, $\epsilon = 10\%$ and $\epsilon = 20\%$ respectively. All tables contain the objective value of both the LP-relaxation and the obtained integer solution, which then can be compared by the optimality gab which is calculated according to equation (7.1). The output parameters are defined in Section 3.6, where the occupancy in a single CC is now defined as the fraction of clients that is invited relative to the number of clients that can be invited when the mean value $\bar{\rho}$ is used.

$$\text{occupancy CC } c = \frac{\sum_{p,t} \tilde{x}^t_{p,c}}{\sum_t I^t_c/\bar{\rho}} \cdot 100\% \tag{7.2}$$

We start with analysing the robust results in the case of the theoretical dimension of the perturbation vector of $L = 790$ for different tolerance levels, see Table 7.4. We see that for all four levels of tolerance the rest group is large, about 13% of the clients cannot be invited in this robust solution. Even when we allow that the probability that the limited capacity constraint is violated in 75% of the cases, we get a bad solution. This is caused by the fact that we built in to much buffer capacity as can be seen by the low occupancy rates. The robust solution in the case of $L = 790$ are comparable with the worst case scenario of the MILP (scenario B) and do not give solutions that are valuable in practice. Concluding, if we immunize against uncertainty in this way, we built in so much buffer capacity that we cannot satisfy the quality criteria on inviting all clients of the target group. In other words, the robust solution based on budgeted uncertainty with $L = 790$ is to conservative and in practice not desirable. This is caused, as explained earlier, by the fact that we do not use all 790 postcode areas for one single CC and one single week.

Table 7.4: The results of robust optimisation with different tolerance levels ϵ and dimensions perturbation vector $L = 790$.

ϵ	Objective LP	Objective Integer	Gab %	Rest group %	Average Travel Time [min]	% not in nearest CC	% not within 22-26 months	Mean Occupancy CCs %
5%	-61,437,974	-58,299,442	-5.1	13.7	19.3	19.5	3.0	89.3
10%	-62,832,376	-59,011,175	-6.1	13.5	20.0	19.6	2.9	89.4
20%	-64,701,854	-60,602,452	-6.3	13.4	20.2	19.8	2.8	89.5
75%	-75,318,141	-70,356,326	-6.6	12.5	22.0	20.6	1.9	90.5

We expected this result and we therefore also look at the budgeted uncertainty robust optimisation with lower dimensions of the perturbation vector. The results of these experiments are shown in Table 7.5, 7.6 and 7.7 for three different levels of tolerance, $\epsilon = 5\%$, $\epsilon = 10\%$ and $\epsilon = 20\%$ respectively.

Table 7.5: The results of robust optimisation with tolerance level $\epsilon = 5\%$ and different dimensions of the perturbation vector L.

L	Objective LP	Objective Integer	Gab %	Rest group %	Average Travel Time [min]	% not in nearest CC	% not within 22-26 months	Mean Occupancy CCs %
7	-100,843,294	-98,360,851	-2.5	9.9	19.6	19.9	0.3	92.7
8	-99,455,384	-96,982,586	-2.5	10.1	19.8	20.0	0.3	92.5

Table 7.6: The results of robust optimisation with tolerance level $\epsilon = 10\%$ and different dimensions of the perturbation vector L.

L	Objective LP	Objective Integer	Gab %	Rest group %	Average Travel Time [min]	% not in nearest CC	% not within 22-26 months	Mean Occupancy CCs %
5	-107,145,183	-105,108,005	-1.9	9.0	19.1	19.3	0.2	93.6
6	-105,226,657	-103,078,284	-2.0	9.3	19.2	19.5	0.3	93.3
7	-103,607,427	-101,264,318	-2.3	9.5	19.3	19.6	0.3	93.1
8	-102,205,154	-99,800,844	-2.4	9.7	19.5	19.7	0.3	92.9

As expected the objective value decreases (gets better) when the tolerance level increases. When the dimension L increases the objective value increases, because more uncertainty is taken into account which results in more buffer capacity. More buffer capacity implies that we can invite less clients and we therefore get a solution which is worse. The improvements in objective value (due to increase of ϵ or decrease in L) is caused mostly by the improvement in the rest group. The rest group in these scenarios with lower L is significantly smaller than the rest group in the scenarios with $L = 790$. Although a rest group of about 9% is still quite high, it is much better than the rest group of 13% in Table 7.4. When the

Table 7.7: The results of robust optimisation with tolerance level $\epsilon = 20\%$ and different dimensions of the perturbation vector L.

L	Objective LP	Objective Integer	Gab %	Rest group %	Average Travel Time [min]	% not in nearest CC	% not within 22-26 months	Mean Occupancy CCs %
4	-113,139,413	-111,292,889	-1.6	8.2	18.7	18.9	0.2	94.5
5	-110,874,153	-108,989,339	-1.7	8.5	18.8	19.0	0.2	94.1
6	-108,987,452	-107,117,162	-1.7	8.8	18.9	19.1	0.2	93.9
7	-107,372,951	-105,371,427	-1.9	9.0	19.1	19.3	0.2	93.6
8	-105,968,866	-103,905,469	-2.0	9.2	19.2	19.4	0.3	93.4

rest group percentage decreases, the mean occupancy of the CCs naturally increases, because if we can invite more clients, we use more capacity of the CC.

The other output parameters that are displayed in the Tables 7.5, 7.6 and 7.7 do not differ very much over the scenarios. The percentage of subsequent round clients that cannot be invited within 22-26 months after their previous invitation is always 0.2 or 0.3 %. This percentage is very small and is acceptable in practice. The values for this output parameter in the case of $L = 790$ are larger and therefore worse. The average travel time is about 19 minutes and increases slightly when the objective value gets worse. Also the percentage of clients who cannot be invited in the nearest CC increases slightly when the objective value increases. Both these effects are visible by increasing L or decreasing ϵ over all possible scenarios. A better objective is smaller (we minimize) and this objective value consists of the different output parameters, so a larger objective (worse) also means output parameter values which are worse/higher.

Concluding, by lowering the dimension of the perturbation vector we get robust solutions which are more useful in practice and have better / more realistic output parameters. The tolerance level also makes a difference. The robust optimisation results make use of a buffer capacity, the magnitude of this buffer capacity depends on the value of the safety parameter γ. If the tolerance level ϵ becomes larger, γ becomes smaller and we will have less buffer capacity. If the dimension of the perturbation vector increases, L becomes larger, then γ also becomes larger and we have more buffer capacity. This buffer capacity is visible in Figure 7.2. The yellow bars are the number of clients that can be invited in a week in CC 'L' if the mean values for participation and referral rate are used. The blue bars indicate how many clients are actual invited in the robust solution, scenario $\epsilon = 10\%$ and $L = 5$. The differences between the bars in one week represent the buffer capacity. The buffer capacity can also be seen in the occupancy rates in the tables, a higher occupancy means a lower buffer capacity.

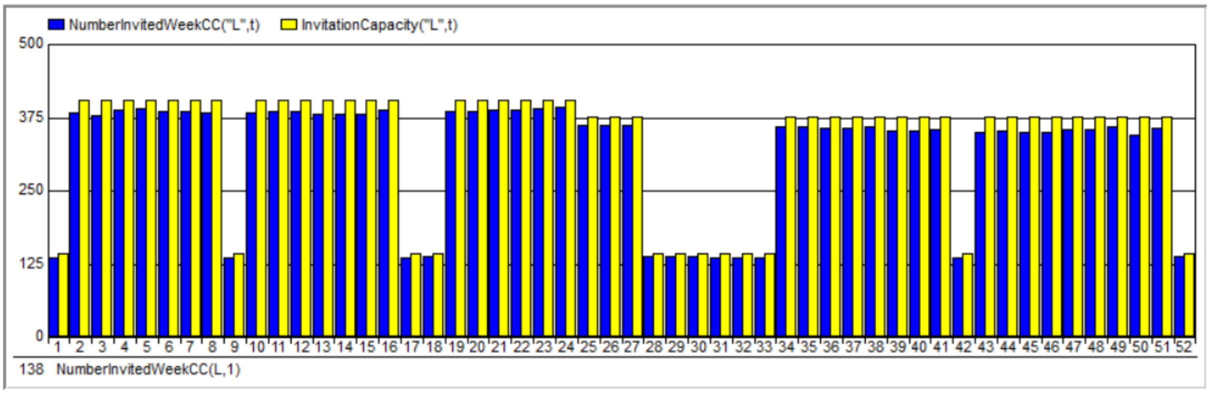


Figure 7.2: The available and used invitation capacity of CC 'L' in the robust scenario with $\epsilon = 10\%$ and $L = 5$.

As explained in Section 7.3 the given integer solutions of the robust optimisation problem are not the optimal integer solutions. We obtained these integer solutions from solving the robust optimisation problem by using an LP-relaxation. The optimality gab value that is given in Tables 7.4-7.7 gives us an indication of how good the obtained integer solution is. For low dimensions of the perturbation vector the optimality gab lies around -2%, which means that the obtained integer solution is at most 2% worse than the best integer solution. For $L = 790$ the optimality gab is larger, namely 5-6%. To explain why the optimality gab values differ per scenario we first should mention that the effect of ϵ on γ is larger when L is large then the effect of ϵ on γ when L is small.

If we look at the optimality gab values in scenarios with low values for L, so Tables 7.5-7.7, we see that if ϵ increases the optimality gab values become smaller in absolute value (closer to zero). This effect is explained in the following way. If ϵ becomes larger, then γ becomes smaller. With smaller γ we have less buffer capacity and therefore a larger solution space. This larger solution space will result in a better objective value which is smaller because we minimize. However we have negative objective values and therefore the absolute value of the objective function becomes larger when ϵ increases. For the relative optimality gab value we divide by this objective value and dividing by a number that is larger in absolute value results in an optimality gab which is smaller in absolute value. Therefore we have different optimality gab values for the different values of ϵ with low L despite the absolute difference in objective value of the LP and integer solutions remains constant. The same holds for the increasing (in absolute value) optimality gab values in Tables 7.5-7.7 when ϵ is constant and L increases. Due to a larger L the objective values becomes smaller in absolute value. Dividing by a smaller value results in a larger relative optimality gab value in absolute value.

In Table 7.4 we see another trend in the optimality gab values. L has the constant value of 790 and when ϵ increases the absolute value of the optimality gab also increases, so the relative difference between LP and integer objective value becomes larger. This effect is different than described above because the effect of ϵ on γ is larger with this large value for L. Due to this large effect the solution space changes significantly by changing ϵ, this change was with small L of minor importance. When the solution space changes the solution space of the LP-relaxation grows significantly more than the solution space of the integer problem. This is because there are more real numbers than integer numbers. By increasing ϵ the absolute difference between the LP objective and the integer objective therefore becomes larger and this results also in a larger relative optimality gab.

7.4.2 Compare MILP and robust solutions

To compare the results of the different scenarios of robust optimisation with the results of the MILP we give a recap of the MILP results in Table 7.8. The mean occupancy of the CCs is also in this table defined by equation (7.2), which is different from the results in Section 5.2.1.

Table 7.8: Recap: The results of the MILP in different scenarios with experiment 5.

Scenario	PR	RR	Objective value	Rest group %	Average Travel Time [min]	% not in nearest CC	% not within 22-26 months	Mean Occupancy CC %
A	73%	4.7%	-145,013,342	3.3	16.3	17.2	0	100
B	76%	5.1%	-52,774,745	14.4	15.8	18.8	3.5	88.3
C	70%	5.1%	-123,615,447	7.0	16.3	17.4	0	96.4
D	76%	4.3%	-161,794,976	0	16.4	18.3	0	103.5
E	70%	4.3%	-167,889,787	0	15.7	15.5	0	106.7

We immediately see that the scenarios of the robust optimisation model with $L = 790$ are comparable with scenario B of the MILP, which is the worst case scenario. This confirms that the robust optimisation model with $L = 790$ does not give desirable results for practice. When we look at the values for the output parameters Average Travel Time and Percentage Not in Nearest CC in Table 7.8, we see that these values are better than the values of the robust scenarios in Tables 7.4, 7.5, 7.6 and 7.7. This is caused by the fact that in robust optimisation all CC and weeks have a planned buffer capacity which implies that clients need to be spread more over the different CCs. By spreading the clients over the different CCs, some clients will have to travel longer and are not linked to the nearest CC.

In order to compare the robust optimisation results with the MILP results in a quantitative way, we calculated the objective value difference between all possible scenarios of the robust optimisation model and the MILP. We use the relative improvement in objective function from MILP to robust optimisation, which is defined as follows.

$$\text{improvement} = \frac{\text{objective scenario robust} - \text{objective scenario MILP}}{\text{objective scenario MILP}} \cdot 100\% \tag{7.3}$$

These improvement percentages are shown in Table 7.9 for all scenarios of robust optimisation and all scenarios of the MILP. Again, we see here that the scenarios of robust optimisation with $L = 790$ are comparable with MILP scenario B, because the relative improvement in objective value are 10,12,15 and 33%. All improvements in the column of scenario B are positive which means that the results of the robust scenarios are better than the worst case scenario results of the MILP. If we compare the robust scenarios with the other scenarios of the MILP (A, C, D and E) we see that the robust solutions are worse than the MILP, the improvements are negative. This is expected, because robust optimisation gives in on capacity due to the reserved buffer capacity.

The most interesting column of Table 7.9 to look at is the column corresponding to MILP A, because this is the scenario where the nominal value for participation and referral rate are used. We see that the robust scenarios with lower values for L come closer to the MILP solution. The robust scenarios are about 20-30% worse than the mean MILP A scenario, but in replace for a worse objective function you get a safe solution which protects the invitation strategy against uncertainty. The probability that clients cannot get their intake appointment in the planned week and CC is in the robust scenarios smaller than in the MILP A scenario. In Section 7.4.3 we say more about this probability of violating the limited capacity constraint.

Table 7.9: The relative improvement [%] in objective function from MILP to robust optimisation.

ϵ	L	MILP A	MILP B	MILP C	MILP D	MILP E
5%	7	-32	86	-20	-39	-41
5%	8	-33	84	-22	-40	-42
5%	790	-60	10	-53	-64	-65
10%	5	-28	99	-15	-35	-37
10%	6	-29	95	-17	-36	-39
10%	7	-30	92	-18	-37	-40
10%	8	-31	89	-19	-38	-41
10%	790	-59	12	-52	-64	-65
20%	4	-23	111	-10	-31	-34
20%	5	-25	107	-12	-33	-35
20%	6	-26	103	-13	-34	-36
20%	7	-27	100	-15	-35	-37
20%	8	-28	97	-16	-36	-38
20%	790	-58	15	-51	-63	-64
75%	790	-51	33	-43	-57	-58

The adherence that arises when solving the robust optimisation problem for all different scenarios has large similarities to the adherence we already found with the MILP. This is to be expected, because we still want to allocate the clients to a CC that is favourable for them, so nearby. Making the solution robust does not change this, because we only can invite less clients in a week. However this holds for all weeks and all CCs and therefore the adherence from postcode areas to CC does not change. The only difference that we do see is that in the robust solutions some postcode areas use very small percentages of capacity in CC that were not used in the MILP. These very small fractions of clients that are linked to other CCs than in the MILP can be declared by the fact that within robust optimisation more shortage of intake capacity is present. Due to this shortage, some clients might divert to other CC where still some capacity is left. However these ”backup” CCs also have shortage and therefore the new adherence linkings are of a very small magnitude.

One of the things that is worth mentioning is the fact that the robust optimisation solutions use more postcode areas in one single week and CC to invite clients from, than the MILP solutions did. In the MILP on average 4 different postcode areas where linked to one single week and CC, whereas the robust optimisation model uses 13 different postcode areas for one week and one CC. This effect can be explained by the fact that by using more postcode areas the uncertainty in the parameter ρ_p can be spread over more postcode areas. Spreading uncertainty over a larger amount of random variables reduces the total uncertainty because the different postcode areas can compensate each other. One postcode area may have a lower ρ_p value than the mean $\bar{\rho}$ and another postcode area can then have a larger ρ_p. Therefore, we can invite more clients when we use more different postcode areas simultaneously. This effect of using more postcode areas simultaneously does not change the adherence numbers, because the linking between postcode area and CC does not change. We only spread the clients from a single postcode area over multiple weeks and use multiple postcode areas in one week in robust optimisation. Whereas in the MILP we invite a whole postcode area in one week and the next postcode area in the next week.

7.4.3 Probability of constraint violation

Now we have robust solutions for inviting clients to weeks and CCs, it is interesting to look at the actual probability of constraint violation. We want to know what the probability is that clients who are invited on intake capacity in a week and CC cannot have their intake appointment in that week or CC, because to many clients need an intake at the same time. Or in other words, given the number of clients we invited for a week in a CC, what is the probability that more clients need an intake appointment than intake slots are available in that week and CC?

Let N be the number of clients that we invited in a week t for a CC c, i.e. $N = \sum_p x_{p,c}^t$. Let Y be the random variable that represents the number of clients that have a positive result and need an intake appointment in week t in a CC c. We know that clients have a participation rate and a referral rate, which gives us a probability of $\bar{\rho} = 0.03443$ that an invited client needs an intake appointment, as described in Section 6.1. We assume that all clients react independently of each other and the results of different clients do not depend on each other. Also participating and referral of one client do not depend on each other. Due to this assumptions we have that the number of clients that need an intake appointment is binomially distributed with the number of invited clients and the probability of intake needed as parameters. In other words, Y is binomially distributed with parameters N and $\bar{\rho}$, as shown in equation 7.4.

$$Y \sim Bin(N, \bar{\rho}) \tag{7.4}$$

When we use this distribution of the random variable Y, we can calculate for each week and for each CC the probability that out of the $N = \sum_p x_{p,c}^t$ invited clients, more than I_c^t need an intake appointment by the following equation.

$$\begin{aligned} \mathbb{P}(\text{constraint } (c,t) \text{ violation}) &= \mathbb{P}(Y > I_c^t) \\ &= 1 - \sum_{i=0}^{I_c^t} \binom{N}{i} \cdot \bar{\rho}^i \cdot (1-\bar{\rho})^{N-i} \\ \text{Where, } N &= \sum_p x_{p,c}^t \end{aligned}$$

For all different scenarios of the robust optimisation model we calculated these probabilities of constraint violation for all weeks and CCs. Table 7.10 shows the mean probability of violating a constraint when we use the binomial distribution for Y. The mean is taken over all weeks and CCs. As we can see the probability of constraint violation lies around 30%, which means that when we invite clients according to the robust solution, the probability that some clients cannot have their intake appointment in the planned week and CC is 30%. This probability decreases when the dimension of the perturbation vector increases, because we then plan more buffer capacity. When the tolerance level ϵ increases the probability of constraint violation increases because we allow to invite more clients.

Table 7.10: The mean probability of violating a limited capacity constraint.

ϵ	L	Binomial distributed
5%	7	0.297
5%	8	0.295
5%	790	0.253
10%	5	0.308
10%	6	0.304
10%	7	0.301
10%	8	0.299
10%	790	0.254
20%	4	0.319
20%	5	0.314
20%	6	0.311
20%	7	0.308
20%	8	0.305
20%	790	0.256
75%	790	0.267

These probabilities of violation of 30% are large and not desirable in practice. Also this 30% does not align with the maximum constraint violation probability of ϵ which is to be guaranteed by robust optimisation. We therefore think that although the binomial distribution is theoretically the correct distribution to describe the number of positive results Y, we might need to use another distribution that better describes practice. The main problem with the binomial distribution is that the variance of this distribution is quite large. We can best explain the problem with an example. Suppose we have invited $N = 300$ clients for a week and CC. In expectation $300 \cdot 0.03443 \approx 10$ clients then need an intake appointment. When we use the worst case value for the probability that an invited client needs an intake, which is $\bar{\rho} + \hat{\rho} = 0.03876$, we would need $300 \cdot 0.03876 \approx 11.5$ intake slots. If we use the binomial distribution to calculate the probability that we need 12 or more intake slots for those 300 clients this is 34%. Although this should not be possible because we said that not more than a fraction of 0.03876 of the clients need an intake appointment which is equivalent with 11.5 intake slots. Figure 7.3 shows the binomial distribution for Y where $N = 300$ in blue. We see that the variance is indeed quite large.

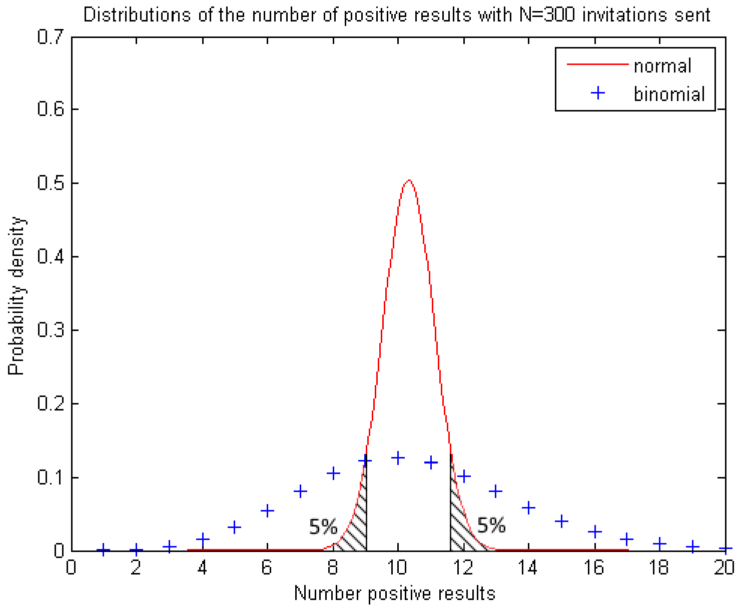


Figure 7.3: The probability density functions for the number of positive results when 300 invitations are sent.

We therefore propose to use a normal distribution for the number of clients that need an intake appointment. When Y is normally distributed we can adapt the standard deviation σ in such a way that the distribution for Y better fits the values that ρ can take and has a smaller variance. The proposed normal distribution for Y by $N = 300$ is given in red in Figure 7.3. For this normal distribution of Y we need the values of two parameters, the mean μ and the standard deviation σ, we then have

$$Y \sim N(\mu, \sigma).$$

The mean of this normal distribution for the number of positive results is the expected number of needed intake appointments when N clients are invited.

$$\mu = N \cdot \bar{\rho} \quad \text{where, } \bar{\rho} = 0.03443 \tag{7.5}$$

The value for the standard deviation (σ) of this normal distribution is based on the number of intakes that we can expect if the border values of ρ are considered. Remember that ρ is defined to vary in $[0.03010, 0.03876]$. The vertical black lines that mark the shaded areas in Figure 7.3 indicate the number of intake slots that is needed when exactly a fraction of $\bar{\rho} - \hat{\rho} = 0.03010$ or $\bar{\rho} + \hat{\rho} = 0.03876$ of the clients need an intake appointment. We want that the normal distribution for Y has a variance that makes sure that the probability of exceeding such borders is 5%. As depicted in Figure 7.3 we want to find σ such that the probability of the shaded areas equals 5% each. We calculate σ in the following way:

$$\begin{aligned}
\mathbb{P}(Y \leq \text{left border}) &= 0.05 && \iff \\
\mathbb{P}(Y \leq N \cdot (\bar{\rho} - \hat{\rho})) &= 0.05 && \iff \\
\mathbb{P}(Z \leq \frac{N \cdot (\bar{\rho} - \hat{\rho}) - \mu}{\sigma}) &= 0.05 && \text{where, } Z \sim N(0,1) \iff \\
\mathbb{P}(Z \leq \frac{-N \cdot \hat{\rho}}{\sigma}) &= 0.05 && \text{where, } Z \sim N(0,1) \iff
\end{aligned}$$

For the standard normal distribution (Z) the inverse of the cumulative distribution function is known, so we should have:

$$\frac{-N \cdot \hat{\rho}}{\sigma} = -1.645 \quad \iff \sigma = \frac{N \cdot \hat{\rho}}{1.645} \tag{7.6}$$

We now have the parameters for the normal distribution of $Y \sim N(\mu, \sigma)$, defined by (7.5) and (7.6), and we can calculate the probability of constraint violation when we invite $N = \sum_p x^t_{p,c}$ clients in week t in CC c by the following function. We use a continuous distribution for a discrete random variable Y. We therefore apply a continuity correction, where we replace I^t_c with $I^t_c + 0.5$ as the number of intake slots that is to be violated.

$$\begin{aligned}
\mathbb{P}(\text{constraint } (c,t) \text{ violation}) &= \mathbb{P}(Y \geq I^t_c + 0.5) \\
&= 1 - \int_{y=0}^{I^t_c + 0.5} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2\right) dy
\end{aligned} \tag{7.7}$$

For all different scenarios of the robust optimisation model we calculated these probabilities of constraint violation for all weeks and CCs, but now with this normal distribution for Y. Table 7.11 shows the mean probability of violating a constraint when we use the binomial distribution for Y and the normal distribution. As we can see the probability of constraint violation with the normal distribution is much smaller than for the binomial distribution. This is caused by the smaller variance of the normal distribution. The probability of constraint violation lies now around 8%. The probabilities also satisfy the tolerance levels ϵ as they are expected to be due to the robust optimisation. When L increases, the probability of violation decreases because we then plan more buffer capacity. When the tolerance level ϵ increases the probability of constraint violation increases because we allow to invite more clients.

Table 7.11: The mean probability of violating a limited capacity constraint.

ϵ	L	Binomial distributed	Normal distributed
5%	7	0.297	0.068
5%	8	0.295	0.065
5%	790	0.253	0.018
10%	5	0.308	0.083
10%	6	0.304	0.078
10%	7	0.301	0.074
10%	8	0.299	0.071
10%	790	0.254	0.019
20%	4	0.319	0.099
20%	5	0.314	0.093
20%	6	0.311	0.088
20%	7	0.308	0.084
20%	8	0.305	0.080
20%	790	0.256	0.022
75%	790	0.267	0.036

We think that the normal distribution for the number of clients with positive results (Y) based on how many we invited (N) is a better fit in practice than the binomial distribution, because with the large number of clients that will be invited we expect a small variance. We only give the mean probability of violating a limited capacity constraint, however the probability of violating a constraint is not the same for all weeks or CCs. The constraint violation probability does depend on how many intake slots are actual available in a week in a CC. In holiday weeks we have fewer available intake slots and we can invite fewer clients. The standard deviation of the normal distribution is also dependent on the number of invited clients and decreases when N decreases. With a smaller standard deviation the probability of violating the constraint becomes also smaller. The same holds when we compare different CCs. A CC with fewer available intake slots will invite less clients, causing a smaller standard deviation and therefore the probability of constraint violation in a small CC will be smaller than in a large CC with many available intake slots.

If you just use the invitation strategy of the average scenario A of the deterministic MILP, the probability that more clients need an intake appointment than intake slots are available in a week and CC will be 22% on average. This probability is calculated in the same way by using the normal distribution in equation (7.7). If we compare this with the probabilities in the normal distribution column of Table 7.11, we see that robust optimisation is really useful in order to reduce the probability of constraint violation. We reduce the probability of constraint violation from 22% to about 8%.

From the results of the robust optimisation model as described in the previous sections we can conclude that it is possible to find a robust / safe solution to the matching problem of inviting clients from postcode areas to CCs and weeks. A robust solution gives in on the quality of the solution in terms of inviting less clients and a higher travel time, but the advantage of a robust solution is that the probability that a client can have his intake appointment at the planned location and time increases. We think that the scenario which gives the best robust solutions for practice are the scenarios with a γ value of 4.7. The corresponding scenarios are $\epsilon = 10\%$ with $L = 5$ and $\epsilon = 20\%$ with $L = 7$. These scenarios give in 30% in the objective value compared to MILP scenario A, but these robust scenarios are 100% better than worst case (MILP B). These robust solutions have a rest group of 9%, an average travel time of 19.1 minutes and 19.3% of the clients cannot be invited in the nearest CC. Only 0.2% of the subsequent clients is not invited within 22-26 months. All these output parameters are acceptable, but a bit to large. However the main advantage of the robust solution in these two scenarios with $\gamma = 4.7$ is that the probability that clients cannot have their intake appointment at the planned week and CC is only 8% if we use the normal distribution for the number of positive results of the invited clients.

Chapter 8

Time uncertainty model

In order to take the time uncertainty such as response times of the participants into account we develop a Stochastic Dynamic Programming (SDP) model. This model determines the moment of sending the invitations in such a way that the possible intake appointment can take place at the desired week in the desired CC. The week and CC of the possible intake appointments are already determined by the MILP and/or the robust optimisation model. First, the throughput times of the system are analysed in Section 8.1 in order to determine the distributions for the time uncertainty. Section 8.2 gives the formulation of the general SDP model where all CCs and postcode areas are evaluated simultaneously and actions also include linking postcode areas to CCs. This general model gives rise to computational problems which are described in Section 8.3. This section also explains the decomposition idea of the general SDP into several single CC SDPs. In Section 8.4 we give the SDP model for one single CC, which results after decomposing the general SDP model. Finally, we explain in Section 8.5 which decomposition methods are known in the literature and what are the similarities to our decomposition approach. We also say something about the approximation error that we make by decomposing the general SDP into several single CC SDPs in this research.

8.1 Throughput times

The data that is used to analyse the throughput times of the system are given by BVO Oost. The dataset contains for all clients over the year 2017 the following dates:

1. The date at which the invitation is set up in ScreenIT. This date is the moment that BVO Oost gives the command to the packing centre to put together an invitation package.
2. The date at which the invitation and test package are actually packed and send to the client by the packing centre.
3. The date at which the test is returned and received by the laboratory.
4. The date at which the test is analysed and the result is reported.

The return date can be empty if the client does not participate, these entries in the data set are not taken into account. According to these dates three different throughput times can be calculated. These throughput times are defined as follows and measured in number of days.

Sending Time: The time that is needed to put together the invitation letter and test and send it to the client. This is defined by the number of days between the invitation date (1) and sending date (2) of the client.

Response Time: The time that a client needs to receive the invitation, take the test and send the test back to the laboratory. This is defined by the number of days between the sending date (2) and the return date (3).

Analysis Time: The time that the laboratory needs to analyse the test and determine the result. This is defined by the number of days between the return date (3) and result date (4).

The samples from the dataset that describe the above mentioned random variables are independently and identically distributed. This is because we assume that each client responses based on his own behaviour, and that clients do not influence each other. Also, the target group is big enough to assume that all clients behave in a similar way. We also assume that the throughput times are constant over time. In other words, sending, response and analysis times do not vary significantly throughout the year.

These assumptions are summarized in Assumption 8.1.1 and 8.1.2.

Assumption 8.1.1. *All clients respond independently from each other, but in the same way.*

Assumption 8.1.2. *The behaviour of clients is the same during the entire year.*

We used the methods as described in Law [2015, Ch. 6] to analyse the dataset and determine the probability distributions for the three different random variables that describe the throughput times. For all three random variables we have made a histogram of the samples from the dataset. These histograms indicate which values of the random variables occur and with which frequency.

Figure 8.1 shows the histogram for the sending time. In most of the cases the invitation is put together 2 days after the invitation is set up, almost 35%. This is the bar between values 2 and 3 on the x-axis. Invitations are set up in ScreenIT on all workdays of the week and the amount of invitations is equally distributed over the week. However, the wide range of values in Figure 8.1 indicates that the invitations are not put together each day of the week by the packing centre. Most of the invitations are put together on Wednesday and Friday. Therefore it can take up to sometimes 7 days until the invitation is send to the client. The average sending time is 3.8 days. This sending time implies that a client whose invitation is set up at the beginning of week n is not likely to have a response within the same week. We therefore have Assumption 8.1.3.

Assumption 8.1.3. *Clients who are invited in week n (set invitation) will not respond in week n and only start to respond from the next week $n+1$ on.*

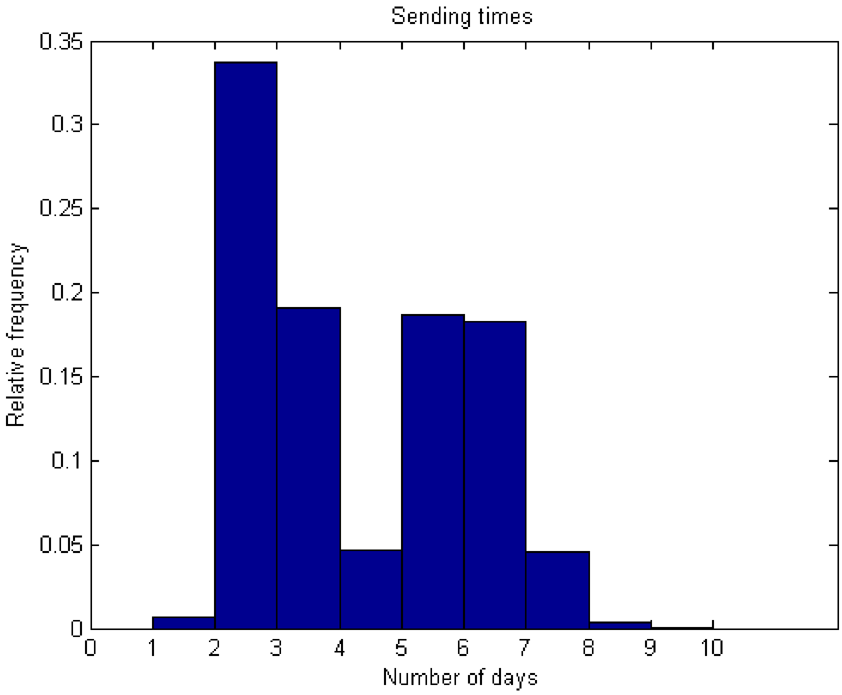


Figure 8.1: A histogram of the sending times.

Figure 8.2 shows the histogram for the response time. The first thing that stands out is that the days that clients need to respond seems to be periodic. Every seven days the same pattern is repeated, only with smaller frequencies. This is explained by the fact the invitations are packed and send mostly on Wednesday and Friday and postal services do not return the tests to the laboratory on Sundays and Mondays, also on Saturday not always tests are returned. For example when an invitation is sent on Friday it cannot be returned on Monday (which is 3 days response time) and an invitation sent on Wednesday cannot be returned on Saturday (also 3 days response time). This is repeated every 7 days again. Therefore, a lower bar in the histogram is present at a response time of 3, 10, 17, 24, ... days. Different combinations of sending and receiving days also imply lower bars after $1+k\cdot 7$, $2+k\cdot 7$, $4+k\cdot 7$ and $5+k\cdot 7$ with $k \in \mathbb{N}$. The magnitude of this effect depends on which sending and receiving day combination is used. This effect cannot appear with response times of $6+k\cdot 7$ or $7+k\cdot 7$ days, therefore these bars are largest in the histogram.

The smallest response time is two days. This is as expected because the test needs at least one day to be delivered by the postal service to the client and it needs again at least one day to be delivered to the laboratory. The average response time is 19.7 days. When the number of days increases the frequency decreases, this is explained by the fact that as time goes by more and more clients already have responded. After about 50 days we see a small increase in number of responses. This is because 42 days after sending the initial invitation a reminder is send to the clients who did not respond yet.

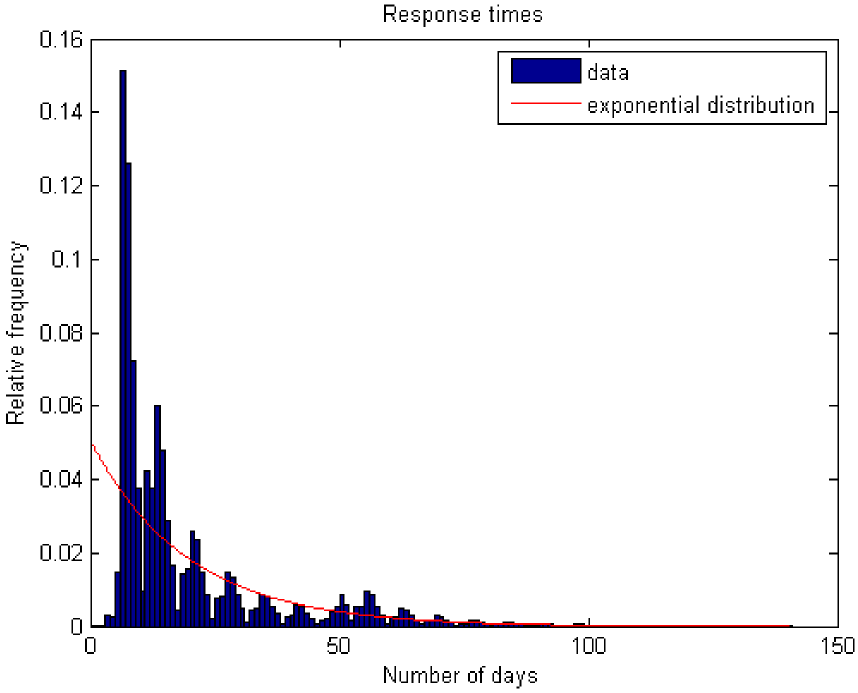


Figure 8.2: A histogram of the response times, including the exponential distribution to approximate.

The response time is measured in number of days, which is discrete, but we use a continues distribution to approximate these times. This is because in theory time is a continuous variable and the number of days that a client may need to respond can become very large. We postulate that the response time of clients is exponentially distributed, because the shape of the histogram has similarities with the probability density function of an exponential distribution. To substantiate this hypothesis we look at some statistics. The sample mean of the response time is 19.7 days, whereas the median is 13 days. The exponential distribution always has a larger mean than median. Furthermore, the skewness of the samples is 2.2 which implies skewness to the right. Finally the coefficient of variation of an exponential distribution is in theory equal to one. The coefficient of variation of the samples is 0.98. All these statistics support the hypothesis of the response times having an exponential distribution and we therefore make the following assumption.

Assumption 8.1.4. *The response times of clients are exponentially distributed.*

The exponential distribution that describes the response times is denoted with $f(x)$ for the density function and $F(x)$ for the cumulative distribution, which are given in (8.1) and (8.2). The stochastic variable for response time is denoted with X, which can take non-negative values x.

$$f(x) = \lambda e^{-\lambda x}, \qquad x \geq 0 \tag{8.1}$$
$$F(x) = \mathbb{P}[X \leq x] = 1 - e^{-\lambda x}, \qquad x \geq 0 \tag{8.2}$$

The parameter λ needs to be estimated from the dataset. This is done by using the Maximum Likelihood Estimator (MLE), as described in Law [2015, Sec. 6.5]. For exponentially distributed random variables the MLE for parameter λ is $1/\bar{x}$, where $\bar{x}$ is the sample mean. Therefore, we have $\lambda = 1/19.7 = 0.051$ which gives the best approximation for the response time. The exponential distribution that is used to approximate the distribution of the response time is plotted in red in Figure 8.2.
The exponential distribution has the memoryless property. This property implies that it does not matter how long a client already has the invitation in his possession, the probability that he responds within the coming t days is equal to the probability that a client responds within the first t days. This memoryless property is very useful when the developing a model that takes time uncertainty into account, because the only thing that we need to keep track of is now how many outstanding invitations are present at a certain time. It is not required to remember how long ago an invitation is sent.

The probability that a client with an outstanding invitation responds within the coming week is called q. This probability is needed in the transition probabilities of SDP as described in Section 8.2 and is given in (8.3). The value of 7.5 is used, because we use a continuous distribution to approximate discrete values. The response time within one week corresponds with the response time within 7 days, but in continuous time this is equivalent with less than 7.5 days.

$$q = \mathbb{P}[\text{ response within coming week }] = \mathbb{P}[X \leq 7.5] = 1 - e^{(-0.051 \cdot 7.5)} \approx 0.31 \tag{8.3}$$

Figure 8.3 shows the histogram for the analysis time. It can be seen that almost all received tests are analysed within one day, which means that they are analysed on the same day as they are received. This is a bit misleading because in reality the tests are scanned as received only just before analysing starts. Therefore, the analysing time is partly included in the response time. With this information we can say that we do not have to incorporate the analysis time individually in the model. However, when a test turns out to be positive (unfavourable) in week n it is not possible to plan an intake appointment for that client in the same week. The ”unfavourable result wait period” needs to be taken into account. This period has a length of 5 workdays (= one week) and is used to inform the client of his positive result.

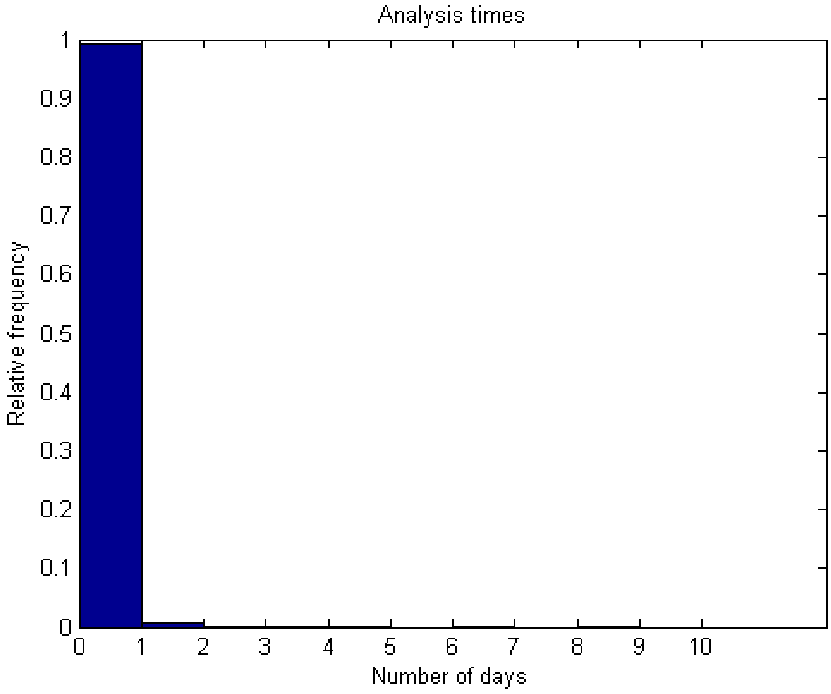


Figure 8.3: A histogram of the analysis times.

Concluding, the response time of clients is modelled as an exponential distribution with an average of 19.7 days. This results in a probability of 0.31 that a response at an outstanding invitation comes within the coming week. The analysing time is neglected, but due to the ”unfavourable result wait period” clients with a positive result in week n can only get an intake appointment in the next week $(n + 1)$. The probability that a client responses within the same week as the week that his invitation is set, equals zero.

8.1.1 Pré-announcement period

The throughput time analysis as described above only concentrates on the throughput times after setting up the invitation in ScreenIT. This invitation is the real invitation that contains the FIT. However, 14 days in advance a pré-announcement is made. In the current process at BVO Oost this pré-announcement is created for all clients and only clients that are invited for the first time will get a pré-announcement letter. Within these 14 days first round clients can unsubscribe, in that case no invitation is set up for this client. Additionally nothing happens within these 14 days. BVO Oost has two reasons for sending a pré-announcement, first they want to inform first round clients about what is coming and second they hope to save some money because fewer FIT have to be send. However mathematically this pré-announcement period only has disadvantages. First, a pré-announcement should be created two weeks earlier than the actual invitation which leads to a larger horizon over which we should predict whether capacity is available in the CCs. Second, a deterministic period of 2 weeks does not have a memoryless property, therefore we should keep track of the pré-invitation moment for all clients. This storing of information takes a lot of memory and can lead to computational problems. We therefore want to ignore the pré-announcement period. To investigate whether ignoring the pré-announcement period is possible we calculate the costs and revenues of sending pré-announcement letters.

The costs of sending pré-announcement letters to all first round clients in region East are approximately €$46,000$. A small percentage of these first round clients unsubscribe after getting the pré-announcment and before receiving an invitation. This implies that BVO Oost saves about €$8,500$ on invitations including tests, that have not to be sent. However this saving is much smaller than the costs for sending all the pré-announcement letters. When the pré-announcement letters will not be send any more, BVO Oost can save €$37,500$ per year. On a national level all screening organisations together can save approximately €$200,000$ each year by discarding the pré-announcement. We think that these numbers motivate enough not to consider the pré-announcement in this research and eventually in practice.

Concluding, the moment of invitation in this research is defined as the moment of setting up the actual invitation in ScreenIT and we neglect the pré-announcement period.

8.2 Stochastic Dynamic Programming general model

A Stochastic Dynamic Program (SDP) is a regularly used model in decision making. In an SDP a finite number of discrete time points is evaluated. At each time point (decision epoch) a decision should be made based on the current state of the system. All possible states of the system are called the state space and a state describes the situation at a given time point. The possible actions that can be taken are called the action space. An action corresponds to a decision that is made which changes the state of the system. Such an action implies in which state the system will be at the next time point. Multiple next states are possible, therefore the current state and taken action give transition probabilities that describe which state will be visited in the next time step. Each state that is visited gives direct costs, which indicate which states are better to be in than others. At all the subsequent time points again an action should be taken based on the current state of the system. This process repeats until the end of the time horizon is reached. The goal is to find optimal actions such that the costs of the (possible) visited states are minimized. More information on SDPs can be found in Puterman [2005].

Based on the assumptions and throughput times as described in the previous section, we can develop an SDP for inviting clients to the colon cancer screening program. All elements of this SDP in this research are described in the remainder of this section. The general idea is that in each week a number of clients should be invited from the postcode areas. These clients are also linked to a CC. This inviting and linking should be done in such a way that all clients are invited at the end of the year and the intake appointments at a CC should fit within the capacity. The goal is to approach the solution of the MILP/robust model, because these solutions have the optimal allocation of clients to CC and weeks. The transition probabilities depend on how many clients will participate, how many clients will respond to the outstanding invitations in a week and the transition probabilities depend also on the number of positive results. As clients behave each in their own way we can make the following assumption.

Assumption 8.2.1. *The number of participating clients, the number of responses within a week and the number of positive results are independent of each other and independently of postcode area, CC and week. The random variable for one CC does not depend on the random variable for another CC.*

Decision Epochs

The decision epochs are the moments at which the system is evaluated and an action should be taken. In this case, the system is evaluated at the beginning of each week of the year. A year consists of 52 weeks, and an extra decision epochs $T = 53$ is added to represent the end of the year. At each decision epoch $n < T$ an action should be taken and no decision is made at $n = T$.

$$\mathcal{N} = \{1, 2, \ldots, T\}, \quad n \in \mathcal{N}$$

The clients live in different postcode areas, to these areas the invitations should be send.

$$\mathcal{P} = \{1, 2, \ldots, P\}, \quad p \in \mathcal{P}$$

For all postcode areas, N_p is the number of clients that should be invited that live in postcode area p. The invitations to the postcode areas are linked to the different CCs, therefore we have the following set of CCs.

$$\mathcal{C} = \{1, 2, \ldots, C\}, \quad c \in \mathcal{C}$$

State space

The entire state space $\mathcal{S}$ consists of all possible state spaces at decision epochs n.

$$\mathcal{S} = \{S_n\}_{n \in \mathcal{N}}$$

The state space at decision epoch n is denoted with S_n, a possible state in this state space is denoted with σ_n. Such a state consists of four quantities which together describe the state of the system.

- $O_n = (O_{1,n}, O_{2,n}, \ldots, O_{C,n})$, where $O_{c,n}$ is the number of outstanding invitations at the beginning of week n that are linked to CC c. Due to Assumption 8.1.4 and the corresponding memoryless property the number of outstanding invitations gives enough information of the system. These outstanding invitations are already sent to clients that live in a postcode area p and linked CC c, but these clients did not yet respond. The number of clients that will not participate is already excluded in this number, so only invitations that will be used are counted.
- $W_n = (W_{1,n}, W_{2,n}, \ldots, W_{C,n})$, where $W_{c,n}$ is the number of positive test results at the beginning of week n that correspond to CC c. This number of clients need to get an intake appointment in CC c in the next week $(n + 1)$.
- $CV_n = (CV_{1,n}, CV_{2,n}, \ldots, CV_{C,n})$, where $CV_{c,n}$ is the total number of sent invitations until week n to clients that are linked to CC c. This includes both the responding and the non-responding clients and does not include the invitations send in week n itself.
- $PV_n = (PV_{1,n}, PV_{2,n}, \ldots, PV_{P,n})$, where $PV_{p,n}$ is the total number of sent invitations until week n to clients that live in postcode area p. This includes both the responding and the non-responding clients and does not include the invitations send in week n itself.

We denote with parameter O_c^0 the number of outstanding invitations corresponding to CC c, which are still open from previous invitation year. W_c^0 is the number of positive results in CC c from previous year that still need an intake appointment in the coming year. The above four quantities together form the state space at decision epoch n.

$$\begin{aligned} S_n = \{\sigma_n = (O_n, W_n, CV_n, PV_n) \quad | \quad & 0 \le O_{c,n} \le O_c^0 + CV_{c,n}, \\ & 0 \le W_{c,n} \le W_c^0 + O_c^0 + CV_{c,n}, \\ & O_{c,n} + W_{c,n} \le W_c^0 + O_c^0 + CV_{c,n}, \\ & 0 \le CV_{c,n-1} \le CV_{c,n} \le \sum_p PV_{p,n}, \\ & 0 \le PV_{p,n-1} \le PV_{p,n} \le N_p, \\ & \sum_p PV_{p,n} = \sum_c CV_{c,n} \qquad \forall c, p\} \end{aligned} \tag{8.4}$$

It is not possible to have more outstanding invitations than $O_c^0 + CV_{c,n}$ in a week, because this is the number of invitations that is send so far this year, increased with the outstanding invitations from previous year. This number, $O_c^0 + CV_{c,n}$, increased with the outstanding positive results from previous year (W_c^0) is the theoretical upper bound for the number of positive results in a week. All invitations that are send up-till week n in CC c (including what is left over from previous year), can have status outstanding or positive result in week n, or they already disappeared from the system by being a non-participant, a negative result or an already planned intake appointment. Therefore, we have the bound $O_{c,n} + W_{c,n} \le W_c^0 + O_c^0 + CV_{c,n}$. The number of send invitations that corresponds to a CC is non-decreasing over the weeks, just like the number of send invitations to a postcode area. The total number of send invitations up-till week n that are linked to CC c is bounded from above by the total number of send invitations to all postcode areas. The total number of send invitations up-till week n in postcode area p is bounded from above by the total number of clients that need to be invited in postcode area p. Both $CV_{c,n}$ and $PV_{c,n}$ indicate how many invitations are send to a CC respectively a postcode area. Therefore the total number of send invitations to all postcode areas should always be equal to the total number of send invitations to all CC, $\sum_p PV_{p,n} = \sum_c CV_{c,n}$.

At the start of the horizon $(n = 1)$ we have that $CV_{c,1} = 0 \quad \forall c$ and $PV_{p,1} = 0 \quad \forall p$, because no invitations are send yet. $W_{c,1}$ will have value W_c^0, because no other test results are known yet. The quantity $O_{c,n}$ will have value O_c^0 at the beginning of the year. These observations give us the state space at decision epoch $n = 1$, which consists of only one state.

$$S_1 = \{\sigma_1 = (O_1, W_1, CV_1, PV_1) \quad | \quad O_{c,1} = O_c^0, W_{c,1} = W_c^0, CV_{c,1} = 0, PV_{p,1} = 0 \qquad \forall c, p\}$$

Action space

The action space consists of all possible actions that are allowed in a given state at a given decision epoch.

$$\begin{aligned}\mathcal{A} &= \{A_n\}_{n\in\mathcal{N}} \qquad (8.5)\\ A_n(\sigma_n) &= \left\{ a_n = \begin{pmatrix} a^n_{1,1} & a^n_{1,2} & \cdots & a^n_{1,C}\\ a^n_{2,1} & a^n_{2,2} & \cdots & a^n_{2,C}\\ \vdots & \vdots & \ddots & \vdots\\ a^n_{P,1} & a^n_{P,2} & \cdots & a^n_{P,C}\end{pmatrix} \mid 0 \le a^n_{p,c} \quad \forall p,c,n \quad \text{and} \quad \sum_c a^n_{p,c} \le N_p - PV_{p,n} \quad \forall p,n \right\}\end{aligned}$$

The actions $a^n_{p,c}$ are the number of new send invitations at the beginning of week n to clients in postcode area p which are linked to CC c. The action space is bounded by 0 from below, because you cannot send a negative number of invitations. It is not possible to send more invitations to a postcode area than there are clients left in that area. $N_p - PV_{p,n}$ is the number of clients that have not received an invitation yet.

Direct costs

The direct costs in decision epoch n when the system is in state σ_n and action a_n is taken are denoted with $k_n(\sigma_n, a_n)$. The $(\bullet)^+$-operator gives value $\bullet$ if this value is positive and value 0 if $\bullet$ is negative, i.e. $(\bullet)^+ = \max\{\bullet, 0\}$. The β factors are used to weight the different costs, based on importance.

$$\begin{aligned}k_n(\sigma_n, a_n) = &\sum_c \beta_w \cdot (W_{c,n} - I_c^{n+1})^+ + \sum_c \beta_i \cdot \left(W_{c,n} - \sum_{n'=n+1}^{n+3} I_c^{n'}\right)^+ + \sum_c \beta_g \cdot |W_{c,n} - G_c^{n+1}|\\ &+ \sum_{p,c} \beta_d \cdot \left(a^n_{p,c} \cdot \hat{D}_{p,c}\right) - \sum_{p,c} \beta_f \cdot \left(a^n_{p,c} \cdot F_{p,c}\right) \qquad \text{for} \quad n = 1, \ldots, T-1\end{aligned} \qquad (8.6)$$

In order to make sure that participants can have their intake in the desired week, costs (β_w) are given to states where the number of positive results in week n exceeds the number of available intake slots in week $n+1$. Namely, positive results of week n should get an intake appointment in week $n+1$. When this difference becomes larger, the costs will increase because this will imply waiting time for clients that need an intake appointment. A quality requirement is that all clients should get an intake appointment within 3 weeks after the result. Therefore costs (β_i) are given to states where the number of positive results in week n exceeds the number of available intake slots in the next three weeks ($n+1, n+2, n+3$). In this way each client that needs to wait a week on top of three weeks has costs β_i. We calculate only costs for number of clients and do not keep track of which client waits. We assume that all clients get an intake appointment according to the First Come First Served policy. Simultaneously, we want to have that the amount of positive results is the same as was planned in the MILP. Therefore, we have costs (β_g) when the number of positive results in week n deviates from the goal in week $n+1$, because positive results need an intake appointment in the next week. By approaching the goal of the MILP in this SDP we want to realize also the desired evenly spread workload in the CCs, as was already determined in the MILP. We define the goal of positive results corresponding to CC c in week n by G^n_c. Where $AH_{p,c}$ is the adherence output from the MILP, $x^n_{p,c}$ is the number of invitations from the MILP and $\bar{\rho}$ is the mean fraction of positive results from the total number of send invitations.

$$\begin{aligned}G^n_c &= \sum_p AH_{p,c} \cdot x^n_{p,c} \cdot \bar{\rho}\\ &= \text{ the number of planned intake appointments in week } n \text{ in CC } c \text{ according to the MILP}\end{aligned}$$

The last two terms of the direct costs $k_n(\sigma_n, a_n)$ are costs that are incurred by taking action a_n. These costs ensure that clients from postcode areas are linked to CCs that are favourable. The term with β_f maximizes the number of clients that is linked to the nearest CC. Due to maximizing this β_f term comes with a -1. The term with β_d minimizes the total travel resistance for all clients.

At the end of the horizon ($n = T$) the costs are as follows.

$$\begin{aligned}k_T(\sigma_T) = &\sum_c \beta_w \cdot (W_{c,T} - I_c^{T+1})^+ + \sum_c \beta_i \cdot \left(W_{c,T} - \sum_{n'=T+1}^{T+3} I_c^{n'}\right)^+ + \sum_c \beta_g \cdot |W_{c,T} - G_c^{T+1}|\\ &+ \sum_p \beta_v \cdot (N_p - PV_{p,T}) + \sum_c \beta_b \cdot (O_{c,T} - \hat{O}_c)^+\end{aligned} \qquad (8.7)$$

The first three terms are the same as for the direct costs in decision epoch n. The next weeks $T+1$, $T+2$ and $T+3$ are the first weeks of the new year. Second, it is desired that all clients are invited during the year. Therefore, costs are incurred when the total number of sent invitations at the end of the year ($PV_{p,T}$) is less than the number of clients N_p, this is done for all postcode areas. To prevent that in the last decision epoch ($n = T-1$) all left clients get an invitation such that no rest group arises, we introduce costs β_b. These costs occur when the number of outstanding invitations at $n = T$ become larger than the maximum number of outstanding invitations for next year, $\hat{O}_c$. This border $\hat{O}_c$ is based on the expected waiting time for intake appointments. Clients at the beginning of next year are not allowed to wait longer than one week in expectation. The costs β_b are larger than the costs for the rest group (β_v).

The scaling factors denoted with β indicate which parts of the direct costs are more important than others. The importance factors are given in Table 8.1 and are determined in consultation with BVO Oost. The different costs are listed from most important to least important. Waiting a week in addition of three weeks waiting time is the most expensive because the quality criterion is that all clients should get an intake appointment within 15 workdays. An other quality criterion is that the rest group at the end of the year is minimized. However, waiting is more important, because these clients are already in the process. As described above costs β_b are more important than β_v in order to prevent that all left clients will be invited at the end of the year. β_f and β_d make sure that invitations of clients are matched to CCs that are desired. In other words we first maximize the number of clients linked to the nearest CC and second we minimize the travel resistance of the clients, in the same way as in the MILP. On the other hand clients do not want to wait long until their intake appointment can take place. Therefore, costs β_w are of the same order of importance as the costs for desirable CC linking. The least important costs are β_g, because these costs are not directly linked to clients. These costs only describe in which state the SDP should be in an ideal situation. In each week the number of positive results is then equal to the positive results as determined in the MILP.

These importance factors need to be normalized because not all direct costs parts are of the same magnitude. All direct costs indicate a number of clients except for the travel resistance part. The travel resistance of a client varies between 0 and 150 and has therefore a size class of 100. This means that the importance factor corresponding the travel resistance should be divided by 100 in order to get the right scaling factor (β_d). All other β's have a size class of 1, so the scaling factor is equal to the importance factor.

Table 8.1: Determination of the scaling factors β in the direct costs function.

β	Description	Importance factor	Size class	Scaling factor
β_i	waiting week above three weeks	100	1	100
β_b	border outstanding invitations end of year	80	1	80
β_v	rest group end of year	50	1	50
β_f	linking to nearest CC	40	1	40
β_w	waiting week	30	1	30
β_d	travel resistance	20	100	0.2
β_g	reach MILP goal per week	4	1	4

Transition probabilities

According to Assumption 8.1.2 we expect the transition probabilities to be the same at each decision epoch, i.e. they do not depend on time. However, the number of available intake slots in a CC varies over time and these intake slots influence the invitation process, so the transition probabilities are not stationary. Given the current state of the system and the chosen action, the transition probabilities are the probabilities that the system is in state (h,i,j,l) next week.

$$\begin{aligned}\mathbb{P}_n[(h,i,j,l) \mid \sigma_n, a_n] &= \mathbb{P}_n[(h_c,i_c,j_c,l_p) \quad \forall p,c \mid \sigma_n, a_n] \\ &= \prod_{c,p} \mathbb{P}_n[(h_c,i_c,j_c,l_p) \mid \sigma_n, a_n]\end{aligned}$$

The second equality holds, because all CCs are evaluated separately, so they are independent of each other (Assumption 8.2.1). When events are independent, you can multiple the probabilities of these events.

State (h_c,i_c,j_c,l_p) for CC c and postcode area p can be reached from σ_n by different number of responses in week n. The stochastic variable $R_{c,n}$ represents the number of responses in week n for CC c from the outstanding invitations. Also the number of positive results and number of participations is needed to determine in which state the system will be in the next week $n+1$. Let $Y_{c,n}$ be the random variable that represents the number of positive results for CC c in week n and let $B_{c,n}$ be the number of participants (based on send invitations $a^n_{p,c}$) for CC c in week n. In order to calculate the transition probabilities we condition on the values that $R_{c,n}$, $Y_{c,n}$ and $B_{c,n}$ can take.

$$\begin{aligned}\mathbb{P}_n[(h_c, i_c, j_c, l_p) \mid \sigma_n, a_n] = \\ \sum_{r_{c,n}=0}^{O_{c,n}} \sum_{y_{c,n}=0}^{r_{c,n}} \sum_{b_{c,n}=0}^{\sum_p a^n_{p,c}} & \mathbb{P}_n[(h_c, i_c, j_c, l_p) \mid R_{c,n} = r_{c,n}, Y_{c,n} = y_{c,n}, B_{c,n} = b_{c,n}, \sigma_n, a_n] \\ & \cdot \mathbb{P}[B_{c,n} = b_{c,n} \mid \sigma_n, a_n] \\ & \cdot \mathbb{P}[Y_{c,n} = y_{c,n} \mid R_{c,n} = r_{c,n}, \sigma_n, a_n] \\ & \cdot \mathbb{P}[R_{c,n} = r_{c,n} \mid \sigma_n, a_n]\end{aligned}$$

The probability for which values $B_{c,n}$ can take does not include the conditions which values $R_{c,n}$ and $Y_{c,n}$ can take because these values give no extra information for the number of participants. In order to be able to determine the number of positive results $Y_{c,n}$ we first need to know the number of responses. The summation of $y_{c,n}$ can only go up to $r_{c,n}$, because there cannot be more positive results than responses. The three random variables $B_{c,n}$, $R_{c,n}$ and $Y_{c,n}$ are all independent of each other, Assumption 8.2.1. All the above mentioned probabilities are known and will be discussed below.

The next state of the system is certain, once the values of the three random variables are known. Namely:

- The number of outstanding invitations increases with the new invitations that correspond to participants, $b_{c,n}$. The outstanding invitations are the outstanding invitations of clients that will participate. On top of that the number of outstanding invitations decreases with the number of responses $r_{c,n}$. This gives the closed form $h_c = O_{c,n} + b_{c,n} - r_{c,n}$.
- The number of positive results at the beginning of the next week is given as shown in the equality $i_c = (W_{c,n} - I^{n+1}_c)^+ + y_{c,n}$. The first term indicates how many positive results are left over from previous week, this is $(W_{c,n} - I^{n+1}_c)$, as long as this number is positive. The second term $(y_{c,n})$ indicates the new positive results that arrive during week n for CC c.
- The total number of sent invitations that are linked to CC c in the next week is the number of sent invitations in the previous week plus the number of new send invitations that is linked to CC c $(\sum_p a^n_{p,c})$. This gives the closed form $j_c = CV_{c,n} + \sum_p a^n_{p,c}$.
- The total number of sent invitations to postcode area p in the next week is of course the number of sent invitations in the previous week (n) plus the number of new sent invitations $(\sum_c a^n_{p,c})$, which gives $l_p = PV_{p,n} + \sum_c a^n_{p,c}$.

Combining these four closed forms, we get the following probability distribution.

$$\mathbb{P}_n[(h_c, i_c, j_c, l_p) \mid R_{c,n} = r_{c,n}, Y_{c,n} = y_{c,n}, B_{c,n} = b_{c,n}, \sigma_n, a_n] = \begin{cases} 1 & \text{if } \; h_c = O_{c,n} + b_{c,n} - r_{c,n} \\ & \text{and } \; i_c = (W_{c,n} - I^{n+1}_c)^+ + y_{c,n} \\ & \text{and } \; j_c = CV_{c,n} + \sum_p a^n_{p,c} \\ & \text{and } \; l_p = PV_{p,n} + \sum_c a^n_{p,c} \\ 0 & \text{otherwise} \end{cases}$$

The probability distribution for the number of participants for CC c is binomial with parameters $\sum_p a^n_{p,c}$ and PR, due to Assumption 8.1.1. Each client has a probability of PR that he will participate. In total $\sum_p a^n_{p,c}$ clients are invited to CC c. Therefore the number of participants is the result of $\sum_p a^n_{p,c}$ random drawings with a probability of success of PR. This gives us the following probability distribution for the number of participants when current state and action are known.

$$\mathbb{P}[B_{c,n} = b_{c,n} \mid \sigma_n, a_n] = \binom{\sum_p a^n_{p,c}}{b_{c,n}} \cdot PR^{b_{c,n}} \cdot (1 - PR)^{\sum_p a^n_{p,c} - b_{c,n}}$$

The probability distribution for the number of responses for CC c is binomial with parameters $O_{c,n}$ and q. The probability that a client responses within the coming week is denoted with q as given in (8.3) in Section 8.1. The number of responses is then the result of $O_{c,n}$ (number of outstanding invitations) random drawings with a success probability of q. Due to Assumption 8.1.3 only clients with an outstanding invitation will be able to respond. This gives us the following probability distribution for the number of responses when current state and action are known. Here we use Assumption 8.1.1.

$$\mathbb{P}[R_{c,n} = r_{c,n} \mid \sigma_n, a_n] = \binom{O_{c,n}}{r_{c,n}} \cdot q^{r_{c,n}} \cdot (1 - q)^{O_{c,n} - r_{c,n}}$$

When we know the number of responses, we can also determine the probability distribution for the number of positive results for CC c. Each of the responses has a probability of RR to have a positive result. Therefore, the number of positive results for CC c in week n is binomially distributed with parameters $r_{c,n}$ and RR, due to Assumption 8.1.1.

$$\mathbb{P}[Y_{c,n} = y_{c,n} \mid R_{c,n} = r_{c,n}, \sigma_n, a_n] = \binom{r_{c,n}}{y_{c,n}} \cdot RR^{y_{c,n}} \cdot (1 - RR)^{r_{c,n} - y_{c,n}}$$

Objective function

The objective function minimizes the direct costs and expected future costs. The objective value at the end of the horizon is known, because this are the final direct costs. Calculating the objective function is done with the following recursion:

$$\begin{aligned} f_n(\sigma_n) &= \min_{a_n \in A_n} \left\{ k_n(\sigma_n, a_n) + \sum_{(h,i,j,l) \in S_{n+1}} \mathbb{P}_n\left[(h,i,j,l) \mid \sigma_n, a_n\right] \cdot f_{n+1}(h,i,j,l) \right\} \quad \forall n < T \\ f_T(\sigma_T) &= k_T(\sigma_T, a_T) \end{aligned}$$

We have defined the state space, action space, direct costs, transition probabilities and the objective function. Therefore, we have the complete formulation of the general SDP for determining the moment of inviting clients from postcode areas and linking them to CCs. Figure 8.4 gives an overview of the process. At a decision epochs n we evaluate a state (O_n, W_n, CV_n, PV_n) in the state space. We take an action a_n and this gives us the number of participants B_n, the number of responses R_n and the number of positive results Y_n. Based on these outputs and the input of available intake slots in the next week I_{n+1}, the next state is known. At the end of the year, $n = 53$ no decision is made.

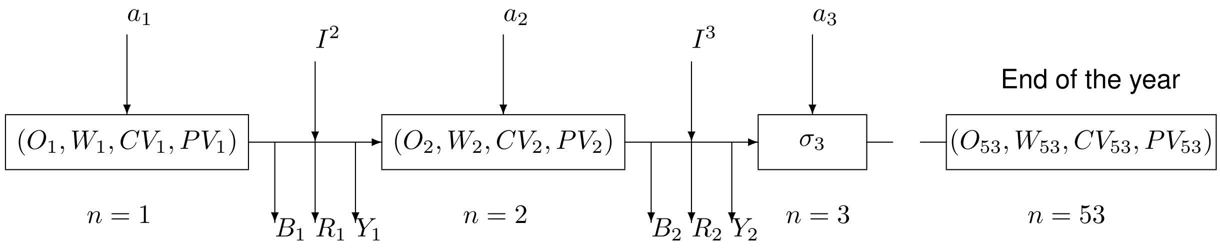


Figure 8.4: Time line of the general Stochastic Dynamic Program.

In order to solve this general SDP model, a backward induction algorithm (Algorithm 2) can be used. However, in practice this is not possible due to the curse of dimensionality. In the following section we explain why we cannot find a solution to the general SDP model. We also describe how we adapt the model such that we can find an invitation strategy for the colon cancer screening in which invitations are send at the right moment in time such that clients can have their intake appointment at the desired CC and week.

```
Set $n = T$ ;
Set $f_T(\sigma_T) = k_T(\sigma_T), \forall \sigma_T \in S_T$;
for $n > 1$ do
    $n := n - 1$;
    for $\sigma \in S_n$ do
        $f_n(\sigma) = \min_{a \in A_n} \{k_n(\sigma) + \sum_{(h,i,j,l) \in S_{n+1}} \mathbb{P}_n[(h,i,j,l) \mid \sigma, a] f_{n+1}(h,i,j,l)\}$;
        $A^*_{\sigma,n} = \text{argmin}_{a \in A_n} \{k_n(\sigma) + \sum_{(h,i,j,l) \in S_{n+1}} \mathbb{P}_n[(h,i,j,l) \mid \sigma, a] f_{n+1}(h,i,j,l)\}$;
    end
end
```

Algorithm 2: The backward induction algorithm that can be used to solve the general SDP.

8.3 Numerical tractability and decomposition

The general SDP model of the previous section is detailed and can handle communicating CCs. We can decide at all decision epochs which postcode areas we link to which CCs. However, these properties come with a disadvantage, namely the curse of dimensionality. The state space and action space of the general SDP are very large, which implies that it will not be possible to find an exact solution to the SDP. Computing the values of the objective function $f_n(\sigma)$ for all decision epochs n and all states σ in the state space using the backward induction algorithm 2 takes a lot of time when action and state space are large. When the state space has size $|\mathcal{S}|$, the action space has size $|\mathcal{A}|$ and we have T decision epochs, the number of computations in the backward induction algorithm is of order $2 \cdot T \cdot \left(|\mathcal{A}||\mathcal{S}|^2 + |\mathcal{A}||\mathcal{S}|\right)$. This number of computations becomes to large to handle when $|\mathcal{A}|$ and $|\mathcal{S}|$ are as large as they are in the general SDP.

To give an idea of the size of the state and action space of the general SDP we look at the case where we start with zero outstanding invitations of previous year as well as zero positive results of previous year, i.e. $W_c^0 = O_c^0 = 0$. When these numbers become larger the state space will also become larger. Section 8.3.1 calculates the number of states in the state space ($|\mathcal{S}|$) of the general SDP and section 8.3.2 calculates the number of actions ($|\mathcal{A}|$) in the general SDP. To overcome these dimensionality issues we propose a decomposition of the general SDP into an SDP that evaluates the CCs individually. Section 8.3.3 explains this decomposition.

8.3.1 State space

To calculate the number of states in the state space (8.4) we first look at a state where the total number of sent invitations up-till then is equal to x. These x invitations are send to P different postcode areas, so we distribute x invitations into P distinct groups. According to Grimaldi [2003] the number of possible ways to do this is denoted below, where $\binom{n}{k}$ denotes the binomial coefficient.

$$\text{\# possibilities to distribute } x \text{ over } P \text{ groups} = \binom{x+P-1}{P-1}$$

This number is the number of possible states in the part for PV_n. On the other hand, these x invitations also need to be linked to CCs and therefore are distributed over C groups, this gives us the number of possible states in the part for CV_n:

$$\text{\# possibilities to distribute } x \text{ over } C \text{ groups} = \binom{x+C-1}{C-1}$$

Each state in the state space has a value for $CV_{c,n}$, these send invitations corresponding to CCs can be either outstanding, positive result received or disappeared from the system. This restriction is given by constraint $O_{c,n} + W_{c,n} \leq CV_{c,n}$ in the state space. The number of possibilities for state parts $O_{c,n}$ and $W_{c,n}$ is then the number of possibilities to distribute the invitations corresponding to CC c, $CV_{c,n}$, over 3 distinct groups. Which should be done for all CCs.

$$\text{\# possibilities to distribute } CV_{c,n} \text{ over } 3 \text{ groups in one single CC } c = \binom{CV_{c,n}+2}{2}$$

By multiplying all these different number of possibilities for the different parts of the states, we have the total number of states when x invitation are send up-till now.

$$\text{\# states when } x \text{ invitations are send in total} = \binom{x+P-1}{P-1} \cdot \binom{x+C-1}{C-1} \cdot \prod_c \binom{CV_{c,n}+2}{2}$$

x is the number of invitations that is send, this can vary between 0 and the total number of clients that need to be invited, which is $X = \sum_p N_p$. Therefore, the total number of states in the state space of the general SDP is:

$$|\mathcal{S}| = \text{\# states in the state space} = \sum_{x=0}^{X} \binom{x+P-1}{P-1} \cdot \binom{x+C-1}{C-1} \cdot \prod_c \binom{CV_{c,n}+2}{2} \tag{8.8}$$

This number becomes very large. We have $P = 790$ and $C = 22$, if for example x has value 22 and each CC has one of these invitations ($CV_{c,n} = 1$) we already have $1.7 \cdot 10^{65}$ states. This number is only the value of the part of the sum that corresponds with $x = 22$, so the actual state space is even larger. Also the total number of clients is much larger than 22, namely $X = \sum_p N_p \approx 400,000$ clients should be invited. When we use these numbers for X, P and C in (8.8), it is clear that the number of states in the state space of the general SDP is far to large.

8.3.2 Action space

The action space, (8.5), is largest at the first decision epoch ($n = 1$) because all clients can be invited at that moment. During the year invitations will be send, which implies that the number of invitations that can be send in a week n will decrease over the year. Therefore, the size of the action space decreases over the year. At the beginning of the year, each postcode area has N_p clients that should get an invitation. These invitation should be linked to one of the C CCs. For a single postcode area the number of actions that can be taken consists of all possible divisions of N_p invitations over $C+1$ groups, where the $+1$ stand for not inviting. According to Grimaldi [2003] this number equals $\binom{N_p + C}{C}$. By taking the product of all these possibilities over all different postcode areas we get the total number of possible actions in the action space.

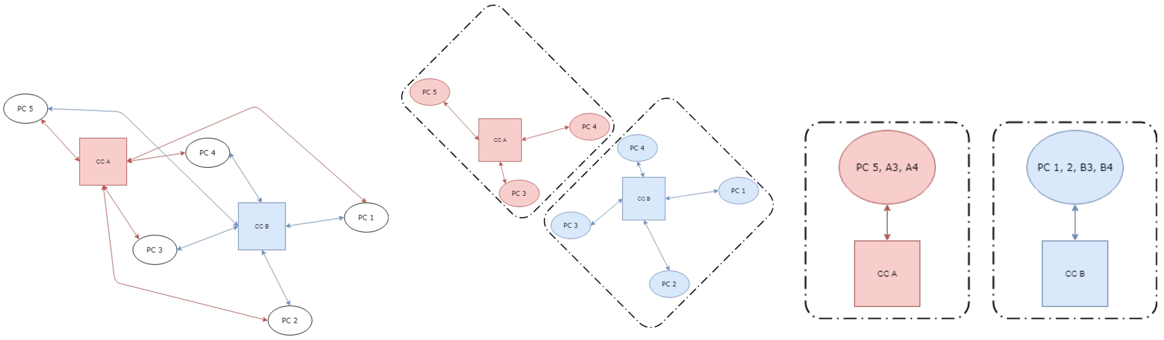
$$|\mathcal{A}| = \text{\# actions} = \prod_p \binom{N_p + C}{C} \tag{8.9}$$

As we have 790 postcode areas and 22 CCs, the number of possible actions will become very large. It grows with power 790 as the number of inhabitants increases.

8.3.3 Decomposition idea

In the previous two sections we see that both state and action space are very large in the general SDP. The size of the state space and action space are determined by equations (8.8) and (8.9) respectively. The computational time of solving the general SDP with the backward induction algorithm 2 will be enormous, also memory issues will make sure that we cannot solve the SDP in this general setting. We therefore propose a method to decompose the general SDP into smaller SDPs by using the deterministic results of the MILP. We split the problem of inviting clients into C individual problems, each corresponding to a single CC.

To illustrate the idea of this decomposition we look at Figure 8.5 where we consider two CCs and five postcode areas. In the general SDP (Figure 8.5a) clients from all postcode areas can be invited to all CCs. We therefore have actions over all possible postcode area and CC combinations, indicated with the arrows. However, in the MILP we already determined how many clients from each postcode area are linked to which CC, namely the adherence numbers. In the example of Figure 8.5 the adherence will indicate that 'PC 5' is linked for 100% to CC 'A', 'PC 1' and 'PC 2' for 100% to CC 'B' and 'PC 3' and 'PC 4' will both be linked for 50% to CC 'A' and 50% to CC 'B'. By applying this adherence we arrive at Figure 8.5b, where both 'PC 3' and 'PC 4' are split into two parts (each with 50% of the clients) and each postcode area is linked to the CC according to the adherence. The actions will now consists only of inviting the clients from the postcode areas to the CC that they are already linked to. However, we want to use the SDP model to determine when we should send invitations in such a way that the intake appointments can take place in the desired CC and week. For this purpose it is not important from which postcode area the client is actually coming from, because we only need to know that the client is linked to the CC that we look at. In other words, we can aggregate all clients from postcode areas which are linked to one single CC into one big group. This is done in Figure 8.5c, where we have for CC 'A' one group of clients from which we invite and for CC 'B' another group. These two single CC SDPs are independent of each other and can therefore be solved consecutively. The SDP for these individual CCs is given in Section 8.4. The goal of such a single CC SDP is to determine when we need to send how many invitations and we do not pay attention to which client the invitations is send specifically.



(a) The general SDP (b) An intermediate step (c) The single CC SDPs.

Figure 8.5: A visualization of the decomposition idea with two CCs.

8.4 SDP single CC

The main difference of this single CC SDP with the general SDP model of Section 8.2 is that we only have areas that correspond to a single CC and that the different CCs do not interact with each other. Therefore both SDPs have many similarities. The notation is introduced for all CCs together, however they do not influence each other any more. Therefore we can also look at one single CC, i.e. $C = 1$. The different elements of the single CC SDP are decision epochs, state space, action space, direct costs, transition probabilities and objective function and are described in the remainder of this section. Finally we give the numerical tractability of this single CC SDP in Section 8.4.1.

Decision Epochs

The decision epochs are the moments at which the system is evaluated and an action should be taken. As in the general SDP we have:

$$\mathcal{N} = \{1, 2, \ldots, T\}, \quad n \in \mathcal{N}$$

where at $n = T = 53$ no decision is made. We have the following set of CCs:

$$\mathcal{C} = \{1, 2, \ldots, C\}, \quad c \in \mathcal{C}$$

The clients that need to be invited are divided over C areas, based on the adherence output of the MILP. This output is $AH_{p,c}$ which gives for all postcode areas p which fraction of the clients from postcode area p go to CC c. By using this adherence the clients are reassigned to an area that corresponds to CC c. Let H_c be the number of clients in such an area corresponding to CC c and N_p is the original number of clients in postcode area p. Then the new number of inhabitants for each area is:

$$H_c = \sum_p AH_{p,c} \cdot N_p \quad \text{where} \quad H_c \in \mathbb{Z}$$

State space

The entire state space $\mathcal{S}$ consists of all possible state spaces at decision epochs n.

$$\mathcal{S} = \{S_n\}_{n \in \mathcal{N}}$$

The state space at decision epoch n is denoted with S_n, a possible state in this state space is denoted with σ_n. Such a state consists of three quantities which together describe the state of the system.

- $O_n = (O_{1,n}, O_{2,n}, \ldots, O_{C,n})$, where $O_{c,n}$ is the number of outstanding invitations at the beginning of week n that corresponds to CC c. These invitations are already sent to clients that live in an area that corresponds to CC c, but these clients did not yet respond. The number of clients that will not participate is already excluded in this number, so only invitations that will be used are counted.
- $W_n = (W_{1,n}, W_{2,n}, \ldots, W_{C,n})$, where $W_{c,n}$ is the number of positive test results at the beginning of week n that correspond to CC c. This number of clients need to get an intake appointment in CC c in the next week $(n + 1)$.
- $V_n = (V_{1,n}, V_{2,n}, \ldots, V_{C,n})$, where $V_{c,n}$ is the total number of sent invitations until week n to clients that live in the area that corresponds to CC c. This includes both the responding and the non-responding clients and does not include the invitations send in week n itself.

We denote with parameter O_c^0 the number of outstanding invitations corresponding to CC c, which are still open from previous invitation year. W_c^0 is the number of positive results in CC c from previous year that still need an intake appointment in the coming year. These three quantities for all CCs together form the state space at decision epoch n.

$$\begin{aligned} S_n = \{\sigma_n = (O_n, W_n, V_n) \quad | \quad & 0 \leq O_{c,n} \leq O_c^0 + V_{c,n}, \\ & 0 \leq W_{c,n} \leq W_c^0 + O_c^0 + V_{c,n}, \\ & 0 \leq V_{c,n-1} \leq V_{c,n} \leq H_c, \\ & O_{c,n} + W_{c,n} \leq W_c^0 + O_c^0 + V_{c,n} \qquad \forall c\} \end{aligned} \tag{8.10}$$

It is not possible to have more outstanding invitations than $O_c^0 + V_{c,n}$ in a week, because this is the number of invitations that is send so far this year, increased with the outstanding invitations from previous year. This number, $O_c^0 + V_{c,n}$, increased with the outstanding positive results from previous year (W_c^0) is the theoretical upper bound for the number of positive results in a week. The number of send invitations that corresponds to a CC is non-decreasing over the weeks and bounded from above by the total number of clients that should get an invitation this year (H_c). All invitations that are send up-till week n in CC c (including what is left over from previous year), can have status outstanding or positive result in week n, or they already disappeared from the system by being a non-participant, a negative result or an already scheduled intake appointment. Therefore, we have the bound $O_{c,n} + W_{c,n} \leq W_c^0 + O_c^0 + V_{c,n}$.

At the start of the horizon ($n = 1$) we have that $V_{c,1} = 0 \quad \forall c$, because no invitations are send yet. $W_{c,1}$ will have value W_c^0, because no other test results are known yet. The quantity $O_{c,n}$ will have value O_c^0 at the beginning of the year. These observations give us the state space at decision epoch one, which consists of only one state.

$$S_1 = \{\sigma_1 = (O_1, W_1, V_1) \quad | \quad O_{c,1} = O_c^0, W_{c,1} = W_c^0, V_{c,1} = 0 \qquad \forall c\}$$

Action space

The action space consists of all possible actions that are allowed in a given state at a given decision epoch.

$$\begin{aligned} \mathcal{A} &= \{A_n\}_{n \in \mathcal{N}} \\ A_n(\sigma_n) &= \{a_n = (a_{1,n}, a_{2,n}, \dots, a_{C,n}) \mid 0 \leq a_{c,n} \leq H_c - V_{c,n} \quad \forall c\} \end{aligned} \tag{8.11}$$

The actions $a_{c,n}$ are the number of new send invitations at the beginning of week n to clients in the area corresponding to CC c. The action space is bounded by 0 from below, because you cannot send a negative number of invitations. It is not possible to send more invitations than there are clients left in area corresponding to c at the beginning of the week. Therefore the upper bound for the action space is set to $H_c - V_{c,n}$, the number of clients minus the number of sent invitations.

Direct costs

The direct costs in decision epoch n when the system is in state σ_n are denoted with $k_n(\sigma_n)$. These costs are comparable with the costs of the general SDP, (8.6). However, in this single CC SDP we do not have costs that correspond to an action.

$$\begin{aligned} k_n(\sigma_n) &= \sum_c \beta_w \cdot (W_{c,n} - I_c^{n+1})^+ + \sum_c \beta_i \cdot \left(W_{c,n} - \sum_{n'=n+1}^{n+3} I_c^{n'} \right)^+ + \sum_c \beta_g \cdot |W_{c,n} - G_c^{n+1}| \\ &\text{for} \quad n = 1, \dots, T-1 \end{aligned}$$

Where,

$$\begin{aligned} G_c^n &= \sum_p AH_{p,c} \cdot x_{p,c}^n \cdot \bar{p} \\ &= \text{the number of planned intake appointments in week } n \text{ in CC } c \text{ according to the MILP} \end{aligned}$$

At the end of the horizon ($n = T$) the costs are as follows, similar to (8.7).

$$\begin{aligned} k_T(\sigma_T) &= \sum_c \beta_w \cdot \left(W_{c,T} - I_c^{T+1}\right)^+ + \sum_c \beta_i \cdot \left(W_{c,T} - \sum_{n'=T+1}^{T+3} I_c^{n'} \right)^+ \sum_c \beta_g \cdot |W_{c,T} - G_c^{T+1}| \\ &\quad + \sum_c \beta_b \cdot (O_{c,T} - \hat{O}_c)^+ + \sum_c \beta_v \cdot (H_c - V_{c,T}) \end{aligned}$$

The scaling factors β that are used in the direct costs of this single CC SDP are given in Table 8.2. The same reasoning as for the general SDP is used, only here the costs that correspond with taken actions are not present. The highest costs are given to the most important quantity.

Transition probabilities

Given the current state of the system and the chosen action, the transition probabilities are the probabilities that the system is in state (h, i, j) next week.

$$\begin{aligned} \mathbb{P}_n[(h, i, j) \mid \sigma_n, a_n] &= \mathbb{P}_n[(h_c, i_c, j_c) \quad \forall c \mid \sigma_n, a_n] \\ &= \prod_c \mathbb{P}_n[(h_c, i_c, j_c) \mid \sigma_n, a_n] \end{aligned}$$

Table 8.2: Determination of the scaling factors β in the direct costs function.

β	Description	Importance factor	Size class	Scaling factor
β_i	waiting week above three weeks	100	1	100
β_b	border outstanding invitations end of year	80	1	80
β_v	rest group end of year	50	1	50
β_w	waiting week	30	1	30
β_g	reach MILP goal per week	4	1	4

State (h_c, i_c, j_c) for CC c can be reached from σ_n by different number of responses in week n. The stochastic variable $R_{c,n}$ represents the number of responses in week n for CC c. Also the number of positive results and number of participations is needed to determine in which state the system will be in the next week $n+1$. Let $Y_{c,n}$ be the random variable that represents the number of positive results for CC c in week n and let $B_{c,n}$ be the number of participants for CC c in week n. In order to calculate the transition probabilities we condition on the values that $R_{c,n}$, $Y_{c,n}$ and $B_{c,n}$ can take. These three random variables are all independent of each other, due to Assumption 8.2.1.

$$\begin{aligned}\mathbb{P}_n[(h_c, i_c, j_c) \mid \sigma_n, a_n] = \sum_{r_{c,n}=0}^{O_{c,n}} \sum_{y_{c,n}=0}^{r_{c,n}} \sum_{b_{c,n}=0}^{a_{c,n}} & \mathbb{P}_n[(h_c, i_c, j_c) \mid R_{c,n} = r_{c,n}, Y_{c,n} = y_{c,n}, B_{c,n} = b_{c,n}, \sigma_n, a_n] \\ & \cdot \mathbb{P}[B_{c,n} = b_{c,n} \mid \sigma_n, a_n] \\ & \cdot \mathbb{P}[Y_{c,n} = y_{c,n} \mid R_{c,n} = r_{c,n}, \sigma_n, a_n] \\ & \cdot \mathbb{P}[R_{c,n} = r_{c,n} \mid \sigma_n, a_n]\end{aligned}$$

All the above mentioned probabilities are known. The next state of the system is certain, once the value of the three random variables are known. Namely:

- The number of outstanding invitations increases with the new invitations that correspond to participants, $b_{c,n}$. On top of that the number of outstanding invitations decreases with the number of responses $r_{c,n}$. This gives the closed form $h_c = O_{c,n} + b_{c,n} - r_{c,n}$.
- The number of positive results at the beginning of the next week is given as shown in the equality with $i_c = (W_{c,n} - I_c^{n+1})^+ + y_{c,n}$. The first term indicates how many positive results are left over from previous week, this is $(W_{c,n} - I_c^{n+1})$, as long as this number is positive. The second term $(Y_{c,n})$ indicates the new positive results that arrive during week n for CC c.
- The total number of sent invitations in the next week is of course the number of sent invitations in the previous week (n) plus the number of new sent invitations $(a_{c,n})$, which gives $j_c = V_{c,n} + a_{c,n}$.

Combining these three closed forms, we get the following probability distribution.

$$\mathbb{P}_n[(h_c, i_c, j_c) \mid R_{c,n} = r_{c,n}, Y_{c,n} = y_{c,n}, B_{c,n} = b_{c,n}, \sigma_n, a_n] \quad = \quad \begin{cases} 1 & \text{if} \quad h_c = O_{c,n} + b_{c,n} - r_{c,n} \\ & \text{and} \quad i_c = (W_{c,n} - I_c^{n+1})^+ + y_{c,n} \\ & \text{and} \quad j_c = V_{c,n} + a_{c,n} \\ 0 & \text{otherwise} \end{cases}$$

The probability distribution for the number of participants is binomial with parameters $a_{c,n}$ and PR. Each client has a probability of PR that he will participate. In total $a_{c,n}$ clients are invited. Therefore the number of participants is the result of $a_{c,n}$ random drawings with a probability of success of PR. This gives us the following probability distribution for the number of participants when current state and action are known.

$$\mathbb{P}[B_{c,n} = b_{c,n} \mid \sigma_n, a_n] = \binom{a_{c,n}}{b_{c,n}} \cdot PR^{b_{c,n}} \cdot (1 - PR)^{a_{c,n} - b_{c,n}}$$

The probability distributions for the number of responses and the number of positive results are both binomial and the same as for the general SDP.

$$\begin{aligned}\mathbb{P}[R_{c,n} = r_{c,n} \mid \sigma_n, a_n] \quad &= \quad \binom{O_{c,n}}{r_{c,n}} \cdot q^{r_{c,n}} \cdot (1-q)^{O_{c,n} - r_{c,n}} \\ \mathbb{P}[Y_{c,n} = y_{c,n} \mid R_{c,n} = r_{c,n}, \sigma_n, a_n] \quad &= \quad \binom{r_{c,n}}{y_{c,n}} \cdot RR^{y_{c,n}} \cdot (1 - RR)^{r_{c,n} - y_{c,n}}\end{aligned}$$

Objective function

The objective function minimizes the direct costs and expected future costs. The objective value at the end of the horizon is known, because this are the final direct costs. Calculating the objective function is done with the following recursion:

$$\begin{aligned} f_n(\sigma_n) &= \min_{a_n \in A_n} \left\{ k_n(\sigma_n) + \sum_{(h,i,j) \in S_{n+1}} \mathbb{P}_n \left[(h,i,j) \mid \sigma_n, a_n \right] \cdot f_{n+1}(h,i,j) \right\} \quad \forall n < T \\ f_T(\sigma_T) &= k_T(\sigma_T) \end{aligned}$$

Combining the state space, action space, direct costs, transition probabilities and the objective function, we have the total formulation of the SDP for individual CCs. Figure 8.6 gives an overview of the process. At a decision epochs n we evaluate a state (O_n, W_n, V_n) in the state space, which represents the number of outstanding invitations, the number of positive results and the number of sent invitations for this CC at the beginning of the week. We take an action a_n (sending new invitations) and this gives us the number of participants B_n, the number of responses R_n and the number of positive results Y_n that arrive during week n. Based on these outputs and the input of available intake slots in the next week I_{n+1}, the next state is known. At the end of the year, $n = 53$ no decision is made.

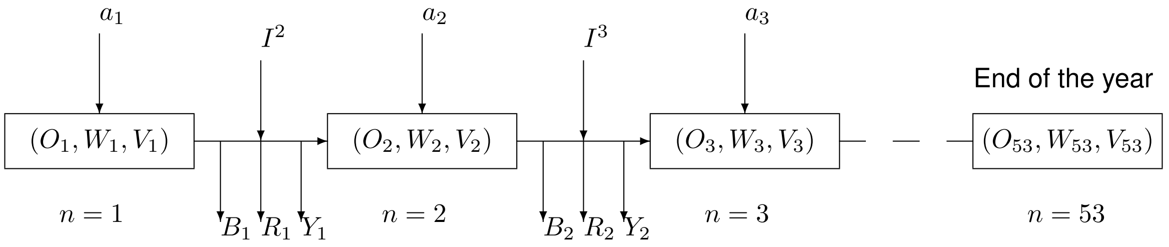


Figure 8.6: Time line of the single CC Stochastic Dynamic Program.

8.4.1 Numerical tractability single CC SDP

The total formulation of the SDP for individual CCs is given in the previous section. This SDP has a smaller state and action space than the general SDP of Section 8.2.

We first look at the size of the state space, (8.10), for one single CC. Each state has 3 quantities which can each take an individual value. We look at an arbitrary decision epoch and a specific CC c with $\sigma = (O, W, V)$. To determine the number of states in the state space we need to calculate how many possibilities we have for the values for O, W and V. First we condition on the value x that V can take, namely 0 up to H_c. In the case that x invitations are send up-till now we can determine which values O and W can take. According to the state space constraint $O_{c,n} + W_{c,n} \leq W_c^0 + O_c^0 + V_{c,n}$ we need to divide $W_c^0 + O_c^0 + x$ over three different groups, namely the number of invitations that is outstanding (O), the number of invitations that turned into a positive result (W) and the invitations that already disappeared from the system. According to Grimaldi [2003] this can be done in $\binom{x + O_c^0 + W_c^0 + 2}{2}$ different ways. However in these combinations, some combinations are not allowed in the state space due to the constraint $O_{c,n} \leq O_c^0 + V_{c,n}$. The combinations were $O > O_c^0 + x$ need to be subtracted. The number of times that this occurs if $O = i$ is equal to the different numbers of values that W can take in that case. These values are $0, 1, \ldots, W_c^0 + O_c^0 + x - i$, so $W_c^0 + O_c^0 + x - i + 1$ combinations need to be subtracted when V has value x and O has value i. With these observations, the number of states in the state space is given below.

$$|\mathcal{S}| = \#\text{ states} = \sum_{x=0}^{H_c} \binom{x + O_c^0 + W_c^0 + 2}{2} - \sum_{x=0}^{H_c} \sum_{i=O_c^0+x+1}^{O_c^0+W_c^0+x} W_c^0 + O_c^0 + x - i + 1$$

By using known expressions for finite sums and some algebraic manipulation we arrive at the following formula for the number of states in the state space when one single CC is evaluated.

$$\begin{aligned} |\mathcal{S}| = \#\text{ states} = &\frac{1}{6} \cdot (H_c + 1) \cdot \left[H_c^2 + (3O_c^0 + 3W_c^0 + 5) \cdot H_c + 3 \cdot (O_c^0 + W_c^0 + 2) \cdot (O_c^0 + W_c^0 + 1) \right] \\ &- \frac{1}{2} \cdot (H_c + 1) \cdot W_c^0 \cdot (W_c^0 + 1) \end{aligned} \tag{8.12}$$

For example, we have a CC with $H_c = 1$ inhabitant and no outstanding invitations or positive results from previous year ($O_c^0 = W_c^0 = 0$). In that case the number of states in the state space is 4, according to equation (8.12). Doing this successively for all 22 CCs we solve 22 times an SDP with only 4 states, i.e. comparable with 88 states. This is much smaller than the $1.7 \cdot 10^{65}$ in the general SDP where the 22 clients could be divided over all 790 postcode areas and 22 CCs, calculated with equation (8.8). This illustrates the great advantage of decomposing the general SDP into C single CC SDPs. Even when we consider a single CC with for example $H_c = 22$, $O_c^0 = 2$ and $W_c^0 = 2$ the state space is of size 3565, which is acceptable.

On top of that the action space (8.11) for one single CC is smaller after the decomposition. The action space of a single CC SDP, (8.11), consists maximally of all numbers between 0 and H_c. In other words we have $H_c + 1$ possible actions at the first decision epoch.

$$|\mathcal{A}| = \text{\# actions} = H_c + 1 \tag{8.13}$$

The number of actions in the action space decreases over the year, because during the year we already sent some invitations. The size of the action space in the single CC SDP, (8.13) grows linearly with the number of inhabitants. Whereas in the general SDP the size of the action space grows with power 790, see equation (8.9).

Both the action space and state space for this single CC SDP are significantly smaller than in the general SDP. Therefore we can now solve this SDP with the help of a backward induction algorithm, as shown in Algorithm 3. The computation time of this algorithm is of order $2 \cdot T \cdot \left(|\mathcal{A}||\mathcal{S}|^2 + |\mathcal{A}||\mathcal{S}|\right)$, where $|\mathcal{S}|$ and $|\mathcal{A}|$ are defined by (8.12) and (8.13) respectively. We programmed this algorithm in Python for one single CC. To solve the single CC SDP for all CCs we can use the backward induction algorithm iteratively for each CC individually.

```
Set $n = T$ ;
Set $f_T(\sigma_T) = k_T(\sigma_T), \forall \sigma_T \in S_T$;
for $n > 1$ do
    $n := n - 1$;
    for $\sigma \in S_n$ do
        $f_n(\sigma) = \min_{a \in A_n} \{k_n(\sigma) + \sum_{(h,i,j) \in S_{n+1}} \mathbb{P}_n[(h,i,j) \mid \sigma, a] f_{n+1}(h,i,j)\}$;
        $A^*_{\sigma,n} = \text{argmin}_{a \in A_n} \{k_n(\sigma) + \sum_{(h,i,j) \in S_{n+1}} \mathbb{P}_n[(h,i,j) \mid \sigma, a] f_{n+1}(h,i,j)\}$;
    end
end
```

Algorithm 3: The backward induction algorithm that is used to solve the single CC SDP.

8.5 Decomposition approximation errors

In the previous sections we described first a general SDP for inviting clients to the colon cancer screening and then decomposed it into independent SDPs for single CC. With this decomposition the size of the state and action space decreased and we are able to solve the SDP for a single CCs. However, due to this decomposition we make a possible approximation error, because we do not take all solutions of the general SDP into account any more in the single CC SDPs. We did not succeed to proof the magnitude of this approximation error, but we can say something about when and where this error occurs. We also did some research in literature to different decomposition methods for SDPs. Section 8.5.1 gives an overview of this literature and explains the similarities and differences with the decomposition in our research. Section 8.5.2 gives a conceptual idea of which approximation errors we make in our decomposition method for the specific colon cancer invitation strategy SDP model.

8.5.1 Literature

A decomposition method is often very specific for the research that is executed. Especially when the research develops an SDP for a specific process in practice, the decomposition method is specialised for only that problem. This is also the case for this research at BVO Oost. However, we can look at the different types of decomposition methods for SDPs. An overview of the different decomposition methods is given in Daoui et al. [2010]. The main reason for decomposing an SDP is to overcome the curse of dimensionality in the state space. Daoui et al. [2010] gives multiple methods to overcome these dimensional issues. The first one is to reduce the state space of the SDP by transforming the model to one with fewer states. However, this is not always possible because the states describe the state of the system.

Another method mentioned by Daoui et al. [2010] is aggregation / disaggregation of the state space, which is useful when the system exists of interacting coupled subsystems. This method is often iterative where sub-models converge to a solution. An example of this aggregation decomposition method is the model used in Archibald [2004]. They model a multi-reservoir control problem where each reservoir has a water level that should be managed. The reservoirs are connected with each other but can be seen as independent sub-systems. Each sub-model consists of a detailed model of one reservoir and an approximation of the other reservoirs. They then solve a number of sub-models which is equal to the number of reservoirs. This example is comparable with our situation at BVO Oost, we also have different models (CCs) that interact with each other. However we cannot split the system in multiple sub-models in a similar way, because we also have all the postcode areas that should be evaluated. We need to decouple all postcode areas and CCs and cannot contain information from other parts in each sub-model.

Sometimes the state space of a system is very large, but only a small set of the states have a large part of the probability mass. In other words many of the states are very unlikely to be visited. Daoui et al. [2010] mention in this case the decomposition method of truncation which only evaluates the states that are likely to be visited. However, with this truncation significant errors might occur because a system mostly reacts differently in rare states.

Next, Daoui et al. [2010] mention methods for decomposing in the case of infinite horizon Markov Decision Processes. MDPs with infinite horizon can be written in a linear programming formulation. Such a large linear program can be divided into several correlated smaller linear programs by using the Dantzig-Wolfe decomposition, [Dantzig and Wolfe, 1960]. Also Benders decomposition can be used to solve large linear programs. Fazel-Zarandi and Beck [2009] address a location-allocation problem where locations of facilities should be determined. On top of that customers and trucks need to be allocated to the facilities. They develop an integer linear programming model and use Benders decomposition to find an optimal solution. Our research uses an SDP with a finite horizon which cannot be rewritten into a linear programming formulation. Therefore we cannot use such Dantzig-Wolfe or Benders decomposition methods.

The Divide-and-Conquer method, mentioned by Daoui et al. [2010], consists of partitioning the state space of a system into different regions. Each region can be seen as a sub-model which can be solved independently of the other sub-models. When solutions to each sub-model are known the idea is to combine these solutions with a simple procedure to a solution for the entire system. This Divide-and-Conquer method is only efficient if (a) each sub-model is solvable (b) the number of sub-models is limited and (c) combining the solutions of the sub-models into a general solution is not to difficult. Parr [2013] use this method of decomposition in their research. They have weakly coupled Markov Decision Problems. Weakly coupled means in this case that the state spaces of the different MDP parts are only linked by a few transitions. If these few transitions are ignored the state space consists of multiple independent regions. Parr [2013] first proposes a partial decoupling method where solutions of the smaller pieces are found independently and then combined in a post processing step. They secondly propose a complete decoupling method where information of the smaller pieces is communicated between the different problems. Also Laroche et al. [2001] use this Divide-and-Conquer method, but then in the context of mobile robotics. A robot needs to reach a goal situated in an environment (office building). This environment is split into different parts, namely corridors and offices. Each corridor or offices is a region. They compute for each region a policy (solution) and then they approximate the state values and policies in the states that link two different regions.

At first sight this Divide-and-Conquer method is the same as our decomposition of the general SDP into SDPs for each single CC. Each SDP for a single CC is solvable and with 22 CC we have a limited number of smaller sub-models. Combining the individual solutions is very easy because the CCs do not have to do anything with each other after our decomposition. However, our decomposition is significantly different from the decompositions of Laroche et al. [2001] and Parr [2013]. When we look at the state space of our general model, we see that the different states are highly coupled and we cannot split the states into different regions. This is caused by the fact that in our system the processes in the different CCs occur parallel instead of consecutively. At a moment in time an action can result in no changes in CC 'A' whereas in CC 'B' many different options are available. These two CC do not have influence on each other but are saved in the same state. Our state space contains all information of both CCs and therefore we cannot see regions of states in the state space. The only way of separating the states is by fixing the adherence numbers and make sure that actions only have influence in one single CC and the states of the other CCs are not evaluated.

Until now we found a lot of decomposition methods for SDPs or MDPs. These decomposition methods looked like our decomposition idea, but also had a lot of differences. As we mentioned earlier a decomposition is very specific for each problem. In our research we actually prefix some of the actions in the SDP, namely we determine on beforehand which postcode areas are linked to which CCs. We therefore looked into literature about prefixing of actions. As we mentioned earlier Cappanera et al. [2017] uses robust optimisation to find an optimal planning in the home care sector. They prefix some of the decision variables by allowing a patient to have only one single care plan, which is the care plan determined earlier. This is comparable with our decomposition of using the MILP adherence to link clients in the SDP only to CCs that are used in the MILP. Such a method is based purely on the characteristics of the specific problem, as in Cappanera et al. [2017], or our research, and it does not give any guarantees in general that prefixing some actions or decision variables gives good results.

Concluding, we did find a lot about different methods of decomposition in MDPs and SDPs, but none of the found literature is similar to our decomposition idea. This makes proving that our decomposition methods works difficult. In the next section we give a global description of which approximation errors we make when applying our decomposition idea.

8.5.2 Conceptual idea of approximation errors

The main thing that we lose when we decompose the general SDP into several single CC SDPs, is that we cannot redistribute clients over the different CCs. The fixed adherence from the MILP assigns all clients to a unique CC and the clients will be invited in that specific CC. This assignment of clients to CCs is optimised but only in a deterministic case. It might happen that due to uncertainty one CC, say CC 'A', gets more clients who need an intake appointment than CC 'B'. In the general SDP it is possible to invite in that case more clients to CC 'B' such that the waiting time in CC 'A' does not become to large. In the single CC SDPs we cannot sent clients initially linked to CC 'A' during the year to CC 'B' because the two SDPs are not coupled. However, sending a client initially linked to CC 'A' now to CC 'B' can cause some extra travel time for that client. Therefore you might win something on waiting time by redistributing the clients, but the travel time might get worse. It is difficult to say in what amount this redistributing effects the waiting time and travel time and when redistributing is optimal in the total objective. This example was for 2 CCs, but a similar thing occurs with multiple CCs.

In the general SDP we evaluate all possible actions of linking clients to CCs at different decision epochs. The objective is then to minimize the sum over all objectives of all CC simultaneously. Whereas in the single CC SDPs we find the optimal invitation strategy for each CC individually (based on deterministic postcode area - CC linking). We therefore obtain the sum over of all minimal objectives per CC. This minimal sum and this sum of minimals does not have to have the same value and therefore the solutions to the general SDP and the single CC SDPs need not be the same. As explained above the decomposition of multiple single CC SDPs might give a worse solution because the option of redistributing clients over CCs is disappeared.
This description of how the approximation errors occur is only conceptual. In order to give a quantitative statement about the effect of the decomposition on our solutions, further research is needed.

Chapter 9

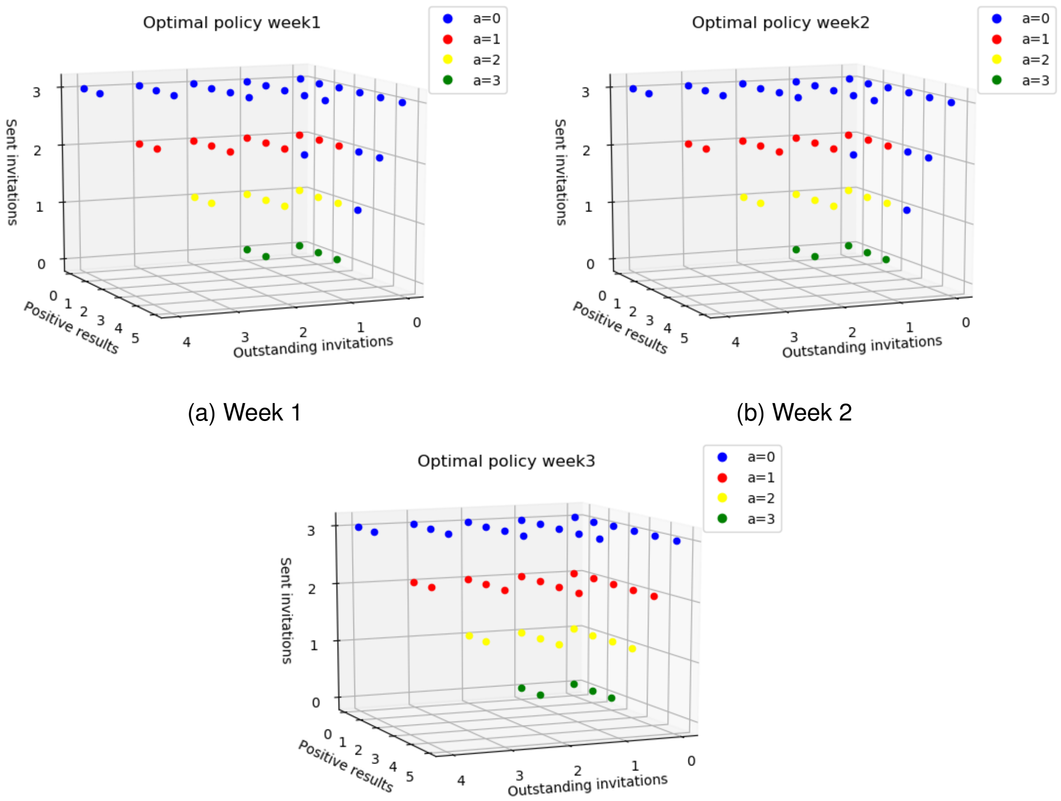
Results single CC SDP

We have programmed the backward induction algorithm for the single CC SDP of Section 8.4 in Python for a single CC, e.g. $C = 1$. This chapter contains the results, which tells us the moment of inviting clients to the screening process. We first look at some small instances which we can solve very fast, in Section 9.1. We use these small instances (with few clients) to check the correctness of the programmed algorithm. Larger instances for one single CC, which will tell us more about the invitation strategy in realistic practical situations with more clients, are still not solvable with our algorithm despite the decomposition. In Section 9.2 we explain why this is not possible and we give some ideas to get insight in more realistic instances. We look in Section 9.3 to instance with few clients but higher participation and referral rates. In this way less clients can be invited on the same number of intake slots. We simulate a more realistic situation and we can hopefully say something about the way an optimal invitation strategy will look like in practice.

9.1 Small instances results

The first small instance that we look at has 3 clients which should be invited. The time horizon is 3 weeks, so $T = 4$. At the beginning of the time horizon we have one outstanding invitation and one positive result from previous year. The CC has each week one available intake slot and the planned intake goal per week is set to $G^n = 1$. At the end of the year the border for number of outstanding invitations equals $\hat{O} = 2$. Figure 9.1 gives a 3D visualisation of the optimal policy that arises when solving the instance with backward induction. The 3D grid represents the state space, with on the axis the number of outstanding invitations, the number of positive results and the number of sent invitations. The coloured dots indicate which action is taken in the state that corresponds to the place of the dot. For example in week 1 (Figure 9.1a), we have a green dot in state $(O, W, V) = (0, 0, 0)$, which means that the optimal action is to send 3 new invitations when we are in that state. In state $(O, W, V) = (1, 3, 2)$ the dot is blue, so the optimal action is to send 0 new invitations.

We see that the last decision (at $n = 3$) always consists of inviting all clients that are left, i.e. $H - V_3$. Even when the number of outstanding invitations is large, we still invite all clients at the end, because the number of outstanding invitations will decrease in the last week, by responding clients. The transition probabilities from a state with a high number of outstanding invitations in week 3 to a high number of outstanding invitations in week 4 are small enough to not have the high costs for violating the border of outstanding invitations at the end of the horizon. Therefore it is more optimal to minimize the rest group. Mostly, this policy also is optimal in weeks $n = 1$ and $n = 2$. This is expected because we have enough intake slots for all clients. Normally you can invite 30 clients on one intake slot. We now invite 3 clients in a week and probabilistic the clients with positive results (in expectation 0) will fit in one intake slot that is available in the first week. In other words, due to the large available intake capacity, we can invite all clients without creating waiting time for the clients with positive results. Also, the number of outstanding invitations during the time horizon is not penalized. In the states where already 3 or more positive results are present, the optimal action in weeks 1 and 2 deviates from the optimal action in week 3. When in weeks 1 and 2 we have 3 or more positive results, the optimal action equals 0 new invitations. This is because the system is already full and we do not want that new invited clients will join the waiting list for an intake appointment in the future. It is therefore more optimal to wait an extra week with inviting new clients, so that we know for sure that these new clients do not have to wait long for an intake appointment. During the current week the number of positive results will decrease because we can plan intake appointments for these waiting clients.



(a) Week 1

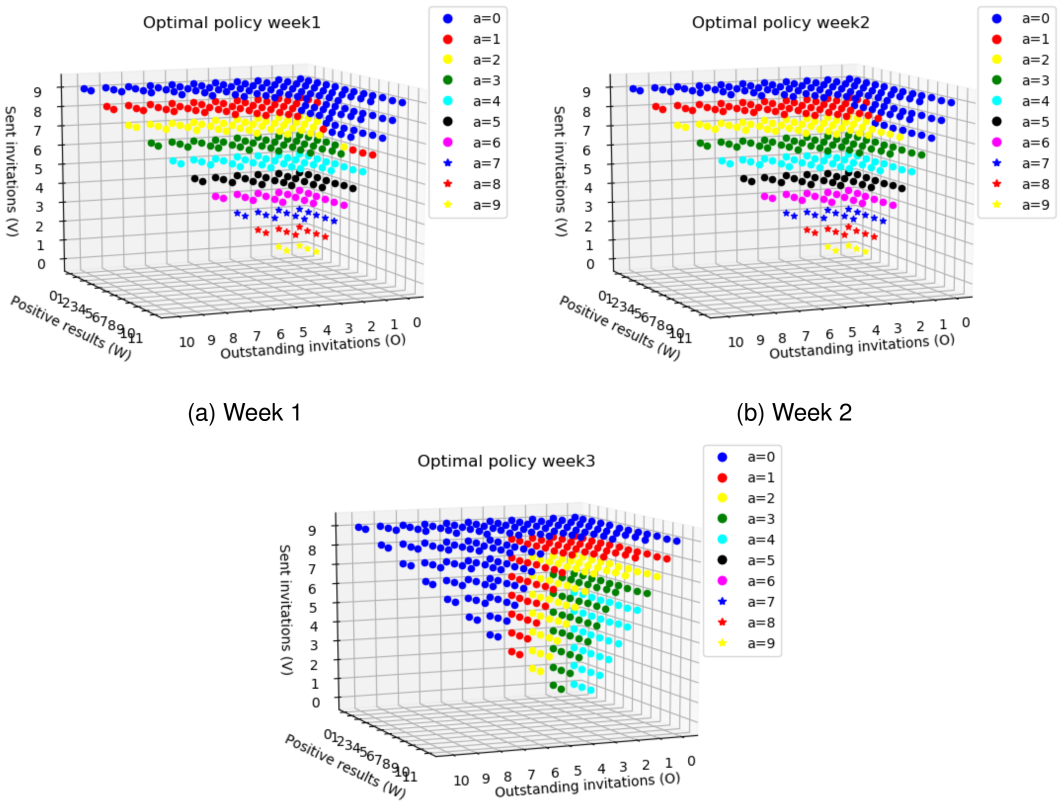
(b) Week 2

(c) Week 3

Figure 9.1: The optimal policies in each decision epoch for the instance with $G^n = 1$, $I^n = 1\ \forall n$, $H = 3$, $O^0 = 1$, $W^0 = 1$ and $\hat{O} = 2$.

Figure 9.2 shows the results when we have 9 clients instead of 3. The optimal policy in this case is of the same kind as the optimal policy with 3 clients. We only have one minor difference, which is not of great significance. When the number of positive results in the current state is large, we send less new invitations because we want the waiting list first to decrease. However, when also the number of outstanding invitations is large we do not invite less clients. This can be seen in Figure 9.2 in the layer of dots where $V = 7$, where from $O \geq 2$ the blue and red dots make place for the yellow dots which correspond to a higher action. Inviting more clients in this case can be explained by the fact that these outstanding invitations already effect the waiting time in the upcoming weeks and therefore one extra invitation does not make much difference. Inviting all clients is in this case more important. This effect of having lower actions in states with a higher number of positive results in only visible in states where already many invitations are sent ($V \geq 6$). With a lower number of sent invitations we just invite the number of new clients that equals the number of clients that is left to invite ($H - V$). This is optimal because at the end of the horizon we want to have invited all clients and this is more important than waiting time for an intake appointment.

Another difference between the instances with 3 or 9 clients is that with 3 clients the optimal actions in week 1 and 2 are the same, whereas with 9 clients the optimal actions in week 1 and 2 differ slightly. In an instance with more clients the time effect plays a bigger role, because with more clients we have more flexibility in the invitation strategy. When the end of the time horizon is closer (n bigger) we invite more clients in the same state than we do in an earlier week, because at the end of the horizon we want to have invited all clients.



(a) Week 1

(b) Week 2

(c) Week 3

Figure 9.2: The optimal policies in each decision epoch for the instance with $G^n = 1$, $I^n = 1\ \forall n$, $H = 9$, $O^0 = 1$, $W^0 = 1$ and $\hat{O} = 2$.

For the instance with $G^n = 1$, $I^n = 1\ \forall n$, $H = 20$, $O^0 = 0$, $W^0 = 0$ and $\hat{O} = 2$ we also get similar results as for the instances with 3 and 9 inhabitants, because 20 inhabitants is still less than the 30 clients which we can invite on one single intake slot. We do not include the graphs because they are unclear with this large number of clients.

Another reason that all clients are invited early in time, is that the horizon used in the above instances is quite short, only 3 weeks. When clients are invited in week n, they will start to respond from week $n+1$ and their result arises in week $n+2$ at the earliest. With a horizon of only 3 weeks you have to invite the clients early in order to make sure they go through the process before the horizon ends. On top of that only a small part of the clients responses directly. In expectation the response time is 3 weeks. This is the expectation in weeks of the exponential distribution for response time as given in Section 8.1, $1/\lambda = 19.7$ days. We can explain the expected response time also from another point of view. It is the expectation of a geometric distribution with parameter $q = 0.31$, the probability that a client responses coming week. The expected response time is than the expected number of weeks until a client responses with the success probability q for each week, so expected response time is $1/0.31 \approx 3$. With this average response time of 3 weeks and the extra week that is needed to sent the invitation we need at least a horizon of 4 weeks in order to have realistic results.

9.2 Computational problems

With the results of the previous section we can conclude that our algorithm to solve the single CC SDP with backward induction does not contain any errors. However, these results come from instances which are not realistic in practice and the results do not give us any information about the optimal invitation strategy in practice. We detected two problems of why our results give no useful information for realistic practical instances. First, we cannot handle enough clients in our algorithm. We need at least 30 clients for one intake slot with the current participation and referral rates. Second, the time horizon is not large enough, we need at least four weeks in order to examine the whole procedure of the invitation process in a realistic way. When the time horizon becomes larger we also need more clients to invite because each week we would like to invite at least 30 clients (a week has at least one intake slot). However, increasing the number of clients is not possible in our current algorithm. Our algorithm in Python can handle about 3000 states in the state space within an acceptable computation time of half an hour per decision epoch. These 3000 states in the state space correspond to a total of 24 clients, according to equation (8.12) in Section 8.4.1. We cannot increase the number of clients to find realistic solutions, so we think of another way to get realistic solutions.

We want to know which policy is optimal when not all clients can be invited in the same week, because this will not fit in one intake slot. Instead of solving an instance with more than 30 clients and the normal participation and referral rates of 73% and 4.7%, we can simulate a realistic situation by increasing the rates such that less clients can be invited on a single intake slot. In this way we will solve instances with a moderate number of clients which our algorithm can handle, but we still get the effect that we would like to have in practice.

In Section 9.3 we will give the results of these instances with higher participation and referral rates. For these instances we will use a horizon of 9 weeks ($T = 10$) in order to have a better view on the time aspect of the invitation process. The other parameters that we should determine for an instance are; the number of clients (H), the number of outstanding invitations at the beginning (O^0), the number of positive results at the beginning (W^0), the border of outstanding invitations at the end of the horizon ($\hat{O}$) and the participation and referral rate (PR and RR). We only have instances where all weeks in the time horizon have 1 available intake slot. The parameters should satisfy some constraints when we want to have a realistic instance which we can solve with our algorithm.

We have $T-2$ intake slots available for the clients that need to be invited and the outstanding invitations at the beginning. Namely, an invited client will respond earliest in week 1, the result is then available in week 2 and therefore the first intake appointment can take place in week 3. The intake slot of week 1 is already used by clients from the period before the time horizon started and the intake slots in week 2 are used for the beginning positive results. We want to have that the amount of clients will fit in these $T-2$ intake slots with the evaluated rates, therefore we want to have:

$$H + O^0 \geq (T-2) \cdot \frac{1}{PR \cdot RR} \tag{9.1}$$

We also want to have that we begin with such a number of outstanding invitations such that the intake slot of week 2 will be used. This gives us the following constraint.

$$O^0 \approx \frac{1}{q \cdot RR} \tag{9.2}$$

We chose to start with one positive result, so $W^0 = 1$, because in this way the first intake slot will be used and no waiting time from before the horizon is present. At the end of the horizon we want that (almost) all clients have completed the entire screening process, so the border value for the number of outstanding invitations should not be to large. We chose to use $\hat{O} = 2$. Together with constraints (9.1) and (9.2) we can now formulate some instances that we will evaluate. The two instances that we look at are given in Table 9.1. The results of these instances are given and explained in Section 9.3.

Table 9.1: The parameters of two scenarios with higher participation and referral rates.

Instance	T	PR	RR	H	O^0	W^0	$\hat{O}$	G^n	I^n
A	10	0.8	0.7	13	3	1	2	1	1
B	10	0.8	0.5	18	5	1	2	1	1

9.3 Results other rates

Figure 9.3 shows the results for instance A of Table 9.1. With these higher rates we can invite on average 1.8 clients on one intake slot. The graphs look a bit different than the graphs in the previous sections, because we now have a larger state space. The graphs in Figure 9.3 only show coloured dots when the optimal action in a state is strictly larger than 1, all states where no action is shown have optimal action $a = 0$. We changed the orientation of the graph to have a better view on the results. From the graphs in Figure 9.3 we see the following structures in the optimal policy for instance A.

- As the time increases (n becomes larger), we invite more new clients in the same state. This higher action is explained by the fact that at the end of the horizon we want to have invited all clients. In the beginning of the horizon we can wait with inviting until next decision epochs, but if the end of the horizon comes closer more clients need to be invited.

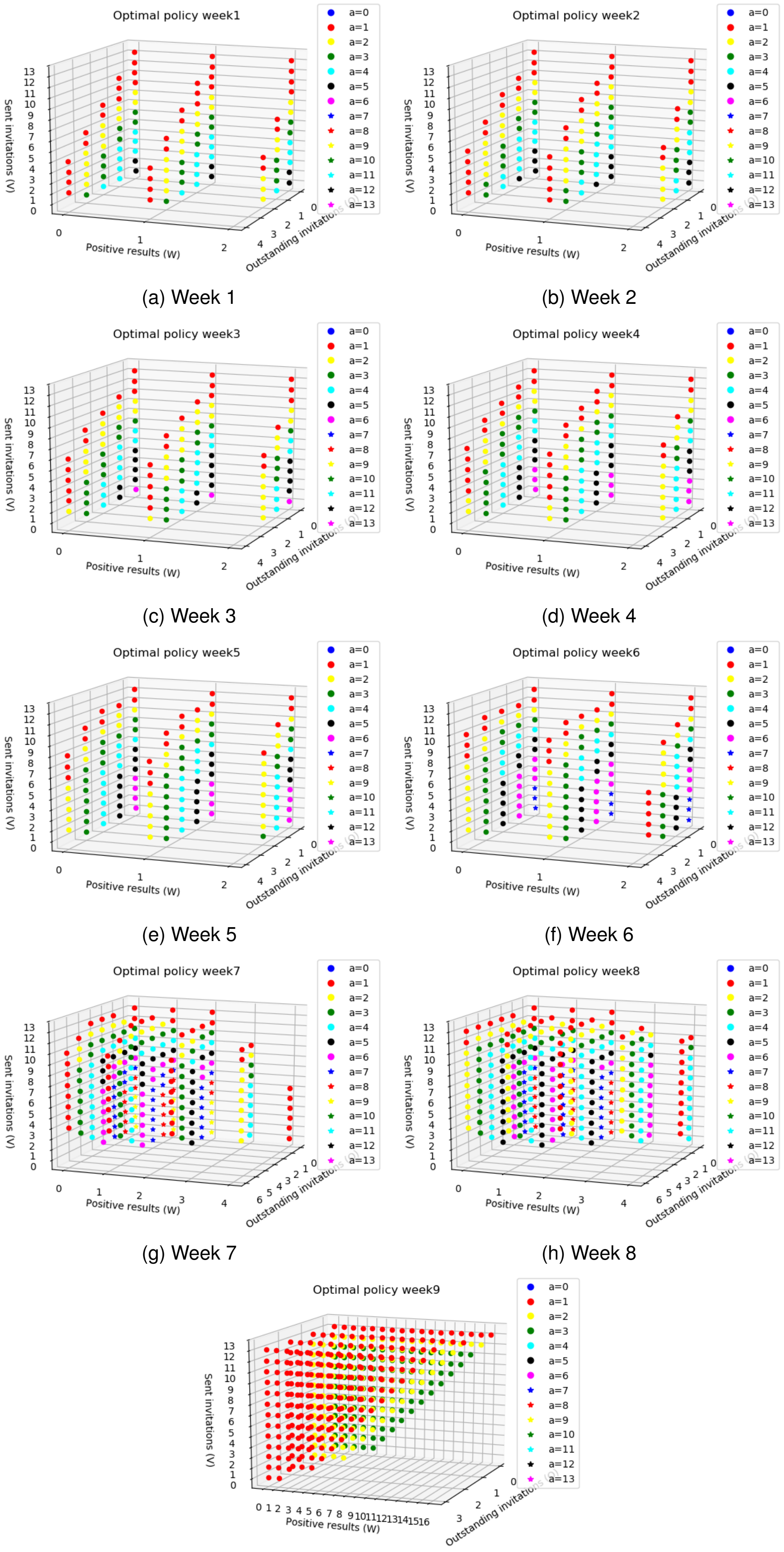
- In the last decision epoch (week 9) the optimal action is such that $O_n + a$ does not become larger than 3, because this will result in large waiting times next year. We want to satisfy the border of $\hat{O}$ in week 10. Only when we have 3 outstanding invitations at this moment (week 9) we send 1 new invitation which results in $O_n + a = 4$. This is because the probability that one of the clients with an outstanding invitation will respond during week 9 is higher when 3 outstanding invitations are present instead of less than 3 outstanding invitations. The responding clients in week 9 do not contribute to violating $\hat{O}$. Even with a high number of positive results we send some new invitations in the last decision epoch (week 9). The waiting time at the end of the year is not penalized extra and sending new invitations does not effect this waiting time directly. Therefore, the situation at the end of the horizon considered the positive results cannot be changed by the action taken.

- When the system is already busy, no more invitations are sent. These actions $a = 0$ are not directly visible in the graphs, because the blue dots are disregarded, but these actions appear in states that do not have a coloured dot. The system is busy when the sum of outstanding invitations and positive results is large ($O_n + W_n \geq d$). The value d depends on the number of sent invitations V and the week n. When $V = 0$ the value d is 5, so when we have sent zero invitations until now and more than 4 clients are present in the system (received invitation or have result), we do not send new invitations in this instance.

- In a state with a large number of positive results ($W_n \geq 3$) no new invitations are sent as long as the end of the time horizon is not nearby. We do not send new invitations with higher W_n, because sending new invitations might result in a waiting list for an intake appointment which is longer than 3 weeks. This is absolutely not desirable and we therefore wait until the large waiting list of $W_n \geq 3$ has decreased before sending new invitations. From week 7 onwards also with a larger number of positive results new invitations will be sent, because minimizing the rest group also becomes important. The probability that waiting time exceeds three weeks with inviting new clients is smaller than the probability that a rest group arises with no new invitations. The latter probability equals 1. Therefore it is optimal to take the risk of violating the three weeks waiting time but certainly have a lower rest group by inviting some clients from week 7 onwards in a state with $W_n \geq 3$.

- In a state with a larger number of outstanding invitations, the optimal action becomes smaller. With more outstanding invitations the probability that clients will respond in the coming week becomes larger and more clients are already present in the system. This means that in the coming weeks the system will be able to operate and additional new invitations are less needed. The larger the number of outstanding invitations is the lower the number of new invitations, because we need less clients to replenish the system. This relation is linear but does not have slope 1, an increase of 1 outstanding invitation does not imply a decrease of 1 in the number of new send invitations. Mostly the decrease in action is higher than 1 per outstanding invitation.

- When the number of sent invitations (V_n) is larger, the optimal action is to send less invitations. Also here the relation does not have slope 1. The number of sent invitations needs to increase with more than one in order for the optimal action to decrease with 1. We send less new invitations when we already have sent more invitations earlier because of two reasons. First, with more sent invitations (V) we have less clients to invite, namely $H - V_n$. Second, with more sent invitations the need of sending new invitations is smaller because the rest group is already smaller. For the fewer client that we should still invite with higher V we have enough time and we therefore might take a lower action at the current moment.



Optimal policy week1

(a) Week 1

Optimal policy week2

(b) Week 2

Optimal policy week3

(c) Week 3

Optimal policy week4

(d) Week 4

Optimal policy week5

(e) Week 5

Optimal policy week6

(f) Week 6

Optimal policy week7

(g) Week 7

Optimal policy week8

(h) Week 8

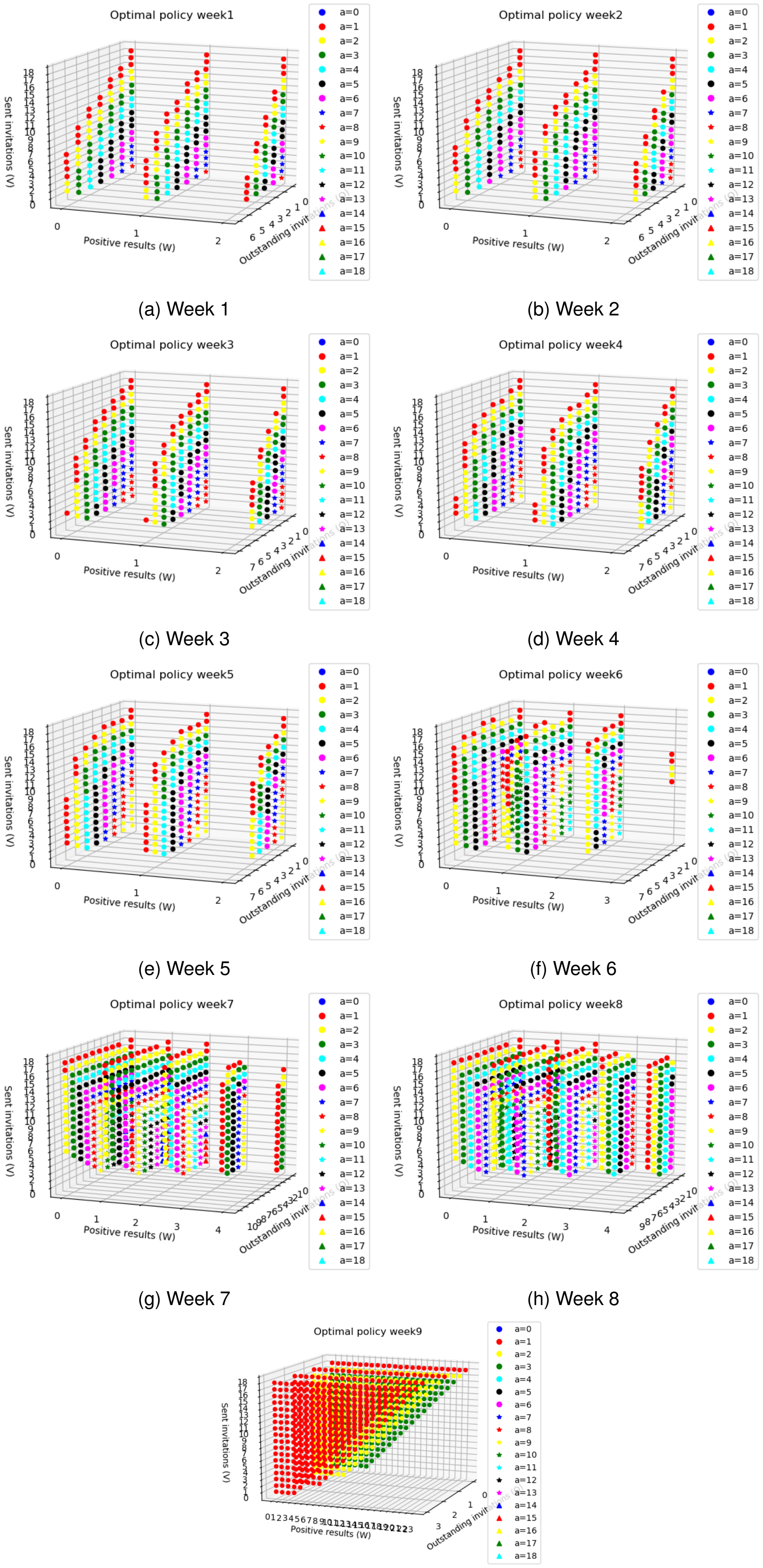
Optimal policy week9

(i) Week 9

Figure 9.3: The optimal policies in each decision epoch for instance A.

Figure 9.4 shows the results for instance B of Table 9.1. With these higher rates we can invite on average 2.5 clients on one intake slot. These graphs show the same structures in the optimal policy as described above. In addition to these structures we mention two things that stand out in the graphs of Figure 9.4.
In week 6 (Figure 9.4f) we have five non-zero actions in states with $W = 3$. These actions do not align with the above described structure of the optimal policies in the instances A and B. On top of that the actions with value 2 (yellow dots) are placed between actions with value 1 (red dots), whereas normally actions are monotone increasing or decreasing. We do not have a proper explanation for the actions in these 5 states, however they are only of a minor significance. These states will only be visited with a very small probability, because three positive results at the same time are very rare. Due to these small probabilities the differences in value function between possible actions are very small and the optimality of the given action might be due to computational inaccuracy.
Second, we see in week 6,7,8 in both Figures 9.3 and 9.4 in the top layer of actions ($V = H - 1$) some gabs in the actions when we look over increasing outstanding invitations. For example in Figure 9.4f the optimal action in state $(1, 1, 17)$ is zero whereas the surrounding states (with other O) have optimal action 1. This gab in the optimal actions is strange, but can be explained in a similar way as above. In these states the differences between the actions are very small and due to computational inaccuracy we make this slight mistake.

We explained that the optimal policies of both instances of a single CC SDP have a certain structure. However, the optimal policy is not a common threshold strategy where we invite clients by supplementing the outstanding invitations to a given threshold. We have consecutive states with the same action or the consecutive states differ more than 1 in the optimal action. It is difficult to give a closed form formula that gives us for each state the optimal action because a possible threshold policy depends on four different quantities. First the state space consists of three quantities which all have influence on the optimal action and second the point in time has influence. We do not have a static optimal policy. These four quantities make finding the optimal policy in a direct way complex. We see that the optimal policy has a structure, but more research is needed to quantitative describe this structure.



(a) Week 1

(b) Week 2

(c) Week 3

(d) Week 4

(e) Week 5

(f) Week 6

(g) Week 7

(h) Week 8

(i) Week 9

Figure 9.4: The optimal policies in each decision epoch for instance B.

Chapter 10

Discussion

In this chapter we will give a reflection on our research. Each following section corresponds to one part of the research. We first discuss the deterministic model part, then the robust optimisation part followed by the time uncertainty model. In Section 10.1 we summarize the suggestions for further research.

Deterministic model

In the results of the MILP we see a specific adherence when two CCs are situated in the same city. For example, all postcode areas West from Nijmegen go to the CC in West-Nijmegen and postcode areas East from Nijmegen are linked to the CC situated in the East of Nijmegen. This is not what is desirable in practice, because this really small travel time difference between the CCs does not play a role in practice. Therefore it might be a good idea to aggregate two CCs in the same city to one single CC. Another option is to use the penalty function to make some distinction between the two CCs.

In this research we gave the possibility of using a penalty function for postcode area - CC combinations that are not desired. However, we did not use this function because we do not have the proper data for this. In order to fine-tune the MILP to reality this penalty function can help. Therefore further research is needed to gather the data that is needed to determine which postcode area - CC combinations are not desirable.

In this research we gathered data for the number of clients in the postcode areas via public available sources and distributed the clients over the ages as good a possible by empirical probability distributions. This data gives reasonable insight in the actual values for the parameters of our model, but when we really want to find an invitation strategy that is suitable for practice we will need the actual real-time data of the number of clients. When we have this data we can easily give the data as input in our model.

Also the data we used for the number of available intake slots per week in each CC, is based on our educated guesses. The total number of intake slots for a CC per year is realistic data, but the distribution of intake slots of the year is not known in practice. Our method of decreasing the number of available intake slots in holiday weeks is reasonable but might not align fully with practice. We therefore need exact numbers of available intake slots in each week for all CCs. Asking the CCs at the beginning of the year to give their available intake capacity in each week is maybe demanding too much of the CCs, but needed for an optimal invitation strategy.

These weekly intake capacities of the CCs also have to align with the intake capacity needed in each area. We saw that in total there is enough capacity in region East, but the capacity is not always situated at the right places. We therefore propose to investigate possibilities for predicting the needed capacity for a CC per week. With this information it is easier for Bevolkingsonderzoek Oost and the CCs to come to a good distribution of available intake slots over the year and region. One of the possibilities to achieve such a prediction of needed capacity can be based on the methods used in Kortbeek et al. [2015]. They predict the bed census on nursing wards per hour based on the Master Surgical Schedule (MSS) and arrival patterns of emergency patients. In developing such a prediction method we might also take into account multiple client types, based on the number of times the client has already participated. Clients from different invitation rounds or even age and home address may have different characteristics as probability of participating and probability of having a positive result. As input for the prediction method we might consider to invite clients on their date of birth and see what effect this has on waiting time and travel time of the clients.

In this research the screening process ends when the intake appointment is scheduled, because the actual colonoscopy takes place under the responsibility of the CC. However, the time that a CC needs to perform a colonoscopy is much larger than the time needed for an intake appointment. It might therefore be interesting to include the actual colonoscopy appointment in the planning process of the invitations for the screening process. The actual colonoscopy is much harder to plan for a CC than an intake appointment. It might be wise to invite clients to the screening process based on colonoscopy capacity instead of intake capacity. However this is more difficult because the colonoscopy should take place after the intake appointment. Another possibility is to use another model to determine the intake slots in a CC based on their available colonoscopy capacity and then use these intake slots in the models of our research.

In practice Bevolkingsonderzoek Oost also sends (new) invitations to clients who lost their test or clients who had a test that was not assessable. We did not take these cases into account, because they are to detailed and do not contribute to a general steady state invitation strategy. We think that the number of new invitations for these clients can be neglected, because they will be compensated by the number of invitations that again get lost or are not assessable. Due to the law of large numbers the amount of invitations that disappear from the invitation process by these reasons is the same as the number of new invitations (caused by these reasons) that enter the invitation process. However when we really want to use the MILP model in practice we should take these new invitations into account.

The last thing that we would like to mention about the MILP model is the aggregation level of the postcode areas. We used PC4 areas which consists of all addresses with the same 4 digits in their postcode, because a smaller aggregation level leads to computational problems. We also do not have the number of clients that live in a smaller area. However in practice we would like to invite individuals to the colon cancer screening program. For this we need a smaller aggregation level and ultimately the matchings variable will depend on p which stands for a single participant. By using a more detailed level of aggregation also the handling of uncertainty in robust optimisation will be more realistic. The uncertainty of participation- and referral rate is actually depended on the client and not on a whole postcode area. The variance in the total number of needed intakes in a week and CC will decrease when we have uncertainty per individual instead of per postcode area. In order to develop the MILP model to a model which can handle individual clients, more research is needed.

Robust optimisation model

Due to the large number of variables in the robust optimisation model we had to use an LP-relaxation to solve the robust counterpart. With this LP-relaxation and our algorithm to obtain an integer solution we are able to find robust solutions to the matching problem of clients from postcode areas to CCs and week numbers. However, these solutions are approximations of the optimal integer solutions. We should keep this in mind, despite the fact that we give a bound on the optimality gabs of the found solutions. Further research might include a search for methods that can give us the optimal integer solutions of the robust optimisation problems.

We tried to find out why the LP-relaxation works so good. During this search we saw that solving the MILP mostly gave integer LP-relaxation solutions. This indicates a possibly Totally Unimodular (TUM) property of the constraint matrix of this specific matching problem of clients to week numbers and CCs. After some research, we were able to prove that the MILP is TUM in the case where no overcapacity distribution constraint is considered. This also holds when all available capacity is used to invite clients. However, this does not say anything about why we can use the LP relaxation to find solutions for the robust model or the MILP including the overcapacity distribution constraint. Adding a constraint or changing a constraint slightly can easily destroy the TUM property (which happens in our case) and you cannot say anything about how good a LP-relaxation is in that case. We therefore did not include the TUM proof in this thesis, because it does not add any valuable information to this research. It only explains why we can solve the MILP very fast when all available capacity should be used, because we then only need to solve the LP problem which gives us an integer solution directly. However in practice this is not valuable. Because in the coming years more available capacity is expected so we will need the overcapacity distribution constraint. In the second place we do not want to use all available capacity due to the uncertain nature of the invitation process, we want to have a robust solution that builds in some buffer capacity.

According to the same reasoning as above the TUM property of the MILP without overcapacity constraint does not give any guarantee that using the LP-relaxation for the robust optimisation problem works good. The robust optimisation has great similarities with the MILP, but the small adaptations made, can give a totally other structure to the problem and the TUM property is not valid any more. Why we do use the LP relaxation in robust optimisation we already explain in Section 7.3.

We may do robust optimisation in a different way by determining in advance the probability that the limited capacity constraint is violated under different numbers of invited clients. We determine N with a desired probability of constraint violation (for example 5%) by using the normal distribution of Section 7.4.3. We take the largest N for which the probability of constraint violation is still below 5%. We then can solve the MILP where the available capacity is N in number of clients instead of I_c^t in number of intake slots. The buffer capacity is then determined before we solve the MILP. With this method we do not have all the auxiliary variables of the robust counterpart and we can find an optimal integer solution.

The disadvantage of this method is that we assume that the uncertainty in participation- and referral rate has exactly the normal distribution. Robust optimisation is more general, because there you only assume in which range the uncertain parameters take their values.

Time uncertainty model

We first notice that we explained why we use the decomposition method from the general SDP to the single CC SDP and why we think that it is a good approximation, but we were not able to say something about the quality of the approximation. For this more research into decomposition methods and error bounds is needed.

In the SDP we used binomial distributions for the number of participations, the number of responses and the number of positive results. However, during the probability calculations of constraint violation in the robust optimisation solutions, we concluded that a binomial distribution may not be the best distribution to describe the uncertainty from practice. Using a normal distribution with adapted variance seemed to describe the uncertainty better for the case of constraint violations. We therefore might use this normal distribution also in our SDP instead of the binomial distributions for the transition probabilities.

We could have looked into instances of the single CC SDP where a week has no intake slot available. It is interesting to see what would be the effect of this, because a drop in available intake slots in a year happens in practice in holiday weeks. We expect that a couple of weeks before the holiday the strategy of inviting clients will become different, less clients will be invited. When the holiday week comes closer we expect that the invitation strategy will go back to normal because these clients will need an intake appointment after the holiday week, which is again a normal week. In general we need more testing instances in order to detect the actual structure in the optimal policy.

We now fixed the problem of non-realistic instances of the SDP by increasing the participation and referral rate, but in the further we want to be able to solve the SDP for realistic instances with normal amount of clients and real participation- and referral rates. The results of the non-realistic instances of our research can give us an idea of the optimal invitation strategy, however this method comes with a disadvantage. With the higher participation- and referral rates, the variance of the binomial distribution for the number of participations and positive results becomes larger. With a higher variance the system of the invitation process can behave differently. In our research we neglected this effect but we should keep it in mind when we want to extend the found optimal policies to realistic instances.

There are different methods available for solving larger instances of the single CC SDP models. The first option is to program the backward induction algorithm more efficiently. However, we already made a small step in this direction during our research. We do not know how many improvement in computational efficiency is possible. A second option to solve larger instances, is to reduce (or truncate) the state space of the single CC SDP. We might say that the number of outstanding invitations does vary between a lower bound larger than 0 and an upper bound smaller than the number of clients, because in a stable invitation process the number of outstanding invitations will be more or less constant. This will also hold for the number of positive results. The bounds of the state space considering the number of positive results can be a lot smaller than the bounds for the number of outstanding invitations because only 4.7% of the participating clients will have a positive result on average. Using these lower- and upper bounds will reduce the state space of the SDP which will reduce the computation time needed for the backward induction algorithm. However, the disadvantage of this method is that determining these bounds is complicated and will mostly rely on assumptions and approximations. On top of that the behaviour of an SDP is often different on the boundary of the state space than in the kernel. By approximating the bounds of the state space, large errors might occur. The third option to solve larger instances is to develop an Approximate Dynamic Program (ADP) for the single CC SDP. ADPs are well known to overcome the ”curse of dimensionality”, however they are also difficult to develop.

Testing solutions

The solutions that we found in our research with the different model types (MILP, robust optimisation and SDP) are based on our assumptions and used data. The quality of the solutions in practice are not fully known yet, we can only expect that the solutions are good, based on our analysis in this research. Before implementing the found invitation strategies in practice we might want to test the solutions better. This testing can be done by using Discrete Event Simulation models. We then implement the found invitation strategies and simulate what will happen in practice. This testing with simulation might help in getting more confidence in the quality of our solutions.

10.1 Further research suggestions

Based on the given discussion we give the following suggestions for further research.

- In order to make the MILP model more realistic to practice, extra/other data should be collected. This data consists of undesired postcode area - CC combinations, actual number of clients and real number of available intake slots for each week of the year.
- A prediction tool for the number of intake slots needed per week and CCs might help in determining which intake capacity is needed.
- The time needed for the actual colonoscopy that takes place after an intake appointment might be taken into account in optimising the invitation process.
- A smaller aggregation level for the postcode areas will help in making the models more detailed and suitable for practice. Further research will here-fore consist of finding more efficient solving methods that can handle the large number of variables that occur when individuals should be invited.
- For actually implementing the robust optimisation model we can investigate the correctness of our LP-approximation or find more efficient solving algorithms for the integer robust counterpart.
- Further research is needed to quantify the approximation error that we make in decomposing the general SDP into smaller single CC SDPs.
- In order to detect the structure of the optimal policy in the single CC SDP more instances can be solved and different types of instances can be tested.
- To be able to solve the single CC SDPs for larger and more realistic instances we might look at better algorithmes, state space reduction methods or Approximate Dynamic Programming (ADP) techniques.
- All found solutions for the MILP, the robust optimisation model and the SDP can be tested whether they are realistic in practical situation, by using Discrete Event Simulation models.

Finally our research concentrated only on one of the 5 screening organisations in the Netherlands. A logical extension of our research would be to include all regions of the Netherlands. This will probably be easy because the mathematical models are quite general. However, with examining the entire Netherlands the curse of dimensionality might play a role because of the fact that the number of clients, postcode areas and CCs will increase.

Chapter 11

Conclusion

In the previous chapters we gave our developed models for optimising the invitations strategy for the colon cancer screening program at Bevolkingsonderzoek Oost. We also gave the results and explained the results in detail. In this chapter we answer the research questions from Section 1.2 and give the conclusions of our research.

A Mixed Integer Linear Program (MILP) is a correct and useful model to find an optimal matching between clients from postcode areas (PC4) to week numbers and colonoscopy centres (CCs). With the developed model we can easily find an invitation strategy which tells us how many clients from which postcode areas should be invited to which colonoscopy centre (CC) and in which week of the year. We take the invitation interval from subsequent round clients into account and make sure that the invitations are evenly distributed over the year while satisfying the available capacity. We minimize the rest group of clients at the end of the year and we minimize the travel time of the clients to their CC, where we maximize the number of clients linked to their nearest CC.
Our model gives other adherence numbers of linking the postcode areas to CCs than the current adherence of Bevolkingsonderzoek Oost. Our optimal matching is 96% better in objective value than the current adherence of Bevolkingsonderzoek Oost. With the current adherence of Bevolkingsonderzoek Oost, we have a rest group of 7.8% and almost 40% of the clients cannot be invited in the nearest CC. With a participation rate 73% and a referral rate of 4.7% it is possible to invite 96.7% of the clients with the available capacity by using the developed MILP model. All subsequent round clients can be invited in their predefined interval of 22-26 month after their previous invitation. The average travel time of the clients to the CC that they are linked to is 16.3 minutes. 17.3% of the clients cannot be invited in the nearest CC, which is much smaller than currently 40%. This reduction in clients not linked to the nearest CC can result in a decrease of rescheduled intake appointments. In total their is enough available capacity in the region East, but the capacity is situated in the wrong places. The distribution of capacity over the entire region East does not align with the distribution of clients over region East. The improvement from to current situation to the developed model results, is caused by using smaller invitation areas, postcode 4 areas instead of councils, and travel times instead of euclidean distances. Also maximising the number of clients that is linked to the nearest CC contributes to the improvement.

With the robust optimisation model we are able to find a ”safe” solution to the matching problem of clients from postcode areas to week numbers and CCs. In this way we can immunize against uncertainty in the participation and referral rates of the clients, which vary in the intervals $[70\%, 76\%]$ and $[4.3\%, 5.1\%]$ respectively. The best suitable method of robust optimisation is budgeted uncertainty, which rules out large deviations from the cumulative number of needed intake appointments in a CC in a certain week. A value of 4.7 for the safety parameter γ seems to be most optimal and desirable in practice. This can be achieved by having a tolerance level of $\epsilon = 10\%$ and a dimension of the perturbation vector of $L = 5$. These values mean that it is guaranteed that a limited capacity constraint is violated in at most $\epsilon = 10\%$ of the cases and at most $L = 5$ different postcode areas, that are linked to one single week and CC, are allowed to have the worst case participation and referral rate. The values $\epsilon = 20\%$ and $L = 7$ give similar results. These safe solutions have buffer capacity in the CCs, so not all available capacity is used, which gives us a rest group of about 9% at the end of the year. However, the probability that a client cannot have his intake appointment in the planned week and/or CC is only 8% instead of 22% when uncertainty is not taken into account. These safe solutions perform better than using the worst case participation and referral rates in the deterministic MILP model, where a rest group of 14.4% arises.
The given results in this research are based on approximations and uncertainty assumptions. The computational complexity of the robust optimisation model makes it difficult to apply this method directly in practice. Further research is needed in this direction but also in the direction of finding better probability distributions that describe the uncertainty of participation and referral rate in practice. However, the given results of this research in robust optimisation are promising.

The uncertainty in time of the screening process can be taken into account by using a Stochastic Dynamic Program (SDP), where we do not take the pré-announcement period into account. We model the response times of clients with an exponential distribution. In this way we only need to keep track of the number of outstanding invitations for a CC instead of all the dates at which clients received their invitation. Due to the large state space and action space of the developed general SDP we were not able to find solutions in practical situations. With a decomposition method that uses the adherence results of the MILP we split the general SDP into smaller sub-models, each representing a single CC. These single CC SDPs can be solved by backward induction and do not depend on each other.
We are only able to find solutions in instances with a small number of clients to be invited and high participation and referral rates. These instances are not realistic in practice, but do give us an idea of the optimal invitation strategy. These solutions suggest a structured optimal policy where we send invitations to clients during the year when enough clients still need to be invited under the conditions that in the current week (1) the number of outstanding invitations is small and (2) the number of positive results is not to large. As mentioned earlier this are only preliminary results and further research is needed to be able to use this SDP model in practical problems.

11.1 Recommendations

This conclusion can be translated to a couple of concrete recommendations, which are listed below.

- By using travel times instead of euclidean distances and maximizing the number of clients that is linked to the nearest CC, more clients can be satisfied. Meaning that the clients can have their intake appointment in the nearest CC. In this way the number of rescheduled intake appointments can be decreased.
- Instead of councils for the invitation areas, we can use postcode 4 areas, or even smaller regions. In this way the invitations and linkings to CCs are more specific for the clients and their home addresses.
- Another intake capacity distribution over the region is needed for a better linking of clients to CCs such that clients can be linked to the nearest CC. For this Bevolkingsonderzoek Oost might discuss the available capacity in more detail with the CCs and they can give the CCs more insight in what capacity is needed in the region.
- By using the invitation interval of 22-26 months after the previous invitation for subsequent round clients, it is possible to have a better distribution of invitations over the year.
- If Bevolkingsonderzoek Oost wants to have a ”safe” invitation strategy, a budgeted uncertainty model for robust optimisation might be useful. A safety parameter of $\gamma = 4.7$ gives the best results according to our research. However, for actually implementing this method in practice more research is needed.
- If Bevolkingsonderzoek Oost does not send pré-announcement letters anymore, they can save €37,500 each year. On top of that it is easier to predict available capacity 2 weeks less in advance.
- The SDP model can be further developed for the use in practical problems.

Bibliography

Rijksinstituut voor Volksgezondheid en Milieu, Ministerie van Volksgezondheid, Welzijn en Sport. Uitvoeringskader bevolkingsonderzoek darmkanker, December 2017.

E. Toes-Zoutendijk, M. E. van Leerdam, E. Dekker, F. van Hees, C. Penning, I. Nagtegaal, M. P. van der Meulen, A. J. van Vuuren, E. J. Kuipers, J. M. G. Bonfrer, K. Biermann, M. G. J. Thomeer, H. van Veldhuizen, S. Kroep, M. van Ballegooijen, G. A. Meijer, H. J. de Koning, M. C. W. Spaander, and I. Lansdorp-Vogelaar. Real-time monitoring of results during first year of dutch colorectal cancer screening program and optimization by altering fecal immunochemical test cut-off levels. *Gastroenterology*, 152(4):767 – 775.e2, 2017.

M. Bretthauer, G. Gondal, I. K. Larsen, E. Carlsen, T. J. Eide, T. Grotmol, E. Skovlund, K. M. Tveit, M. H. Vatn, and G. Hoff. Design, organization and management of a controlled population screening study for detection of colorectal neoplasia: Attendance rates in the norccap study (norwegian colorectal cancer prevention). *Scandinavian Journal of Gastroenterology*, 37(5):568–573, 2002.

D. Bertsimas and I. Popescu. Revenue management in a dynamic network environment. *Transportation Science*, 37(3):257–277, 2003.

A. Gosavi, E. Ozkaya, and A. F. Kahraman. Simulation optimization for revenue management of airlines with cancellations and overbooking. *OR Spectrum*, 29(1):21–38, Jan 2007.

A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, 2009.

V. Gabrel, C. Murat, and A. Thiele. Recent advances in robust optimization: An overview. *European Journal of Operational Research*, 235(3):471 – 483, 2014.

T. Aouam, K. Geryl, K. Kumar, and N. Brahimi. Production planning with order acceptance and demand uncertainty. *Computers & Operations Research*, 91:145 – 159, 2018.

Chen Chen and Yu sha Zhou. Robust multiobjective portfolio with higher moments. *Expert Systems with Applications*, 100:165 – 181, 2018.

B. Addis, G. Carello, A. Grosso, and E. Tànfani. Operating room scheduling and rescheduling: a rolling horizon approach. *Flexible Services and Manufacturing Journal*, 28(1):206–232, Jun 2016.

P. Beraldi, M.E. Bruni, and D. Conforti. Designing robust emergency medical service via stochastic programming. *European Journal of Operational Research*, 158(1):183 – 193, 2004.

P. Cappanera, M. G. Scutellà, F. N., and L. Galli. Demand uncertainty in robust home care optimization. *Omega*, 2017.

Topicus Zorg. Functioneel ontwerp screenit darmkanker, January 2017.

Nationaal Georegister. Lijst van woonplaatsen per gemeente en provincie, March 2018. URL https://data.overheid.nl/data/dataset/lijst-van-woonplaatsen-per-gemeente-en-provincie.

CBS. Buurt, wijk en gemeente 2017 voor postcode huisnummer, September 2017. URL https://www.cbs.nl/nl-nl/maatwerk/2017/38/buurt-wijk-en-gemeente-2017-voor-postcode-huisnummer.

CBS. Bevolking per viercijferige postcode op 1 januari 2017, September 2017. URL https://www.cbs.nl/nl-nl/maatwerk/2017/39/bevolking-per-viercijferige-postcode-op-1-januari-2017.

CBS StatLine. Bevolking; geslacht, leeftijd, burgerlijke staat en regio, 1 januari, January 2017. URL http://statline.cbs.nl/Statweb/.

Object Vision. Reisweerstandtabel pc4, September 2011. URL http://www.objectvision.nl/gallery/themes/bereikbaarheid/reisweerstandtabel/pc4.

Creative Commons. Cc-sa-by-nl-30 licence. URL https://creativecommons.org/licenses/by-sa/3.0/nl/.

Imergis. 2018-nl-gemeenten-basis, March 2018. URL http://www.imergis.nl/asp/44regios.asp.

D. Bertsimas and A. Thiele. A robust optimization approach to inventory theory. *Operations Research*, 54(1):150–168, 2006.

D. Bertsimas and A. Thiele. A robust optimization approach to supply chain management. In *Integer Programming and Combinatorial Optimization*, pages 86–100. Springer Berlin Heidelberg, 2004.

A. M. Law. *Simulation Modeling and Analysis*. McGraw-Hill, fifth edition, 2015.

M.L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 2005.

R. P. Grimaldi. *Discrete and Combinatorial Mathematics, An Applied Introduction*. Pearson, 2003.

C. Daoui, M. Abbad, and M. Tkiouat. Exact decomposition approaches for markov decision processes: A survey. *Advances in Operations Research*, 2010:19, 2010.

T. W. Archibald. Modelling the operation of multireservoir systems using decomposition and stochastic dynamic programming. 2004.

G. Dantzig and P. Wolfe. Decomposition principle for dynamic programs. *Operations Research*, 8: 101–111, 1960.

M. M. Fazel-Zarandi and J. C. Beck. Solving a location-allocation problem with logic-based benders' decomposition. In *CP*, 2009.

R. Parr. Flexible decomposition algorithms for weakly coupled markov decision problems. *CoRR*, abs/1301.7405, 2013.

P. Laroche, Y. Boniface, and R. Schott. A new decomposition technique for solving markov decision processes. In *ACM symposium on Applied computing*, 2001.

N. Kortbeek, A. Braaksma, H. F. Smeenk, P. J. M. Bakker, and R. J. Boucherie. Integral resource capacity planning for inpatient care services based on bed census predictions by hour. *Journal of the Operational Research Society*, 66(7):1061–1076, 2015.