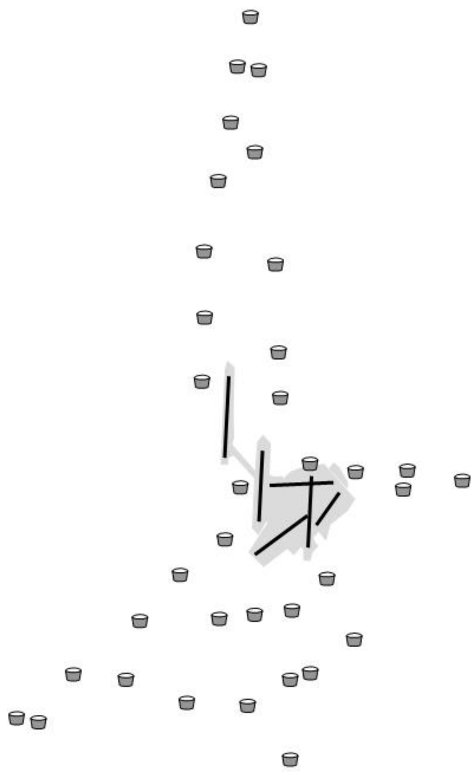
Noise load management at Schiphol

A stochastic dynamic approach



T.R. Meerburg
0008974
UT TW

March 24, 2006

LVNL ATM/R&D/PIA

University of Twente
Enschede - The Netherlands
UT TW/SOR

Status page

Document data	
Title:	Noise load management at Schiphol
Subtitle:	A stochastic dynamic approach
Volume:	1
Document number:	
Version number:	1.0
Status:	Completed
Version date:	24-03-2006
Number of pages:	94

Authored by:	
Name:	**Function:**
T.R. Meerburg	Student at UT TW/SOR on internship in the period from 17-05-2005 till 17-04-2006 at LVNL ATM/R&D/PIA (Stationsplein 1001, 1117 VC Schiphol-Oost)

Accepted by (coach LVNL):			
Name:	**Function:**	**Initials:**	**Date:**
M.J.A.L. van Kraaij	Performance Expert		

Accepted by (coach UT):			
Name:	**Function:**	**Initials:**	**Date:**
R.J. Boucherie	Professor		

Change log

Version:	Version date:	Chapters changed:	Remarks:
0.1	14-06-2005		document creation
0.2	02-08-2005	2,3	additions
0.3	01-09-2005	4,5	additions
0.35	15-09-2005	1,2,4	additions, revision
0.4	15-10-2005		additions, revision
0.5	31-12-2005	6	
0.6	20-01-2006	5,6,7	
0.7	09-02-2006	5,7	
0.8	03-03-2006	all	
0.9	18-03-2006	8,9	
1.0	24-03-2006	all	

Summary

Bounds for noise load are important environmental restrictions at Schiphol. Noise load is calculated for a number of enforcement points surrounding the airport. Each one has a standard that may not be exceeded. The noise load in the enforcement points is managed during the year by the use of preference lists for runway utilization. Monthly it is determined what preference list is best to implement for the coming period. The current method to determine this preference list does not take into account the stochastic nature of the weather and produces inaccurate values for exceedance probabilities. The main goal of this project is to find and develop a new mathematical framework that can be used for decision-making in noise load management.

A framework is devised that is based on realization possibilities instead of the expected realization. The essence of noise load management is captured by the stochastic dynamic programming method: successive decision-making under uncertainty. Using this method, the decisions are based on all noise load realization possibilities. Therefore it is a better method than the current method to determine preference lists. It also offers the opportunity to investigate some interesting alterations to the Schiphol operation, like the increase in decision moments for the implementation of preference lists.

Noise load management was modeled, so that stochastic dynamic programming could be applied. Numeric results for the exceedance probabilities and effects of doubling the number of decision moments were obtained for scenarios with three and four modeled enforcement points. Due to computation time issues, the implementation of the method is done in a discrete setting. This resulted in discretization errors or calculated values.

It was found that, using the current implementation, values for the exceedance probability could be found to within 6 times the discretization interval. The found exceedance probability for 2006 is between 4.59% and 16.26%. A doubling of decision moments was found to result in a significant decrease in exceeding probability. It is therefore recommended that the number of decision moments is increased.

The calculation of the exceedance probability is more insightful and mathematically founded. It allows for a better utilization of preference lists and thus a more efficient handling of traffic at Schiphol. Stochastic dynamic programming is therefore an interesting method for Air Traffic Control the Netherlands to use. It is recommended to further research the possibilities and implementation of this framework.

Preface

This research project was issued by the Performance & Incidents Analysis department (PIA) of Air Traffic Control the Netherlands (LVNL). They provided the general assignment, which was further specified by myself.

This project also serves as my graduation project. It completes my studies at the Stochastic Operation Research chair at the Applied Mathematics department of the Faculty of Electrical Engineering, Mathematics and Computer Science at the University of Twente. The graduation was in the form of a combined internship and final thesis. This report describes both; the accent of the internship lays with Sections 2 and 3 and Appendix F and G.

For his constant demanding questions and coaching at LVNL I would like to thank Maurice van Kraaij. It has been a pleasure to do research with him. In addition, I would like to thank the rest of the department Performance & Incident Analysis for their time and support. Thanks to Jasper Daams, who approved my application for the internship and gave me a chance to work at LVNL.

Furthermore, I would like to thank Richard Boucherie for his coaching at the University of Twente. His guidance and hard working mentality encouraged me to complete a project that I can be proud of.

Many people have contributed to this report in some way. The encouragements of Robert Ossevoort, Conny Schiere, Folkert van Vliet and Tom Coenen should not go unmentioned.

For the support on home base, I thank Ben Meerburg and Jeannet Harms for their continuous backing and reviewing of preliminary reports. A special thanks to Enif Stevens for listening to all my complaints and frustrations and still wanting to share her free time with.

Tristan Meerburg
March 2006

Contents

1 Introduction

This report describes the investigation into a method that optimizes the operation at Schiphol within the bounds for produced noise load. By altering the operation during the year, the noise load is managed so that bounds are not exceeded. The current framework lacks the ability to base decisions for alterations to the operation on accurate data. This report proposes a method that can be used for the decision-making and is intended as a starting point for further development and improvement in noise load management.

The report is organized as follows. In this section, air traffic control is introduced and the relevant parties are described. In Section 2 the background of the current situation is presented. Additionally the problem is described and interesting aspects for investigation are pointed out. The method that forms the basis for the new framework is discussed in Section 3, after which the model is described and formulated in Section 4. A theoretical analysis on some aspects the method and the model brought forth is given in Section 5. For the practical part, the implementation process of noise load management is discussed in Section 6 Obtained numeric results can be found in Section 7. Concluding remarks and some recommendations are given in Section 8. A number of appendices conclude this report. The coherence and arrangement of sections is graphically represented in Figure 1.

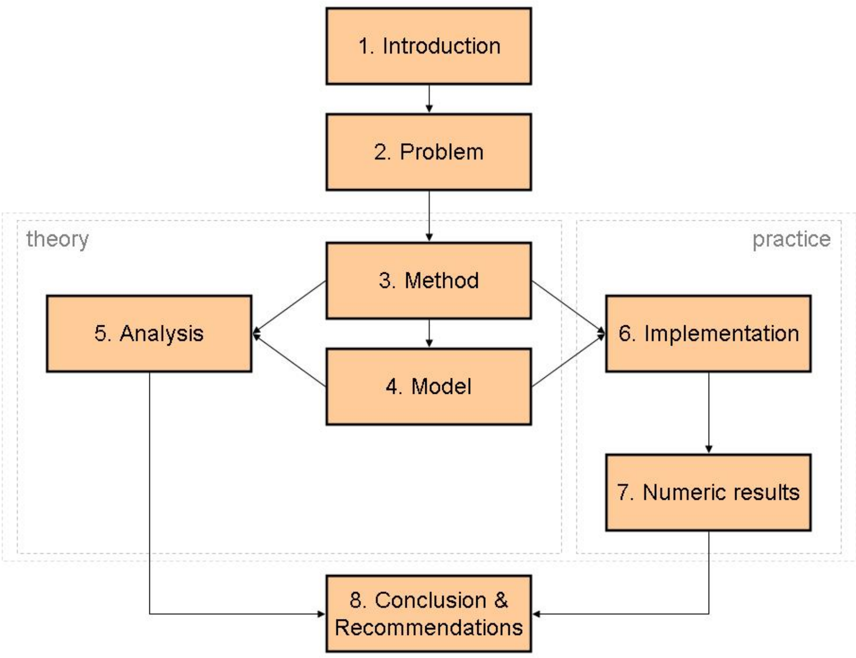


Figure 1: Section arrangement

1.1 Air traffic control

Air traffic control is a service provided by ground-based controllers who direct aircraft in the air and on the ground to ensure safe, orderly and efficient traffic flow. Control is offered by issuing clearances and directions to pilots, who are by law obligated to obey these instructions. Air traffic control services are provided throughout the majority of airspace over the Netherlands.

Air traffic control can be subdivided in sub-specialties: En-route control, Appoach control and Terminal control.

- En-route control is concerned with the handling of traffic in the upper layers of airspace; the overpassing traffic. Specialized Air Route Traffic Control Centers provide this service.
- Approach control is responsible for the separation of traffic on specified air routes departing from, arriving in and flying over an area below a certain flight level (usually 24.000 ft). It is concerned with the guidance of departing aircraft to the air routes and arriving aircraft from the air routes to the immediate vicinity of an airport. So-called TRACON (Terminal Radar Approach CONtrol) facilities provide these services.
- Terminal control is responsible for all traffic on the airport including airborne in- and outbound traffic in the immediate vicinity of the airport. This service is provided from the tower at an airport.

Eurocontrol at Maastricht provides the En-route control in the Netherlands. Approach and Terminal control at Schiphol airport are provided by LVNL respectively at Schiphol-Oost and the tower at Schiphol-Center. This report involves operational decisions for Terminal control.

1.2 LVNL

Luchtverkeersleiding Nederland (LVNL), or Air Traffic Control the Netherlands, is a non-profit organization that is engaged in the air traffic control of the Dutch airspace. Its main activity is to provide air traffic control services to civil air traffic in the Amsterdam Flight Information Region (FIR), the airspace the Netherlands is held responsible for. In a year approximately 400.000 flight movements are managed. About 1.100 flights are handled during a day, with up to 110 an hour during busy periods. Other activities include the design, acquiring and maintenance of infrastructure to provide air traffic control, the distribution of aeronautical information, training of controllers and internal supervision of Safety, Efficiency and Environment. All tasks have been laid down in the Aviation Act (Luchtvaart Wet).

Services provided by LVNL are based upon a dialogue with its stakeholders. These are parties in the aviation industry like airport Schiphol and airlines, but also local residents and the State department. A balance between their interests and demands are taken in account.

LVNL is an Independent Administrative Body ("Zelfstandig Bestuursorgaan") since 1993 and its main office is located at Schiphol-Oost.

An organogram of LVNL is shown in Figure 2, which depicts the structure of LVNL at directorate- and department-level. The Performance & Incident Analysis department (PIA), the issuer of this research project, is a sub-department of the Research & Development department (R&D). This department falls under the directorate Air Traffic Management (ATM), which is managed by the board of LVNL. A compete profile of LVNL is given in Appendix F, where its directorates, departments and their activities are described.

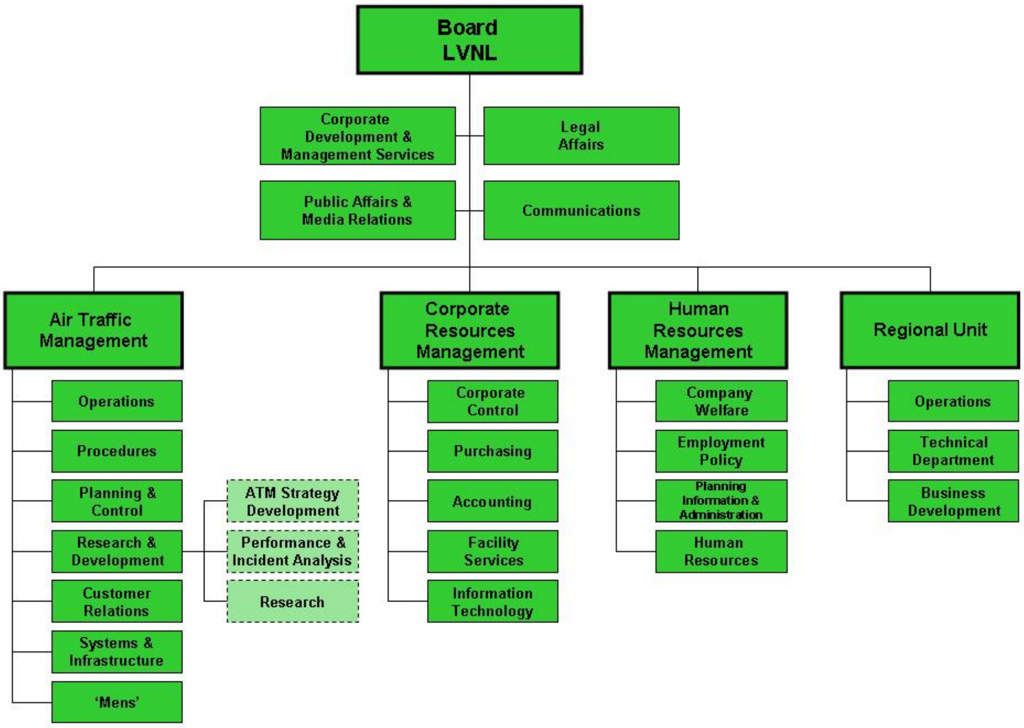


Figure 2: Organogram of LVNL

1.2.1 Schiphol

LVNL provides Terminal control for Schiphol. Amsterdam Airport Schiphol (AAS) is one of the five primary hubs in Europe, providing various forms of transportation and a wide range of services and facilities for its passengers. As a primary hub, it serves as a mainport, a junction in the national and international network of transportation of people, goods and services. It is generally considered an important drive for the Dutch economy.

LVNL supports the development of the mainport. Within the limiting conditions for safety aimed is at the optimal utilization of the airspace. The expansion of Schiphol through the years has lead to an increase in complaints about noise pollution in the surrounding area. This has lead to limiting conditions concerning noise load produced at Schiphol as well.

1.2.2 E^2MC

An example of the consultation with the stakeholders is the Environmental and Economic Management Committee (EEMC or E^2MC). This is a consultative body containing LVNL, AAS and the airlines operating from Schiphol. On a monthly basis E^2MC monitors the development of the environmental aspects noise load, pollution of the air, smell and third-party risk. It has the power to enforce a change in the operations at Schiphol when it is concluded that certain environmental limits might be exceeded. The consultation is designed to minimize the disruption of operations.

2 Problem

In this section, the problem is discussed. First, some background information is given on the elements that the situation consists of. The current situation is explained and its shortcomings will be discussed. Improvement possibilities are presented and the research objectives are formulated. The approach to tackle the problem is discussed at the end of this section.

2.1 Background

The main task of air traffic control is to prevent collisions between aircraft. With the increase of air traffic through the years a new task appeared: the efficient handling of traffic to ensure high capacity in the sky. All over the world, Safety and Efficiency are two aspects that characterize the services provided by air traffic control centers. In the Netherlands, also a third aspect is taken into account, namely: Environment.

With the introduction of the Polderbaan in 2003, a new Aviation Act was introduced, aiming at the growth of the mainport Schiphol on one hand and the protection of the environment on the other. The idea behind this law is that it bounds the produced noise and that within these bounds the air transport sector can optimize the Schiphol operation.

Schiphol has six runways at its disposal[1]. Each runway has a name and an alphanumeric code. This alphanumeric coding is used worldwide to designate runways. The runways of Schiphol and their designations can be found in Table 1. The numeric part represents the heading of the runway using a magnetic compass (the last number is skipped). The letter represents the relative position of parallel runways; R for Right, L for Left and C for Center. At Schiphol there are 3 parallel runways, as can be seen in Figure 3. Depending on the supply of inbound and outbound traffic, two runways are used for landing and one for take-off or vice versa. For a short moment of time, it is also possible that four runways are in use, during transitions between utilizations of runway combinations. With the runways available at Schiphol, many combinations at most three runways can be made to handle the traffic. Not all theoretically conceivable runway combinations are possible for utilization due to safety issues.

The choice for a runway combinations to handle traffic at a particular moment is limited in the first place by the weather. Secondly, preference lists for runway combinations are developed to stay within the environmental bounds imposed by the Aviation Act. Both will be discussed.

2.1.1 Weather

Current weather conditions limit the runway utilization. Wind- and visibility conditions are key factors in the choice of a runway combination. For wind, the choice is based on

[1]Runway 04 - 22 (Schipholbaan) is a relatively short runway and is mainly used for small aircraft and business jets; only during very specific weather conditions is it used for larger civil aircraft.

Runways at Schiphol			
Name	Designation		
Schipholbaan	04	-	22
Kaagbaan	06	-	24
Buitenveldertbaan	09	-	27
Aalsmeerbaan	18L	-	36R
Zwanenburgbaan	18C	-	36C
Polderbaan	18R	-	36L

Table 1: Runway designation at Schiphol

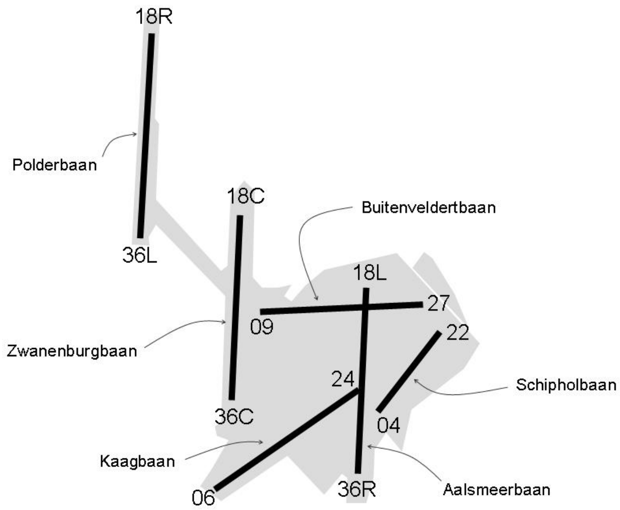
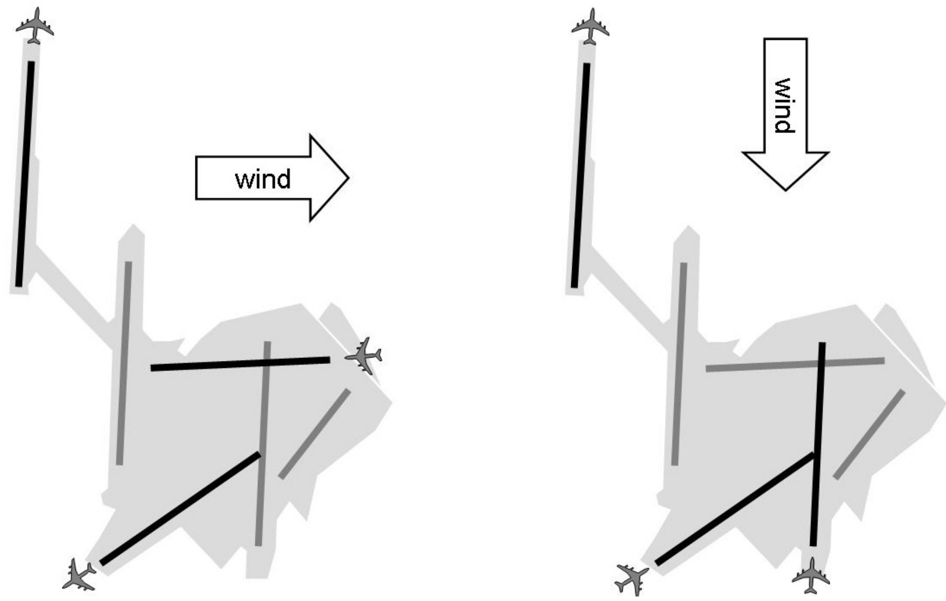


Figure 3: Runway designation at Schiphol

direction and speed. For visibility it is based on horizontal visibility and cloudbase[2] for vertical visibility. Other factors in the choice are the presence of thunderstorms, snow, severe gusts and fog, which can limit the use of runways significantly.

Wind Aircraft preferably take-off and land against the wind. Take-off and landing are, although limited, also possible with a crosswind. The amount of crosswind depends on many factors of current conditions and aircraft characteristics. An aircraft needs sufficient speed relative to the prevailing wind when accelerating for take-off. Against the wind, the amount of used runway is considerably smaller than with tail wind. Landings with tail wind or with strong crosswind are considered to be not safe. Aircraft would touch down with too much speed and possibly overshoot or be blown from the runway. Considering the safety, limits on crosswind and tailwinds exists at Schiphol, thus excluding runway combinations under certain wind conditions. For a runway combination that is in use, the wind conditions are within the limits for all utilized runways.

[2]The cloudbase is the height of the bottom of the cloud cover.



(a) Outbound peak, western wind (b) Inbound peak, nothern wind

Figure 4: Examples of utilization of runway combinations

For instance, when the wind is blowing moderately to the east, runways are in use that are headed west, as in Figure 4(a). When the wind is blowing to the south, runways are in use that are headed north, as in Figure 4(b).

Visibility Visibility conditions influence the choice of runway combinations and the capacity of Schiphol as well. For safety reasons, the distance between landing aircraft is increased when visibility drops. When two runways are in use for landing, the operation on one can influence the operation on the other; they are dependent. When this is the case, aircraft that break off their landing can end up in each other's flight paths. Runway combinations with dependency between runways can only be used when visibility conditions are good, while pilots and controllers are able to observe unsafe situations and perform actions to counter it. The use of two parallel runways for landing is independent.

2.1.2 Preference lists

The use of preference lists is a direct result of the environmental restrictions concerning noise load. A preference list is an ordered list of runway combinations. The order of this list determines what runway combination is preferred over other runway combinations. When more than one runway combination is possible under the current weather conditions, air traffic controllers must choose the runway combination with the highest position on the list. The preference lists are predefined and formulated by E^2MC.

At four briefings during the day, at which a meteorological assistant and representatives from parties in the E^2MC are present, the runway combination for the coming hours is determined.

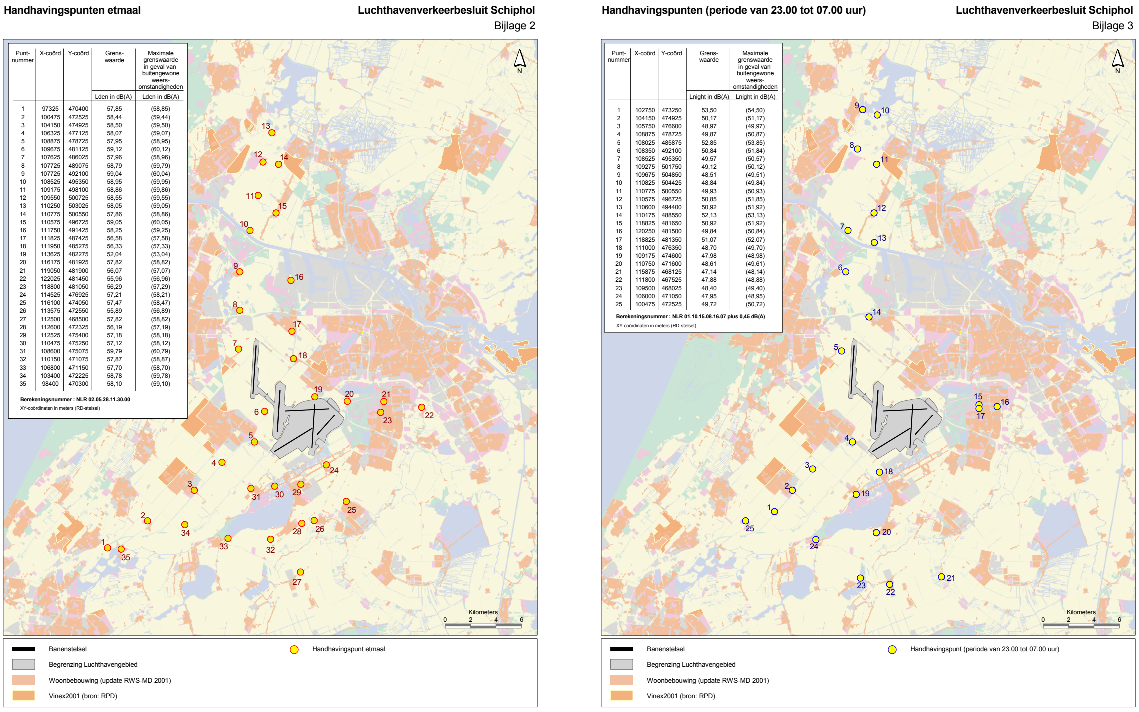
2.2 Noise load management

The Aviation Act lays down two standards with respect to the noise load produced. The first standard limits the total noise load per operational year. The other standard limits the noise load in a number of enforcement points surrounding Schiphol, also per operational year. The latter is discussed in this project. Both standards consist of two components, a maximum with respect to the noise load during the whole dag, the day-evening-night period, and a maximum with respect to the noise load during the night, the night period.

There are 60 enforcement point located around Schiphol at which noise is calculated and accumulated during the year; 35 for the day-evening-night period, and an additional 25 for the night period. The location of these points can be found in Figure 5. Each enforcement point has a standard that may not be exceeded at the end of the year. They can be seen as buckets in which noise load is accumulated during the year. Every arrival and departure at Schiphol contributes an amount to the buckets. None of the buckets may overflow. When a standard does exceed, the government executes measures that will restrict the amount of traffic at Schiphol further or financially penalize the sector. This will have a negative effect on the sector as a whole. Hence, it is of importance that noise loads in the enforcement points stay within their limits and exceedance is avoided.

To make sure the standards are not exceeded, the noise load development needs to be managed during the year by means of so called steering measures. This is carried out by the use of different preference lists that can be implemented at fixed moments during the year. By using the optimal preference list for given situations at these moments, the noise load in the enforcement points is managed, so that the exceedance probability remains as low as possible. This brings control over the Environment aspect into the operation at Schiphol.

For instance, when the noise load realization in enforcement points in the north are relatively high, these points must be relieved. Since an exceedance in a single enforcement point must be avoided, it is aspired to obtain balanced noise load realizations in all enforcement points, which will have a lower exceedance probability. By operating more to and from the south, a more balanced noise load is created, without shootouts in the north.



(a) Day Evening Night (b) Night

Figure 5: Enforcement points around Schiphol

Enforcement points that are in line with the Polderbaan (18R - 36L) or the Kaagbaan (06 - 24) have the highest standards, hence these runways offer the highest capacity numbers with the lowest noise load contribution. Runway combinations with these runways are therefore preferred and have high places in the preference lists.

2.3 Current situation

Currently the choice for a preference list is made at monthly meetings of the E^{2}MC. In advance, the preference list is calculated that is expected to perform best in the coming period. The calculation method used and the decision-making process during the year will be illustrated.

2.3.1 Calculation method

The calculation method used to determine the best preference list is based on the realized noise load so far and the expected noise load in the future. A representation of the year is used as can be found in Figure 6 (upper left). The year is divided into 12 decision epochs (12 months). At each epoch there are several preference list alternatives, one of which is chosen. Successive choices for preference lists form a path through the year representation. A genetic algorithm is used to determine the 'optimal path' through the remainder of the year based on the expected weather[3], hence based on the expected noise load realization. The exceedance probability forms the basis for optimality in this algorithm. A path is executed in each individual meteo year of data and it is determined if the noise load is exceeded. The exceedance probability then becomes:

$$\mathcal{P}_{exc} = \frac{\#\text{ meteo years in which a standard is exceeded using the path}}{\#\text{ meteo years on data}}$$

The algorithm determines if there is a better path, hence a path with a lower exceedance probability. When the algorithm is done, the path is found that minimizes the defined exceedance probability. The preference list for the first epoch in the path is the best choice for the first coming period.

2.3.2 Decision-making process

At the beginning of an operational year the realized noise load so far equals zero and the optimal path through the 12 coming epochs is determined. The preference list in the first epoch of this path is used for the operation at Schiphol.

At the end of an epoch a certain noise load has been realized in reality. This noise load realization is subtracted from the standards and a new optimal path is calculated for the remaining 11 epochs. The preference list in the first epoch of this new path is the best choice for the coming month. If this preference list is different from the preference list currently in use, this change is implemented: the steering measure.

This process is continued until the end of the year: at each epoch a new optimal path is determined based on the realization so far and the expected noise load realization of the future. An example of the first steps of the decision process is given in Figure 6.

2.4 Shortcomings

Through experience it is found every month a new optimal path is calculated that does not coincide with the previously calculated path. The exceedance probabilities are near 100% when calculations are done at the beginning of the process. Using the current method, the possibility of altering the path during the remainder of the year is not taken into account when calculating the exceedance probability. Since a path has direct influence on the exceedance probability, it is clear that altering this path does have effect. The exceedance probability is therefore overestimated and decisions are based on inaccurate values of it.

The reason why optimal paths calculated in different months do not coincide is because the weather during a month rarely behaves exactly as expected; it is stochastic. The assumption was made that the expected weather is correctly predicted by the meteo years on

[3]The expected weather is based on 30 years of meteo data ranging from 1971 till 2000.

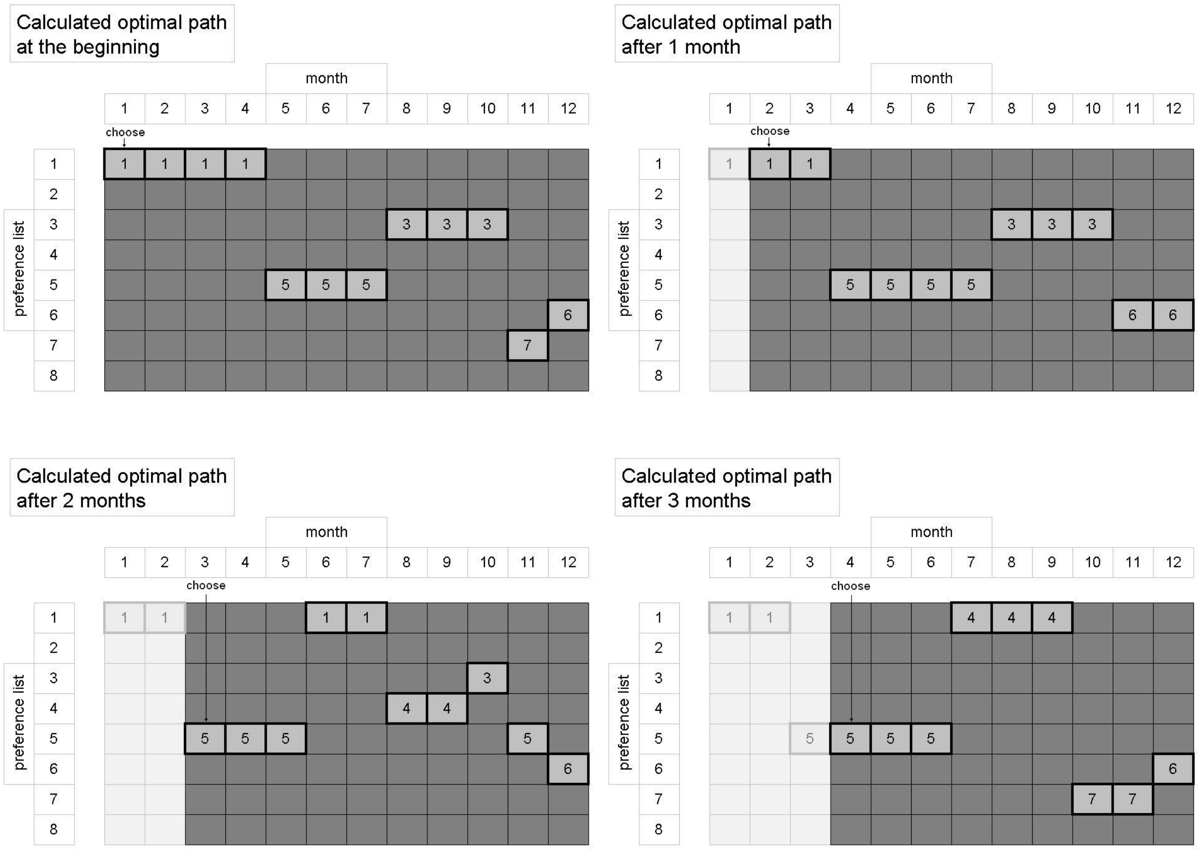


Figure 6: Decision process

record. The current method lacks the ability to base the exceedance probability on different realization possibilities during epochs; it is based on expected realizations only and therefore a deterministic method instead of a stochastic method.

Since the current method is little insightful on the basis for the choice, operational personnel can have difficulty understanding why a certain change in preference list needs to be implemented. This can lead to friction between operational and non-operational personnel. It is important that air traffic controllers back the proposed decisions, since they have to carry out the implementation.

2.5 Research

Possibilities for improvement for the current situation are researched in this project. First, the research objectives are formulated. The scope and the approach to this problem are discussed thereafter.

2.5.1 Objectives

The main goal of this project is to find and develop a new mathematical framework that can be used for decision-making in noise load management. A theoretical, insightful framework has to be found that models noise load management as it is currently in use and additionally takes in account stochastic realizations. Implicitly this means that a calculation method must be constructed that does not solely depend on the expected noise load realization, but also an on different realization possibilities.

Currently there are 12 decision moments at which a preference list is chosen. LVNL likes to know the effect on the exceedance probability if there are more such moments. The goal in this project is to draw conclusions on the effect on the exceedance probability when the number of decision moments is increased. Altering the decision-making may also have effect and a method can be proposed that has a positive effect on noise load management.

When implementing the new framework into a numeric calculation program, errors will be made due to choices that are made to limit the computation time. The aim is to conclude to what extend the implementation produced reliable outcomes and to what degree the new framework can be used as a tool for decision-making in the future.

In short, the three main questions in this project are:

- What mathematical theory applies to noise load management and takes its relevant characteristics into consideration? How can this theory be applied to noise load management to optimize the Schiphol operation within the environmental restrictions? What choices are made with the implementation?
- What is the effect of increasing the number of decision moments? Can the decision-making be altered with any positive effect? What is the influence on the exceedance probability?
- How large are the errors made when a numeric calculation method is implemented? What is the significant value of the output that is found?

2.5.2 Approach

Noise load management will be modeled so that a new method for decision-making can be applied. Since noise load management requires the making of multiple decisions this method will be in the lines of dynamic programming. Dynamic programming is a mathematically founded theory of which its input parameters can be altered as pleased, so that calculations to determine the answers to the other research objectives are able to be executed.

Relevant characteristics that have to be taken into consideration are the stochastic and dynamic nature of noise load management. Preceding this project, a literature review has been done on possible methods that could apply to noise load management. From the study a method was selected that fits noise load management and has the ability to answer the other research questions.

When modeling noise load management the standards on the noise load in the enforcement points are the only standards taken in account; these restrictions pose the largest effect on the operation and will be the pivot of this project. The total noise load per operational year is therefore not in the scope of this project. In addition, the operation during the day-evening-night period most defines the operation; the night period will not be considered.

The department R&D/PIA of LVNL uses the Safety Efficiency Environment framework (SEE framework). This framework ensues from the different quality aspects of the ATM system the multiple stakeholders of LVNL request. Due to the complexity of the ATM system trade-offs exist between safety, efficiency and environment. The research takes this framework into consideration. It will focus on the Environment branch of it and consider the other branches.

The large-scale implementation for numeric calculation of the new method is not within the scope of this project. However, to determine to the effect of different alterations to model and to give insight into the feasibility of large-scale implementation, sub-models are programmed in a numeric computing environment to obtain some numeric results. To this end parameters are constructed using data from LVNL on the current operation. Statements with respect to the exceedance probability can be given based on results.

The first step toward an increase in decision moments is a doubling of decision moments. This will be researched and implemented in this project. A theoretical analysis of this change and of a further increase is given in the form of a literature review. The input parameters of the implemented dynamic program are altered and the effect is observed in executed calculations. Also, an alteration to the decision-making is proposed that could have a positive effect on noise load management. It is implemented and its effect observed. Statements on the effect of doubling the decision moments and altering the decision-making will be given for the implemented models.

The errors made due to the numeric implementation are found by detailed theoretically analysis of the model. Results are applied to the models that were executed. They give insight into the significant value of the results obtained from the numeric calculation method.

Based on the theoretical analysis and the practical implementation conclusions are drawn on the utilization of the new method.

3 Method

In this section, the mathematical theory is discussed that was found to apply to noise load management and will be used for decision-making in this project. First, an overview is given of the literature search preceding the project, which provided the appropriate theory. Stochastic dynamic programming, the method chosen, is formulated and relevant aspects to this project are discussed. The reason why this method is chosen and the opportunities it offers are discussed thereafter.

3.1 Literature review

A literature study was done into several areas of mathematics on the relevance to noise load management. The study was concerned with giving an overview of techniques that could be used to tackle the noise load management situation. The reviewed topics were stochastic scheduling, effective bandwidth utilization and revenue management. The complete study can be read as a separate document [11]. The dynamic properties of the theory used in revenue management offered the most promising perspectives. It was found that stochastic dynamic programming applied to the situation best, since it has the appropriate characteristics in common with noise load management.

Further investigation into stochastic dynamic programming was performed to investigate its compatibility with noise load management and its opportunities. Findings will be discussed in Sections 3.3 and 3.4. Books that were consulted include [1], [7] and [14] which cover the material on dynamic programming. These books also briefly address the stochastic variant. This is discussed in more detail in [2] and [13], where many variants, algorithms and applications are discussed. These books can be consulted for further research in this area. The description of the mathematical theory is based on the appropriate parts of the theory from these books and some articles that all reference to them.

3.2 Stochastic dynamic programming

When there is a decision to be made, there are a number of different alternatives (actions) to choose from. Choosing the best action requires thinking about more than just the immediate effects of the actions. The immediate effects are easy to see, but the long-term effects are not always as clear. What usually makes the choice particularly difficult is that there is a lot of uncertainty about the future. The outcomes of certain actions in the future are not entirely predictable. Stochastic dynamic programming is a method that is used in problems so that the process of decision making in uncertain environments is automated. Many problems in daily life are stochastic and dynamic: Stochastic because the future cannot be predicted and dynamic because several decisions must be made during the period looked at. The theory of stochastic dynamic programming provides a method for making optimal decisions in situations with uncertainty and multiple decisions. These two characteristics are present in the noise load management setting.

Stochastic dynamic programming (SDP) exists in a number of varieties, among others:

- finite vs. infinite horizon
- discrete vs. continuous time
- discrete vs. continuous decision
- discrete vs. continuous state
- discounted vs. non-discounted rewards

The finite horizon non-discounted variant with discrete time and decision will be discussed, because these characteristics can be found in the noise load management setting, which will be discussed in Sections 3.3 and 4.1. The choice for discrete state was done on the basis of computational issues, which will be addressed in Sections 3.2.2 and 5.3. In the following section the continuous state version in discussed.

3.2.1 SDP with finite horizon

In this section a general description of the stochastic dynamic programming method with finite horizon problem is provided. Dynamic programming is a method for solving sequential decision problems using recursion. Optimization can be done in two ways: maximization and minimization. The minimization variant will be discussed, while ultimately noise load needs to be minimized. The goal of a stochastic minimization problem with finite horizon of N stages (thus with N decision to be made) is to minimize the expected sum of contributions over the whole planning horizon:

$$\min \mathbb{E}\left[\sum_{n=1}^{N} c_n(i_n, d_n)\right]$$

Here c_n is the contribution, i_n the state and d_n the decision made at stage n. To find this minimum an optimal value function and recursion relation is used of the form:

$$f_n(i) = \min_{d \in D_n(i)} \left[c_n(i,d) + \int f_{n+1}(j) \cdot dP_n(j|i,d)\right], \quad i \in S$$

in which

$f_n(i)$	=	Minimal expected sum of contributions in stages $n, n+1, \dots, N$ when in state i before the decision at stage n is made,
S	=	Continuous set of possible states (state space),
$D_n(i)$	=	Discrete set of possible decisions at state i in stage n (decision space),
$c_n(i,d)$	=	Single step contribution during stage n when choosing decision d in state i,
$dP_n(j\|i,d)$	=	Probability density for the transition from state i to state j under decision d.

In essence, stochastic dynamic programming splits the problem into smaller problems per decision moment. These are called the stages and are defined as discrete moments in time. Solutions to these subproblems are used to construct the minimum for the goal function by using the recursion relation.

It can be seen that $f_n(\cdot)$ is expressed in terms of $f_{n+1}(\cdot)$, hence it is called a recursion. Solving the recursion equation is done backwards. Values for $f_N(i), i \in S_N$ are given since at the end of the process no more decisions have to be made and values are known with certainty. Starting with $f_N(\cdot)$ successively values for $f_{N-1}(\cdot)$, $f_{N-2}(\cdot)$, ..., $f_0(\cdot)$ can be determined using the recursion. The goal function is build up stage by stage calculating these optimal values for the states. In the first stage there is only one state, while there are no multiple jumping-off points (the situation at the beginning is observed and is hence unique). The value for $f_0(\cdot)$ equals the optimum for the goal function. The value signifies the minimal expected result for the whole planning horizon given that optimal decisions are made.

When the value for $f_0(\cdot)$ is found also the optimal decision for the stage is found. Decisions at later stages cannot be determined while the states at those stages are not known in advance because of the stochastic transitions. However an optimal strategy $\pi = (\pi_1, \ldots, \pi_N)$ can be found that gives the optimal decision for all possible states $i \in S_n$ in stage n. The rule $\pi_n : S_n \rightarrow D_n$ is the decision rule for stage n.

A central concept in this method is the state. The state describes the characteristics of a situation the system is in. Knowledge of the current state provides all the necessary information for the future behavior of the system. All information of the past is represented in the state. In other words, it does not matter how the system ended up in a certain state, decisions are based on the current state only and not on the previously visited states. This is called the Markov property.

Decisions are made in every state (except in the last stage). Depending on the type of problem, a single step contribution might be incurred. Probabilities of going from state to state depend on the decision also. These are called transition probabilities. They represent for each state the distribution of going to states in the next stage.

3.2.2 From continuous to discrete state space

In spite of the growth in computing power multi-dimensional continuous-state dynamic programming problems are still challenging to solve or infeasible. The difficulty lies with the computation of the value functions: values are in the Banach space, which is more difficult to work with than the Euclidean space. A discrete approximation, or discretization, of the state space is necessary to solve numeric problems. The values are in the much better computable Euclidean space. The continuous state space is discretized into a finite number of grid points and a resulting finite-state dynamic program can be solved numerically. The optimal value function for points in the state space that are not on the defined grid points can be calculated using interpolation.

When the state space is discretized into a grid $\hat{S} = [0, \epsilon, 2\epsilon, 3\epsilon, \ldots]^k$, with ϵ the discretization increment of the grid that is chosen, the recursion relation will alter with it. Instead of integrating the recursion relation, it can now be considered as a sum of values over the grid points:

$$\hat{f}_n(i) = \min_{d \in D_n(i)} \left[c_n(i,d) + \sum_{j \in S_{n+1}} \hat{f}_{n+1}(j) \cdot \hat{P}_n(j|i,d) \right] \qquad i \in S_n \quad \forall n$$

with

$$\hat{P}_n(j|i,d) = \int_{j-\frac{\epsilon}{2}}^{j+\frac{\epsilon}{2}} dP_n(j|i,d)$$

The probability distribution is now a discrete one resulting in subsequent approximation of the optimal value function $\hat{f}_n(i)$. State space $S_n \in \hat{S}$ can be defined while there are only so much transition possibilities per stage above a certain lower limit of probability (the rest is cut off). Hence S_n is countable $\forall n$.

The appeal of the discrete method is that it reduces an infinite-instance problem (solving $f_n(i)$ for all $i \in \mathbb{R}^k$) to a finite-instance problem with a countable number of calculations. This is needed to obtain numeric results from the model. In addition, it is needed for the strategy iteration that is discussed in Section 3.2.3.

A discretization error occurs when implementing this approach. This will be discussed in Section 5.3.

A graphic example of the state and decision space and transitions is given in Figure 7. An overview of requisites for discrete time, decision and state stochastic dynamic programming is given:

	Overview	
Stages	n	Successive moments at which decisions are made: $n \in \{1, \ldots, N\}$
State space	S_n	Set of possible states in stage n
Decision space	$D_n(i)$	Set of possible decisions in stage n at state i
Singe step contribution	$c_n(i,d)$	Single step contribution during stage n when choosing decision d in state i
Transition probability	$\hat{P}_n(j\|i,d)$	Probability that decision d in state $i \in S_n$ leads to state $j \in S_{n+1}$
Recursion relation	$\hat{f}_n(i) =$	$\min_{d \in D_n(i)} \left[c_n(i,d) + \sum_{j \in S_{n+1}} \hat{f}_{n+1}(j) \cdot \hat{P}_n(j\|i,d) \right]$

Table 2: Requisites for stochastic dynamic programming

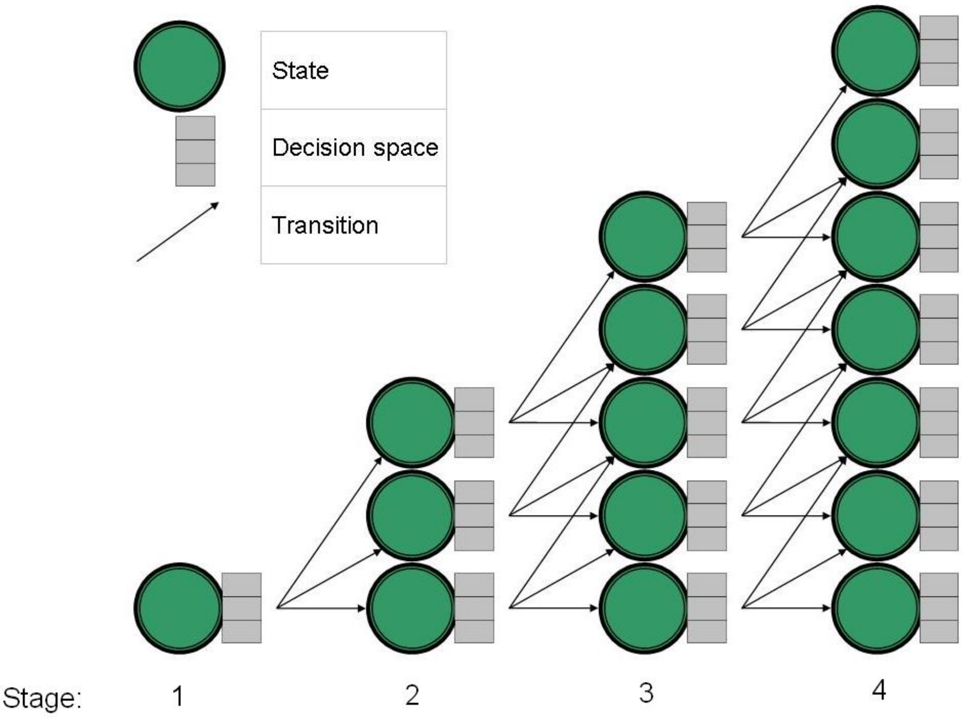


Figure 7: Graphic representation

3.2.3 Strategy iteration

The technique of strategy iteration can be used to develop a better strategy with respect to the goal function given a initial strategy. The strategy iteration algorithm manipulates the strategy directly, instead of finding it through the calculation of the optimal value function. It consists of two stages: Strategy evaluation and Strategy improvement. In the evaluation stage, the values for the optimal value function are calculated for a given strategy. In the improvement stage, the strategy is improved greedily where possible.

If an optimal strategy is already found, there is no use for this algorithm. It is used to calculate a better, and in little steps the optimal, strategy given a initial strategy. Using this algorithm the whole SDP network need not be calculated completely; the strategy improvement is found directly in less time than calculating the whole model. The stages are discussed in detail, with a slight alteration of notation for clarity:

Strategy evaluation Computing the value functions for all states given a certain strategy is done with backward recursion. Given a strategy π, that assigns an action $\pi(i)$ to each state $i \in S_n \quad \forall n$, the value functions can all be calculated. The structure of the process is completely known; it is in fact a linear system of size $\sum_{n=1}^{N} |S_n|$ with the same amount of unknowns (the $f_n(i), \quad i \in S_n \quad \forall n$), hence the calculation is straightforward.

Strategy improvement The evaluation is carried so that it can be compared to a different strategy that might be better. Suppose that for some state i it is of interest to know whether to deviate from the current strategy by choosing decision $a \neq \pi(i)$ or not. A way to determine this is to consider selecting decision a in this state i and thereafter following the existing policy π. This produces a different value function

$$g_n^{\pi}(i,a) = c_n(i,a) + \sum_{j \in S_{n+1}} f_{n+1}^{\pi}(j) \cdot P_n(j|i,a)$$

If strategy π is used all the time the value function is $f_n^{\pi}(i)$; if in state i decision a is chosen and thereafter strategy π is followed the value function is $g_n^{\pi}(i,a)$. These values can be compared; if the latter value is smaller (more toward an optimum) it is logical to choose a instead of $\pi(i)$. More general, if π and π' are strategies, such that for all $i \in S$

$$g_n^{\pi}(i, \pi'(i)) \leq f_n^{\pi}(i)$$

then policy π' must be at least as good as π. This is measured by the optimal value functions for all $i \in S$:

$$f_n^{\pi'}(i) \leq f_n^{\pi}(i)$$

This is called the policy improvement theorem.

Strategy iteration algorithm The method discussed above is described in the following algorithm:

- Choose strategy π'
- Loop
  - Assign $\pi = \pi'$
  - Compute the value functions of strategy π by solving the linear equations:
    $f_n^{\pi}(i) = c_n(i, \pi(i)) + \sum_{j \in S_{n+1}} f_{n+1}^{\pi}(j) \cdot P_n(j|i,d) \qquad \forall i \in S_n \quad \forall n$
  - Improve the strategy at each state:
    $\pi'(i) = \arg\max_{d \in D} \left[c_n(i, \pi(i)) + \sum_{j \in S_{n+1}} f_{n+1}^{\pi}(j) \cdot P_n(j|i,d) \right]$
- until $\pi = \pi'$

3.3 Correspondence with noise load management

Stochastic dynamic programming is a mathematical method for successive decision making under uncertainty. These two aspects are the essence of noise load management.

- Successive decision making: Choices for preference lists occur monthly during the year. Choices in consecutive months have effect on the noise load realization.
- Uncertainty: The weather cannot be predicted and is hence stochastic. There are a number of realization possibilities during a month; it is not certain which one will be realized.

SDP bases its decisions on different realization possibilities. By defining all noise load realization possibilities beforehand and basing choices and optimal values on their future realizations possibilities a balanced decision is made that is not solely based on the expected realization (as is done currently). Alteration to the path in the current method are taken into account, since, in essence, all possible paths are evaluated with SDP. The theory, as described, can be used to model noise load management.

The decision-making is based on the current noise load realization and not on how this realization had evolved. This is consistent the Markov property that is present in stochastic dynamic programming.

When a new year starts the accumulated noise load of the previous year is put to zero; a new year is started with a clean sheet. This corresponds with a finite horizon variant of SDP. All noise load realizations during a year weight equally in the accumulation of the noise load in the enforcement points. This corresponds to the non-discounted variant of SDP, that weighs all values in successive stage equally. The utilization of predefined preference list at fixed times during the year corresponds to discrete decision and discrete time variant of SDP.

3.4 Opportunities

Stochastic dynamic programming optimizes a goal function using a generally defined value function. It hence offers the opportunity to define the value function for different utilizations and the process is optimized accordingly. Most interesting in noise load management will be to determine the minimal realization and minimal exceedance probability. These variants will be discussed in Sections 4.2 and 4.3. Other variants may be considered and could be implemented in later projects.

By altering input parameters of SDP, the method offers the opportunity to investigate the effect of concepts for future operations. Possible effects relevant to this project to investigate can be the effect of . . .

. . . an increase in decision moments

. . . altering the optimal strategy

. . . changing the starting date of the aviation year

The increase in decision moments and the altering of the optimal strategy are within the scope of this project. By altering the number of stages, the effect on the value function can be analyzed compared to the original setting. This will be discussed in Section 4.5.2. The change of the starting date of the year in the aviation industry was not covered in this project.

For a given strategy, the value functions can be determined. The effect of altering the optimal strategy can hence be analyzed. Altering the optimal strategy can be done to achieve properties beside an (sub-)optimal goal function. In Section 4.5.3 a construction is proposed where the optimal strategy from one model is iterated in another model. The value function is analyzed on its effect; it will be sub-optimal in both models, but also have characteristics of both models.

There also exists a possibility to compare the current situation to proposed situation. The feasibility is the question, while the current situation is not dynamic and hence the different paths should be evaluated for different realizations. It will not be possible to perform all these calculations. Only an indication of improvement can be provided.

4 Model

Due to its corresponding characteristics it plausible that the stochastic dynamic programming method can be applied to noise load management. Noise load management is executed in real-life and needs to be model such that the method can be applied. Not all of its facets can be taken into account; choices will be made on the characterization of properties, which will be discussed in this section. The modeling is done based on the requisites for SDP. Later the SDP is formulated separately, using the model of noise load management, which is general enough to work with theoretically. Noise load management can be seen as an instance or application of the formulated SDP. Theoretical results of SDP can then by applied to noise load management.

4.1 Modeling the characteristics of noise load management

4.1.1 Enforcement points

In total, there exist 60 enforcement points for which noise load is calculated. For 35 enforcement points, noise is calculated during the day-evening-night period. Let EP_{den} be the set of these 35 enforcement points. During the night the noise load is calculated in 25 additional enforcement points on different locations than the ones in EP_{den}. Figure 5 shows the location of these points around Schiphol. Only the day-evening-night period is within the scope of this project, so EP_{den} is considered. A detailed map of these points is given in Appendix A.5.

Ideally all enforcement points are taken in account. A subset of enforcement points can be considered to model reality. The subset $EP \subseteq EP_{den}$ with $|EP| = k$ is considered to base the model upon. Disadvantage of this approach is that a subset may not represent reality; results in enforcement points that are not in the subset may develop uncontrollable. Therefore, the enforcement points in the subset must reflect reality as well as possible; they must be chosen with care, based on the experience of an expert. It will be necessary to perform checks to determine what the effect is in the enforcement points that are not modeled.

Noise load management is further modeled with the requisites for stochastic dynamic programming in mind.

4.1.2 Decision moments

In noise load management decisions are made at meetings of the E^2MC; these are the moments the possible change of preference list is determined.

Say the implementation of these decision made at an E^2MC-meeting takes place at the first day of the month. The decision is valid until the implementation of the decision of the next meeting. Define $Moments$ the set of dates at which decisions are implemented. With the current decision frequency (monthly) this set becomes[4]:

[4] A year in the aviation industry starts 1 November and ends 31 October

$$\begin{array}{rcllll} Moments & = & \{ \; 1 \text{ November}, & 1 \text{ December}, & \ldots, & 1 \text{ October} \; \} \\ & = & \{ \quad 0, & 1, & \ldots, & 11 \quad \} \end{array}$$

These dates can be shifted or its number adjusted as the decision frequency changes. The number of elements in $Moments$ is the number of decision moments, thus $N = |Moments|$. Currently there are 12 E^{2}MC meetings, hence $N = 12$. If decisions would be made on a half-monthly basis then $N = 24$. This N is the number of stages in the stochastic dynamic programming method.

4.1.3 Noise load realizations

Noise load realization in the enforcement points from the set EP are modeled by a k-dimensional vector with values that represent the noise load in each enforcement point. At LVNL, the noise load is characterized by the percentage of the standard that is realized. The standard, L, is a vector in $\mathbb{R}^k_+$. A realization of noise load so far is also a vector in $\mathbb{R}^k_+$; call it crl_ep, the cumulative realized load in enforcement point ep. A percentage of the standard can then be calculated for each enforcement point:

$$\frac{crl_{ep}}{L_{ep}} \cdot 100\% \quad \forall ep \in EP$$

Since all enforcement points are observed simultaneously a certain realization of noise load, rl, is characterized by the vector:

$$rl = \begin{bmatrix} \frac{crl_1}{L_1} \\ \frac{crl_2}{L_2} \\ \vdots \\ \frac{crl_k}{L_k} \end{bmatrix} \cdot 100\%$$

This way the current situation in the k modeled enforcement points can be characterized; it is used to characterize a state in SDP. The set of all possible states rl represents the state space S. Since $\frac{crl_{ep}}{L_{ep}}$ is in $\mathbb{R}_+$, there would be an infinite number of states, which makes numeric analysis impossible. The state space needs to be discretized.

Discretization A k-dimensional grid, or multigrid, can be laid over the space in $\mathbb{R}^k_+$. A grid block can be considered as a state; all possible realizations that fall into the grid block belong to the state. The size of the grid blocks depend on the size of the increments that were chosen. Increments can be chosen such that the model is accurate enough to represent the real world; the smaller the increments the more the model represents reality. Call the size of the increment $\epsilon \in \mathbb{R}_+$. The state is characterized by the middle values of the block it represents. State i is represented by a k-dimensional vector. Element $i_{ep} = m \cdot \epsilon \quad \forall ep$ with $m \in \mathbb{N}$ is the value that represents the location of the grid block on axis ep; it is the middle value on the ep-axis of the block. The boundaries of the block depend on the size of the increment that was chosen. A certain observed realization of noise load, rl, is in state i if all values are between the boundaries of block i :

$$rl \text{ is in state } i$$

$$\Leftrightarrow$$

$$i_{ep} - \frac{\epsilon}{2} \leq rl_{ep} < i_{ep} + \frac{\epsilon}{2} \quad \forall ep$$

The state space is then spanned by:

$$S = \left\{ i \in \mathbb{R}^k \;\middle|\; \begin{array}{l} i_{ep} = m \cdot \epsilon \\ m \in \mathbb{N} \\ \forall ep \in \{1, \ldots, k\} \end{array} \right\}$$

This characterization produces an infinite number of states while $\mathbb{N}$ is countably infinite. This will be the state space used in the stochastic dynamic programming method.

4.1.4 Decision alternatives

Decisions are made at each stage. Decisions are of the form of a choice for a preference. The preference lists are build up as follows:

Runway Let R be the set of runways that are used at Schiphol. This set contains an extra dummy element $\times$, describing a situation for which no runway is in use. The designation of runways and layout of Schiphol was discussed earlier and can be seen in Figure 3. The set of runways is

$$R = \left\{ \begin{array}{l} 04, 06, 09, 18L, 18C, 18R, \\ 22, 24, 27, 36R, 36C, 36L, \times \end{array} \right\}$$

Runway combination Let RC be the set of runway combinations that can be used at Schiphol at various points in time. An element of RC describes what runways are used for take-off and landing. Maximally there are four runways active during a day; the set R^4 is considered in which the first and second element describe the runways that are used for take-off and the third and fourth for landing. Typically 2 or 3 runways are in use; the dummy element in R is used when an operation does not take place during the utilization of the runway combination. Many runway combinations are not possible due to the layout (for instance, the dependencies of runways discussed in Section 2.1.1) and location of the airport, hence the used set $RC \subset R^4$.

A way of distinguishing between phases during a day is provided by the ATM system. The traffic inbound and outbound rate is not constant at Schiphol, but traffic arrives and departs in waves. Hence, there are several traffic modes: inbound-peak, outbound-peak, off-peak and night. A day consists of several inbound-, outbound- and off-peaks and starts and ends with the night-mode. For each mode, a runway combination can be chosen to handle the traffic. While there are four modes the traffic of a day can be handled by an element from the set RC^4. Different combinations of runways contribute differently in the enforcement points.

Preference list A preference list is an ordered set of elements from RC^4. The runway combination with the highest order-number is the most preferred combination, when using a certain preference list.

All possible preference lists are combined in a set of decisions D from which one is chosen per stage. Different preference lists will have different contributions in the enforcement points; one of these preference lists will be optimal to choose. The set of preference lists that are used, are defined in the "Plan of Operations" formulated by the E^2MC and can be found in Appendix A.4. These 8 are the preference lists possible for usage during the year. For a general idea, preference lists 1 to 4 contribute more in the northern enforcement points, preference lists 5 to 8 more in the southern ones. The decision space that will be used in the stochastic dynamic programming method is described as follows:

$$D = \{\text{preference list } 1, \ldots, \text{preference list } 8\}$$

4.1.5 Probability of noise load contributions

Transitions between states are represented by the probability of their occurrence. The probability of going from the current noise load realization $i \in S_n$ to any $j \in S_{n+1}$ is the probability of making noise load contribution $x = j - i$. Hence, the notation $\hat{P}_n(j|i,d)$ or $\hat{P}_n(x,d)$ is used, since, given a certain decision d, a noise load contribution is independent from state i.

It is assumed that contributions have a multivariate normal distribution. The contributions depend on the weather, whose quantitative phenomena are frequently modeled with the normal distribution. Since underlying causes of the weather are often unknown, the use of the normal distribution is justified by the Central Limiting Theorem[5]; many small effects added together have a normal distribution.

Since the state space is discrete, discrete probabilities are needed for the transitions between states. However, the multivariate normal distribution is a continuous distribution. By integrating the continuous distribution over the appropriate interval, discrete probabilities are found.

Multivariate normal distribution In general, a random vector $X = [X_1, \ldots, X_k]^T$ follows a multivariate normal distribution if it satisfies the following condition. There is a random vector $Z = [Z_1, \ldots, Z_l]^T$, whose elements are independent standard normal variables with $\mu \in \mathbb{R}^k$ and $A \in \mathbb{R}^{k \times l}$ a matrix such that $X = AZ + \mu$. The vector μ is the expected value of X and $\Sigma = AA^T \in \mathbb{R}^{k \times k}$ a symmetric, positive definite (co)variance matrix of the

[5]Central Limiting Theorem (idea): if the sum of variables has a finite variance, then it will be approximately normally distributed. Since many natural processes have distributions with finite variance, the normal distribution is used frequently for natural phenomena.

components of X. Say

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix}, \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_k \end{bmatrix} \text{ and } \Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \dots & \sigma_{1k} \\ \sigma_{21} & \sigma_2^2 & & \sigma_{2k} \\ \vdots & & \ddots & \vdots \\ \sigma_{k1} & \sigma_{k2} & \dots & \sigma_k^2 \end{bmatrix}$$

Here μ_i is the expected value and σ_i^2 the variance of variable X_i and σ_{ij} the covariance between variables X_i and X_j. The probability density function is then described by

$$f_X(x_1, \dots, x_k) = \frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$$

with $|\Sigma|$ the determinant of Σ. The cumulative density function $F_X(a, b)$ is defined as the probability that all values in a random vector X are between the values of vectors a and b, both elements from $\mathbb{R}^k$. It is found by integrating the probability density function:

$$F_X(a, b) = \int_{a_1}^{b_1} \cdots \int_{a_k}^{b_k} f_X(x_1, \dots, x_k) d(x_1, \cdots, x_k)$$

With this distribution, the mean, variances and covariances of the contributions x in the k modeled enforcement points is taken into account. The mean (μ) describes the expected noise load contribution for all k enforcement points. The variance (σ^2) indicates how far from the expected contributions the real contributions typically are for one of the k enforcement points. The covariance (σ_{ij}) is the measure of how much two variables vary together; the covariance becomes more positive for each pair of values which differ from their mean in the same direction, and becomes more negative with each pair of values which differ from their mean in opposite directions. It describes the correlation between any two enforcement points.

Based on these parameters the main characteristics of a noise load contribution are modeled. It is therefore a good distribution to model the probabilities. Parameters μ and Σ are based on observations of noise load, based on the weather of the meteorological years on record.

Much is already known about normal distribution, which makes it easy to implement. However, values for the integration of the probability density function in multiple dimension needs to be estimated, since no closed form expression exists for more than 3 dimensions. Often a Monte Carlo method is used to estimate its value.

The multivariate normal distribution for k variates is written in the following notation:

$$X \cong N_k(\mu, \Sigma)$$

Discrete probabilities The transition probability of making contribution x can easily be found when the cumulative density function is known. The increment, ϵ, that were chosen to form the grid for states, are the region over which to integrate the density function, as was discussed in Section 3.2.2. Hence, the discrete probability of contribution x equals (for any decision):

$$\begin{aligned} \hat{P}(x) &= \int_{j-\frac{\epsilon}{2}}^{j+\frac{\epsilon}{2}} dP(x) \\ &= F_X\left(x-\tfrac{\epsilon}{2}, x+\tfrac{\epsilon}{2}\right) \end{aligned}$$

4.1.6 Remaining requisites for SDP

This section models the remaining requisites for stochastic dynamic programming. They are the single step contribution and the initial values for the recursion relation. The recursion relation of SDP can be formulated in two different ways: realization-based and probability-based. They serve a different goal. The realization-based model gives a measure of the amount of exceedance. The probability-based model gives the expected exceedance probability.

Realization-based In this approach, the optimal value function for a certain state is defined as the expected amount of noise load above the standard, given an optimal strategy is used in the remaining period. The goal of this approach is to minimize the amount of exceedance. The single step contribution in the recursion relation in the realization-based model is characterized by:

$$c_n(i,d) = \sum_{j \in S_{n+1}} \left[\sum_{\{ep|j_{ep}>l_{n+1}\}} (j_{ep} - i_{ep})^\alpha \right] \cdot \hat{P}_n(j|i,d)$$

with l_{n+1} the bound at which it is certain that a remaining path from a state under this bound will never become exceeding any more. The α-term is a constant that can be chosen to evaluate different aspect of the problem. If $\alpha = 1$ the real amount of exceedance is calculated. If $\alpha \neq 1$ then the value function will not represent the true amount of exceedance, but a measure for the exceedance. With this α, the severity of certain situation can be indicated[6]. The contribution term represents the expected (measure of) noise load contribution when in state i making decision d in stage n. The contribution is 0 when it is certain that the path from state i can be argued never to be exceeding. The process adds all contributions when exceedance is still possible from states, resulting in a term for the expected exceedance at the beginning of the year. This corresponds with the calculation of $f_0(0)$.

At the last stage, there is no contribution any more, since it is the end of the year:

$$\hat{f}_N(i) = 0 \quad \forall i \in S_N$$

[6]For instance when $\alpha = 2$ a doubling of exceedance is four times as bad as the original exceedance.

Probability-based The goal of this approach is to minimize the exceedance probability. In the probability-based model the optimal value function is defined differently. It now represents values for probabilities of ending up in specified states at the end of the process. There is no single step contribution, but the other properties remain the same. The optimal value function for a certain state represents the probability of ending up in an exceeding state at the end of the process, given an optimal strategy is used in the remaining period.

The recursion relation is build up as follows. The process stops after N steps and while no decision has to be made anymore, it can be said with certainty that:

$$\hat{f}_N(i) = \begin{cases} 1 & \text{if } i \text{ is exceeding} \\ 0 & \text{if } i \text{ is NOT exceeding} \end{cases}$$

The probability of ending up in a undesired state at the end of the process is known, because it can be observed. For $1 \leq n < N$ the recursion relation is defined as:

$$\hat{f}_n(i) = \min_{d \in D} \left[\sum_{j \in S} \hat{f}_{n+1}(j) \cdot \hat{P}_n(j|i,d) \right]$$

This way the optimal value function $\hat{f}_n(i)$ represents the value for the probability of ending up in an exceeding state at the end of the process when in state i at decision moment n given that an optimal strategy is used. Using the backward recursion $f_0(0)$ is found which represents the probability of ending up in an exceeding state when at the beginning of the year given an optimal strategy is used, hence:

$$\mathcal{P}_{exc} = f_0(0)$$

4.2 Formulation of SDP: Realization-based

The goal of a stochastic minimization problem with finite horizon of N periods (thus with N decisions to be made) is to minimize the expected sum of contributions over the whole planning horizon:

$$\min \mathbb{E}\left[\sum_{n=1}^{N} c_n(i_n, d_n)\right]$$

Here c_n is the contribution, i_n the state and d_n the decision made at stage n. The optimal value function represents the amount of remaining expected exceeding noise load at the end of the year given that an optimal strategy is used in the remaining period. Its recursion is of the form

$$\hat{f}_n(i) = \min_{d \in D}\left[c_n(i,d) + \sum_{j \in S_{n+1}} \hat{f}_{n+1}(j) \cdot \hat{P}_n(j|i,d)\right], \quad i \in S_n$$

in which

$$\begin{aligned}
S_n &= \left\{ i \in \mathbb{R}^k \;\middle|\; \begin{array}{l} i_{ep} = m \cdot \epsilon \\ m \in \mathbb{N} \\ \forall ep \in \{1, \ldots, k\} \end{array} \right\} \\
D &= \{ \text{ preference list } 1, \ldots, \text{ preference list } 8 \} \\
c_n(i,d) &= \textstyle\sum_{j \in S_{n+1}} \left[\sum_{\{ep | j_{ep} > l_{n+1}\}} (j_{ep} - i_{ep})^{\alpha}\right] \cdot \hat{P}_n(j|i,d) \\
\hat{P}_n(j|i,d) &= F_X(j - i - \tfrac{\epsilon}{2}, j - i + \tfrac{\epsilon}{2}), \quad X \cong N_k(\mu(n,d), \Sigma(n,d))
\end{aligned}$$

4.3 Formulation of SDP: Probability-based

The goal of a stochastic minimization problems with finite horizon of N periods (thus with N decision are to be made) is to minimize the expected sum of contributions over the whole planning horizon. While there are no single step contributions, this method results in the expected exceedance probability for the whole year: $f_0(0)$.

The optimal value function represents the optimal value function represents the probability of ending up in an exceeding state at the end of the process when in a certain state given that an optimal strategy is used in the remaining period. Its recursion is described by:

$$\hat{f}_n(i) = \min_{d \in D}\left[\sum_{j \in S_{n+1}} \hat{f}_{n+1}(j) \cdot \hat{P}_n(j|i,d)\right], \quad i \in S$$

in which

$$S_n = \left\{ i \in \mathbb{R}^k \,\middle|\, \begin{array}{l} i_{ep} = m \cdot \epsilon \\ m \in \mathbb{N} \\ \forall ep \in \{1, \ldots, k\} \end{array} \right\}$$

$$D = \{ \text{ preference list 1, } \ldots \text{ , preference list 8 } \}$$

$$\hat{P}_n(j|i,d) = F_X(j - i - \tfrac{\epsilon}{2}, j - i + \tfrac{\epsilon}{2}), \quad X \cong N_k(\mu(n,d), \Sigma(n,d))$$

4.4 Assumptions

Modeling noise load management certain assumptions were made. It is a mathematical model and it will differ from the real world. In order to stay aware of any differences between the model and reality, the model assumptions are clearly defined:

- The standards in the enforcement points during the day-evening-night period are the only enforced environmental restrictions
- The chosen subset of enforcement points represents all enforcement points
- There eight more preference lists
- Noise load realizations and contributions are discrete
- Noise load contributions have a multivariate normal distribution
- The meteo years can be used to predict all weather possibilities

4.5 Application and form of results

In this section the applications of the method to answer the research objective questions will be explained. A short description and substantiation is given of the calculation methods and the results that are obtained.

4.5.1 Exceedance probability

When executing stochastic dynamic programming, a linear system of size $\sum_{n=1}^{N} |S_n|$ with the same amount of unknowns is solved. The goal is to find $f_0(0)$, which is a straight forward calculation.

For the current situation, the model is executed for $N = 12$. During the calculation, results for optimal value functions and decisions are stored for each state in the state space and are used in successive calculation steps. All the decisions together form the used strategy for the calculated situation. When the calculations are done the exceedance probability and the optimal strategy are found.

4.5.2 Effect of an increase in decision moments

One of the research objectives is to investigate the effect of an increase in decision moments on the exceedance probability. To this end, a situation is examined where there are twice as many decision moments; hence, going from $N = 12$ to $N = 24$. The characterization of the state space will remain the same, but the states per decision moment alter. In addition, the transition probabilities change.

Strategy iteration is used to find the strategy for $N = 24$. The optimal strategy found for $N = 12$ is modified into an arbitrary strategy for $N = 24$. This is done by assigning the decision from $N = 12$ to the even moments in $N = 24$ and assigning intermediary states decisions that are in line with the ones on the even moments. This constructed strategy is (most probably) not optimal. It is used as the initial strategy in the strategy iteration algorithm. The algorithm will not need many iteration steps to converge to an optimum; by construction the strategy already makes similar choices for similar situations. For each iteration step, the optimal value functions and decisions for the current strategy and are stored again. When the algorithm is done the optimum is found and the exceedance probability can be given for $N = 24$. It is compared to the value found for $N = 12$. This process is graphically represented in Figure 9(a).

4.5.3 Altering the strategy

Another research objective was to determine if an alteratinf of the decision-making has any positive effect on noise load management. To this end an alteration to the optimal strategy is proposed.

The probability-based SDP determines the optimal strategy, π^p, that minimizes the exceedance probability. Still, when this probability is very small the actual exceedance can be very high. For instance, there is an enforcement point where the noise load contribution is extremely high when a certain type of rare weather occurs. Especially because an exceedance in a single enforcement point must be avoided, this is not a very stable situation; when a few rare events occur, the probability-based optimal strategy for the exceedance probability is not very good.

In this case, a more evenly distributed noise load would be preferred. The realization-based SDP determines the optimal strategy, π^r, that minimizes the amount of exceedance; it implicitly forms balanced amounts of exceedance. This method also has its drawbacks; the minimal amount of exceedance might have (and most probably has) a very high probability of occurring.

A composition of the two methods is considered. An optimal strategy can be found with one method. This strategy can be altered slightly by few strategy iteration steps of the other method, so that the altered strategy possesses characteristics from both methods.

Since LVNL is mainly interested in a low exceedance probability, the probability-based model is executed initially. Few strategy iteration steps are performed with the realization method, which yields a strategy π^{pr} for which by definition:

$$f_n^{\pi^{pr}}(i) \geq f_n^{\pi^{p}}(i) \quad \forall i$$

Therefore, the exceedance probability of the new strategy will be higher. An example of possible noise load realizations is given in Figure 8 With all the assumption and jitter in the model it could well be that even when all expected noise load realization exceed the limit slightly, the real realization will not. Since the probability-based strategy can produce a large expected realization, it will probably exceed no matter what. The altered strategy is hence more robust than the original. This process is depicted in Figure 9(b).

With a slight increase in exceedance probability a more robust strategy is found. Exceedance probabilities and expected realizations are found and it can be compared to what extend they differ.

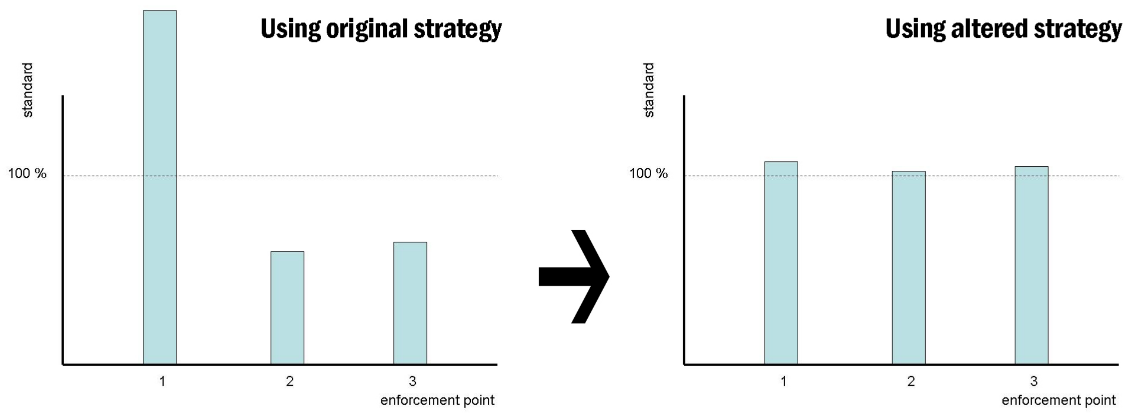
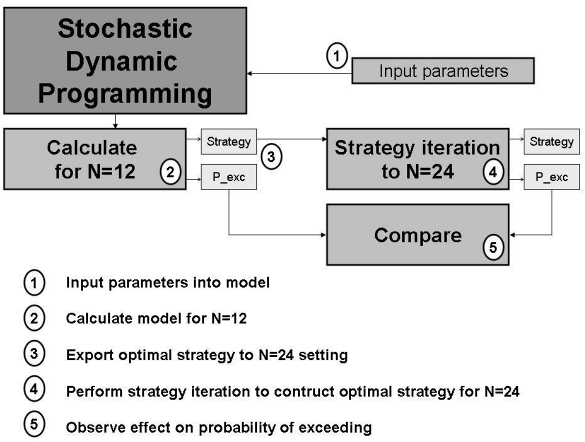
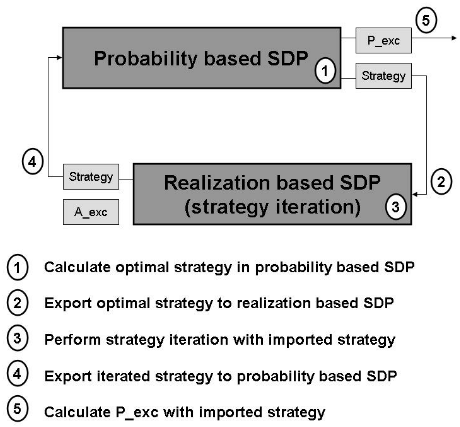


Figure 8: Realisations with different strategies



(a) Effect of doubling the decision moments



(b) Altering the strategy

Figure 9: Applications of methods

5 Analysis

In this section, an analysis is given of different features of the model. First, the extend to which the model represents reality is examined. When not all of reality is modeled, certain checks must be performed to ensure the results in the not modeled part stay within bounds. These checks will be discussed. The effect of an increase in decision moments is discussed theoretically. An overview is given of methods and results already described in literature. The expected result is discussed for noise load management. Due to the discretization of the state space a discretization error occurs. This error is described and analyzed at the end of this section.

5.1 Checks

The subset of enforcement points chosen in Section 4.1 must approximate the effect on all enforcement points as well as possible. It is still important to know what happens in the other enforcement points. To make sure results have not grown beyond the standard in the enforcement points that were not modeled, a check is performed. The expected noise load realization in all 35 enforcement points is analyzed and it can be determined if one (or more) exceeds its standard at the end of the year. The calculation of this check is presented in this section.

This check is performed after the optimal strategy, π is found for a certain model. Since all decisions are known for all states in all stages, all transition probabilities between states are also known. Therefore, the probability of being in a state during a stage can be determined for all states in a stage. Different paths through the network can end up in a certain state. Define a path from the origin i_0 to state i_n in stage n as $(i_0 \rightarrow i_1 \rightarrow \ldots \rightarrow i_n)$. The probability of being in state i_n during stage n is then as follows:

$$p(i_n) = \sum_{\substack{\text{all paths} \\ \text{to } i_n}} \prod_{m=1}^{n} P_m(i_m | i_{m-1}, \pi(i_m))$$

Probabilities of states in a stage that perform the same decision can be added; the probability of making a certain decision in a certain stage is found. Choosing preference list pl in stage n has a probability:

$$P_{n,pl} = \sum_{i \in S_n} \mathbb{1}\left(\pi(i_n) = pl\right) \cdot p(i)$$

Other requisites are the mean noise load realization values for all 35 enforcement points. This is a vector $\mu \in \mathbb{R}^{35}$ of mean values per enforcement point. The calculation from data is discussed in Appendix A.3 with $k = 35$. These expected noise loads can be determined for all months and all preference lists, hence the notation $\mu_{n,pl}$ is used for the mean noise load realization in month n using preference list pl.

Summing all mean values $\mu_{n,pl}$ multiplied by their accompanying probability $P_{n,pl}$ the expected value for the total mean noise load realization is found:

$$\mathbb{E}(NL) = \sum_{n=1}^{N} \sum_{pl=1}^{8} \mu_{n,pl} \cdot P_{n,pl}$$

It can be analyed if any of the enforcement points exceeds its standard (exceeds 100%). For the points that were modeled this will most probably not be the case (the exceedance probability would equal 1), but the values in the remaining points may or may not exceed. Analyzing these points it can be determined to what degree the model with the used subset of enforcement points represents reality.

5.2 Effect of an increase in decision moments in literature

As the number of decisions grows, theoretically and intuitively, it is impossible that the exceedance probability increases using SDP. A situation with less decision moments can be optimized and a strategy be found. The strategy can be modified into a strategy for a situation with more decision moments. Theoretically, this modified strategy could be optimal, but this will rarely be the case, since the modified strategy is guessed and not mathematically determined with the method. The modified strategy is most probably sub-optimal. This means that the optimal strategy with less decision moments is sub-optimal in a situation with more decision moments; the accompanying exceedance probably is not optimal also. When optimizing the situation with more decision moments a lower exceedance probability will thus be found. Hence, in the worst-case scenario the exceedance probability remains the same, otherwise the exceedance probability will decrease. This is also discussed in [12] where theorems are presented which indicate that the value function is non-increasing in the horizon length N [7]:

> **Lemma 2. (Monotonicity of N-period-to-go value function in N).** *The value function $V^N(\cdot)$ of the N-period-to-go problem is non-decreasing in the number of periods to go N. For $q \geq \alpha$, it is strictly increasing in N with*
>
> $$V^N(q) > V^{N-1}(q) + \beta^{N-1}(2f_0(q) - 1)$$

This is however a (further) truncation of an infinite horizon horizon problem; transition probabilities are not altered when increasing N, as is the case in noise load management. Approximating techniques are discussed in [4]. The topic of increasing the horizon length is

[7] More accurately: the article determines the value function is non-decreasing in N with maximization as optimality criteria. Hence, in the setting of this project, the value function is non-increasing while here the optimality criterion is minimization. Also, β must be chosen 1, while this is the discount factor.

discussed. The conditions under which this can be executed and accompanying shortcomings are discussed.

Statements on the convergence and the extent of the decrease are of interest. The ultimate situation that can be reached is a decision process in a continuous time setting. In noise load management this means that decisions for the use of preference lists can be implemented all the time instead of after one of the E^2MC-meetings. Increasing the number of decision moments effectively converses the problem from a discrete problem to a continuous problem as $N \to \infty$. It converses the finite horizon variant to an infinite horizon variant of the problem. This is discussed in [3]. It is shown that a continuous time value is always smaller than the discrete time value. The discrete time value converges to the optimal continuous time value as the epoch between decisions tends to zero. This is shown in a theorem on the convergence:

Theorem 4.1, iii) (idea). *Let $S_N^{(t)}$ be the set of initial states for which, for a receding horizon problem, a solution can be approximated then $S_N^{(t)}$ converges to S as $N \to \infty$ and $t \to 0$. Meaning that forall $\epsilon > 0$ there exist N and t such that for $\tau < t$ and $N^* > N$ there is*

$$(1 - \epsilon)S \subset S_N^{(t)} \subset S$$

Furthermore, an approach is presented that detaches the calculation of transition probabilities from the number of decisions. The calculation of the transition probabilities is based on a different time interval than the epoch between decisions. This approach is applied in receding horizon problems, which are also discussed in [10] and [5]. These articles presents an interesting starting point for future research in this area. When this approach is implemented, the obtained exceedance probability for the continuous time case can be compared to the current exceedance probability. The quotient represents the maximum improvement in exceedance probability; all steps toward a continuous time model (increase of N) will have effects smaller than that quotient. Unfortunately, no numeric results were given that coincides with the setting of this project. Concluding it can be said that an increase in decision moments decreases the value functions and it converges to some level, which was not determined.

Increasing the number of decision moments will also have a decreasing effect on the discretization error; the values become more accurate. This is discussed in [8]. Here theorems are formulated on the order of magnitude of the error as a function of the horizon length. However, it was also shown that calculations for certain accuracy might not have bounded planning horizons. Hence, in the discussed setting of this project increasing the number of decision moments will results in the approach of a lower bound for the maximum of the discretization error.

Increasing the number of decision moments also has drawbacks. While the decisions in noise load management are still made by man and not based solely on mathematics in the near future, there exists a cognitive effect to the increase of decision moments. In [9] it is discussed how errors are made in decision making processes. Dynamic consistency exists when the planned (optimal) strategy is executed. It was found that the dynamic consistency decreases rapidly when the horizon length increases. The deviation from the optimal strategy in this project leads to an increase in exceedance probability. This effect may negate the positive effect of the increase in decision moments on the exceedance probability. This is, however, outside the scope of this project.

5.3 Discretization error

The discrete-state process approximates the continuous-state process. When discretized, the process suffers from inevitable loss of information and its resulting errors. Therefore, the conversion results in a discretization error. In the following, the discretization error will be discussed and measured as well as possible.

First, the local error of each state is determined when calculating its optimal value. The local error is defined as the error that is made in the value function with one transition. An upper bound to this error is determined. For further calculation a refined estimate of the local error is used, which is discussed thereafter. Then the global error of each state is determined, which is the accumulation of errors in previous calculations.

5.3.1 Upper bound to local error

The notation in this section is as follows. The following norm is used:

$$||f_n|| = \max_{i \in S_n} |f_n(i)|$$

The state space is divided into a grid, with grid blocks $S_{x,n}$, so that $\cup_x S_{x,n} = S_n$ and $\cap_x S_{x,n} = \oslash$. The optimization term in the backward recursion in the continuous state process is (the real value):

$$g_n(i,d) = \int_{S_{n+1}} f_{n+1}(j) \cdot dP_n(j|i,d)$$

The optimization term in the backward recursion in discrete state process is (the approximate value):

$$\bar{g}_n(i,d) = \sum_x f_{n+1}(j_x) \cdot \hat{P}_{x,n}(j|i,d)$$

with j_x the state that represents grid block $S_{n,x}$; the middle point.

The discretization error in a certain state is the difference between the optimal value of the continuous-state process and the value found by the discrete-state process. It can be formulated by:

$$\bar{f}_n(i) - f_n(i) \quad = \quad \min_d \bar{g}_n(i,d) - \min g_n(i,d) > 0$$

An upper limit to this term is constructed, as this represents the worst-case realization possible and gives insight into the maximum size of the error.

Theorem: Upperbound to local error 1. *An upperbound to the error term for states i in stage n in the discrete SDP defined in Section 3.2.2 is given by:*

$$\bar{f}_n(i) - f_n(i) \leq 2 \cdot \max_d \left(\bar{g}_n(i,d) - g_n(i,d)\right)$$

Proof: The proof of this theorem can be found in Appendix B.1

A rougher upperbound is found if the following term is observed:

$$\zeta_n^{\text{upper}} = ||\bar{f}_n - f_n|| = \max_{i \in S_n} \left(\bar{f}_n(i) - f_n(i)\right) \leq 2 \cdot \max_{i \in S_n} \max_d \left(\bar{g}_n(i,d) - g_n(i,d)\right)$$

This ζ_n^{upper} represents the worst error that can be made in stage n. For this an expression for $g_n(i,d) - \bar{g}_n(i,d)$ is needed. This is provided by the following theorem.

Theorem: Upperbound to local error 2. *If f_{n+1} is Lipschitz-continuous with constant M $\forall n$ then $\bar{g}_n(i,d) - g_n(i,d) \leq M \cdot \epsilon$*

Proof: The proof of this theorem can be found in Appendix B.2.

The error term ζ_n^{upper} can now be given as:

$$\zeta_n^{\text{upper}} \leq 2 \cdot \max_{i \in S_n} \max_d \left(M \cdot \epsilon\right) = 2\epsilon \cdot M$$

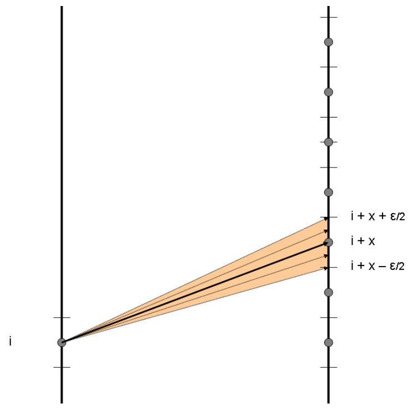
This constant M is based on the calculated value functions. This means that the discretization error can only be calculated after the calculation of the model. Beforehand no comment can be made on the discretization error.

5.3.2 Mean local error

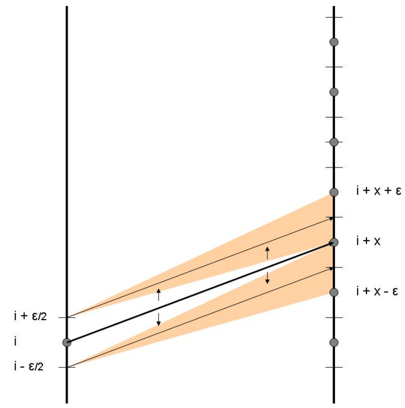
Erroneous decisions can be made within a discretization interval, which yields an error in the value function. The upper bound to this error was discussed above. A refined estimate of this error is derived in this section.

In the upper bound case, the value function is analyzed for all other decisions. It is determined which decision is worst and what the difference in value function is when this decision is applied. However, there are only so many 'logical' decisions that are made within the bounds of the discretization interval. The sets of states that are reached from states in a discretization interval will be similar. It is plausible that decisions based on similar sets are confined to a subset of decisions that have similar effect on the value function. These decisions are most probably made, when the optimal decision is not chosen. Hence, this approach focuses on the construction of a set $\hat{D} \subset D$, which contains the decisions that is most probably made within the bounds of the discretization interval.

The subset $\hat{D}$ for state i is constructed as follows. In essence, the interval is analyzed in all point for all variations of the transition possibilities. A transition to another realization is never exactly as the discrete transition prescribes; realizations fall within a discretization interval in the next stage, see Figure 10(a) for an example of one particular transition. In addition, the realization that is started from will never be exactly the real realization. Hence, there are several starting points within the interval, see Figure 10(b). This holds for all transitions from a certain state.



(a) For one transition



(b) From an interval for one transition

Figure 10: Realization possibilities

The set $\hat{D}$ is given by:

$$\hat{D} = \left\{ d \;\middle|\; \begin{array}{l} \sum_x P(x,d) f(i^* + x^*) \text{ minimal} \\ \\ \forall x^*_{ep} \in \left[x_{ep} - \frac{\epsilon}{2}, x_{ep} + \frac{\epsilon}{2}\right] \\ \forall i^*_{ep} \in \left[i_{ep} - \frac{\epsilon}{2}, i_{ep} + \frac{\epsilon}{2}\right] \\ \forall ep \end{array} \right\}$$

Remember that the backward recursion was constructed like this:

$$f_n(i) = \min_d \int_{S_{n+1}} f_{n+1}(j) dP(j|i,d)$$

By definition this is an expected value, hence it can be rewritten (notation) as

$$f_n(i) = \mathbb{E}_d f_{n+1}(X + i)$$

For the calculation of the discretization error the value functions when making a decisions from $\hat{D}$ are compared to the optimal value function (the optimal decision). The largest difference in value function is chosen as the local discretization error (with d^* the optimal decision in state i):

$$\zeta_n^{\text{mean}}(i) = \max_{d \in \hat{D}} \left[\mathbb{E}_d f_{n+1}(X + i) - \mathbb{E}_{d^*} f_{n+1}(X + i)\right]$$

The local error for a whole stage is determined by using the following norm:

$$||\zeta_n^{\text{mean}}|| = \frac{\sum_{i \in S_n} |\zeta_n^{\text{mean}}(i)|}{|S_n|}$$

Using this norm ζ_n represent the mean local error for states in stage n.

5.3.3 Construction of global error

The global error ξ_n in $f_n(i)$ is an accumulation of all local errors. This is made clear in the following theorem:

> **Theorem: Construction of global error.** *The error in $f_n(i)$ in the discrete SDP defined in Section 3.2.2 is calculated by $\xi_n = \sum_{i=n}^{N-1} \zeta_i$.*

Proof: The proof of this theorem can be found in Appendix B.3.

Hence, the global error in the value function in stage 0 is:

$$\xi_0 = \sum_{i=0}^{N-1} \zeta_i$$

This result can be used to cumulate the upper bound of the local errors or the mean local errors. Using the expressions found the upperbound for the global error for the whole planning horizon is:

$$\xi_0^{\text{upper}} \leq 2\epsilon \cdot N \cdot M$$

The upper bound of the global error will most probably be very large.

The mean global error gives insight into the magnitude of the real error that is made. While the calculation of the mean global error is very time consuming, the global error is estimated by the worst errors that are made in the stages: the local error in the last and one-to-last stage. While the difference between values of the optimal value function in adjacent states are most severe at the end of the process (either 0 or 1), the error that is made when going to the wrong state is thus also most severe. As the stages progress backwardly in time the values for the optimal value function flatten out and errors of going to the wrong state decrease; the two most severe local errors that are made are found in the last and one-to-last stage. The local error of the one-to-last stage is of the same magnitude as (but probably smaller than) the local error in previous stages. The mean global error is estimated as follows:

$$\xi_0^{\text{mean}} \approx \zeta_{N-1} + 11 \cdot \zeta_{N-2} \geq \sum_{i=0}^{N-1} \zeta_i$$

The discretization errors represent the deviation from the real exceeding probability given a situation where constantly wrong decisions are made. The real exceeding probability is therefore lower than the calculated probability. An interval is constructed for which it is certain that it contains the real exceeding probability; its lower bound is therefore the calculated value minus the error:

$$\text{interval containing real } \mathcal{P}_{exc} = \{f_0(0) - \xi_0, f_0(0)_{exc}\}$$

6 Implementation

The model of noise load management was implemented in the numeric computing environment Matlab. A computing program was written that executes stochastic dynamic programming on the model. In this section the modeling of the characteristics and choices for the implementation for numerical computation are discussed first. For clarification graphical representations are given for one dimension (hence for one enforcement point). The execution of the different applications is discussed thereafter.

6.1 Modeling for numeric calculation

Requisites for stochastic dynamic programming that can be used as they were modeled, are the Stages and the Decision space; these need not be specified further then was already done in Section 4. Other requisites need to be specified further for implementation for numeric calculation are discussed hereafter.

6.1.1 Selection of enforcement points

The stochastic dynamic programming method grows exponentially in computation time when adding new dimensions. Due to this characteristic, also known as the 'curse of dimensionality', it is not possible to model all enforcement points. A subset of enforcement points is constructed for which the model will be executed. These are chosen strategically, so that they represent all the enforcement points as well as possible.

To achieve this, looked is at the layout of Schiphol. The points are in different directions relative to Schiphol. If points were all chosen in the same direction, it is most likely that preference lists are chosen that contributes little in this single direction, which will not yield a realistic decision-making. To prevent this from happening, as many of the directions as possible need to be covered by the modeled enforcement points.

The enforcement point in a certain direction that experiences the largest effect on the noise load is chosen to represent that direction. In reality these points have also proved to be often at critical levels at the end of the year. This way a selection of enforcement points can be constructed that covers all the directions and is representative for all other states.

6.1.2 State space

The process of reducing the state space is illustrated. The state space is reduced as much as possible to reduce computation time. The state space is characterized by the value ϵ. It describes the increment that is used to make the continuous state space (as in Figure 11(a)) into a discrete one (as in Figure 11(b)). Hence,

$$S = \left\{ i \in \mathbb{R}^k \;\middle|\; \begin{array}{l} i_{ep} = m \cdot \epsilon \\ m \in \mathbb{N} \\ ep \in \{1, \ldots, k\} \end{array} \right\}$$

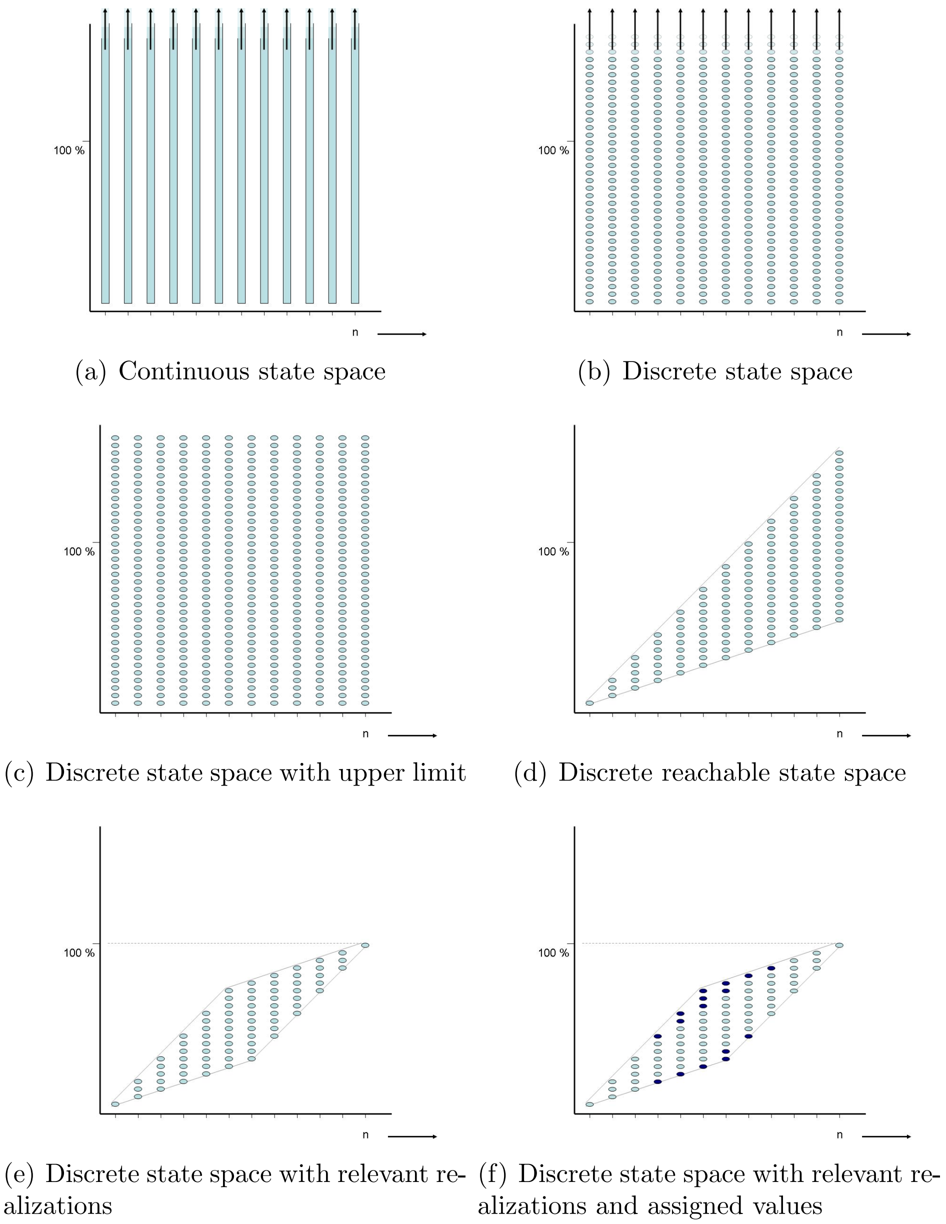


Figure 11: Construction of state space for numeric computing

The optimal value function cannot be calculated for all states in the state space, since it is still infinitely large. A selection of the relevant portion of the state space is considered. The value $t_2 \cdot \epsilon$ describes maximum realization possible for a single stage. Effectively the upper part of the realization possibilities, for which it is plausible the probability of being there is extremely small, is cut off. Since there are 12 phases to consider, the state that represents the largest realization is found at $(12 \cdot t_2)$ with realized load value $(12 \cdot t_2) \cdot \epsilon$. This is the upper limit of the realizations; all realizations during the year, including the end, are below this upper limit, hence the state space encompasses all possible realizations. The state space that is generated (as in Figure 11(c)) now is:

$$S = \left\{ i \in \mathbb{R}^k \;\middle|\; \begin{array}{l} i_{ep} = m \cdot \epsilon \\ 12 \cdot t_2 \geq m \in \mathbb{N} \\ ep \in \{1, \dots, k\} \end{array} \right\}$$

In addition, the lower part of the realization possibilities can be cut off, resulting in a smallest contribution per stage of t_1 and a largest of t_2. Hence, a structure like the one in Figure 11(d) is constructed. Let $t = t_2 - t_1 + 1$ the number of possible transitions per enforcement point; Section 6.1.3 discussed how t_1 and t_2 need to be chosen. These are all the reachable states from the origin. This state space is used for numerical calculations of the realization-based model.

Since in the probability-based model, for particular states it can be said if they will become exceeding (or not) with certainty, optimal value functions of these states need not be calculated. These states need not be part of the state space that is used in a numeric computing program. It can be argued what value these states have (0 or 1), without calculation. Since there are t realizations per enforcement point per phase it can be reasoned what the upper- u_n en lower-limits l_n are of the realizations are with optimal values strictly between 0 and 1. Per phase this 'reduced state space' now becomes (as in Figure 11(e)):

$$l_n = \max\left(t_1 \cdot n, \left(\frac{100}{\epsilon}\right) - t_2 \cdot (N - n)\right)$$

$$u_n = \min\left(t_2 \cdot n, \left(\frac{100}{\epsilon}\right) - t_1 \cdot (N - n)\right)$$

$$S_n = \left\{ i \in \mathbb{R}^k \;\middle|\; \begin{array}{l} i_{ep} = m \cdot \epsilon \\ m \in \mathbb{N} \\ l_n \leq m \leq u_n \\ ep \in \{1, \dots, k\} \end{array} \right\}$$

In this reduced state space there are a lot of states of which in advance it can be said what value they will probably have. These states will not have to be calculated, which saves time. When all states reachable from the a certain state have optimal values larger than 0.95, the state is near the border of the considered state space. It is most likely it will continue on its course and become exceeding, since the probability of returning to a state with an optimal value less than 0.95 is very small. Vice versa with optimal values less than 0.05. States with these properties are assigned optimal values respectively 1 and 0 in advance and calculated. The inaccuracy that is generated using this approach is small and justified by the fact that the contribution of these states to the value function is also very small; there is a low probability of being in states that lay on the boundary for any state, which can be can be calculated with a the $p_(i_n)$ from Section 5.1. Figure 11(f) shows the states that are assigned the values in advance.

This leaves a reasonably reduced state space for numeric computing. For all the states the optimal value needs to be calculated. For states that are reached outside the reduces state space the values can be given. Therefore calculations based on all reachable states are feasible. Note that all figures in Figure 11 are a 1 dimensional simplified examples of state spaces.

6.1.3 Transition probabilities

The probability of going from state to state is given by the probability of the accompaying noise load contribution. A matrix of all contribution probabilities is determined in a few steps. In Section 4.1.5 it was determined that noise load contributions have a multivariate normal distribution. The parameters of this distribution are estimated using data from LVNL; this is step one.

The discrete probabilities are found by integrating the continuous distribution over the increment of the discretization. Using a piece of Matlab code (given in Appendix C.1 and based on article [6]) a Monte Carlo method is implemented that estimates the cumulative distribution function; this is step two.

Step one: Estimating the parameters The parameters of the model have to estimated using data from LVNL. To define vectors μ and matrices Σ for any preference list *pl* and epoch n the program DAISY is used. DAISY is a web-based airport environment toolkit developed by Frontier. It consists of a set of interrelated modules that help study the impact of air traffic in and around airports. DAISY provides insight into the effect of changes to the operations and procedures on the environment.

One of the applications of DAISY is to produce values for the noise load in enforcement points given a preference list, a period of a certain meteo year (one of 30) and a supply of traffic. This way DAISY is used to generate noise load observations. For a certain control cycle a prognosis of the traffic supply can be made based on findings of earlier years. All requisites are present to perform the calculations. Obtained values can be considered as observations of noise load. A description of the noise load calculation and more specifications are given in Appendix A.2.

Input for DAISY consist of among other thing, the location of runways, airways and enforcement points. All these settings for input are used as they are currently in use for calculations. Relevant characteristics of input to this project are as follows:

- Periods: The epoch length is set to half a month.

- Compass rose: The 8 preference lists are translated into a distribution rule over the runways depending on the weather conditions.
- Traffic Prognosis: The prognosis for 2006 is used. This consists of 436.731 specific flight movements.
- Noise: Noise load contribution factors[8] of 2005 are used.

With this input the year 2006 is simulated. The method for calculating the parameters of the multivariate normal distribution from observation data can be found in Appendix A.3.

With these settings DAISY produces observations for half-monthly noise load realizations. Therefore the distribution with the constructed parameters gives probabilities for half-monthly noise load realizations.

Step two: Calculating probabilities There is a fixed number of possible transitions, namely the number of possible contributions $(t_2-t_1+1)^k$. The transition probability matrix is hence finite. While it concerns a continuous distributions the tails, which are cut off at a certain point, contain a small probability mass. Choosing t_1, t_2 and ϵ appropriately, it is ensured that over 95% of the probability mass is integrated for all desision d and stage n; all relevant possible realizations are characterized. Hence, t_1,t_2 and ϵ are chosen such that

$$\int_{a_1}^{b_1} \cdots \int_{a_k}^{b_k} f_X(x_1, \ldots, x_k) d(x_1, \cdots, x_k) \geq 0.95$$

with

$$a_{ep} = t_1 \cdot \epsilon - \frac{\epsilon}{2}, \quad b_{ep} = t_2 \cdot \epsilon + \frac{\epsilon}{2} \qquad \forall ep$$
$$X \cong N(\mu(n,d), \Sigma(n,d) \qquad \forall n, d$$

Probabilities for all contributions are calculated using the Monte Carlo method for integrating multivariate normal distributions. In Figure 12 the situation and the terms t_1, t_2 and ϵ can be seen graphically.

Probabilities are calculated for all realization possibilities x, for all epochs n, for all decisions d and are put into an array $P_{N=24}(x_1, \ldots, x_k, n, d)$ accordingly:

$$P_{N=24}(x_1, \ldots, x_k, n, d) = F_X\left(x - \frac{\epsilon}{2}, x + \frac{\epsilon}{2}\right)$$

with

$$x = \left\{ \begin{array}{l} t_1 \cdot \epsilon \leq x_{ep} \leq t_2 \cdot \epsilon \\ x_{ep} = m \cdot \epsilon, \quad m \in \mathbb{N} \\ ep = \{1, \ldots, k\} \end{array} \right\} \quad \text{and} \quad X \cong N_k(\mu(n,d), \Sigma(n,d))$$

The sum of probabilities for a certain stage and decision must sum to 1 for it to be a proper probability function. Since the integrated area was already put at at least 95%, the effect of normalization is small. The calculated probabilities are normalized so that:

[8]Noise load contribution factors determine how a certain event contributes in the enforcement points

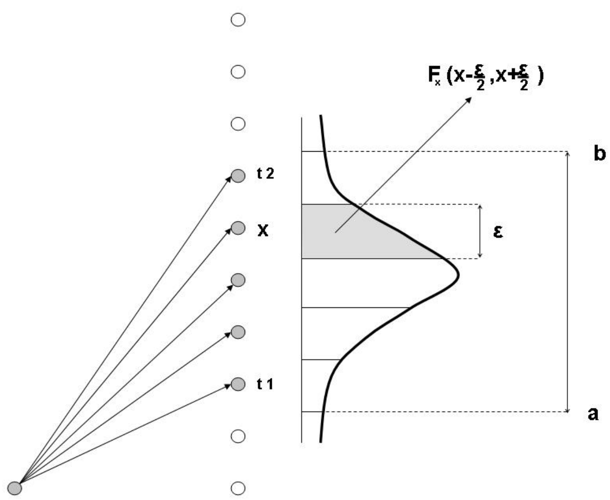


Figure 12: Probability of realization x in 1 dimension

$$\sum_{x_1=t_1}^{t_2} \cdots \sum_{x_n=t_1}^{t_2} P_{N=24}(x_1, \ldots, x_k, n, d) = 1, \qquad \forall n, d$$

The realizations are based on half months, since DAISY produced half-monthly observations. To obtain the monthly transition probabilities the two halves of the month need to be evaluated. A monthly transition consist of two half-monthly transitions, as is depicted in Figure 13. If in the second half for all states the same decisions are made as in the first month the situation represents a monthly transition for that decision. Transition probabilities for both halves can be calculated and summed up appropriately:

$$P_{N=12}(x, n, d) = \sum_{\{(x_1, x_2) | x_1 + x_2 = x\}} P_{N=24}(x_1, n, d) \cdot P_{N=24}(x_2, n+1, d)$$

All transition probabilities can be calculated and a transition matrix is constructed for 12 decision moments. These arrays correspond with the $P_n(j|i, d)$ (or $P_n(x, d)$ with $x = j - i$) that was discussed in the theory.

6.1.4 Recursion relation

The set of all reachable states was given in Figure 11(d). Optimal values and decisions are determined for the state space shown in Figure 11(f). When calculating the recursion relation for a state in a certain stage, first it must be determined what the values are of states that can be reached in the next stage. If this set contains states that are not present in the reduced state space their values were not recorded; it needs to be argued what their value is. In the probability-based model, their value is either 0 or 1.

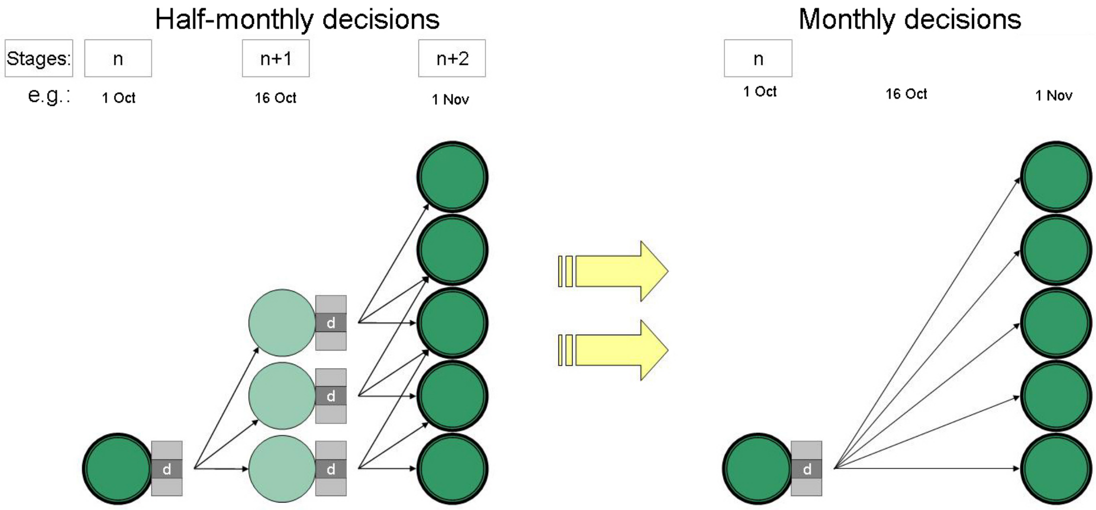


Figure 13: Construction of monthly transition probabilities

The recursion relation is the pivot of the stochastic dynamic programming method. It is hence the most relevant part of the code in the implemented calculation method in Matlab; it is given in Appendix C.2.

6.2 Execution

The concept for calculation for the different aspects was discussed in Section 4.5. The execution of the calculations will be discussed shortly. The discrete model was executed using Matlab on a 1.7 GHz computer. Computation time was the limiting factor when the calculations were made. Therefore, a model with four enforcement points ($k = 4$) was only calculated to give an indication of the exceedance probability. Models with three enforcement points ($k = 3$) were calculated to investigate the other aspects.

For values

$$\epsilon = 0.02$$

$$t_1 = 2$$

$$t_2 = 6$$

it was calculated that they satisfy the posed condition in Section 6.1.3. This means that the discretization interval is 2% and transition probabilities range from 3% to 13% (with 5 representative states per enforcement points), hence 125 transition possibilities per month. With this the following calculations were performed:

6.2.1 Exceedance probability

The exceedance probability is calculated with the above defined requisites for SDP for 12 decision moments ($N = 12$). This was done for the set of four enforcement points {9,19,21,31}.

Calculations for the increase in decision moments and altering the strategy, and the checks and discretization error were not performed due to calculation time issues.

For three enforcement points this was done for different sets of enforcement points containing points 5, 9, 18, 19, 21, 22, 25, 31. These points cover the different directions relative to Schiphol, as can be seen in Figure 14. In this project the model was executed for 8 different sets, given in Table 3 and depicted Figure 15. Optimal decisions and optimal values for all states were calculated and resulted in exceedance probabilities for all sets. The set of enforcement points with the highest exceedance probability estimates the real exceedance probability for all enforcement points best. If there was an addition to this set, the exceedance probability would never become lower; all observes exceedance probabilities are below the real exceedance probability. The largest exceeding probability is therefore closest to the real exceedance probability.

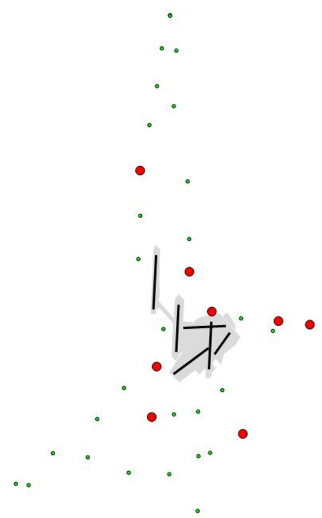


Figure 14: Enforcement points in different directions

nr.	enforcement points		
1.	5	19	21
2.	5	21	25
3.	9	19	25
4.	9	22	31
5.	18	19	21
6.	18	19	25
7.	18	19	31
8.	19	21	31

Table 3: Sets of enforcement points

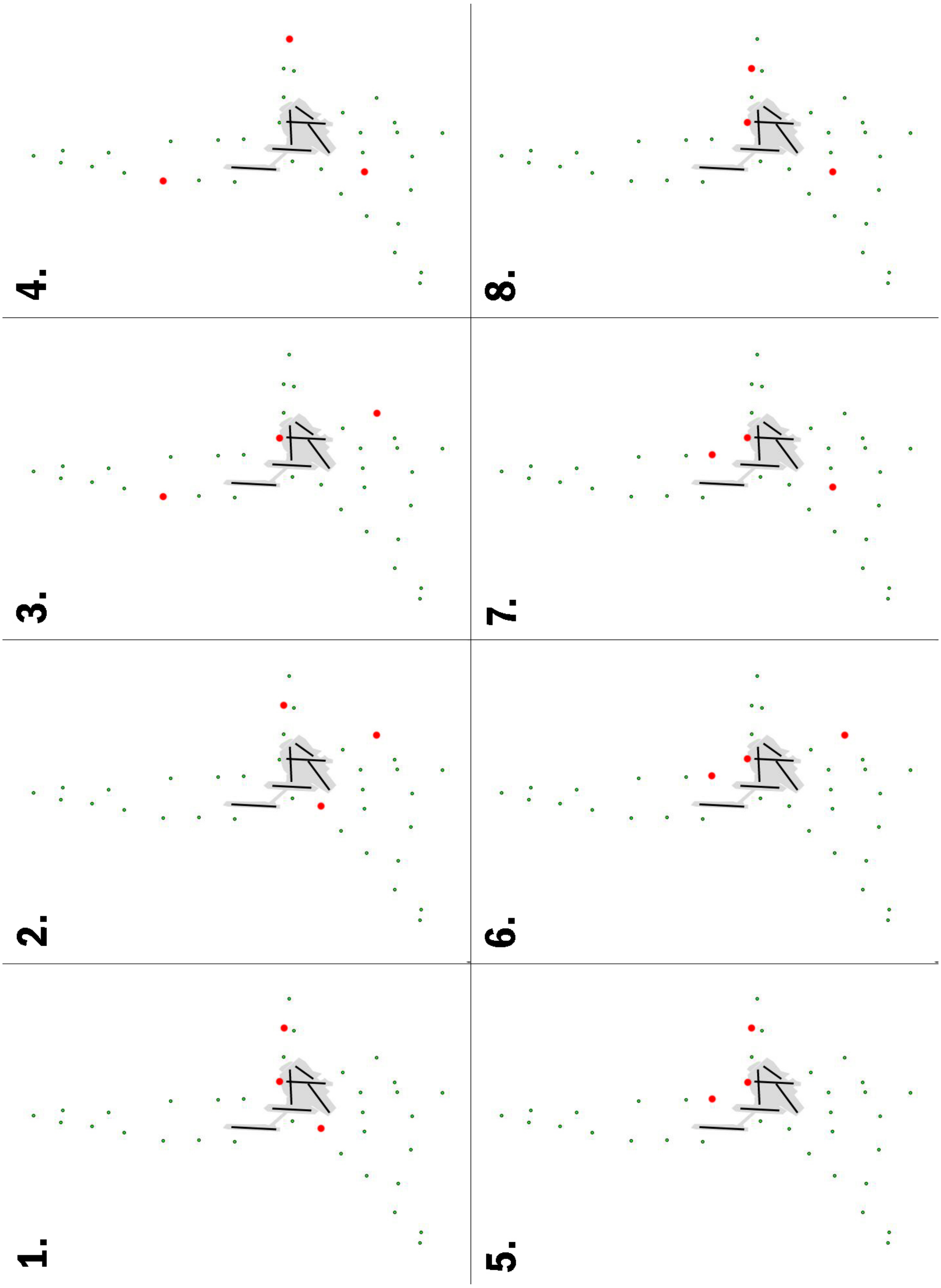


Figure 15: Location of the enforcement points

6.2.2 Effect of an increase in decision moments

For the 8 sets of enforcement points, strategy iteration was performed on the optimal strategy found for $N = 12$. This resulted in a new strategy and new optimal values for all states, hence also a new exceedance probability. The improvement factor is calculated, which indicates the decrease in exceedance probability:

$$\frac{\mathcal{P}_{exc,old} - \mathcal{P}_{exc,old}}{\mathcal{P}_{exc,old}}$$

6.2.3 Altering the strategy

The optimal strategy for $N = 12$ was altered by strategy iteration of the realization based-model. It is then determined what the accompanying exceedance probability is of this new strategy. This was performed for sets 4,7 and 8.

6.2.4 Discretization error

The discretization error was calculated for 6 of the 8 sets of enforcement points; crucial data for the calculation of the error was not stored in the other 2 cases.

The upper bound to the local error is calculated as was based on findingsa in Section 5.3.1. A value for the Lipschitz-constant M needs te be determined. It can be approximated by evaluating the calculated value functions at all pairs of discrete states in the domain and determining the maximum:

$$M = \max_{\substack{\text{all pairs} \\ (j_x, j_y) \\ \forall n}} \frac{|f_{n+1}(j_y) - f_{n+1}(j_x)|}{|j_y - j_x|}$$

This way M is numerically calculated and afterward the upper bound to the local and global errors can be calculated.

The mean local discretization error was calculated by the construction of $\hat{D}$ as in Section 5.3.2. To this end the intervals for i^*_{ep} and i^*_{ep} were slit op in 5 subintervals. For every conceivable combinations the optimal decision was determined; all different decisions form the set $\hat{D}$, which is needed for the calculation of the mean local error. After this the mean global error can be determined as was discussed in Section 5.3.3.

6.2.5 Checks

The total expected noise load in all enforcement points was calculated where needed. When calculating the exceedance probability for $N = 12$ it was calculated to check if the sets were chosen correctly. When altering the strategy the effect on the noise load realization needs to be determined. These calculations were performed.

7 Numeric results

In this section the numeric results obtained from the calculations discussed in Section 6.2 are presented. First, some general results and insight into the structure of results will be discussed. Then specific results are presented for 4 enforcement points and a wide range of results for 3 enforcement points.

7.1 Structure

In the probability-based model the values for optimal value function $f_n(i)$ were calculated for all states in all stages. Since the model was executed in more than two dimensions the graphical results cannot be given in one glance. However, combining some insightful graphical results can sketch a picture of the overall results.

Looking at only two enforcement points, given that the other realizations are fixed, a three-dimensional representation can be given of values of the optimal value function. In the probability-based model the optimal value function represents the exceedance probability of a certain state. Values were used from calculations for enforcement points {5,21,25}. Figure 16(b) shows these probabilities for all states in the 7th (of 12) month of the process. From left to right the figures show the development in exceedance probability (z-axis) for pairs of realizations (x- and y-axis) of the state space. The optimal value function resembles a cumulative distribution function. In Figure 16(a) the optimal values can be found for the 2nd month of the process. It can be seen that the exceedance probabilities vary less between the states than at the end of the process. This signifies that an implementation becomes increasingly important toward the end of the year.

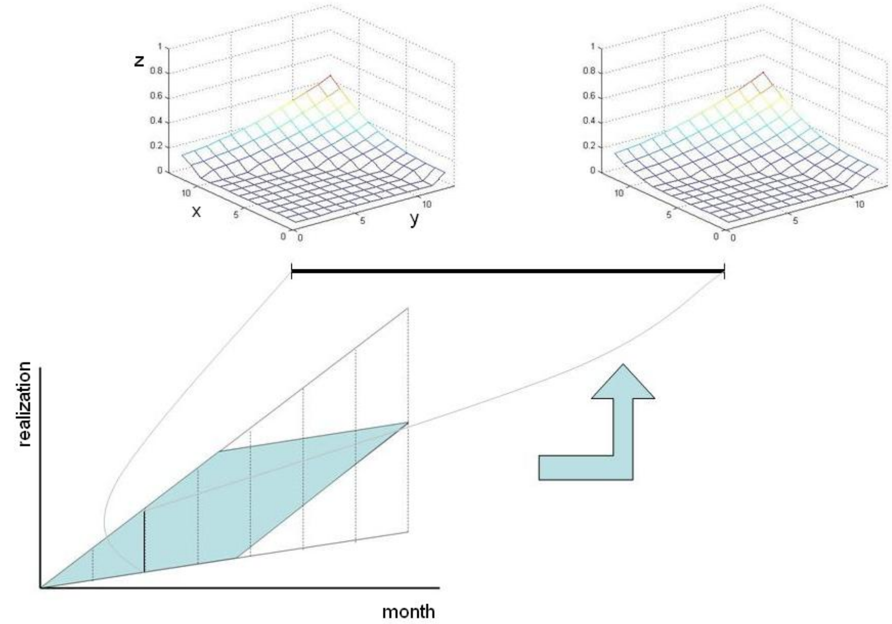
7.2 Four enforcement points

The model was executed for the set enforcement points {9,19,21,31}. To show that the calculations are consistent with reality, several possible scenarios are evaluated more in-depth. Chosen is for some extreme situations, because then the results are most explicit. In this section ↑ signifies take-off and ↓ signifies landing.

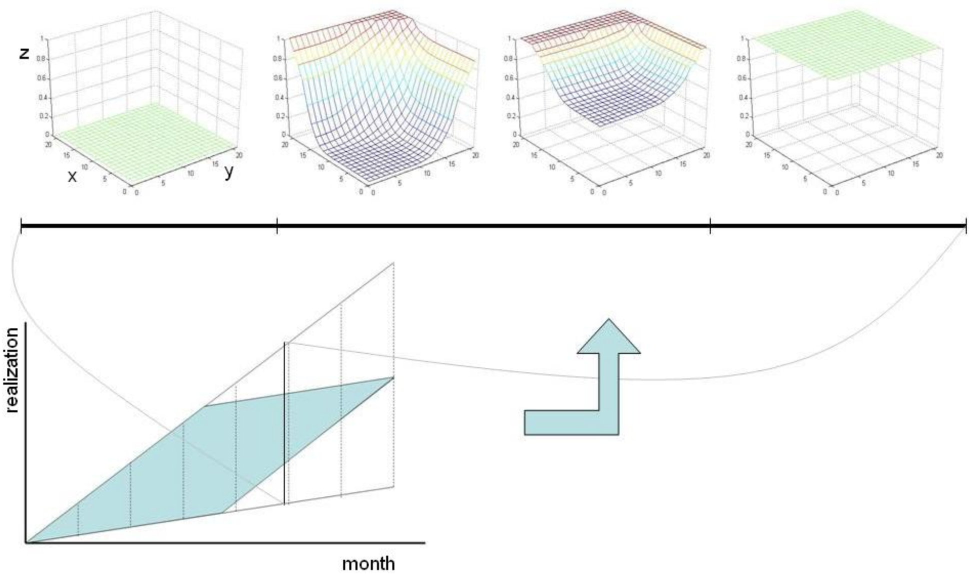
7.2.1 Exceedance probability

Only one full calculation was completed with enforcement points in four directions. The result for the exceedance probability can be found in Table 4. The performed calculations gave insight into the development of noise load, triggered better implementation steps for smaller subsets of enforcement points and were used to construct some examples. These examples show the plausibility for the use of the current model and stochastic dynamic programming: results were found that are consistent with reality.

Example 1: Realistic decisions Take the following scenario: A decision has to be made for August, which is month 9 in the model (with 3 months to go until the end of the year).



(a) Optimal values in month 2



(b) Optimal values in month 7

Figure 16: Optimal value functions

nr.	points				$N = 12$ $\mathcal{P}_{exc}$
1.	9	19	21	31	79.23%

Table 4: Exceedance probability for a set of 4 enforcement point

The realized noise load percentages for the 4 enforcement points equal:

$$rl = \begin{bmatrix} 85.8 \\ 66.0 \\ 65.8 \\ 66.6 \end{bmatrix} \%$$

It is a situation with a high noise load in point 9 and low noise loads in point 19, 21 and 31. Point 9 is most effected by take-offs from 36L and landings on 18R. The decision that has to be made must prevent a high contribution in point 9: a preference lists must be chosen where 36L↑ and 18R↓ have a low positions relative to the other lists.

Because of the discretization with increments of 2% this situation is represented by state:

$$i = \begin{bmatrix} 86 \\ 66 \\ 66 \\ 66 \end{bmatrix}$$

The model prescribes the use of preference list 7 at this state. This is consistent with the expected decision, because the mentioned runways do have equal or lower positions on the list relative to other preference lists. See Appendix A.4: While take-offs are noisier than landings focus on a lower position of 36L↑; this means that lists 1 to 4 are eliminated, while in lists 5 to 8 36L↑ is always lower. Then, 18R↓ must also be lower on the list; preference list 8 looks like the best option, but apparently list 7 has the best effect on the whole situation. From this state, at this time the exceedance probability equals: $\mathcal{P}_{exc} = 93.93\%$.

If the realized load would be (already discretized):

$$i = \begin{bmatrix} 66 \\ 66 \\ 66 \\ 86 \end{bmatrix}$$

It is a situation with a high noise load in point 31 and low noise loads elsewhere. Point 31 is most effected by take-offs from 24 and 18C, and landings on 06 and 36C. The decision that has to be made must prevent a high contribution in point 31: a preference lists must be chosen where 24 ↑ and 06 ↓ have low positions relative to the other lists. The model prescribes the use of preference list 2 at this state, hence more contribution in the north. This is consistent with the expected decision, because the mentioned runways do have equal or lower positions on the list relative to other preference lists. From this state, at this time the exceedance probability equals: $\mathcal{P}_{exc} = 65.95\%$.

Example 2: Transition probabilities Take the following scenario: A decision has to be made for March, which is month 4 in the model (with 8 months to go until the end of the year). Say, the process is in state:

$$i = \begin{bmatrix} 18 \\ 36 \\ 36 \\ 18 \end{bmatrix}$$

It is a situation with a high noise load in points 19 and 21, and low noise loads in point 9 and 31. Due to the choices made in Section 6 to noise load contributions range from 3% to 13% of the standard per stage divided into 5 intervals of increment 2%. This means the state in the next stage is between j_{lb} and j_{ub} defined by:

$$j_{lb} = \begin{bmatrix} 22 \\ 40 \\ 40 \\ 22 \end{bmatrix} \text{ and } j_{ub} = \begin{bmatrix} 34 \\ 52 \\ 52 \\ 34 \end{bmatrix}$$

For instance, when choosing preference list 5 the following two noise load contributions could be made:

$$x_1 = \begin{bmatrix} 4 \\ 8 \\ 8 \\ 10 \end{bmatrix} \text{ with } p(x_1) = 0.0168,\ x_1 = \begin{bmatrix} 10 \\ 8 \\ 8 \\ 4 \end{bmatrix} \text{ with } p(x_2) = 0.0021,$$

This is consistent with reality while preference list 5 contributes relatively more in the south as can be deduced from the preference list in Appendix A.4. The probability of making a high contribution in enforcement point 31 (in the south) is larger than a high contribution in point 9 (in the north).

When adding all the probabilities of possible contributions multiplied by the values of the optimal value function in the next stage the expected exceedance probability is found. It is found that from the current state the exceedance probability $\mathcal{P}_{exc} = 53.97\%$ and the optimal decision in preference list 3. This is consistent with reality. Points 19 and 21 are most effected by take-offs from 18L and operations on 09-27. It can be seen in Appendix A.4 that on preference lists 1 to 4 18L↑ has lower preference than on lists 5 to 8. Also, on preference list 3 09↑ is lowest of preference lists 1 to 4. Even though 27↓ has relative high preference, this can be countered by the fact that landings are less noisy than take-offs. Hence, the choice for preference list is consistent with reality.

7.3 Three enforcement points

In this section the results for the eight different subsets of three enforcement points will be discussed. During the calculations crucial information was not stored for sets 1 and 3. This is why some results are not present for these sets. The effect of altering the strategy was performed with three subsets, namely 4, 7 and 8.

7.3.1 Exceedance probabilities

Numeric results for the calculation of the exceedance probability for the current number of decision moments can be found in Table 5 under $N = 12$. For the found strategy the check is performed as discussed in Section 5.1. Results of the checks for all sets of enforcement points can be found in Table 9 in Appendix D. It can be seen that modeled sets of enforcement points, indicated in white, never exceed their standards; all values are under 100%. However, there are some enforcement points for which the total expected noise load does exceed. The number of exceeding enforcement points and their amount of exceedance indicates how well all enforcement points are modeled with the subset.

For subsets 2, 4, 5 and 8 there are at most three exceedances. Subset 4 has three small exceedances in different directions of the airport; therefore the noise load is globally high, which is not desirable. Subset 2 has two exceedances, one of which is very large; it is also not desirable. This leaves subsets 5 and 8; these are good subsets to model all enforcement points, since in both cases there is a single small exceedance in the total expected noise load.

Subsets 5 and 8 have exceedance probabilities of respectively 16.26% and 15.19%. The estimate of real exceedance probability for all enforcement points 16.26% given by subset 5.

7.3.2 Effect of an increase in decision moments

Numeric results for the effect on a doubling of decision moments for the different subsets can also be found in Table 5. The improvement factor is also given and represents the decrease in exceedance probability.

It can be seen that most decrease factors depend heavily on the sets of enforcement points. It was found in the previous section that subsets 5 and 8 represent all enforcement points best. Their improvement are respectively 0.13 and 0.15, which signifies an approximate 14% improvement in exceedance probability when the number of decision moments is doubled.

7.3.3 Altering the strategy

By altering the strategy so that one iteration step is taken toward an optimal strategy in the realization based setting, the accompanying probabilities are found to be as in Table 6. First α was chosen 1, which resulted in a large increase in exceedance probability. Later α was chosen 2, which resulted in smaller increase exceedance probabilities. The resulting realizations were calculated with the check that can be found in Table 10 in Appendix D. It

nr.	enforcement points			$N = 12$ $\mathcal{P}_{exc}$	$N = 24$ $\mathcal{P}_{exc}$	Improvement factor
1.	5	19	21	10.10%	3.97%	0.61
2.	5	21	25	3.68%	2.49%	0.32
3.	9	19	25	1.51%	0.42%	0.72
4.	9	22	31	8.47%	2.93%	0.65
5.	18	19	21	16.26%	14.12%	0.13
6.	18	19	25	2.67%	2.21%	0.17
7.	18	19	31	2.34%	1.98%	0.15
8.	19	21	31	15.19%	12.98%	0.15

Table 5: Exceedance probabilities for N=12 and N=24 and improvement factor

can be observed that a more balanced noise load arises in the modeled enforcement points, since in subset 7 two expected realization decrease and one increases, and vice versa in subsets 4 and 8. It can also be observed that the noise load in enforcement points that were not modeled have very conflicting results. For instance, after the alteration many of the points that already showed an expected exceedance obtain an even higher exceedance and many low realizations become even lower; the goal of the method was to level the realizations. Therefore, at this scale (number of enforcement points in the subset) this method does not produce a more balanced noise load with a slight increase in exceedance probability.

nr.	enforcement points			Original $\mathcal{P}_{exc}$	Altered $\mathcal{P}_{exc}$ $\alpha = 1$	Altered $\mathcal{P}_{exc}$ $\alpha = 2$
4.	9	22	31	8.47%	19.18%	13.79%
7.	18	19	31	2.34%	4.81%	3.87%
8.	19	21	31	15.19%	26.64%	22.61%

Table 6: Exceedance probabilities with altered strategies

7.3.4 Discretization errors

The local mean discretization errors per stage of this process, as discussed in Section 5.3, are calculated for the last and one-to-last step. The local error in the one-to-last step is of the same magnitude as the step before that and it is used as an estimate for the error those previous steps. Results are given in Table 7. The intervals in which the real exceedance probabilities lie are also given in this table (probabilities cannot become negative, therefore the lower limit of the intervals equals 0).

It can be observed that the global mean discretization error is around 12% for the various subsets. This indicates that the calculated exceedance probability estimates the real

Error nr.	enforcement points			ζ_n^{mean} n=11	ζ_n^{mean} n=10	ξ_0^{mean}	interval for $\mathcal{P}_{exc}$	
2.	5	21	25	2.80%	0.76%	11.16%	0%	3.68%
4.	9	22	31	3.63%	1.01%	14.74%	0%	8.47%
5.	18	19	21	2.87%	0.80%	11.67%	4.59%	16.26%
6.	18	19	25	3.32%	0.95%	13.77%	0%	2.67%
7.	18	19	31	2.85%	0.81%	11.76%	0%	2.34%
8.	19	21	31	2.43%	0.75%	10.68%	4.51%	15.19%

Table 7: Local and global mean discretization errors

exceedance probability to within 12%. This is 6 times the discretization interval ϵ.

Intervals for subsets 2, 4, 6 and 7 show very small exceedance probabilities. It was already determined that they do not represent all enforcement points as well as should. Since subsets 5 and 8 represent reality best and subset 5 estimates the exceedance probability best, it can be said that the exceedance probability when modeling three enforcement points is between 4.59% and 16.26%.

The upper bound to the global error is calculated also. Results show that the errors are so large, that any values for the real exceedance probability is always between 0 and the calculated exceedance probability. Results can be found in Table 8.

Error nr.	enforcement points			ξ_0^{upper}
2.	5	21	25	285.35%
4.	9	22	31	397.98%
5.	18	19	21	371.71%
6.	18	19	25	350.42%
7.	18	19	31	399.18%
8.	19	21	31	282.93%

Table 8: Upper bound global discretization errors

8 Conclusion and Recommendations

In this section concluding remarks on the project are made. Moreover, recommendations concerning the operation at Schiphol and the future development of the proposed method are presented.

8.1 Conclusion

The current method for the calculation of the exceedance probability is conceptually flawed. Therefore the estimate for the exceedance probability has a large error bound. The proposed method of stochastic dynamic programming takes in account the effect of stochastic realizations and therefore gives a better estimate of the exceedance probability.

Stochastic dynamic programming is based on noise load realization possibilities. Every conceivable realization during the year is analyzed in advance and accompanying exceedance probabilities are based on all possible paths from there on. It is a more insightful method for the choice of preference lists, since it can be made clear exactly what the choice is based upon. The current method does not offer this feature to this extent, since future calculations are not taken into account. Using stochastic dynamic programming results in less friction between operational and non-operational personnel, since there is a better understanding of noise load management for both parties.

Implementation of stochastic dynamic programming resulted in plausible decisions for calculated scenarios. This was illustrated with some examples. Therefore, it is theoretically and practically founded and plausible that it can be applied to noise load management.

Applying stochastic dynamic programming to noise load management enables better choices for preference lists compared to the current situation. A lower exceedance probability implicitly means that the allowed noise load in the enforcement points is utilized more efficiently. The operation at Schiphol is optimized within the environmental restrictions. This implies that, compared to the current situation, more noise load volume remains to be utilized within the current environmental restrictions. Therefore more flights can be handled during the year, which will have a positive economic effect on the aviation industry at Schiphol.

Discretization errors determine the significant value of the numeric calculations. The calculated exceedance probability estimates the real exceedance probability to within 6 times the discretization interval.

The exceedance probability for the year 2006 is estimated between 0.0459 and 0.1626. Calculations show a 14% improvement in exceedance probability when the number of decision moments is doubled to 24 moments. Theoretical results indicate a further decrease in exceedance probability when increasing the number of decision moments; the exceedance probability will eventually approach a lower bound.

The proposed alteration to the optimal strategy produces a more balanced noise load with a small increase in exceedance probability in the modeled enforcement points. With the current number of modeled enforcement points (3) this method does not produce a similar effect in the other enforcement points.

This study offers LVNL a method for better estimating the exceedance probability compared to the current method. It is interesting to further investigate its possibilities.

Discussion of the assumptions

To develop the model certain assumptions were made. In order to make the model more realistic, the following could be done in future development of the model:

- The standards in the enforcement points during the day-evening-night period are the only enforced environmental restrictions: The set of modeled enforcement points could also include points from the night period. Further environmental restrictions can be implemented and creates extra limiting conditions to the model.
- The chosen subset of enforcement points represents all enforcement points: The subset of enforcement points can be chosen larger. It will not be possible to model all enforcement points. Study must be done into the adequate number of enforcement points that cover the surroundings of Schiphol as well as possible. It is estimated that six points cover the different directions well enough.
- There are eight preference lists: More preference lists can be thought of and their transition probabilities can be calculated. The set of preference lists can be expanded, so that a preference lists exists that has a positive effect for all noise load realizations.
- Noise load realizations and contributions are discrete: Discretization error occur when a continuous process is estimate by a discrete one. Either discretization intervals can be chosen smaller so that the error decreases or continuous state space and realizations can be implemented so that an error will not occur. The feasibility of this continuous model is questionable, but this assumption could be dropped by implementing these continuous elements to the model.
- Noise load contributions have a multivariate normal distribution: Study can be done into the weather and her facets. If a method is constructed that accurately determines probabilities for weather conditions, a more realistic noise load contribution distribution can be made.
- The 30 meteo years on record can be used to predict all weather possibilities: Weather services, like the KNMI[9], can be consulted on developments in predictions of the weather. Ideally, this concerns a model that does not depend on meteorological history.

[9]KNMI=Koninklijk Nederlands Meteorologisch Instituut, Royal Dutch Meteorological Institute

8.2 Recommendations

The stochastic dynamic programming method is a better method for calculation of the exceedance probability. It should therefore be implemented at LVNL and future decisions for the preference list should be based on this method. For further development of the method, it is recommended to investigate the feasibility of calculations for bigger models. Since the problem complexity and computation time increases rapidly when adding extra constrains, it is recommended that it is investigated what additions to the model provide the most extra information and contribute to most to represent reality.

It was found that a doubling of decision moments resulted in a significant decrease in exceedance probability. It is therefore recommended to increase the number of times a decision is made on the preference list. Further research can be done into the lower bound of the exceedance probability and an optimal that can be reached when increasing the decision frequency.

9 Bibliography

References

[1] R.E. Bellman, 1957. *Dynamic programming*, Princeton University Press.

[2] D.P. Bertsekas, 1987. *Dynamic programming: Deterministic and stochastic models*, Prentice-Hall.

[3] F. Blanchini & S. Miani & F.A. Pellegrino, 2003. Suboptimal receding horizon control for continuoustime systems. *IEEE Transactions on Automatic Control*, 48, 1081-1086.

[4] A.R. Cassandra, 1995. *Optimal policies for partially observable Markov decision processes*. Technical Report CS 94-14, Department of Computer Science, Brown University.

[5] C.E. Garcia & D.M. Prett & M. Morani, 1989. Model predictive control theory and practice - a survey. *Automatica*, 25, 335-348.

[6] A. Genz, 1992. Numerical computation of the multivariate normal probabilities. *Journal of Computational and Graphical Statistics*, 1, 141-150.

[7] F.S. Hillier & G.J. Lieberman, 1995. *Introduction to operations research*, McGraw-Hill International Editions.

[8] M. Jacobson & N. Shimkon & A. Shwartz, 1999. Piecewise stationary Markov decision processes. Submitted to *Mathematics of Operations Research*.

[9] J.G. Johnson & J.R. Busemeyer, 2001. Multiple-stage decision-making: The effect of planning horizon length on dynamic consistency. *Theory and Decision*, 51, 217-246.

[10] D.Q. Mayne, 2001. Control of constrained dynamic systems. *European Journal of Control*, 7, 87-99.

[11] T.R. Meerburg, 2005. An overview of literature for a study into noise load around Schiphol.

[12] M. Ottivani, 1996. Monopoly pricing with social learning. In: *Social Learning in Markets*. Ph.D. thesis, MIT.

[13] M.L. Puterman, 1994. *Markov decision processes: Discrete stochastic dynamic programming*, John Wiley & Sons, Inc.

[14] W.L. Winston, 1993. *Operations Research*, Duxbury Press.

Appendices

A Background information

A.1 Abbreviations and glossary

Abbreviations	
AAS	Amsterdam Airport Schiphol
ACC	Area Control Center
APR	Approach
ATC	Air Traffic Control
ATM	Air Traffic Management
EEMC/E^{2}MC	Environmental and Economic Management Committee
FIR	Flight Information Region
LVB	Luchtverkeersbeveiliging
LVNL	Luchtverkeersleiding Nederland
	Air Traffic Control the Netherlands
PIA	Performance and Incident Analysis
R&D	Research and Development
RLD	Rijksluchtvaartdienst
SDP	Stochastic Dynamic Programming
SEE	Safety, Efficiency, Environment
SOR	Stochastic Operations Research
TRACON	Terminal Radar Approach CONtrol
TW	Toegepaste Wiskunde
	Applied Mathematics
TWR	Tower
TWR-W	Tower West
UT	Universiteit Twente
	University of Twente

Glossary	
$c_n(i,d)$	Single step contribution during stage n when choosing decision d in state i
crl	Vector with cumulative realized load values per enforcement point
$D_n(i)$	Discrete set of possible decisions at state i in stage n (decision space)
$dP_n(j\|i,d)$	Probability density for the transition from state i to state j under decision d
ϵ	Discretization interval
EP	Set of all enforcement points
EP_{den}	Set of enforcement points used during the day, evening and night period
$f_X(x)$	Probability density function for random vector X
$f_n(i)$	Minimal expected sum of contributions in stages $n, n+1, \ldots, N$ when in state i before the decision at stage n is made
$f_n^\pi(i)$	Minimal expected sum of contributions in stages $n, n+1, \ldots, N$ when in state i before the decision at stage n is made when following strategy π
$\hat{f}_n(i)$	Approximation of minimal expected sum of contributions in stages $n, n+1, \ldots, N$ when in state i before the decision at stage n is made
$F_X(a,b)$	Cumulative distribution function for random vector X between limits a and b
k	Number of evaluated enforcement points
L	Vector with standard per enforcement point
$Moments$	Set of dates at which decisions are implemented
N	Number of decision moments
π	Optimal strategy
$\pi_n(i)$	Decision rule for for state i stage n
$p(i_n)$	Probability of being in state $i_n \in S_n$
$P_{n,d}$	Probability of making decision d in stage n
$\hat{P}_n(j\|i,d)$	Probability that decision d in state $i \in S_n$ leads to state $j \in S_{n+1}$
$\mathcal{P}_{exc}$	Exceeding probability of a standard in the enforcement points
R	Set of runways at Schiphol
RC	Set of runway combinations used at Schiphol
rl	Vector with realized load in a period per enforcement point
S	Continuous state space
$\hat{S}$	Discrete state space
S_n	Discrete state space in stage n
ξ_n	Global error for states in stage n
ζ_n	Local error for states in stage n

A.2 Noise load calculation

This section analyzes how landing and departing aircraft on Schiphol produce noise load as it is done by LVNL and described in their reports. The pyramid in Figure 17 shows the hierarchical realization of noise load.

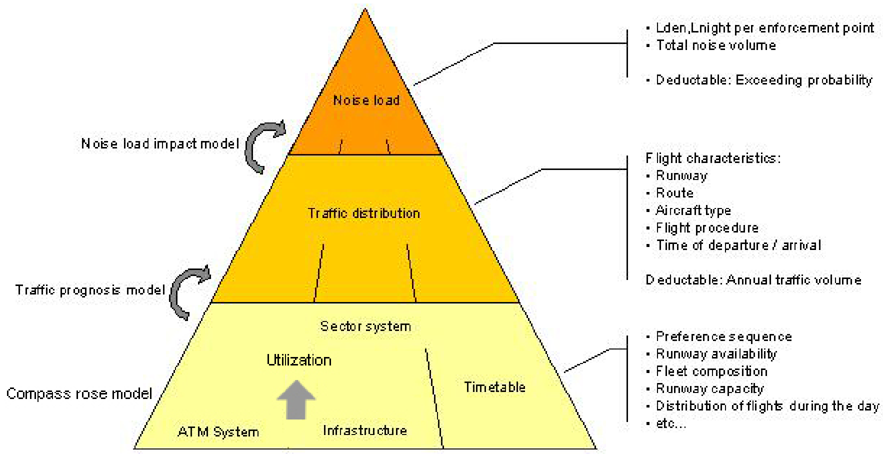


Figure 17: Noise load pyramid

The sector system, consisting of the ATM system, Infrastructure and the Timetable, determines the traffic distribution on Schiphol, influenced by the weather conditions. The traffic distribution determines the noise load of the operation.

The ATM system is the system with which the supply of air traffic is processed. It consists of a human, a machine and a procudure component. The infrastructure is the configuration of runways, taxiways, aprons, etc. on Schiphol that is used to process the air traffic. The timetable is a list of departures and arrivals on Schiphol, based on expectations of the development of the airliner market. Per flight, the aircraft type, the time of departure or arrival and the airport of destination if departure is specified.

The analysis of noise load produced by the Schiphol operation is application independent. It can be used to measure and analyze the noise load of the current operation, but also to estimate the noise load of envisaged operation. To estimate the noise load, three models must be used:

- Compass rose: based on the specifications of the sector system this model determines the rules according to which the traffic will be handled at Schiphol
- Traffic prognosis: based on the compass rose and the timetable
- Noise load impact: based on a certain traffic distribution this model calculates the noise load per enforcement point

Their main relevant specifications will are discussed further:

Compass rose DAISY is used to determine the rules according to which the traffic will be handled at Schiphol. Compass rose determines what runway combinations are used under what conditions; the utilization. The following inputs are used for this project:

- Meteorological data from 1971 to 2000
- Limiting conditions for cross and tail wind
- Preference list from the 'Plan of Operations 2006'
- Infrastructure of Schiphol and utilization of the ATM system for 2006
- Periods of inbound-, outbound- and offpeaks (modes)

Traffic prognosis With this model, the traffic distribution over the runways is estimated. The following inputs are used for this project:

- Timetable of flights for 2006 delivered by the aviation sector
- Flight characteristics as aircraft type, landing procedures etc.
- Route assignment
- Utilization (from Compass rose)

Noise load impact With this model, the noise load impact is estimated. The following inputs are used for this project:

- Location of the enforcement points
- Noise load contribution factors
- Weight factors (penalty factors for day, evening and night operation)
- Traffic distribution (from Traffic prognosis)

A cluster is a collection of flights with the same values for runway, route, aircraft type and flight procedure. The traffic distribution is defined by the number of flights per cluster. Noise load is calculated through the addition of all noise load contribution factors of clusters multiplied by the size of the clusters.

A.3 Calculation of parameters for the multivariate normal distribution

In advance, a database is made with values for observations of noise load for all control cycles, for all meteo years, for all preference lists and for all enforcement points. Filtering this database for a certain control cycle n, for the selection of k enforcement points and for a preference list pl produces the following dataset:

$$\text{Data} = \begin{bmatrix} z_{1,1} & z_{1,2} & \cdots & z_{1,k} \\ z_{2,1} & z_{2,2} & \cdots & z_{2,k} \\ \vdots & \vdots & & \vdots \\ z_{30,1} & z_{30,2} & \cdots & z_{30,k} \end{bmatrix} \quad \begin{matrix} \uparrow \\ \text{meteo years} \\ \downarrow \end{matrix}$$

$$\longleftarrow \text{enforcement points} \longrightarrow$$

The elements of the expected value vector μ are calculated by:

$$\mu_{ep} = \frac{\sum_{y=1}^{30} z_{y,ep}}{30}$$

Hence:

$$\mu = \frac{1}{30}\left[\sum_{y=1}^{30} z_{y,1}, \ldots, \sum_{y=1}^{30} z_{y,k}\right]^T$$

The covariance between to variables is calculated by:

$$\sigma_{ij} = \frac{1}{30-1}\sum_{y=1}^{30}(z_{y,i}-\mu_i)(z_{y,j}-\mu_j)$$

Hence:

$$\Sigma = \frac{1}{29}\begin{bmatrix} \sum_{y=1}^{30}(z_{y,1}-\mu_1)^2 & \sum_{y=1}^{30}(z_{y,1}-\mu_1)(z_{y,2}-\mu_2) & \cdots & \sum_{y=1}^{30}(z_{y,1}-\mu_1)(z_{y,k}-\mu_k) \\ \sum_{y=1}^{30}(z_{y,2}-\mu_2)(z_{y,1}-\mu_1) & \sum_{y=1}^{30}(z_{y,2}-\mu_2)^2 & & \sum_{y=1}^{30}(z_{y,2}-\mu_2)(z_{y,k}-\mu_k) \\ \vdots & & \ddots & \vdots \\ \sum_{y=1}^{30}(z_{y,k}-\mu_k)(z_{y,1}-\mu_1) & \sum_{y=1}^{30}(z_{y,k}-\mu_k)(z_{y,2}-\mu_2) & \cdots & \sum_{y=1}^{30}(z_{y,k}-\mu_k)^2 \end{bmatrix}$$

These parameters can be estimated for any selection of enforcement points, control cycle n and preference list pl. This means there will be a different dataset for different situations and hence different parameters to describe it.

A.4 Used preference lists

For a general idea, preference lists 1 to 4 contribute more in the northern enforcement points, preference lists 5 to 8 more in the southern, while the first position in the lists consists of runway combinations in that direction. Lists 2 to 4 and 6 to 8 are variations of respectively 1 and 5; changes can be found in the runway combination of the outbound peak and consist of changes in the first 6 positions in the lists.

↑ = take-off
↓ = landing

Preference list 1										
Peak:	Inbound			Outbound			Off		Night	
nr.	↑	↓		↑		↓	↑	↓	↑	↓
1	36L	06	36R	36L	36C	06	36L	06	36L	06
2	24	18R	18C	24	18L	18R	24	18R	24	18R
3	18L	18R	18C	18L	18C	18R	18C	18R	18C	18R
4	36L	36R	36C	36L	36C	36R	36L	36C	36L	36C
5	24	27	18R	36L	09	06	09	18R	06	06
6	24	18R	22	24	36L	27	09	06	24	18C
7	18L	18R	22	24	18L	27	24	27	24	27
8	09	06	09	24	27	27	36L	27	36L	27
9	36L	06		36L		06	24	24	24	24
10	24	18R		24		18R	27	27	09	09
11	18L	18R		18L		18R	09	09	06	06
12	36L	36R		36L		36R	24	22	09	18R
13	09	18R		09		18R	06	06		
14	09	06		09		06				
15	24	27		24		27				
16	36L	27		36L		27				
17	24	22		24		22				
18	27	27		27		27				
19	09	09		09		09				
20	24	24		24		24				
21	06	06		06		06				

Preference list 2										
Peak:	Inbound			Outbound			Off		Night	
nr.	↑	↓		↑		↓	↑	↓	↑	↓
1	36L	06	36R	36L	09	06	36L	06	36L	06
2	24	18R	18C	36L	36C	06	24	18R	24	18R
3	18L	18R	18C	24	18L	18R	18C	18R	18C	18R
4	36L	36R	36C	18L	18C	18R	36L	36C	36L	36C
5	24	27	18R	36L	36C	36R	09	18R	06	06
6	24	18R	22	24	36L	27	09	06	24	18C
7	18L	18R	22	24	18L	27	24	27	24	27
8	09	06	09	24	27	27	36L	27	36L	27
9	36L	06		36L		06	24	24	24	24
10	24	18R		24		18R	27	27	09	09
11	18L	18R		18L		18R	09	09	06	06
12	36L	36R		36L		36R	24	22	09	18R
13	09	18R		09		18R	06	06		
14	09	06		09		06				
15	24	27		24		27				
16	36L	27		36L		27				
17	24	22		24		22				
18	27	27		27		27				
19	09	09		09		09				
20	24	24		24		24				
21	06	06		06		06				

Preference list 3										
Peak:	Inbound			Outbound			Off		Night	
nr.	↑	↓		↑		↓	↑	↓	↑	↓
1	36L	06	36R	24	36L	27	36L	06	36L	06
2	24	18R	18C	36L	36C	06	24	18R	24	18R
3	18L	18R	18C	24	18L	18R	18C	18R	18C	18R
4	36L	36R	36C	18L	18C	18R	36L	36C	36L	36C
5	24	27	18R	36L	36C	36R	09	18R	06	06
6	24	18R	22	36L	09	06	09	06	24	18C
7	18L	18R	22	24	18L	27	24	27	24	27
8	09	06	09	24	27	27	36L	27	36L	27
9	36L	06		36L		06	24	24	24	24
10	24	18R		24		18R	27	27	09	09
11	18L	18R		18L		18R	09	09	06	06
12	36L	36R		36L		36R	24	22	09	18R
13	09	18R		09		18R	06	06		
14	09	06		09		06				
15	24	27		24		27				
16	36L	27		36L		27				
17	24	22		24		22				
18	27	27		27		27				
19	09	09		09		09				
20	24	24		24		24				
21	06	06		06		06				

Preference list 4										
Peak:	Inbound			Outbound			Off		Night	
nr.	↑	↓		↑		↓	↑	↓	↑	↓
1	36L	06	36R	36L	09	06	36L	06	36L	06
2	24	18R	18C	24	36L	27	24	18R	24	18R
3	18L	18R	18C	36L	36C	06	18C	18R	18C	18R
4	36L	36R	36C	24	18L	18R	36L	36C	36L	36C
5	24	27	18R	18L	18C	18R	09	18R	06	06
6	24	18R	22	36L	36C	36R	09	06	24	18C
7	18L	18R	22	24	18L	27	24	27	24	27
8	09	06	09	24	27	27	36L	27	36L	27
9	36L	06		36L		06	24	24	24	24
10	24	18R		24		18R	27	27	09	09
11	18L	18R		18L		18R	09	09	06	06
12	36L	36R		36L		36R	24	22	09	18R
13	09	18R		09		18R	06	06		
14	09	06		09		06				
15	24	27		24		27				
16	36L	27		36L		27				
17	24	22		24		22				
18	27	27		27		27				
19	09	09		09		09				
20	24	24		24		24				
21	06	06		06		06				

Preference list 5										
Peak:	Inbound			Outbound			Off		Night	
nr.	↑	↓		↑		↓	↑	↓	↑	↓
1	24	18R	18C	24	18L	18R	24	18R	24	18R
2	36L	06	36R	36L	36C	06	36L	06	36L	06
3	18L	18R	18C	18L	18C	18R	18C	18R	18C	18R
4	36L	36R	36C	36L	36C	36R	36L	36C	36L	36C
5	24	27	18R	36L	09	06	09	18R	06	06
6	24	18R	22	24	36L	27	09	06	24	18C
7	18L	18R	22	24	18L	27	24	27	24	27
8	09	06	09	24	27	27	36L	27	36L	27
9	36L	06		36L		06	24	24	24	24
10	24	18R		24		18R	27	27	09	09
11	18L	18R		18L		18R	09	09	06	06
12	36L	36R		36L		36R	24	22	09	18R
13	09	18R		09		18R	06	06		
14	09	06		09		06				
15	24	27		24		27				
16	36L	27		36L		27				
17	24	22		24		22				
18	27	27		27		27				
19	09	09		09		09				
20	24	24		24		24				
21	06	06		06		06				

Preference list 6										
Peak:	Inbound			Outbound			Off		Night	
nr.	↑	↓		↑		↓	↑	↓	↑	↓
1	24	18R	18C	24	18L	18R	24	18R	24	18R
2	36L	06	36R	24	36L	27	36L	06	36L	06
3	18L	18R	18C	36L	36C	06	18C	18R	18C	18R
4	36L	36R	36C	18L	18C	18R	36L	36C	36L	36C
5	24	27	18R	36L	36C	36R	09	18R	06	06
6	24	18R	22	36L	09	06	09	06	24	18C
7	18L	18R	22	24	18L	27	24	27	24	27
8	09	06	09	24	27	27	36L	27	36L	27
9	36L	06		36L		06	24	24	24	24
10	24	18R		24		18R	27	27	09	09
11	18L	18R		18L		18R	09	09	06	06
12	36L	36R		36L		36R	24	22	09	18R
13	09	18R		09		18R	06	06		
14	09	06		09		06				
15	24	27		24		27				
16	36L	27		36L		27				
17	24	22		24		22				
18	27	27		27		27				
19	09	09		09		09				
20	24	24		24		24				
21	06	06		06		06				

Preference list 7										
Peak:	Inbound			Outbound			Off		Night	
nr.	↑	↓		↑		↓	↑	↓	↑	↓
1	24	18R	18C	24	18L	18R	24	18R	24	18R
2	36L	06	36R	36L	09	06	36L	06	36L	06
3	18L	18R	18C	36L	36C	06	18C	18R	18C	18R
4	36L	36R	36C	18L	18C	18R	36L	36C	36L	36C
5	24	27	18R	36L	36C	36R	09	18R	06	06
6	24	18R	22	24	36L	27	09	06	24	18C
7	18L	18R	22	24	18L	27	24	27	24	27
8	09	06	09	24	27	27	36L	27	36L	27
9	36L	06		36L		06	24	24	24	24
10	24	18R		24		18R	27	27	09	09
11	18L	18R		18L		18R	09	09	06	06
12	36L	36R		36L		36R	24	22	09	18R
13	09	18R		09		18R	06	06		
14	09	06		09		06				
15	24	27		24		27				
16	36L	27		36L		27				
17	24	22		24		22				
18	27	27		27		27				
19	09	09		09		09				
20	24	24		24		24				
21	06	06		06		06				

Preference list 8										
Peak:	Inbound			Outbound			Off		Night	
nr.	↑	↓		↑		↓	↑	↓	↑	↓
1	24	18R	18C	24	18L	18R	24	18R	24	18R
2	36L	06	36R	36L	09	06	36L	06	36L	06
3	18L	18R	18C	24	36L	27	18C	18R	18C	18R
4	36L	36R	36C	36L	36C	06	36L	36C	36L	36C
5	24	27	18R	18L	18C	18R	09	18R	06	06
6	24	18R	22	36L	36C	36R	09	06	24	18C
7	18L	18R	22	24	18L	27	24	27	24	27
8	09	06	09	24	27	27	36L	27	36L	27
9	36L	06		36L		06	24	24	24	24
10	24	18R		24		18R	27	27	09	09
11	18L	18R		18L		18R	09	09	06	06
12	36L	36R		36L		36R	24	22	09	18R
13	09	18R		09		18R	06	06		
14	09	06		09		06				
15	24	27		24		27				
16	36L	27		36L		27				
17	24	22		24		22				
18	27	27		27		27				
19	09	09		09		09				
20	24	24		24		24				
21	06	06		06		06				

A.5 Detailed enforcement points

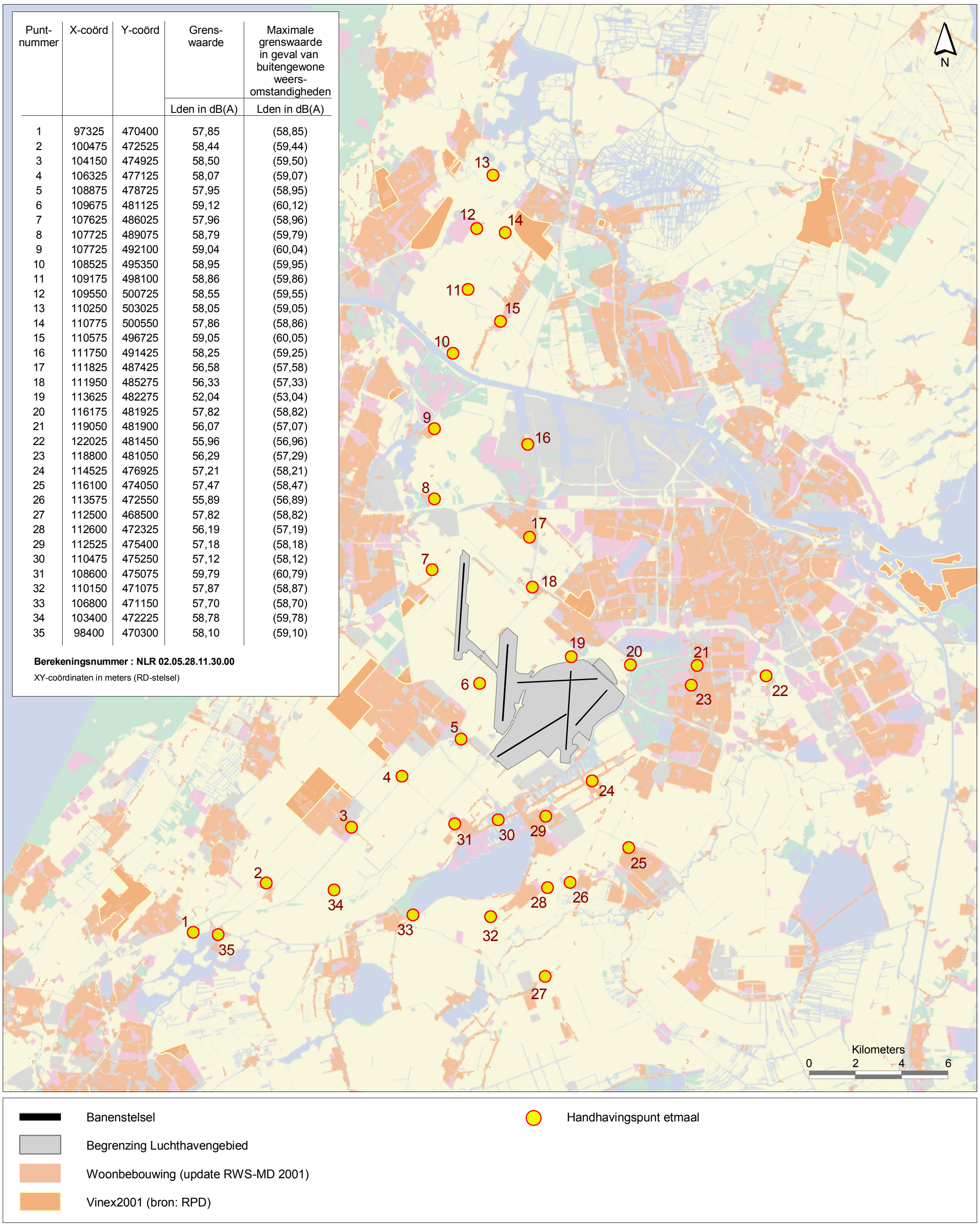
Handhavingspunten etmaal

Luchthavenverkeerbesluit Schiphol
Bijlage 2

Punt-nummer	X-coörd	Y-coörd	Grens-waarde	Maximale grenswaarde in geval van buitengewone weers-omstandigheden
			Lden in dB(A)	Lden in dB(A)
1	97325	470400	57,85	(58,85)
2	100475	472525	58,44	(59,44)
3	104150	474925	58,50	(59,50)
4	106325	477125	58,07	(59,07)
5	108875	478725	57,95	(58,95)
6	109675	481125	59,12	(60,12)
7	107625	486025	57,96	(58,96)
8	107725	489075	58,79	(59,79)
9	107725	492100	59,04	(60,04)
10	108525	495350	58,95	(59,95)
11	109175	498100	58,86	(59,86)
12	109550	500725	58,55	(59,55)
13	110250	503025	58,05	(59,05)
14	110775	500550	57,86	(58,86)
15	110575	496725	59,05	(60,05)
16	111750	491425	58,25	(59,25)
17	111825	487425	56,58	(57,58)
18	111950	485275	56,33	(57,33)
19	113625	482275	52,04	(53,04)
20	116175	481925	57,82	(58,82)
21	119050	481900	56,07	(57,07)
22	122025	481450	55,96	(56,96)
23	118800	481050	56,29	(57,29)
24	114525	476925	57,21	(58,21)
25	116100	474050	57,47	(58,47)
26	113575	472550	55,89	(56,89)
27	112500	468500	57,82	(58,82)
28	112600	472325	56,19	(57,19)
29	112525	475400	57,18	(58,18)
30	110475	475250	57,12	(58,12)
31	108600	475075	59,79	(60,79)
32	110150	471075	57,87	(58,87)
33	106800	471150	57,70	(58,70)
34	103400	472225	58,78	(59,78)
35	98400	470300	58,10	(59,10)

Berekeningsnummer : NLR 02.05.28.11.30.00

XY-coördinaten in meters (RD-stelsel)



B Theorems and proofs

B.1 Upperbound to local error 1

Theorem: Upperbound to local error 1. *An upperbound to the error term for states i in stage n in the discrete SDP defined in Section 3.2.2 is given by:*

$$\bar{f}_n(i) - f_n(i) \leq 2 \cdot \max_d \left(\bar{g}_n(i,d) - g_n(i,d)\right)$$

Proof (by contradiction) For convenience, let $g_n(i,d) = g(d)$ and $\bar{g}_n(i,d) = \bar{g}(d)$ for all n and i and let $m = \max_d[g(d) - \bar{g}(d)]$. Let $d^* = \arg\min_d g(d)$ and $\bar{d}^* = \arg\min_d \bar{g}(d)$. Let $G = \{d | g(d^*) + 2 \cdot m \geq \hat{g}(d)\}$. Suppose $\bar{d}^* \notin G$ then

$$\begin{array}{rcll} g(d^*) + 2 \cdot m & < & g(\bar{d}^*) & \text{since } \bar{d}^* \notin G \\ & \leq & \bar{g}(\bar{d}^*) + m & \text{since } \bar{g}(\bar{d}^*) + m = \min \bar{g} + \text{largest error} \\ & \leq & \bar{g}(d^*) + m & \text{since } \bar{g}(\bar{d}^*) \leq \bar{g}(d^*) \\ & \leq & g(d^*) + 2 \cdot m & \text{since } \bar{g}(d^*) \leq g(d^*) + m = \min g + \text{largest error} \end{array}$$

Contradiction!

Hence, $\bar{d}^* \in G$ and

$$\bar{f}_n(i) - f_n(i) = \bar{g}_n(i, \bar{d}^*) - g_n(i, d^*) \leq 2 \cdot \max_d \left(\bar{g}_n(i,d) - g_n(i,d)\right) \quad \forall i, n$$

QED

B.2 Upperbound to local error 2

Theorem: Upperbound to local error 2. *If f_{n+1} is Lipschitz-continuous with constant M* $\forall n$ *then* $\bar{g}_n(i,d) - g_n(i,d) \leq M \cdot \epsilon$

Proof:

$$\begin{aligned}
\bar{g}_n(i,d) - g_n(i,d) &= \textstyle\sum_x f_{n+1}(j_x) \cdot \int_{S_{x,n+1}} dP_n(j|i,d) - \int_{S_{n+1}} f_{n+1}(j) dP_n(j|i,d) \\
&= \textstyle\sum_x \int_{S_{x,n+1}} f_{n+1}(j_x) dP_n(j|i,d) - \sum_x \int_{S_{x,n+1}} f_{n+1}(j) dP_n(j|i,d) \\
&\leq \textstyle\sum_x \int_{S_{x,n+1}} [f_{n+1}(j_x) - f_{n+1}(j)] dP_n(j|i,d) \\
&\leq \textstyle\sum_x \left[\max_{j \in S_{x,n+1}} f_{n+1}(j) - \min_{j \in S_{x,n+1}} f_{n+1}(j)\right] \cdot \int_{S_{x,n+1}} dP_n(j|i,d) \\
&= \textstyle\sum_x \left| f_{n+1}\left(j_x - \frac{\epsilon}{2}\right) - f_{n+1}\left(j_x + \frac{\epsilon}{2}\right)\right| \cdot \hat{P}_{x,n}(j|i,d)
\end{aligned}$$

The last step here is made based on the assumption that optimal value function $f_n(i)$ is strictly increasing in i; this was proved by (among other) Lemma 1. in [12]. Therefore, maximum and minimum values of the value function are on the boundaries of the grid block.

In general, a function defined on an interval of real numbers with real values is called Lipschitz continuous if there exists a (Lipschitz-)constant $M \geq 0$ such that for all x and y:

$$|f(y) - f(x)| \leq M \cdot |y - x|$$

The function f_{n+1} is Lipschitz continuous, since all its values are between 0 and 1. A value M exists for any two possible points from the domain. The largest value is the Lipschitz constant. In particular it also holds:

$$\left| f_{n+1}\left(j_x - \frac{\epsilon}{2}\right) - f_{n+1}\left(j_x + \frac{\epsilon}{2}\right)\right| \leq M \cdot \epsilon$$

Using this expression the $\bar{g}_n(i,d) - g_n(i,d)$-term is further estimated:

$$\begin{aligned}
\bar{g}_n(i,d) - g_n(i,d) &\leq \textstyle\sum_x M \cdot \epsilon \cdot \hat{P}_{x,n}(j|i,d) \\
&= M \cdot \epsilon \cdot \textstyle\sum_x \hat{P}_{x,n}(j|i,d) \\
&= M \cdot \epsilon
\end{aligned}$$

QED

B.3 Construction of global error

Theorem: Construction of global error. *The error in $f_n(i)$ in the discrete SDP defined in Section 3.2.2 is calculated by $\xi_n = \sum_{i=n}^{N-1} \zeta_i$.*

Proof: For clarity the complete notation of terms is reduced to the necessary notation needed for this proof. Say:

$$\hat{f}_n = \int_{S_{n+1}} \hat{f}_{n+1} d\hat{P}_n$$

$$\bar{f}_n = \int_{S_{n+1}} f_{n+1} d\hat{P}_n$$

$$f_n = \int_{S_{n+1}} f_{n+1} dP_n$$

Here $\hat{f}_n$ is the discretized optimal value function when using optimal values from the discrete setting $\hat{f}$ and the discrete probability function $\hat{P}$. The $\bar{f}$-term represents the optimal value constructed from the optimal values from the continuous setting f and the discrete probability function. The optimal value function in the continuous function f represents the real optimal value. Let,

$$\begin{aligned}
\xi_n &= ||f_n - \hat{f}_n|| \\
&\leq ||f_n - \bar{f}_n|| + ||\bar{f}_n - \hat{f}_n|| \\
&= \zeta_n + || \textstyle\int_{S_{n+1}} f_{n+1} d\hat{P}_n - \int_{S_{n+1}} \hat{f}_{n+1} d\hat{P}_n || \\
&\leq \zeta_n + \textstyle\int_{S_{n+1}} ||f_{n+1} - \hat{f}_{n+1}|| d\hat{P}_n \\
&= \zeta_n + \xi_{n+1} \cdot \textstyle\int_{S_{n+1}} d\hat{P}_n \\
&= \zeta_n + \xi_{n+1}
\end{aligned}$$

Due to this recursive relation and $\xi_N = 0$ (while the values in stage N are correct) the values for ξ_n are determined by:

$$\xi_n = \sum_{i=n}^{N-1} \zeta_i$$

QED

C Relevant programming code

C.1 Monte Carlo method

This piece of Matlab-code was implemented to find the cumulative distribution function for a multivariate normal distribution. It is based theory in article [6].

P = MVNCDF(X,MU,SIGMA,ERRMAX,CI,NMAX) uses additional control parameters. The difference between P and the true value of the cumulative distribution function is less than ERRMAX CI percent of the time. NMAX is the maximum number of iterations that the algorithm makes. Values were chosen ERRMAX=0.0005, CI=99, NMAX=10000 for the calculation of the transition probabilities.

```
function [Intsum, error, N] = mvncdf(a, b, mu, Sigma, epsilon, ci, Nmax)
m1 = length(a);
m2 = length(mu);
[m3,m4] = size(Sigma);
m = m1;
a = a(:) - mu(:);
b = b(:) - mu(:);
C = chol(Sigma)';
Intsum = 0;
N = 0;
Varsum = 0;
d = normcdf( a(1) / C(1,1) );
e = normcdf( b(1) / C(1,1) );
f = zeros(m,1);
f(1) = e(1) - d(1);
y = zeros(m,1);
error = 2 * epsilon;
while ( error > epsilon ) & ( N < Nmax )
  w = unifrnd(0,1,m-1,1);
  for i = 2:m
  y(i-1) = norminv( d(i-1) + w(i-1) * ( e(i-1) - d(i-1) ) );
  q = 0;
  for j = 1:i-1
   q = q + C(i,j)*y(j);
  end;
  d(i) = normcdf( (a(i) - q) / C(i,i));
  e(i) = normcdf( (b(i) - q) / C(i,i));
  f(i) = ( e(i) - d(i) ) * f(i-1);
  end;
  N = N + 1;
  delta = (f(m) - Intsum) / N;
  Intsum = Intsum + delta;
```

```
  Varsum = (N-2) * Varsum / N + delta^2;
  error = alpha * sqrt(Varsum);
end;
```

C.2 Recursion relation

This part of the program implemented in Matlab is the pivot of the calculation method. It is a simplified version of the real implemented code.

```
waarde=0;
N=12;
Z=zeros(t2,t2,t2);
for n=11:-1:0
bovena = round( min( t2*n,(100/Nstap)-t1*(N-n)));
  ondera = round( max( t1*n,(100/Nstap)-t2*(N-n)));
  x=bovena-ondera+1;
  M=[];
for i1 = 1:x
  clear M
  clear D
  for i2 = 1:x
  for i3 = 1:x
      if (waarde>.9) & (i3>1)
          %if the previous value was larger than 0.9, the current value is 1
          waarde=1;
          M(i2,i3)= waarde;
          D(i2,i3)= actie;
      else
  statenu1=(ondera+i1-1);
  statenu2=(ondera+i2-1);
  statenu3=(ondera+i3-1);
  statenu=[statenu1 statenu2 statenu3];
  %what states are reached from the current state
  bovenb = round( min( t2*(n+1), (100/Nstap)-t1*(N-(n+1))      ));
  onderb = round( max( t1*(n+1), (100/Nstap)-t2*(N-(n+1))      ));
%determines the set Z(x1,x2,x3) of values of states that can be reached
  mogelijkerealisaties;
%determines the optimal value
  if ( min(min(min(Z(t1:t2,t1:t2,t1:t2)))) > .9 )
      waarde=1;
      M(i2,i3)=waarde;
      D(i2,i3)=actie;
  elseif ( max(max(max(Z(t1:t2,t1:t2,t1:t2)))) < .1 )
      waarde=0;
      M(i2,i3)=waarde;
```

```
      D(i2,i3)=0;
  else
      OVF=[];
for pl=1:8
ovf=0;
  %calculates the optimal value
    for x1=t1:t2
    for x2=t1:t2
    for x3=t1:t2
kansje = Kans(x1,x2,x3,n+1,pl);
        ovf = ovf + kansje * Z(x1,x2,x3);
end
    end
    end
OVF = [OVF ovf] ;
end
[waarde,actie] = min(OVF);
      M(i2,i3) = waarde;
      D(i2,i3) = actie;
          end
        end
      end
      end
    end
    %stores the optimal values and decisions for each state
    pad     = fullfile(computerpad,sprintf('M12_%d',n),sprintf('V%d.txt',i1));
    dlmwrite(pad,M(:,:),'delimiter', ' ', 'precision', '%.18f')
    pad     = fullfile(computerpad,sprintf('M12_%d',n),sprintf('D%d.txt',i1));
    dlmwrite(pad,D(:,:),'delimiter', ' ')
  end
end
```

D Output data

Total expected noise load for all enforcement points

ep	2. [%]	4. [%]	5. [%]	6. [%]	7. [%]	8. [%]
1	56,72	42,22	46,82	57,42	57,42	53,23
2	72,65	59,29	63,03	72,08	72,08	68,91
3	82,38	67,98	72,99	83,76	83,76	79,02
4	77,97	98,52	93,65	77,42	77,42	83,21
5	80,71	106,07	101,60	84,09	84,09	88,62
6	62,45	61,09	60,60	60,36	60,36	61,57
7	82,40	58,12	67,22	91,71	91,71	78,41
8	90,82	61,27	72,38	102,25	102,25	85,91
9	90,93	64,93	75,22	102,57	102,57	87,20
10	77,06	64,69	69,38	83,18	83,18	75,28
11	75,16	75,27	73,92	75,08	75,08	74,45
12	75,16	83,29	78,88	71,97	71,97	75,78
13	61,73	72,17	67,39	58,02	58,02	63,10
14	58,79	63,14	60,38	57,02	57,02	58,88
15	65,24	60,19	61,30	66,94	66,94	63,94
16	62,61	49,68	54,29	65,70	65,70	59,94
17	58,61	47,20	42,92	40,94	40,94	51,52
18	126,75	95,85	80,54	69,76	69,76	105,41
19	102,88	103,87	98,24	89,37	89,37	98,48
20	78,11	76,57	76,92	79,23	79,23	77,79
21	90,07	87,49	97,08	114,22	114,22	96,02
22	87,33	86,32	99,90	122,26	122,26	96,86
23	72,43	70,42	76,91	88,91	88,91	76,38
24	79,23	92,76	83,31	60,62	60,62	76,66
25	86,97	102,58	92,95	66,29	66,29	84,93
26	65,80	68,69	65,03	59,79	59,79	64,52
27	60,63	73,63	71,85	66,71	66,71	65,69
28	61,92	62,93	60,70	59,89	59,89	61,49
29	71,90	82,93	75,78	57,80	57,80	70,44
30	87,00	94,92	93,95	89,23	89,23	89,80
31	73,94	97,49	93,70	79,18	79,18	81,76
32	75,62	97,85	95,41	85,77	85,77	84,10
33	75,62	86,89	84,87	71,68	71,68	71,41
34	63,15	67,85	64,94	58,66	58,66	62,78
35	72,94	51,08	58,12	74,67	74,67	67,84
P_exc	2.49%	8.47%	16.26%	2.21%	2.34%	15.19%

Table 9: Results from check

ep		4
	original	$a=2$
1	45.06	42.22
2	62.01	59.29
3	70.88	67.98
4	94.60	98.52
5	101.08	106.07
6	61.20	61.09
7	63.00	58.12
8	67.17	61.27
9	70.02	64.93
10	67.14	64.69
11	75.51	75.27
12	81.91	83.29
13	70.21	72.17
14	62.48	63.14
15	61.33	60.19
16	52.25	49.68
17	48.41	47.20
18	98.59	95.85
19	105.24	103.87
20	78.71	76.57
21	89.56	87.49
22	87.45	86.32
23	72.36	70.42
24	90.41	92.76
25	99.82	102.58
26	68.11	68.69
27	71.01	73.63
28	62.61	62.93
29	80.93	82.93
30	93.39	94.92
31	92.93	97.49
32	93.55	97.85
33	81.61	86.89
34	67.19	67.85
35	55.33	51.08

P_exc 8.47% 13.79%

ep		7
	original	$a=2$
1	57.57	57.42
2	72.23	72.08
3	83.77	83.76
4	76.85	77.42
5	83.12	84.09
6	60.36	60.36
7	91.42	91.71
8	101.99	102.25
9	102.31	102.57
10	83.09	83.18
11	75.18	75.08
12	72.16	71.97
13	58.20	58.02
14	57.14	57.02
15	66.97	66.94
16	65.66	65.70
17	40.97	40.94
18	69.89	69.76
19	94.60	89.37
20	84.27	79.23
21	115.88	114.22
22	120.81	122.26
23	91.38	88.91
24	61.64	60.62
25	67.40	66.29
26	60.10	59.79
27	66.00	66.71
28	59.94	59.89
29	58.44	57.80
30	88.80	89.23
31	78.14	79.18
32	84.60	85.77
33	70.28	71.68
34	58.74	58.66
35	74.88	74.67

P_exc 2.34% 3.87%

ep		8
	original	$a=2$
1	55.61	53.23
2	71.20	68.91
3	81.40	79.02
4	79.80	83.21
5	84.22	88.62
6	61.83	61.57
7	82.34	78.41
8	90.66	85.91
9	91.28	87.20
10	77.25	75.28
11	74.66	74.45
12	74.70	75.78
13	61.54	63.10
14	58.37	58.88
15	64.87	63.94
16	62.06	59.94
17	53.82	51.52
18	111.76	105.41
19	98.72	98.48
20	78.00	77.79
21	95.86	96.02
22	96.19	96.86
23	76.30	76.38
24	75.03	76.66
25	82.97	84.93
26	64.22	64.52
27	63.31	65.69
28	61.36	61.49
29	69.05	70.44
30	88.41	89.80
31	77.68	81.76
32	80.14	84.10
33	66.60	71.41
34	62.27	62.78
35	71.40	67.84

P_exc 15.19% 22.61%

Table 10: Results from altering the strategy

E Figures and tables

List of Figures

List of Tables

F Profile LVNL

Luchtverkeersleiding Nederland (LVNL), or Air Traffic Control the Netherlands, is a non-profit organization engaged in the air traffic control of Dutch airspace and at Dutch airports. In this section a profile of the organization is given. In addition, the department of my internship and final thesis within the organization and its activities are discussed.

Brief history

LVNL originates from the directorate of the Department of Air Traffic Control (Luchtverkeersbeveiliging, LVB). In 1993, it was corporized, lifted from the Department of Civil Aviation (Rijksluchtvaartdienst, RLD), and continued as an Independent Administrative Body (Zelfstandig Bestuursorgaan). When the new building at Schiphol-Oost was put into use in 1998, the name was changed to LVNL.

In the Netherlands air traffic control exists since 1918 when two aircraft collided over Soesterberg. It initiated the introduction of air traffic regulation. Air traffic communication was provided by the military station at Soesterberg, later by the radio station in Rotterdam, and mainly consisted of information. Radar did not exist yet; keeping track of aircraft was done by verbal communication.

The RLD originates from the Department of Aviation (Luchtvaartdienst) that was founded in 1930 after the Aviation Act of 1927 was put into use. In 1940, the LVB was founded, which consisted of only three sub-departments: a message service provider, air traffic control and a meteorological unit. After the Second World War, air traffic increased rapidly. Nowadays air traffic is handled procedurally. Technological developments make it possible to guide aircraft using radar. At Schiphol new towers, runways and air traffic control systems are build to handle the traffic. In 2003, the Polderbaan, the latest expansion of Schiphol, came into use. Nowadays LVNL employs around 1000 people with a wide range of responsibilities.

Activities and responsibilities

The main activity of LVNL is to provide air traffic control services to civil air traffic in the area the Netherlands are held responsible for: the Amsterdam Flight Information Region (FIR). Other activities include the design, acquirement and maintenance of infrastructure to provide air traffic control, the distribution of aeronautical information, training of controllers and internal supervision of Safety, Efficiency and Environment. In short, LVNL is responsible for the control of civil airspace and everything that is related. All tasks have been laid down in the Aviation Act (Luchtvaart Wet).

Air traffic control is provided by order of the government. Performance and policy have to be reported to the minister of Transport and Water Management. LVNL serves as an important link between the government and other parties involved in air traffic control. Based on a dialogue with parties involved in air traffic, air traffic control is performed as well as possible, with a balance between Safety, Efficiency and Environment.

The operational activities of LVNL are performed 'op zaal' (in the hall) in the main building at Schiphol-Oost and in the towers at Schiphol and other Dutch airports. In the hall at Schiphol-Oost air traffic controllers direct the aircraft over the Netherlands (Area Control Center, ACC) and line them up for the runways (Approach, APR). In the towers at Schiphol (Tower, TWR and Tower-West, TWR-W) take-off, final approach and ground traffic is managed. All other departments are situated at the main building at Schiphol-Oost.

Positioning

Concerning the environmental restrictions LVNL is positioned as an equal party in the aviation sector by the Aviation Act. The aviation sector has the joint responsibility handle the traffic within the enforced limitations. The aviation industry consists of Amsterdam Airport Schiphol (AAS) and the airlines operating from Schiphol.

Philosophy

Services provided by LVNL are based upon a dialogue with its stakeholders. A balance between their sometimes conflicting interests and demands are taken in account. Decisions are made such that they are beneficial to all stakeholders.

LVNL operates as an independent, authoritative, professional service provider within the aviation industry. The position LVNL takes toward stakeholder demands with respect to Safety, Efficiency and Environment is transparently substantiated.

Schiphol serves as a mainport in the Netherlands, a junction in the national and international network of transportation of people, goods and services. It is of great importance for the Dutch competitive position in the world. LVNL fully and actively supports the mainport objective. Within the limiting conditions for Safety and Environment the optimal utilization of the airspace is aimed at.

In order to be well prepared for future changes and to be able to anticipate to them flexibly, a client-oriented organization with an efficient management is required. With a view to maintaining and extending existing knowledge and experience, LVNL also wants to deploy this in its role of consultant of foreign governments and sister organizations.

LVNL aims to be a significant and qualified player in the future European air traffic management system. The 'Single European Sky' is the relevant EU policy aimed at meeting future demands for air traffic. These demands are related to the capacity, cost-effectiveness and the environmental efficiency of increasing air traffic.

Collaborations

All LVNL activities are part of processes. There are three kinds: Primary, Supporting and Managing processes.

- Primary processes consist of all activities that contribute directly to the air traffic management services: 'Provide ATM service,' 'Modification ATM systems' and 'Maintenance ATM systems' are the activities in this process.
- Supporting processes consist of activities required to facilitate the Managing and Primary processes: among others this includes activities as 'Training new ATC staff' and 'Maintenance administrative workstations.'
- Managing processes direct the Primary and Supporting processes: 'Managing LVNL' is the main activity in this process.

Processes involve different departments, that carry out activities that contribute to the process.

Structure

An organogram of LVNL is shown in Figure 18, which depicts the structure of LVNL at department and directorate level. LVNL consists of four directorates managed by the board. These are: Air Traffic Management, Corporate Resource Management, Human Resource Management and Regional Unit. A description follows.

Board LVNL The board itself is assisted by four supporting departments: Corporate Development & Management Services, Legal Affairs, Public Affairs & Media Relations and Communications. Legal Affairs is mainly responsible for consultancy in a wide range of legal areas, such as administrative law, civil law, air (traffic) law, labor law, European and international law. It plays a major role in litigation and legislation procedures, closely cooperating with the Ministry of Transport, Public Works and Water Management and the Transport and Water Management Inspectorate. Whenever LVNL is in the news, the departments of Communication and Public Affairs & Media Relations are involved. Activities that are performed contribute in the Managing processes.

Corporate Resource Management The directorate Corporate Resources Management is concerned with the internal provision of services regarding housing, finance, information services, and third party acquirement of goods and services. The Purchasing department is the intermediary between the internal client and the external supplier. The department adds to safeguarding continuity, quality and the lowest integral costs for LVNL. The Information Technology department develops, implements and maintain all systems not directly involved in the operational system (air traffic control systems). These are supporting processes. This directorate employs about 100 people.

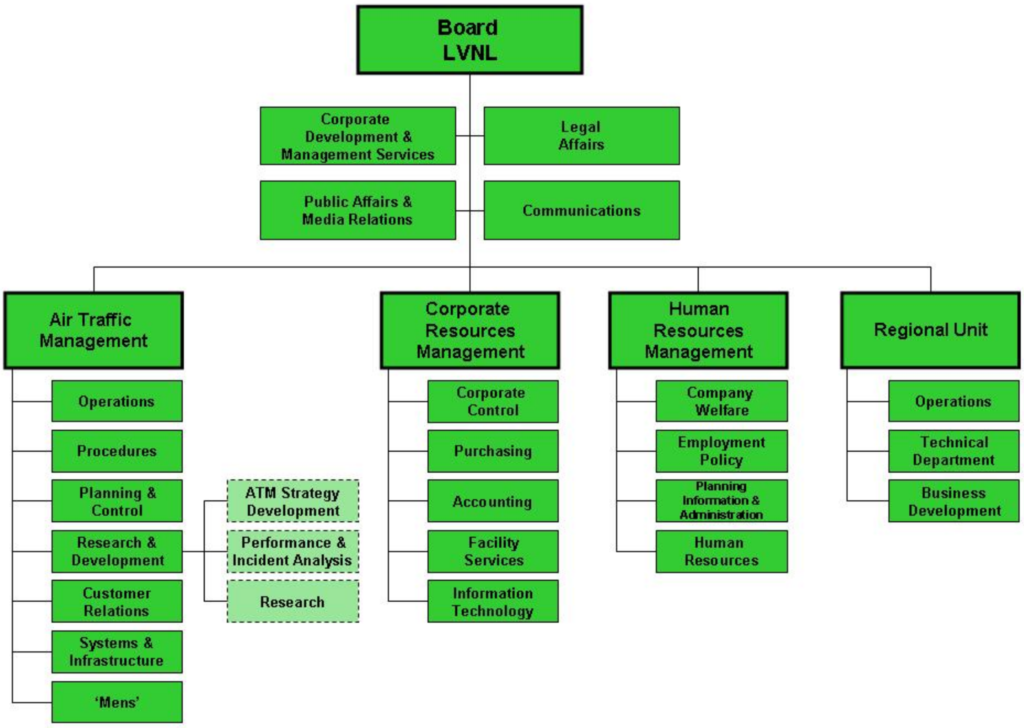


Figure 18: Organogram of LVNL

Human Resource Management The directorate Human Resources Management is concerned with the support regarding the development of Human Resources policy: quantitative and qualitative inflow of personnel as well as training and development possibilities for all personnel. It advises management in the implementation of the Human Resources policy and is responsible for the realization of this policy. These are also supporting processes. About 100 people are employed in this department.

Regional Unit The directorate Regional Unit is responsible for high quality ATM services on and around the three regional airports in the Netherlands[10]. Furthermore, the training of air traffic control personnel for ATM Schiphol, the provision of technical services to third parties, and the development of new services to third parties are important activities. It warrants the continuity of air traffic control services at the regional airports. This department employs about 100 people.

Air Traffic Management The directorate Air Traffic Management is responsible for high quality ATM-services in the Amsterdam Flight Information Region and in particular at and

[10]The regional airports in the Netherlands are: Maastricht Aachen Airport (Beek), Groningen Airport Eelde (Eelde) and Rotterdam Airport (Rotterdam).

around Schiphol. The Flight Service Center is also part of this directorate. Flight Service Center is involved in the pre-flight support of aircraft at Schiphol. The operational air traffic control system consists of three components that can be identified in the departments: 'human' (Operations department), 'machine' (Systems & Infrastructure department) and 'procedure' (Procedures department). These departments collaborate to create high quality ATM-services. The Operations department consists of the air traffic controllers. The Systems & Infrastructure department is concerned with the design and maintenance of computer systems and systems in the field, such as runways lights, cabling, etc. The Procedures department maintains and develops the operating procedures air traffic controllers work with. The management and development of ATM-concepts are also activities of this directorate. The Research & Development department is one of the responsible departments for such developments.

These ATM-concepts lead to tools that ensure the safe, efficient and environmentally friendly handling of air traffic. These tools can be divided into competent air traffic control personnel, supporting technical systems and procedures, and information concerning airport and airspace utilization. Department 'Mens' is concerned with the recruitment, selection and training of air traffic controllers. These activities all contribute to Primary processes. The Customer Relations department provides LVNL services and products to external clients. They act as client toward the internal organization, where the expertise of the services to be provided is at hand. Toward the external client they act as discussion partner and contact for the LVNL. The Air Traffic Management directorate is the biggest directorate in LVNL and employs about 650 people. This includes the 300 air traffic controllers, of which there are about 50 in the building at all time.

My place in the organization during my internship and final thesis was with Air Traffic Management. I worked at the Performance & Incident Analysis, which is a sub-department of Research & Development. These departments are specifically depicted in Figure 18 These departments are discussed in more detail below.

Department: Research & Development Research & Development studies new air traffic management concepts to meet future customer demands. It is occupied with inventing new solutions and ideas for the continuing improvement process of the primary LVNL product: air traffic control. It brings together the LVNL development department, controllers, stakeholders and knowledge institutes to define system solutions in a united effort. This is done in a safe and efficient way, in which LVNL operates within the environmental standards laid down by the government. In this way, it makes an important contribution to the development of systems at LVNL, in order to maintain a strong position of mainport Schiphol. There are approximately 40 employees in this department.

Sub-department: Performance & Incident Analysis Performance & Incident Analysis is a sub-department of Research & Development. It quantifies the factors Safety, Efficiency and Environment (SEE) and makes them insightful by conducting SEE Effect Reportings and maintaining the SEE Management Information Systems. Due to the complexity of these interrelated factors as well as air traffic control itself, this is a challenging task. The department is thus concerned with the performance analysis of the executed operation as well as the concepts for operation, which are provided by the sub-departments R&D/Research and R&D/ATM Strategy Development. Each Performance Expert in this department has a specific expertise corresponding to the factors. After a recent merger of departments, air traffic related incidents (e.g. proximity violations and runway incursion) are also investigated in this department. This department consists of 18 employees.

On a regular basis, experts from other companies are brought in to assist the research of this department. These are organisations like the Nationaal Lucht- & Ruimtevaart Labartorium (NLR, National Aerospace Laboratory) and Frontier Information Technologies (software company that delivers ATM related tools).

Culture

LVNL aspires to be a good employer for its employees. There is a general relaxed atmosphere at LVNL. The organization structure is flat, meaning that from any point in the organization it takes few steps to reach the board. The exchange of information and ideas is therefore easy and results in pleasant a working environment. A challenging, divers working environment and interest in aviation are the main reasons to work at LVNL for its employees. Many employees praise the great liberty in performing their duties and the good development opportunities.

G Reflection

In the period from 17 May 2005 till 17 April 2006 I carried out an combined internship and final thesis at Luchtverkeersleiding Nederland at Schiphol-Oost. My interests in the aviation business and air traffic control in particular originated in my high school years. In the first year of my studies I even applied for a job as an air traffic controller, but was rejected. I continued my studies in Applied Mathematics and hoped to be involved in the air traffic control in some other way later in life. The university offered some courses about the aviation industry, that I was glad to attend. When the final year of my studiesapproached, it was not hard to think of a place to perform my internship. It turned out to be no problem to find a suitable place for me at the department of Research & Development / Performance & Incident Analysis of LVNL.

Now that I had a found a place to complete my graduation project, I had to find a place to live and sleep. Traveling time between Enschede and Schiphol is 2,5 hours by train; it would not be possible to live in Enschede and work at Schiphol. A friend from high school provided me with a room in his house in Amsterdam. This room was very small and the house had numerous deficiencies. On more than one occasion did I have to use the neighbors' bathroom. Due to several change-overs of trains and busses, the traveling time to Schiphol-Oost was still 45 minutes by public transportation. Therefore I decided to cycle to my office if the weather was nice enough. Although it did not save me any time, I enjoyed the exercise and the thrill of traveling in a straight line. During my stay with LVNL I moved to a very pleasant student house in Leiden that was found through WWW.KAMERNET.NL. Traveling time remains the same, but cozy Leiden is certainly worth occasional delayed train. The bus journey from Schiphol to Schiphol-Oost takes me along the arriving and departing aircraft; these I can watch every day!

LVNL

My work place at LVNL was very complete; everything I needed was present. I could make free use of the copier, printer and coffee machine. A big desk, plenty of paper, pens and pencils, a computer, a telephone and a white-board were the items I used most intensively in my office. A disadvantage of the computer was that no additional software could be installed. I mainly used my own laptop computer on which relevant software was already installed, such as Matlab and a LaTeX-compiler. Also a link for downloading articles through the library of Twente could not be made. In one of the first weeks new computers were installed, running Windows XP, which made file transfer between the office computer and my laptop extremely easy. For the better part of my internship at LVNL I shared my work place with two employees of the NLR who temporarily worked for LVNL. One of them had studied mathematics in Twente, the early days; the other is a former pilot. The rest of my department was at the end of the hall, but they never forgot to call me when they went for lunch. In the last month I moved to a solitary work place directly opposite my direct colleagues, which finally included me in their coffee circuit.

Security at LVNL is very tight to prevent terrorist attacks. Preceding my stay at LVNL a background check was performed by the Ministery of Justice. Interns are given a temporary admission card that -unfortunately- has to be collected at and returned to the security desk in the main entrance hall every day. Frustratingly, the bus stop was right next to my work place at the rear of the building; every single day I had to walk up and down the building to deal with security and there was no room in the building farther away from the main entrance than mine. Furthermore, guests needed to be properly announced and accompanied through the building.

In the first week already a part-time controller from my department took me 'op zaal', the place the actual air traffic controlling takes place. I was introduced to several controllers who told me about their interesting line of work. I attended a 'vision' presentation of LVNL, that gave insight into the future development of LVNL. On another occasion I attended an information meeting on noise pollution for local residents in Nieuw-Vennep at which LVNL was present; this was directly linked with my project. All in all, I tried to experience LVNL as much as possible.

There were several academic interns active when I joined LVNL (with backgrounds in Economics, Communications and Business Management). An intern receives much freedom at LVNL. Although I signed a contract for a 36-hours week, attendance was never checked. A day off or doing some work at home could be done as pleased. LVNL is a 24-hour business; working late was never a problem. This was supported by an in-house restaurant offering tasty meals at dinner-time. Every intern is paid a very reasonable compensation. It is dependent on the level of education and the purpose of the internship. The payment is done at the end of each month, which coincided with the payment of my rent.

Project

The goal of a combined internship and final thesis is to apply the knowledge and skills acquired from the studies solving a problem in a practical working environment with adequate mathematical depth. My project was on noise load management at Schiphol. Noise load restrictions are a hot topic in the media; I was glad to work on a practical, real-life situation, for which results could have real meaning.

Using a variant of dynamic programming, I devised a model with which traffic can be handled more efficiently within the current restrictions. Defining and formulating the exact problem I thought was the hardest; I wanted to investigate too many aspects. In addition, I realized late in the project that too much time was lost with programming and calculation time of the implementation, whereas the project was aimed at a theoretical model, with only small-scale numeric results required for general insight. Planning all activities

I conducted two presentation during my stay at LVNL. The first after three months, when it was decided what model and methods would be used. The second at toward the end of my project, presenting the results of the implementation. These presentations forced me to think about my work critically and about the way to present my findings to a group of people without much mathematical background. Although my coach at LVNL had a mathematical background, it was still hard to convince him of some of my ideas. He taught me to properly prepare my work not just for big presentations by even for the smallest meetings.

Epilogue

Time has really flown during my stay with LVNL; I really enjoyed myself. I can therefore certainly recommend an internship at Schiphol-Oost. My advice is to start looking for a internship placement about half a year in advance. I think it is important to be active in a field you really like and therefore it is not a bad thing to be a little critical. As a mathematician, you can even afford to be picky about your internship, since there are a lot of places to choose from.

My graduation was in the form of a combined internship and final thesis. That equals 9 months of hard work! I thought it was a good idea to take a 3 week break in the middle, when I finished part of my research. This immediately alleviated the stress level. I recommend not to finish the graduation as fast as possible, but to relax a little in between.

My stay with LVNL has broadened my interest in the aviation industry. Currently, I am looking for a interesting job at Schiphol.