Thesis Applied Mathematics
Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS)

Analyzing the solution of a Linear Program

The effect of normal distributions in costs and constraints

Marjan van der Velde (s0166391)

Assessment Committee:
Prof. dr. R.J. Boucherie (UT)
Dr. ir. J. Goseling (UT)
Dr. ir. N. Gademann (ORTEC)
Prof. dr. J.L. Hurink (UT)

June 19, 2013

UNIVERSITEIT TWENTE.

ORTEC
ADDING CREDIBILITY

ABSTRACT

Suppose the optimization of distribution network can be modeled as a Linear Program. This work considers multivariate normally distributed cost and constraint vectors. A method is developed to compare alternative basic solutions on optimality, feasibility and outliers. The basic solution is used instead of the full problem because of the corresponding invertible data matrix. This work contributes in four ways. First, an overview is provided of methods that optimize Linear Programs under uncertainty or analyze its solution. As no current method has the desired properties, requirements for such a method are stated. Second, expressions are derived for normal distributions in the cost and constraint vectors. These provide probabilities for optimality, feasibility and outliers for a solution of a Linear Program. In that way, the robustness of a solution can be determined. Third, a method is developed to systematically evaluate solutions of a Linear Program for varying costs and constraint values. This method provides a comprehensive approach to compare alternative solutions on optimality, feasibility and outliers. Finally, the method is applied to a small test case and a real world fuel distribution test case. The results show that the obtained basic solution is robust and outperforms the alternative basic solutions under changes in the demand for fuel.

PREFACE

Can we estimate the effect of data changes in a Linear Program? If we decide on a strategy, what are the consequences on the costs? How can we decide what strategy to use given these consequences? These are the questions that ORTEC asked and that motivate this work. It is the result of my final project for the master's specialization of Stochastic Operations Research (SOR) within Applied Mathematics at the University of Twente (UT).

Six years ago I started studying at the University of Twente for a bachelor's degree. The reason for that was not in the beauty of fundamental mathematics, but in the wide range of applications. More specifically, in the ability it provides to analyze and structure the world. Out of all the Dutch universities Twente offered the most application possibilities therefore I decided to start my studies there. However, that broad range of applications available led to a serious decision problem. About once a year, I changed my opinion on which application is the most appealing. During the start of the economic crisis, I was fascinated by the world of financial products and confident that it would become my specialization. My minor a year later, however, was on traffic theory in Civil Engineering. However relevant those topics are, I then decided that the application of mathematics in health care was much more interesting due to its societal relevance. The common factor in these three applications is the modeling of uncertainty: we know that many results are uncertain, so we should anticipate this uncertainty and analyze it. This led to my choice of Stochastic Operations Research as a master's specialization. By that time, however, the Netherlands had become too small and I went to South-Africa to study mobile communication networks as an internship.

Now the time had come to choose a final project. After having tried all the above applications, my decision problem had hardly become smaller. However, all the studying so far had been at universities so I decided to conduct my thesis at a company. At ORTEC I was given the opportunity to study Supply Chain Optimization (yet another application) within the consulting department. During my stay there I got a valuable insight into working life, consulting and business.

A thesis, however, is not something that I can conduct on my own. At ORTEC, Noud Gademann supervised me and we had several brainstorm sessions, where the returning question was 'what is the practical value of this theory?' At the UT Richard Boucherie supervised me, where the returning question was 'what is the theoretical value of this?' The trade-off between those two was a challenge, but the result contains both. Jasper Goseling, my second UT supervisor, supported me in the last months of the project by balancing the priorities and helped me to find the balance between that two.

Such a long-term full-time project has to be altered by some social distraction. Fortunately, I could stay at the Boot family in Nootdorp who gave me a very pleasant stay. Furthermore, my mother and brother were of great value at the time I did not like my thesis anymore. But those are not the only people that supported me, I enjoyed being at ORTEC with the lovely colleagues. The last weeks I worked at SOR and that was very pleasant as well. Most of all I thank God, who made me and gave me all the talents I have. He has guided me all of my life, and will continue to guide me in the years to come.

Marjan van der Velde
Marjanvdvelde@gmail.com
June 19, 2013

CONTENTS

CHAPTER 1

INTRODUCTION

In this introduction we give a problem description in Section 1.1, then give a company description in Section 1.2. As this research is about Linear Programming, we give a summary of basic theory in Section 1.3. We then give the goals and contributions of this thesis in Section 1.4.

1.1 Problem description

Often decisions on long-term investments have to be made without a precise knowledge of the future. As the investment costs are usually high, information on potential changes in costs and demands is valuable. A typical example of a Supply Chain Optimization problem is in distribution networks. Consider the owner of a fuel company with several hundreds of customers. These can be gas pumps, companies or farms. There are several products that have to be delivered from a couple of possible depot locations. A decision has to be made regarding the used depots and the customers they serve. A strategic decision in this is the choice of depots. When a choice for depots is made, this choice should be robust for future changes in data and possible inaccuracy of data. Note that the optimality of a solution is not always the main goal of a Supply Chain study. In the fuel company example, the main question is to what extend the different scenarios lead to different costs: what is the potential of a change? Furthermore, the gathering of precise data and future estimations is expensive if possible at all. Thus, to what extend does the potential improvement depend on the precise knowledge of the data?

The above is an example, but it is typical for many companies. This thesis is therefore motivated by the following question:

> "Which influence has a change in data on the cost and feasibility of the solution of a Linear Program?"

Each Supply Chain Optimization problem has its own characteristics. However, there are similarities: uncertainty often arises in cost coefficients and demand values. Furthermore, often a couple of alternative solutions is available. In this work, we therefore develop a method to evaluate the impact of data uncertainty on one solution and a way to compare several alternatives.

As there are various approaches to incorporate uncertainty, the objectives have to be specified explicitly. The questions posed are mainly evaluations of alternative solutions:

1. Suppose we have chosen a solution that has optimal cost. Can we be sure that costs will not increase too much when data changes?
2. As data changes, it might be possible that the chosen solution does not remain feasible. For example, a factory has to produce more than its capacity. How sure are we that the chosen solution stays feasible under a change of data?
3. Suppose we have chosen a solution that is optimal using expected values of data. Can we give a guarantee that it will stay optimal when data changes? Thus, that although costs may rise, the costs are still lower than that of alternative solutions.

Note that 'solution' is a broad term. It both includes the strategic decisions involved and the exact values on an operational level. Therefore we can distinguish two parts of a solution:

- A **strategy** concerns the strategic decisions that are made. These include which depots are used, which flows are assigned, and where slack is available. A strategy does not include precise values of variables, but denotes which variables have a positive value and which have value zero.
- An **exact value solution** gives the precise numbers assigned to all variables.

In the above stated questions we thus evaluate different strategies. If we choose a strategy and data changes, the involved exact values have to be changed. We then ask ourselves how that influences cost, if such a change is possible and if the adjusted exact values are optimal.

1.2 Company description

This thesis is initiated by ORTEC, a large provider of advanced planning and optimization solutions and services. ORTEC was founded in 1981 and has offices all over the world with in total 700 employees. ORTEC develops advanced software solutions to support and optimize operational planning for several business applications. Amongst others these include workforce scheduling, vehicle load optimization, and fleet routing and dispatch.

Besides from these operational standardized software packages, ORTEC provides logistics consultancy for individual customers on a strategic or tactical level. This is supported by dedicated logistics decision support systems, developed to meet individual customer needs. Furthermore, ORTEC conducts network studies and has developed software tools which are developed for internal use.

This research is initiated by the consultancy business unit Consulting and Information technology Services (CIS). This department provides customized software solutions for a large number of customers, mainly in logistics. Furthermore, this department developed the software tool BOSS. This is an AIMMS interface dedicated to strategic and tactical distribution network decision making. It can model typical supply chains resulting in a Mixed Integer Linear Program (MILP) or a Linear Program (LP). Often the resulting model is intuitive, easy to adapt and solvable within a couple of minutes. Therefore the main question for this thesis is how we can investigate the effect of data uncertainty using the available deterministic models.

1.3 Basics of Linear Programming

Often, a distribution problem or supply chain optimization problem can be modeled as a Linear Program. In this section we now recall the basic theory of Linear Programming. This theory can be found in textbooks on Linear Programming, for example [45].

Suppose we want to optimize an objective value y, which is a linear combination of n decision variables which form a vector x. Each decision variable x_i has a cost coefficient c_i assigned, which form a vector c. This optimization is due to m constraints, which are linear combinations of the decision variables. The linear combinations are specified in the data matrix A and should be equal to a constraint vector b. Furthermore, all decision variables are nonnegative. Thus, we have the following problem:

$$\begin{array}{lrcl} \min & y & = & c^T x \\ \text{s.t.} & Ax & = & b \\ & x & \geq & 0 \end{array} \tag{1.1}$$

This is the Linear Program that is studied in the remainder of this thesis. However, in many applications not only equality constraints but also inequality constraints will arise. Such an inequality constraint is easily replaced by an equality constraint by adding an extra variable. This extra variable then represents the slack value of the constraint. Another generalization is the maximization problem. Suppose we want to maximize the function $y = c^T x$. That is the same as min $y = -c^T x$.

There are n variables that are due to m constraints. Without loss of generality, we can state that $m \leq n$ and that there is no redundancy in the constraints: suppose $n < m$. In that case, there are more constraints than variables. Then there is redundancy in the constraints: at least one of these constraints is a linear combination of the other constraints. Thus, it can be removed from the LP without a loss of information. In the remainder of this thesis, we therefore assume that this redundancy is not present.

Definition 1. *A solution x to (1.1) is a feasible solution if and only if:*

$$Ax = b, \;\; x \geq 0$$

A feasible solution is a vector x that satisfies all constraints. The feasible set consists of all possible vectors x which are feasible. As the constraints are linear, the feasible set is a convex set. As a consequence, a local minimum is also a global minimum. When the feasible set is not empty and bounded, the global minimum exists. Denote by y^* the optimal value of (1.1) and x^* the corresponding solution. There can be more then one possibility for x^*, but there is always at least one x^* that is an extreme point of the feasible set. As the rank of A is m, there are $n - m$ free variables: they can have any value.

Definition 2. *Consider a standard Linear Program. A subset B of the variables is a basic set if $|B| = m$ and if the columns A_j with $j \in B$ are linear independent. Furthermore, x is a basic solution if all x_i with $i \notin B$ have value zero.*

We can write the solution x^* such that there are m basic variables which have a value greater or equal to zero and $n - m$ non-basic variables which have value zero. Denote by B the set of indices corresponding to the basic variables and by N the indices of the non-basic variables. We can take the columns of A belonging to the basic variables and denote this as the matrix A_B. The columns corresponding to non-basic variables form the matrix A_N. We can make a vector of the basic variables x_B and a vector of non-basic variables x_N. The cost coefficients c can be split up in c_B and c_N. As the non-basic variables are all zero, we have

$$\begin{array}{rcl} c_B^T x_B & = & y \\ A_B x_B & = & b. \end{array} \tag{1.2}$$

The matrix A_B has m independent rows, and therefore m independent columns. A_B is invertible when the basic variables are chosen such that the columns are linear independent.

A common way to find the optimal solution for a Linear Program is the Simplex Method, which was designed by Dantzig [12]. Although in theory the computational complexity of this method is exponential, the average-case complexity is polynomial and therefore often used. Suppose each feasible basic solution is represented by a vertex. Then the feasible set is the convex set spanned by all these vertices. The Simplex Method jumps from vertex to vertex until it finds the optimal solution. The idea of the Simplex method that we rewrite problem (1.1) in an equivalent form that either suggests further reformulations or makes the solution obvious.

The reformulations consist of dividing the variables and corresponding columns of A in two sets: the basic set B and the non-basic set N. If we separate x and A into two sets consisting of basic and nonbasic variables and collect these on different sides of the equation, we can write

$$A_B x_B = b - A_N x_N. \tag{1.3}$$

As A_B is a square matrix with linearly independent columns, it is invertible. Thus, we can rewrite (1.3) into

$$x_B = A_B^{-1} b - A_B^{-1} A_N x_N. \tag{1.4}$$

Moreover, we can write y in terms of the basic and nonbasic sets:

$$y = c_B x_B + c_N x_N. \tag{1.5}$$

Combining (1.4) and (1.5), we can then write:

$$\begin{aligned} y &= c_B(A_B^{-1}b - A_B^{-1}A_N x_N) + c_N x_N \\ &= c_B A_B^{-1} b + (c_N - c_B A_B^{-1} A_N) x_N. \end{aligned} \tag{1.6}$$

Suppose that all elements of the vector $c_N - c_B A_B^{-1} A_N$ are nonnegative. In that case, we can minimize y over all choices of $x_N \geq 0$ by letting $x_N = 0$. Then for x to be feasible, x_B is fixed as $A_B^{-1}b$. Therefore, the following expression is known as the optimality criterion:

$$c_N \geq c_B A_B^{-1} A_N. \tag{1.7}$$

The simplex method exploits this knowledge to find the optimal solution. In case one or more non-basic variables do not satisfy (1.7), at least one of them should be in the basis. As the size of the basis is fixed to m, one of the variables that is currently in the basis has to be moved to the non-basic variables. A next basis is chosen until (1.7) is satisfied and thus the optimal solution is found.

1.4 Research goals and contribution

Assume a Linear Program with stochastic data A, b, and c. For A we have a probability density function $f_A(a)$, for b we have $f_b(b)$ and for c we have $f_c(c)$. In this thesis we study the influence that a change in data has on the cost and feasibility of the solution of a Linear Program.

In Chapter 2 we discuss the wide range of models and approaches that are available for Linear Programming under uncertainty. References are given for a more detailed description. The contribution in this chapter is the comparison of approaches in Table 2.1. The last column of this table describes the properties of a model that is able to answer the questions stated in Section 1.1.

As none of the described methods has these properties, we make and motivate assumptions in Chapter 3. The contribution in this chapter is the insight that the study of a basic solution more important than exact knowledge of values in a solution. We can then sharpen the questions of Section 1.1 to the obtain expressions for the performance measures. What is the probability that a solution is feasible? Suppose we have chosen a solution x^* that is optimal using expected values of data. Can we give a guarantee that it will stay optimal when data changes? Suppose we have chosen a solution that has an expected cost y^*. What is the probability that the realized cost y will be at most $\alpha\%$ more then the expected cost? In Section 3.5 we formulate these questions more precisely.

In Chapter 4 we derive these probabilities for general distributions in Section 4.1 and derive means and expectations for multivariate normal vectors in LP's in Section 4.2. These results are then combined in Section 4.3. The contribution in this chapter is in Lemma 3, Lemmas 5-8, Theorems 9-13 and Table 4.1.

In Chapter 5 we propose a method to systematically evaluate solutions of a linear program. It is summarized in Figure 5.1. We then illustrate how the theory is used in a small example. After that, the method is applied to a real-world case.

We conclude the work in Chapter 6. Finally, appendices are given. In Appendix A an approach is discussed that seemed promising, but turned out to have little practical value. In Appendix B details on the implementation are provided. An overview of used symbols can be found in Appendix C and finally a list of references is provided.

Summarizing, the main objectives of this study are:

1. Investigate which methods are available in the literature. Describe their properties and determine to which extend they are able to answer the questions as stated in Section 1.1.
2. Derive expressions for the following probabilities:
   a) As data changes, it might be possible that the chosen solution does not remain feasible: it does not satisfy all constraints. What is the probability that a solution is feasible:
   $$P(Ax = b, x \geq 0)$$
   b) Suppose we have chosen a solution x^* that is optimal using expected values of data. Can we give a guarantee that it will stay optimal when data changes? If X is the set of all feasible solutions calculate:
   $$P(y^* = c^T x^* \leq \min_{x \in X} c^T x)$$
   c) Suppose we have chosen a solution that has an expected cost y^*. What is the probability that the realized cost y will be at most $\alpha\%$ more then the expected cost:
   $$P(y \geq (1 + 0.01\alpha)y^*)$$
3. Develop a method to systematically evaluate solutions of a Linear Program for varying costs and constraint values.
4. Test the derived theory and method on a real-world case.

CHAPTER 2

LITERATURE: UNCERTAINTY IN LINEAR PROGRAMS

In this chapter we will discuss the wide range of models and approaches that are available for Linear Programming under uncertainty. Some will incorporate the uncertainty in the model before the optimization step, others evaluate the quality of the solution of a standard LP. Some are easy to implement, others require extensive calculations. The goal of this chapter is to name the most common approaches and give a quick insight in how they work. For a more detailed description of those methods we refer to the more appropriate literature. In Section 2.8, the approaches will be compared to see which approach is the most appropriate for the questions stated in the introduction of this work.

Uncertainty occurs when decision makers cannot estimate the outcome of an event or the probability of its occurrence [34]. There are various causes of uncertainty, which can be divided into three levels [29]. A common approach is to distinguish a strategic (long-term), tactical (mid-term) and operational (short-term) level. Data is estimated or an expected value is used, but in both cases the realization can be different from the used value.

According to [18], there are three basic methods to describe uncertainty in the optimization model:

- Bounded form: the value of a parameter will be within an interval.
- Probabilistic description: uncertainties are characterized by the probabilities associated with events.
- Fuzzy description: instead of probability distributions, these quantities make use of membership functions, based on possibility theory.

There are several applications where uncertainty in a network arises. Extensively studied is the supply chain. This is done in [11, 22, 29, 34, 36]. More uncertainty is within reverse logistics, as the demand and return can vary a lot [19, 30, 31, 35]. A field with a lot of uncertainty is chemical process scheduling. A review of the models used there is given in [18], and an example of such a model in [21]. Finally, power system networks are studied as they need to cope with future changes [10, 39, 43]. These are all papers where the network is optimized with uncertain data. In [15, 25, 26] approaches are presented to gather and classify uncertain data.

2.1 Robust Optimization

Robust optimization models uncertainty in a bounded form. The goal is to minimize the effects of disruptions on the performance measure and tries to ensure that the predicted and realized schedules do not differ drastically, while maintaining a high level of schedule performance. A solution to an optimization is considered to be solution robust if it remains close to the optimal for all scenarios, and model robust if it remains feasible for most scenarios [18]. The modeler has the capability to control the trade-off between cost and robustness [5].

There are several textbooks available on Robust Optimization. The following text is based on [2]. Consider an optimization problem where the objective function and its constraints are linear. As stated

in Section 1.3, we can write an LP in both equality and inequality form. Within Robust optimization the inequality form is used:

$$\min_x \{c^T x : Ax \leq b\},$$

where x, c and b are vectors and A a matrix. Now consider the case that the data A, b and c are not known exactly but that they can be all members of an uncertainty set U. Then the objective is not just to minimize $c^T x$, but to find x such that the value of the worst-case scenario is minimized. Thus, the best possible robust feasible solution is the one that solves the optimization problem

$$\min_x \left[\sup_{(c,A,b)\in U} \{c^T x : Ax \leq b \quad \forall (c, A, b) \in U\} \right].$$

As the vector c is part of the uncertainty set, both the objective and constraints are uncertain. We can rewrite this to have only uncertain constraints. This is often formulated in the following form, which is called the Robust Counterpart:

$$\min_{x,t} \left[t : c^T x \leq t, Ax \leq b \quad \forall (c, A, b) \in U \right].$$

Solvers like CPLEX can be used to find the optimum for x and t. Aside from the size of the problem, often a problem has some non-linearity. Then if the problem is still convex, there are efficient solution methods available. Many heuristics have been developed for problems which are hard to solve.

In Robust Optimization, a solution should be feasible for all possible realizations of the uncertainty set. The bigger this set is, the more conservative the solution will be. As stated by [4], each choice of an uncertainty set has a corresponding risk measure and vice versa. A larger uncertainty set may lead to higher costs, but to a smaller risk that the chosen solution is infeasible.

2.2 Stochastic Programming

Stochastic programming is an approach to model data uncertainty by assuming scenarios for the data occurring with different probabilities [5]. This can model the uncertainty more precise then by only using bounds on the data, but the computational complexity becomes challenging [18, 31]. Therefore, most stochastic programming approaches are only suited for a small number of scenarios [36].

This section is based on [16]. The common approach in this method is an inequality formulation of an LP:

$$\min_x \{c^T x : Ax \leq b\}$$

The real values of (A, b) are not known, but there is a finite set S of possible scenarios. These scenarios are constructed based on the probability distribution of A and b:

$$Pr\{(A, b) = (A^s, b^s)\} = p_s, \quad s = 1, \ldots, S.$$

To be able to solve such a problem, one should decide what an optimal solution looks like. A straightforward approach is replace $Ax \leq b$ by $A^s x \leq b^s$ for all s, so the solution satisfies all constraints in all scenarios and one has to solve a huge deterministic LP. The construction of this LP is easy, but due to the size it is difficult to solve. Another straightforward approach is to replace $Ax \geq b$ by the expected values of A and b, which gives a relatively small deterministic LP but gives a solution that will be feasible for only few scenarios.

An approach that explicitly incorporates risk is a probabilistic or chance constraint. Replace $Ax \leq b$ by $Pr\{Ax \leq b\} \geq \alpha$. Then $1 - \alpha$ is the maximal acceptable risk. There are various choices possible for this α. The total risk of infeasibility can be chosen, an individual constraint can get a maximum risk of being violated, or some joint constraints can be considered. Each individual or group of constraints can get a value for α assigned. A disadvantage of this is that the resulting LP is a possibly non-convex model, which is harder to solve.

Another Stochastic Programming approach is to impose a penalty when a constraint is violated. Thus, risk is not taken as a qualitative but as a quantitative approach. A common method for this is a two-stage recourse model, where corrective actions if a constraint is violated are stated explicitly. For each scenario s, a cost y^s is imposed if the constraints $A^s x \leq b^s$ are violated. This cost specified by W. Thus, for each scenario s we have the constraint $A^s x + W y^s \leq b^s$. Then the recourse model for discrete distributions is the following:

$$\min_{x,y^s} \left[c^T x + \sum_{s=1}^{S} p_s q y^s : A^s x + W y^s \leq b^s \ \ \forall s \right]$$

Thus, risk is taken care of explicitly, but again this model is difficult to solve.

The chance constraints and recourse models have a different model for risk: the one qualitative and the other quantitative. However, in a real-world case both types of risks might be good to involve. Thus, some constraints can have a penalty if violated and others a maximal acceptable risk of being violated. These can be combined using integrated chance constraints. Thus, instead of the constraints $A_i x \geq b_i$ for all constraints i we replace them with

$$\int_0^\infty Pr\{A_i(\omega)x - b_i(\omega) > t\}dt = E_\omega \left[(A_i(\omega)x - b_i(\omega))^+ \right] \leq \beta_i$$

where ω is the random variable representing the uncertainty and β_i is a positive risk aversion parameter to be specified in advance. Then β_i can be chosen in such a way that it represents a penalty or a chance of violation. The computation of this expectation is not easy in general, but not more difficult then in the chance constraint optimization or the recourse models. Models with integrated chance constraints are convex in general, and in many cases a reduced form is known for the corresponding feasibility sets. In principle these models can therefore be solved by any standard optimization algorithm for (non-linear) convex problems.

A more general form of a chance constraint is a joint risk constraint, as studied in [24]. For one or multiple constraints we can formulate it as

$$\rho^m(Ax - b) \leq 0$$

where m is the dimension of the measure. Then an example of this measure is the probability that a constraint is violated, but there are more possibilities.

2.3 Fuzzy Programming

A third approach to handling uncertainty is based on fuzzy set theory. Possibility distributions can be used to model the lack of knowledge on certain parameters [30, 39]. Their limitations are related to the simplicity of the production/distribution models usually used [22]. Some constraint violation is allowed and the degree of satisfaction of a constraint is defined as the membership function of the constraint [18]. Fuzzy models are suggested to be combined with simulation [29].

Fuzzy programming can be used if constraints or data are not crisp values, but are somehow distributed around a value. Fuzzy modeling is mostly used for linguistic uncertainty [29]. For example consider the cost of the production of one unit. A linguistic uncertainty would be that 'we prefer the costs not to exceed 35', and a strict constraint would be that $x \leq 40$. This linguistic uncertainty usually arises in the modeling phase. While a we need a sharp value to write a constraint, the modeler is not sure which value to use. At this point a fuzzy constraint can be introduced. Membership functions μ are introduced to specify the extent to which a constraint is satisfied. In such a function, zero means strongly violated and one means accepted, and the values in between the level of violation. In the above example, $\mu = 1$ corresponds to a cost of at most 35, while $\mu = 0$ corresponds to a cost of at least 40. Since this fuzziness can be described with any monotonous function, there are various models

possible. Such a model should then be translated into a crisp model, that can be solved using regular optimization techniques.

As an example [42], consider the case where the membership function is either zero, one or linear and where the values of the constraints are fuzzy. Thus, in the standard LP min $_x\{c^T x : Ax \leq b\}$ we replace each b_i by b_i' and b_i'', where b_i' is the preferred value and b_i'' the maximum value. In the above example, this would mean $b_i' = 35$ and $b_i" = 40$. The linear membership function $\mu_i(A_i x)$ is then for each constraint i:

$$\mu(A_i x) = \begin{cases} 1 & \text{if } A_i x \leq b_i' \\ 1 - \dfrac{A_i x - b_i'}{b_i'' - b_i'} & \text{if } b_i' \leq A_i x \leq b_i'' \\ 0 & \text{if } A_i x > b_i.'' \end{cases}$$

For the objective value such a membership function has to be constructed as well: the value $c^T x$ is expected to be at least z' but at most z''. The membership functions are then inserted in the original LP min $c^T x$ s.t. $Ax \leq b$. Then the problem to solve can be formulated as

$$\begin{array}{lrcll} \max & \alpha & & & \\ st. & c^T x & \leq & z'' + \alpha(z' - z'') & \\ & A_i x & \leq & (1-\alpha)b_i'' + \alpha b_i' & \forall i. \end{array}$$

This is a problem which has only one variable and constraint more then the non-fuzzy variant. Once one has described the desired membership functions, these should be transformed into a crisp model that can be solved. A major drawback of fuzzy programming is the simplicity of the models that are usually used [22]. When one wants to use more divers or complicated membership functions then just linear or triangular functions, the translation into a crisp model becomes very difficult.

2.4 Sample Average Approximation

The scenario set used in Stochastic Programming in Section 2.2 can be infinitely large and thus the problem becomes too big to solve efficiently. Optimizing for individual scenarios, however, can give solutions that are not a global optimum or different scenarios can give totally different solutions.

Therefore, a sample average approximation (SAA) method can be used. A random sample of scenarios is generated and the expected value function, which is hard to compute in the stochastic programming techniques, is approximated by the corresponding sample average function. The obtained sample average optimization problem is solved, and this procedure is repeated until a stopping criterion is satisfied [17].

Thus, it is comparable to the Stochastic Programming approach. The difference is that the model is solved for only a few scenarios at once, instead of all scenarios at once. This is done repeatedly for samples of scenarios that are randomly chosen. In [36] it is claimed that solving the SAA problem repeatedly with independent samples solves the problem more efficiently then by increasing the sample size. In [17] the authors prove that the SAA scheme converges with probability one to the optimal solution of the original problem. The procedure is summarized in [38].

2.5 Propagation of errors within a Linear Program

Already in 1966 Prékopa [32] addressed the influence of random variables in a LP. He assumed that A, b and c are random in the standard linear problem:

$$\begin{array}{rcl} y & = & \max c^T x \\ \text{s.t.} & & Ax = b \\ & & x \geq 0 \end{array}$$

Now assume that A is a square matrix and nonsingular. Then the inverse exists and thus the optimal value for y is given by:

$$\mu = c^T A^{-1} b$$

Thus, y is a function of the variables A, b and c. Then if $R = A^{-1}$ and we change the element a_{ik} to $a_{ik} + \xi$ it follows from matrix theory that R' is given by:

$$R' = R - \frac{\xi}{1 + r_{ki}\xi} \begin{pmatrix} r_{1i} \\ \vdots \\ r_{mi} \end{pmatrix} (r_{k1} \cdots r_{km})$$

Assume each element a_{ij} has a corresponding change ξ_{ij}. Then Ξ is the matrix denoting the changes in A and vectors γ and β are given for changes in c and b. By introducing derivatives of y with respect to a_{ik}, b_i and c_i a second-order Taylor expansion is given for the difference in the expected value for y and the achieved value. The variance of this difference is also computed. The first and second moment are needed for this. Conditions are given for which the optimal value of y has an asymptotic normal distribution.

In 1976, Bereanu [3] addresses the continuity of the optimal value in a LP. A regularity condition is stated and proven which is sufficient but not necessary for the optimal value to be defined and continuous in any open subset of the space where A, b and c are defined. However, this does not assure the continuity of the set of optimal solutions: when the optimal value changes little, it might need a totally different solution to achieve that value.

Suppose that we introduce a probability measure on the space of parameters. Let $A(\xi)$, $b(\xi)$ and $c(\xi)$ be random matrices on a given probability space. Then the stochastic linear problem is given by

$$\begin{array}{rrcl} & \gamma(\xi) & = & \sup c(\xi)^T x \\ \text{s.t.} & x & \in & X(\xi) \\ \text{where} & X(\xi) & = & \{x | x \in R^n, A(\xi)x \leq b, x \geq 0\} \end{array}$$

The distribution problem consists of finding the probability distribution and/or some moments of $\gamma(\xi)$, subject to some a priori probability distribution of $\mathscr{A} = (A(\xi), b(\xi), c(\xi))$. The stochastic linear problem has an optimal value if and only if the following implications are valid almost everywhere:

$$\begin{array}{rcl} (A(\xi)v \leq 0, v \geq 0) & \Rightarrow & (c(\xi)v \leq 0) \\ (wA(\xi) \geq 0, w \geq 0) & \Rightarrow & (c(\xi)w \geq 0) \end{array}$$

Furthermore, it is called positive if almost everywhere the columns of $A(\xi)$ and $b(\xi)$ are semi positive vectors. In that case and if the random vector ξ has an absolutely continuous distribution, then $\gamma(\xi)$ is an absolutely continuous, random variable. Other papers that cite this result point out that the conditions are rarely satisfied – and therefore often not applicable. Thus, it is mainly of theoretical value.

When binary or integer variables are involved, the problem becomes harder. Even in recent papers, "finding the exact distribution for general mixed 0-1 LP problem appears to be impossible" [28]. Either the number of random variables should be low, or only one of the matrices A, b and c should be random, or approximation techniques have to be used. For example, in [27] the persistence value is studied. This is the probability that the value of a variable stays optimal when parameters change. Thus, it is a measure how good a solution is given uncertainty. A method is given to find the persistence value of a model where the values of c are random and the values of x are integers in a specified range. According to the paper, this probability that the optimal solution will be of the same shape, can be approximated by formulating a concave maximization model. However, most papers have their main focus on the distribution of the optimal value and not on the persistence problem.

2.6 Bounds on the optimal value

The approaches named so far all assume one particular optimal solution x^*. The reason is that only one solution can be implemented in practice: we cannot switch strategy at every instance. Thus, we have to choose one solution and can then analyze how good or bad it performs under changing circumstances.

A means for that is to calculate the optimal cost of each possible scenario as if it would be deterministic. If then one solution is chosen to be implemented, it can be compared to these values. This does not give us a direct method to find an optimal solution x^*, but it does give us an impression how well a solution performs under uncertainty.

Assume we want to choose a solution x which is chosen based on the deterministic data A and b, and a probability distribution of c. Thus, we do not know the realization of c when choosing x. Using that information, can we give a bound on the optimal objective value? Thus if you have some fixed feasible x, how good is it compared to the fictional situation that we can adapt the solution at every instant of time? A clarifying example for this is given in Figure A.1 in Appendix A.

A bound that is explained in [40] is the Dyer-Frieze-McDiarmid inequality, which was first introduced in [13]. Assume that the cost coefficients $c_1, \ldots, c_n$ are independent nonnegative random variables. Furthermore, assume that a constant $\beta \in (0, 1]$ exists such that the following relation holds for all j and all $h > 0$ with $P(c_j \geq h) > 0$:

$$E[c_j | c_j \geq h] \geq E[c_j] + \beta h. \tag{2.1}$$

As an example, if $c_j \sim U(0, 1)$ then $\beta = 1/2$, and if $c_j \sim exp(\lambda)$ then $\beta = 1$. If we could change our solution at every instant of time, this would give a stochastic objective value y_{DFM}. The information on the conditional expectation is used in the proof of the following bound:

$$\beta E[y_{DFM}] \leq \max_{S:|S|=m} \sum_{j \in S} x'_j E[c_j] \tag{2.2}$$

Thus, the set S consists of all combinations of x'_j such that their cardinality is the size of the basis. The proof of this bound combines (2.1) with the optimality criterion (1.7) from the simplex method. Note that this bound does not give the optimal value, nor the optimal solution that corresponds to it. In this thesis we examined the practical value of the DFM-bound and investigated to which extend the techniques in the proofs can be used to find a robust solution to an uncertain LP. The result of this, however, is that the DFM-bound is not of use in this work. Details on this are given in Appendix A.

2.7 The Dual

In several Linear Programming techniques the dual of a Linear Program is considered. The following information is summarized from [6]. The dual of our standard LP (1.1), which is also named the primal problem, is given by:

$$\begin{array}{lrcl} \max & v & = & b^T w \\ \text{s.t.} & A^T w & \leq & c \\ & w & & \text{unrestricted} \end{array} \tag{2.3}$$

Note that the same data A, b and c are used as in the standard LP, they only have changed position. There is one dual variable w_i for each primal constraint, and one dual constraint for each primal variable. The dual variables w_i correspond to the shadow prices of (1.1). Furthermore, each LP has a unique dual and the dual of the dual is again the original LP. As the dual is again a Linear Program, it is of great value in an efficient calculation of the solution of the primal problem: the dual value v and the primal value y are equal if and only if v and y are optimal.

According to [23], the various approaches proposed in Robust Optimization can roughly be divided into two distinct categories, depending on whether the underlying uncertainty model refers to possible

fluctuations on the row vectors of the constraint matrix (we call this 'rowwise uncertainty'), or on column vectors (we call this 'columnwise uncertainty'). Most robust techniques use rowwise uncertainty, while a dual formulation could be useful for columnwise uncertainty or for uncertainty in the constraint vector b. The author, however, shows that one cannot use standard duality theory to convert a columnwise uncertain robust Linear Program into a rowwise uncertain robust Linear Program while preserving equivalence.

In [38], the authors consider the use of duality in Stochastic Programming. As mentioned before, a stochastic problem is of large computational complexity. To make up for that, the Sample Average Approximation was proposed. The authors propose a heuristic that solves a group of scenarios in an SAA iteration by using duality decomposition as proposed in [9] for individual scenarios. This is then used to find an initial search direction to solve the group of scenarios together. Concluding we have that the dual of a Linear Program is not a standalone method to incorporate uncertainty in an LP, but it can be used as a tool within other methods.

2.8 A Comparison of Approaches

In this section we will compare the approaches and models that have been explained in Chapter 2. The properties that will be compared are the following:

1. **Exact or not.** Determine how the solution or value given by the approach is calculated. Is there a way to give an exact answer? If not, are efficient heuristics or approximations available? Another possibility is that the solution or value is obtained by simulation. Finally, a bound on the the solution or value can be given.
2. **Division in subproblems.** Often, we have a deterministic Linear Program and some measure of uncertainty on the corresponding data. Thus, we have to formulate criteria or risk measures and we have to solve a Linear Program. There are various approaches in this. We can first formulate criteria or risk measures and then incorporate these in the Linear Program and optimize. We can also first solve the Linear Program and then analyze the effect of uncertainty on this solution.
3. **Numerical effort.** The computational complexity is discussed here.
4. **Recognizability of LP.** If no uncertainty was present, we would have to solve a deterministic Linear Program. As uncertainty is incorporated, the problem is rewritten to an optimization problem that includes the uncertainty. How much rephrasing is necessary? Can we recognize the original LP in the rephrased optimization problem? This is of practical value as a modeler often adjusts the model in some iterations to validate it.
5. **Type of uncertainty.** What kind of uncertainty can be modeled? A bound on the values, a probability distribution, moments of a probability distribution or some other kind of information.
6. **Type of optimization model.** This work is on the standard Linear Program. In some applications, however, Integer Linear Programs (ILP), Mixed Integer Linear Programs (MILP) or other types of optimization models can be more appropriate. To what extend can these be used?
7. **Place of uncertainty.** Uncertainty can be present in the data in A, b and c. However, not all models can deal with all of them. At what places can we have uncertain data?

The approaches are compared on these properties in Table 2.1. The approaches that have been researched most are those where risk measures are defined a priori. Then the LP is rewritten to include these risk measures and the resulting optimization problem is solved. However, these techniques give one optimal solution. As said in Chapter 1, we want to compare the risks of alternative scenarios.

Techniques that analyze a particular solution are less extensively researched. Furthermore, many of those approaches focus on special cases and not on a generic approach. Thus, we want to develop a generic method that can be used for multiple cases.

	Robust Optimization	Stochastic Programming	Fuzzy Programming	Sample Average Approximation	Propagation and Persistence	Bounds on the optimal value	Required in this project
(1) Exact or not	Exact for small problems, heuristic for large instances.	Exact for small problems, heuristic for large instance.	Dependent on the uncertainty modeling. For easy uncertainty and small models exact, otherwise heuristics.	Simulation with guaranteed convergence.	Exact or approximation.	Bound.	Exact for special cases, efficient simulation.
(2) Division in subproblems	Give bounds on data, give robust LP formulation, calculate Robust Counterpart, optimize.	Give bounds or penalties on constraints, formulate deterministic equivalent, optimize.	Give fuzzy description, formulate deterministic equivalent, optimize.	Give bounds or penalties on constraints, then optimize by iterating over samples of scenarios.	Solve LP, analyze solution.	At once: calculate bound of optimum of all scenarios.	Give probabilities on data, generate alternative solutions, compare alternative solution on optimality, feasibility and outliers.
(3) Numerical effort	Extensively researched, many efficient techniques available.	More complex then RO, but still efficient techniques available.	Heavily dependent of the uncertainty model. Techniques available for easy models.	Faster then SP, adjustable by sample size.	No generic techniques.	No generic techniques.	Efficient technique available
(4) Recogniz-ability of LP	Depends on the amount of uncertainty. Reformulation always necessary, but fairly recognizable.	Depends on the amount of uncertainty. Reformulation always necessary, large increase in problem size.	Hardly recognizable, complex reformulation.	Dependent on sample size, reformulation always necessary.	Original LP.	Original LP.	Basic solutions: use part of LP
(5) Type of uncertainty	Bounds.	Probability distribution.	Possibility distribution: 'about' description.	Probability distribution.	First and second moment.	Some probability distributions.	Probability distribution
(6) Type of optimization model	LP, ILP, MILP, and more.	LP, ILP, MILP, and more.	LP, ILP for some possibility distributions.	LP, ILP, MILP, and more.	LP and special cases.	LP.	LP, possibly (M)ILP
(7) Place of uncertainty	A, b, c.	A, b, c.	b, c.	A, b, c.	Depends per model. More randomness gives less possibilities for analysis.	c.	b , c

Table 2.1: The properties of several models and approaches that deal with uncertainty in Linear Programming.

CHAPTER 3

METHOD AND ASSUMPTIONS

In this chapter we will formulate and motivate the assumptions that are made for the remainder of this thesis. After that, the questions stated in Section 1.4 are considered again. The assumptions made are the following:

1. **A basic solution.** The knowledge of a basis B is enough to determine the corresponding strategic and tactical decisions. The exact values of the variables are mainly of operational value.
2. **Uncertainty in b and c.** Uncertainty is assumed to be in b and c. Uncertainty in distribution network studies often arises there and uncertainty in A can be transferred to b.
3. **Normality assumption.** Uncertainty in b and c is assumed to have a normal distribution.
4. **LP assumption.** An LP is used to model the distribution problem.

These assumptions are evaluated in Section 6.2. Furthermore, possibilities for generalization are given there.

3.1 A basic solution

In the Section 1.3 we discussed the basic theory of Linear Programming. One of the goals of this thesis is to apply this theory in Supply Chain optimization problems, with a focus on distribution networks. We will now discuss what information we need to answer the related questions.

Consider the following typical example. There is a large number of customers that have to be served from a couple of depots. There are k customers and each customer i has a demand d_i. There are l depots and each depot j has a maximal capacity f_j. All customers can be served from all depots. Then the flow from depot j to customer i is given by decision variable x_{ij}. This x_{ij} is called a flow variable. Each combination of customer and depot has a cost associated. We want to minimize costs. The resulting LP then is the following:

$$\begin{array}{rcll} \min y & = & c^T x & \\ \text{s.t.} & & \sum_{i=1}^{k} x_{ij} \leq f_j & \forall j \\ & & \sum_{j=1}^{l} x_{ij} = d_i & \forall i \\ & & x \geq 0. & \end{array} \tag{3.1}$$

By introducing l slackvariables we can rewrite (3.1) to an equality problem:

$$\begin{array}{rcll} \min y & = & c^T x & \\ \text{s.t.} & & \sum_{i=1}^{k} x_{ij} + s_j = f_j & \forall j \\ & & \sum_{j=1}^{l} x_{ij} = d_i & \forall i \\ & & x \geq 0. & \end{array} \tag{3.2}$$

This problem has $k+l$ constraints and $k \cdot l + l$ variables. If we optimize this using the simplex method, this will lead to a basis of $k + l$ variables. At most $k + l$ will have a positive value, the others have value zero.

Now consider the options for a variable:

- A variable is non-basic. If it is a flow variable, it means that customer i is not served via depot j. If it is a slack variable, it indicates that the corresponding depot j is used to its full capacity.
- A variable is basic. If it is a flow variable, it indicates that there is a flow assigned from depot j to customer i. If it is a slack variable, it indicates that not all capacity of depot j is currently in use.

In most distribution network studies an advice is desired about strategical or tactical questions. For example what depots should be used, and which customers should be served from what depot. The exact quantities are less important, since they may vary operationally. Thus, if we know which variables are in the basis and which variables are not, we can take the strategical and tactical decisions. Recall from Section 1.1 that a solution contains both a strategic information and exact values. We are in this work mainly interested in the strategic information of the solution. The exact values are, however, useful for operational decisions and to investigate the available slack of resources.

3.2 Uncertainty in b and c.

There are three data components in a standard Linear Program: A, b and c. From this data we derive x and y. Studies often address randomness in only one or two of these data components. In this section we will show how uncertainty can be transfered from one data component to another.

From c to A. Suppose we have a probability distribution P_c for the cost vector c. Define a new variable x_{n+1} with cost coefficient c_{n+1}. Then define as new data A', b' and c':

$$A' = \begin{bmatrix} A & 0_m \\ c^T & -1 \end{bmatrix} \quad b' = \begin{bmatrix} b \\ 0 \end{bmatrix} c' = \begin{bmatrix} 0_n \\ 1 \end{bmatrix}.$$

In this expression, 0_i denotes a column vector with i zeros. Now c' is a deterministic vector and the uncertainty is transferred to the last row of A'.

From b to A. A similar approach can be used for uncertainty in b. Define a new variable x_{n+1} with cost $c_{n+1} = 0$ and add a constraint to ensure $x_{n+1} = 1$. The right-hand side uncertainty in b is now transferred to the last column of A':

$$A' = \begin{bmatrix} A & -b \\ 0_n^T & 1 \end{bmatrix} \quad b' = \begin{bmatrix} 0_m \\ 1 \end{bmatrix} c' = \begin{bmatrix} c \\ 0 \end{bmatrix}.$$

From A or b to c. This is generally not possible. The data in A and b deal with the feasibility of a solution. Thus when data in A or b changes, the corresponding constraints change. As these need to be satisfied, they cannot be transferred to the cost vector. With the dual problem (2.3) it is possible to interchange b and c, but the interpretation of the dual variables is different from the interpretation of the variables of the primal problem.

From A to b. The uncertainty in A can be transferred to b by using the inverse of A. Recall from Section 1.3 that we denote the basic matrix of A by A_B. As a basic matrix A_B is invertible, the inverse A_B^{-1} exists. Decompose A_B in $A_B = A_B^0 + A_B^\delta$: a deterministic part A_B^0 and a stochastic part A_B^δ. Then we have the following relation:

$$\begin{aligned} x &= A_B^{-1} b \\ A_B^0 x &= A_B^0 A_B^{-1} b. \end{aligned}$$

Thus we can rewrite the basic problem (1.2) to:

$$A'_B = A_B^0, \quad b' = A_B^0 A_B^{-1} b, \quad c'_B = c_B.$$

The general case A is more complicated as this matrix is possibly non-square and thus the inverse need not exist. It might be possible to use some generalized inverse G to extend the above manipulation

to all matrices A. However, we do not elaborate on this as it is not the core of this work.

The use of stochastic A_B^{-1} is not straightforward. If A_B is stochastic, the full rank of A_B is not guaranteed anymore. Neither is then the existence of the inverse A_B^{-1} guaranteed. The effect of perturbations on a matrix has been studied extensively in [1]. However, this is not the main of focus of this thesis and thus we do not further study the impact of a stochastic matrix A. In case For completeness we include stochastic A in the remainder of this work, but will assume we can transfer the uncertainty to b and c. In that case, the remaining matrix A_B is deterministic and always invertible.

3.3 Normality assumption

In the next chapter we will derive expressions for feasibility, optimality and outliers. Numerical results can only be given if a probability distribution of the uncertain data is specified. A natural choice for this is the normal distribution. Most of the probability mass will be around the estimated value, while values further away from the estimated value have a lower probability. Furthermore, we have the Central Limit Theorem which states that under certain conditions the mean of a sufficiently large number of independent random variables will be approximately normally distributed. Thus if the data considered exists of many independent components, we can assume that a normal distribution is a good approximation.

3.4 LP assumption

In the remainder of this work we will consider LP-formulations only. However, realistic problems can include some integer or binary variables. In Section 6.2 we discuss how this work can be extended to more general (M)ILP problems.

3.5 Rephrasing research questions

In Section 1.4 we stated the research goals of this thesis. With the assumptions of this chapter we thus derive in Chapter 4:

1. The probability that a basis B is feasible:
$$P\left(A_B^{-1}b \geq 0\right). \tag{3.3}$$
2. The probability that a basis B is optimal:
$$P\left(y_B \leq \min{}_{B'}\{y_{B'} | y_{B'} \in \mathbb{B}\}\right). \tag{3.4}$$
3. The probability distribution of y. Then we can calculate the probability that under a basis B the cost y exceeds a value y':
$$P\left(y_B \geq y'\right). \tag{3.5}$$
4. The probability that the cost y under a basis B is less or equal to the cost y' of some other basis B':
$$P\left(y_B \leq y_{B'}\right). \tag{3.6}$$

CHAPTER 4

NORMAL DISTRIBUTIONS IN A LINEAR PROGRAM

In this chapter we derive expressions for the research questions as stated in Section 3.5. In Section 4.1 this is done for the standard Linear Program for probability distribution for b and c. If a generalization is straightforward, it is also done for stochastic A. We will, however, restrict ourselves to basic solutions. In Section 4.2 the properties of a normal distribution in b and c will be studied. These results will then be combined in Section 4.3. There we provide a Monte Carlo method to calculate the probabilities and provide exact answers for some special cases.

In this chapter we will extensively use the properties of A_B and the Linear Transformation Property. These are given in Lemma 3 and 4:

Lemma 3. *Assume B is a basis to a standard Linear Program. Then we have:*

1. *The matrix A_B is square.*
2. *The matrix A_B is of full rank.*
3. *The inverse A_B^{-1} exists.*

Proof. From Definition 2 we have that $|B| = m$ and therefore A_B has m columns. From the LP formulation (1.1) we have that the LP has m constraints and therefore A_B has m rows. Thus, A_B is square. Furthermore, we assumed that the LP has no redundancy in its rows and therefore the row rank of A_B is m. As by Definition 2 the column rank is also m, the matrix A_B is of full rank. A standard result from Linear Algebra is that a square matrix of full rank is invertible, and therefore A_B^{-1} exists. □

Lemma 4. [41] Linear transformation of a multivariate normal random vector. *Let X be a $K \times 1$ multivariate normal random vector with mean μ and covariance matrix V. Let E be an $L \times 1$ real vector and F an $L \times K$ full-rank real matrix. Then the $L \times 1$ random vector Y defined by*

$$Y = E + FX$$

has a multivariate normal distribution with mean

$$E[Y] = E + F\mu$$

and covariance matrix

$$Var[Y] = FVF^T.$$

4.1 Optimality and feasibility under stochasticity

In this section we will derive general expressions for the probabilities stated in the research questions on page 17. Recall from Section 1.3: there is always a solution that is both basic and optimal. The corresponding matrix A_B is square and of full rank and therefore invertible. Therefore, we have the following relations:

$$\begin{aligned} x_B &= A_B^{-1}b \\ y &= c_B^T x_B \\ &= c_B^T A_B^{-1} b. \end{aligned} \tag{4.1}$$

Assume A is stochastic and has a probability density function $f_A(A)$. Furthermore, assume that the inverse of the basic matrix A_B exists and has a probability density function $f_{A_B^{-1}}$. Assume b is stochastic and has a probability density function $f_b(b)$. Then the probability density function $f_{x_B}(x_B)$ of x_B is uniquely determined by $f_{A_B^{-1}}$ and $f_b(b)$. Assume c_B is stochastic and has a probability density function $f_{c_B}(c_B)$. Then y is also stochastic and its probability density function $f_y(y)$ is determined by $f_{x_B}(x_B)$ and $f_{c_B}(c_B)$.

As data changes, the chosen basis B might not satisfy the feasibility conditions given in Definition 1. Assume we can derive the probability distribution of x_B by exploiting $x_B = A_B^{-1}b$. Then we obtain for feasibility:

Lemma 5. The probability that a basis B is feasible. *Assume x_B is a basic solution to (1.1) with probability density function $f_{x_B}(x_B)$. Then the probability that x_B is feasible is given by:*

$$\begin{aligned} P(x_B \textit{ feasible}) &= P(x_B \geq 0) \\ &= \int_0^\infty \cdots \int_0^\infty f_{x_B}(x_{B1}, \ldots, x_{Bm}) dx_{B1} \cdots dx_{Bm}. \end{aligned} \tag{4.2}$$

Proof. From Definition 1 we have that a solution x is feasible if and only if:

$$Ax = b, \;\; x \geq 0.$$

From Definition 2 we know:

$$b = Ax = A_B x_B,$$

and in (4.1) we state that x_B is set to $A_B^{-1}b$. Thus, we have obtain:

$$b = Ax = A_B x_B = A_B A_B^{-1} b = b.$$

Concluding, a basic solution x_B is feasible if and only if $x_B \geq 0$. Then the result follows. □

The simplex method provides in (1.7) a criterion for a basis B to be optimal. A basis B is optimal if and only if:

$$c_N \geq c_B A_B^{-1} A_N.$$

We now use this to derive the probability that a basis is optimal.

Lemma 6. The probability that a basis B is optimal. *Assume c has a probability distribution specified by $f_c(c_B, c_N)$. Assume A is deterministic. Then the probability that B is an optimal basis is given by:*

$$P\left(c_N \geq c_B A_B^{-1} A_N\right) = \int_D f_c(c_B, c_N) dc_B dc_N, \tag{4.3}$$

where D is given by

$$D = \left\{ \begin{array}{l} (c_B^1, \ldots, c_B^m, c_N^1, \ldots c_N^{n-m}) | \\ \quad c_N^1 \geq (A_B^{-1}A_N)_1 c_B, \ldots, c_N^{n-m} \geq (A_B^{-1}A_N)_{n-m} c_B \end{array} \right\}.$$

Proof. A basis B is optimal if and only if $c_N \geq c_B A_B^{-1} A_N$ holds. Therefore, we integrate the probability density function for all values of c_N and c_B for which this criterion holds. □

A more general expression would involve stochastic A_B^{-1} and A_N. As the inverse of a stochastic matrix need not exist, we do not consider it here. Instead, we assume that A_B^{-1} and A_N are deterministic.

In the next lemma we derive the probability distribution of the cost y. There are two approaches to this:

1. Specify the cost y by the equation

$$y = c_B^T x_B = c_1 x_1 + \cdots + c_m x_m.$$

   We then use the probability distribution of c_B and x_B to derive the probability distribution of cost y.

2. Condition on feasibility. The above equation holds, but the interpretation is tricky. Costs in a Linear Program are not only given by $y = c_B^T x_B$, but are due to constraints. In Lemma 5 we see that a basis is feasible if and only if $x_B \geq 0$. Thus, if we calculate the cost y we need to condition on feasibility.

The value of the first approach is the ease of the expression. In Section 4.3 we derive that for special cases there are exact evaluations possible, while the second expression can only be calculated using simulation techniques. Thus if $P(x_B \geq 0) \approx 1$, the first expression is a good approximation.

Lemma 7. The probability that under a basis B the cost y exceeds a value y'. *Assume x_B and c_B are stochastic and have a joint probability density function $f_{c_B,x_B}(c_B, x_B)$.*

1. *Assume negative values for x_B are allowed. Then the probability density function of y is given by*

$$\begin{aligned} f_Y(y) &= P(y = c_1 x_1 + \cdots + c_m x_m) \\ &= \int_D f_{c,x}(x_1, \ldots x_m, c_1, \ldots, c_{m-1}, \frac{y - \sum_{i=1}^{i=m-1} c_i x_i}{x_m}) dc_B dx_B, \end{aligned} \tag{4.4}$$

   where

$$D = \{(x_1, \cdots x_m, c_1, \cdots c_{m-1}) \in R^{2m-1}\}.$$

   Then the probability that y exceeds a value y' is given by

$$P(y \geq y') = \int_{y=y'}^{\infty} f_Y(y) dy. \tag{4.5}$$

2. *Assume negative values for x_B are not allowed. Assume that an infeasible system corresponds to infinite costs. Then the probability density function of y conditioned on $x_B \geq 0$ is given by:*

$$f_{Y|x_B \geq 0}(y | x_B \geq 0) = \int_D f_{c,x}(x_1, \ldots x_m, c_1, \ldots, c_{m-1}, \frac{y - \sum_{i=1}^{i=m-1} c_i x_i}{x_m}) dc_B dx_B, \tag{4.6}$$

   where

$$D = \{(x_1, \cdots x_m, c_1, \cdots c_{m-1}) | (x_1, \cdots x_m) \in R_+^m, (c_1, \cdots c_{m-1}) \in R^{m-1}\}.$$

   Then the probability that y exceeds a value y' is given by

$$P(y \geq y') = 1 - P(x \geq 0) + \int_{y=y'}^{\infty} f_Y(y) dy. \tag{4.7}$$

Proof. The value of y is determined by the values of c_B and x_B through $y = c_B^T x_B$. Thus in (4.4) the probability density function of y is obtained by integrating $f_{c_B,x_B}(c_B, x_B)$ over all values of x_B and c_B that yield a value y. Then (4.6) includes the interpretation of y: it is the cost due to a set of constraints. As the total mass of $f_{Y|x_B \geq 0}(y | x_B \geq 0)$ is equal to $P(x \geq 0)$, this is compensated through $1 - P(x \geq 0)$ in (4.7). This assumes that an infeasible system corresponds to an infinite cost. □

In Lemma 6 the global optimality of a basis is studied: the cost of a basis B is at least as good as all other possible bases. In an application, however, we can have a couple of bases that are all (close to) optimal for some realization of the data. The following lemma gives an expression to obtain the probability that the cost y^1 of basis B^1 is less or equal to the cost y^2 of basis B^2:

Lemma 8. The probability that the cost y under a basis B is less or equal to the cost y' of some other basis B'. *Assume basis B^1 and basis B^2 with corresponding stochastic solutions x^{B1} and x^{B2}. Assume a stochastic cost vector c and assume x^{B1}, x^{B2} and c have a joint probability density function $f_{c,x^{B1},x^{B2}}(c, x^{B1}, x^{B2})$. Assume that the desired probability is not defined if at least one of the components of x^{B1} and x^{B2} is negative. Then the probability the cost y_1 corresponding to B^1 is less or equal to the cost y_2 corresponding to B^2 is given by:*

$$\begin{aligned} P(y_1 \leq y_2) &= P\left(\sum_{i\in B_1} c_i x_i^{B1} \leq \sum_{j\in B_2} c_j x_j^{B2}\right) \\ &= \int_D f_{c,x^{B1},x^{B2}}(c, x^{B1}, x^{B2}) d(x^{B1}, x^{B2}, c), \end{aligned} \tag{4.8}$$

where D is given by

$$D = \left\{ \begin{array}{ll} x_i^{B1}, x_j^{B2}, c_k | & i \in B_1, j \in B_2, k \in (B_1 \cup B_2),\ x_i^{B1} \geq 0, x_j^{B2} \geq 0, \\ & \sum_{i\in B_1} c_i x_i^{B1} \leq \sum_{j\in B_2} c_j x_j^{B2} \end{array} \right\}.$$

Proof. The motivation of (4.8) is similar to that of the previous lemmas. We have a probability density function and the desired probability is the integral over the area in which $y_1 \leq y_2$ is true. Note that in general the stochastic vectors x^{B1} and x^{B2} are dependent as they are both derived from the vector b. □

4.2 Normal distributions in b and c

In this section we will study the standard Linear Program where the constraint vector b and the cost vector c are random variables with a normal distribution. There are several cases to be distinguished with increasing complexity, these will be studied in increasing order.

Theorem 9. A normal distribution for c. *Assume the cost-vector c is a multi-variate normal vector with $c \sim N(\mu^c, \Sigma^c)$ where μ^c denotes the expectation of c and Σ^c denotes the covariance matrix of c. Assume b and A_B are deterministic. Then the cost y is a normal random variable with:*

$$\begin{aligned} \mathbb{E}[y] &= \sum_{i,j} \mu_j^c (A_B^{-1})_{ij} b_i, \\ \mathrm{Var}[y] &= (A_B^{-1} b)\ \Sigma^c\ (A_B^{-1} b)^T \\ &= \sum_{k,l} ((A_B^{-1})_k \cdot b)((A_B^{-1})_l \cdot b)\Sigma_{k,l}^c, \end{aligned} \tag{4.9}$$

where $(A_B^{-1})_k$ denotes the k^{th} row of A_B^{-1}.

Proof. Only basic solutions are considered. As b and A_B^{-1} are deterministic, through (4.1) x_B is deterministic and fixed as $x_B = A_B^{-1} b$. As $y = c_b^T x_B$, y is a linear combination of normal random variables and the Linear Transformation Property (Theorem 4) can be applied to obtain the result. □

Theorem 10. A normal distribution for b. *Assume the constraint coefficients vector b is a multivariate normal vector with $b \sim N(\mu^b, \Sigma^b)$ where μ^b denotes the expectation of b and Σ^b denotes the covariance matrix of b. Assume c and A_B are deterministic. Then x_B has a normal distribution with parameters*

$$x_B \sim N\left(A_B^{-1}\mu_b, A_B^{-1}\Sigma^b(A_B^{-1})^T\right),$$

and y has a normal distribution with parameters

$$y \sim N\left(c_B^T A_B^{-1}\mu_b, c_B^T A_B^{-1}\Sigma^b(A_B^{-1})^T c_B\right).$$

Proof. The distributions of x and y can again be determined using Theorem 4. These expressions can also be calculated using the basic rules of expectation and variance of random variables. If b_i denotes a random variable, we can compute the expectation of y using equation (4.1):

$$\begin{aligned} \mathbb{E}[y] &= \mathbb{E}\left[\sum_{i,j} c_{Bj}(A_B^{-1})_{ij} b_i\right] \\ &= \sum_{i,j} c_{Bj}(A_B^{-1})_{ij}\mathbb{E}\left[b_i\right] \\ &= \sum_{i,j} c_{Bj}(A_B^{-1})_{ij}\mu_i^b. \end{aligned} \tag{4.10}$$

We can also derive the variance of y, where $c_B^T \cdot (A_B^{-1})_l$ denotes the inner product of c_B^T and the l^{th} column of A_B^{-1}:

$$\begin{aligned} \mathsf{Var}[y] &= \mathsf{Var}\left[\sum_{i,j} c_{Bj}(A_B^{-1})_{ij} b_i\right] \\ &= \mathsf{Var}\left[\sum_{i} (c_B^T \cdot (A_B^{-1})_i) b_i\right] \\ &= \mathsf{Cov}\left[\sum_{k} (c_B^T \cdot (A_B^{-1})_k) b_k, \sum_{l} (c_B^T \cdot (A_B^{-1})_l) b_l\right] \\ &= \sum_{k,l} (c_B^T \cdot (A_B^{-1})_k)(c_B^T \cdot (A_B^{-1})_l)\Sigma^b_{k,l}. \end{aligned} \tag{4.11}$$

In similar fashion we can also derive the expectation and variance of the different components of the solution. Due to the correlation within b, the solution values will be correlated as well.

$$\begin{aligned} \mathbb{E}[x_i] &= \sum_j (A_B^{-1})_{ij}\mu_j^b\,, \\ \mathsf{Var}[x_i] &= \sum_{k,l} (A_B^{-1})_{ik}(A_B^{-1})_{il}\Sigma^b_{k,l}, \\ \mathsf{Cov}[x_i, x_j] &= \sum_{k,l} (A_B^{-1})_{ik}(A_B^{-1})_{jl}\Sigma^b_{k,l}. \end{aligned} \tag{4.12}$$

□

Thus, with only b or c as a random variable the derivations are straightforward and the result is a normally distributed solution and cost.

Theorem 11. A normal distribution for b and c - independent case. *Assume b is a multivariate normal vector with $b \sim N(\mu^b, \Sigma^b)$ and c is multivariate normal vector with $c \sim N(\mu^c, \Sigma^c)$. Assume independence between b and c. Assume A_B is a constant. Then x_B is a multivariate normal vector with*

$$x_B \sim N\left(A_B^{-1}\mu_b, A_B^{-1}\Sigma^b(A_B^{-1})^T\right).$$

The cost y is not a normal random variable but has expectation

$$\mathbb{E}[y] = \sum_{i,j} \mu_j^c (A_B^{-1})_{ij} \mu_i^b \tag{4.13}$$

and variance

$$\mathrm{Var}[y] = \sum_{k,l}\sum_{p,q} (A_B^{-1})_{kl}(A_B^{-1})_{pq}[\mu_k^b\mu_p^b\Sigma_{l,q}^c + \mu_l^c\mu_q^c\Sigma_{k,p}^b + \Sigma_{l,q}^c\Sigma_{k,p}^b]. \tag{4.14}$$

Proof. First consider the distribution of x_B. As this is only dependent on A_B and b, the proof is the same as in Theorem 10. The cost y, however, is different as Theorem 4 cannot be applied: $c_B^T x_B$ is not a linear combination of normal random variables, but a sum of products of normal random variables. □

So far, we have assumed that the cost coefficients c and the constraint values b are independent. In most distribution problems this is realistic, as for example the wages of truck drivers need not depend on the demand for a certain product. However, for a more complete derivation we will include the dependent case here.

Theorem 12. A normal distribution for b and c - dependent case. *Assume b and c are dependent multivariate vectors. Assume A_B is a constant. Then x_B is a multivariate normal vector with*

$$x_B \quad \sim \mathcal{N} \quad \left(A_B^{-1}\mu_b, A_B^{-1}\Sigma^b(A_B^{-1})^T\right) \tag{4.15}$$

and y is not a normal random variable but has expectation

$$\mathbb{E}[y] = \sum_{i,j} (A_B^{-1})_{ij} \left(\mathit{Cov}(c_j, b_i) + \mu_j^c\mu_i^b\right) \tag{4.16}$$

and variance

$$\mathrm{Var}[y] = \sum_{k,l}\sum_{p,q} (A_B^{-1})_{kl}(A_B^{-1})_{pq}\mathit{Cov}[c_l b_k,\ c_q b_p] \tag{4.17}$$

with

$$\begin{aligned} \mathit{Cov}[c_l b_k,\ c_q b_p] \quad &= \quad \mathbb{E}\left[c_l b_k c_q b_p\right] - \mu_l^c\mu_k^b\mu_q^c\mu_p^b + \mu_l^c\mu_k^b\mathit{Cov}[c_q,\ b_p] \\ &\quad + \mu_q^c\mu_p^b\mathit{Cov}[c_l,\ b_k]. \end{aligned} \tag{4.18}$$

Proof. As in Theorem 11, a stochastic c has no influence on the distribution of x_B. Therefore the expression and proof is the same as in Theorem 10. As y is now a sum of products of dependent normal random variables. The expectation of y now includes the dependency between b and c:

$$\begin{aligned} \mathbb{E}[y] \quad &= \quad \mathbb{E}\left[\sum_{i,j} c_j (A_B^{-1})_{ij} b_i\right] \\ &= \quad \sum_{i,j} \mathbb{E}\left[c_j (A_B^{-1})_{ij} b_i\right] \\ &= \quad \sum_{i,j} (A_B^{-1})_{ij} \mathbb{E}\left[c_j b_i\right] \\ &= \quad \sum_{i,j} (A_B^{-1})_{ij} \left(\mathrm{Cov}(c_j, b_i) + \mu_j^c\mu_i^b\right). \end{aligned} \tag{4.19}$$

The derivation of the variance of y requires some more work:

$$\begin{aligned} \text{Var}[y] &= \text{Var}\left[\sum_{i,j} c_j (A_B^{-1})_{ij} b_i\right] \\ &= \text{Cov}\left[\sum_{k,l} c_l (A_B^{-1})_{kl} b_k,\ \sum_{p,q} c_q (A_B^{-1})_{pq} b_p\right] \\ &= \sum_{k,l}\sum_{p,q} (A_B^{-1})_{kl}(A_B^{-1})_{pq}\text{Cov}\left[c_l b_k,\ c_q b_p\right]. \end{aligned} \tag{4.20}$$

The last covariance term in (4.20) requires some more specification:

$$\begin{aligned} \text{Cov}\left[c_l b_k,\ c_q b_p\right] &= \mathbb{E}\left[(c_l b_k - \mu_l^c \mu_k^b)(c_p b_p - \mu_q^c \mu_p^b)\right] \\ &= \mathbb{E}\left[c_l b_k c_q b_p\right] - \mu_l^c \mu_k^b \mu_q^c \mu_p^b + \mu_l^c \mu_k^b \text{Cov}\left[c_q,\ b_p\right] \\ &\quad + \mu_q^c \mu_p^b \text{Cov}\left[c_l,\ b_k\right]. \end{aligned} \tag{4.21}$$

Thus, we have derived the expectation and variance of the cost y when b and c are multivariate normal and dependent. □

4.3 Calculation of probabilities

Lemmas 5 - 8 in Section 4.1 are all of the following form:

$$P(\cdot\cdot) = \int_D f_V(v)dv, \tag{4.22}$$

where V is a set of stochastic variables and D is a subset of the domain of the variable set V. The function $f_V(V)$ is the joint probability density function of the stochastic variables V. In the lemmas the variables c_B, x_B and y are used. As $y = c_B^T x_B$, we have that all Lemmas can be evaluated using the probability distributions of c_B and x_B.

From Theorems 9 - 12 in Section 4.2 we can conclude that in all studied cases, the vectors c_B and x_B are either a constant or have a multivariate normal distribution. If we thus want to study the effect of multivariate distributions in the cost and demand vectors and evaluate the Lemmas in Section 4.1 we have:

Theorem 13. *Let V be a multivariate normal random vector with expectation μ_V and covariance matrix Σ_V. Let D be a subset of the domain of V. Let $f_V(V)$ be the joint probability density function of V. Then all probabilities in Section 4.1 are of the form:*

$$P(\cdot\cdot) = \int_D f_V(V)dv. \tag{4.23}$$

Proof. Follows directly from Lemmas 5 - 8 and Theorems 9 - 12. □

For some special cases we can find an exact expression for the answer. In general, however, we need a simulation technique. We first discuss a simulation technique. Then we discuss the several cases and give an exact answer if possible.

Monte Carlo Simulation

The integral in equation (4.23) has no expression in elementary functions, as a simplified form of the following expression is not known:

$$\int e^{-t^2} dt.$$

We therefore need a numerical technique or simulation method to evaluate (4.23). Numerical techniques for the evaluation of integrals of multivariate normal distributions are available, but are mostly suited for low (< 10) numbers of variables. Numerical models are amongst others proposed in [20, 14, 37]. An alternative approach is in Monte Carlo Simulation. Monte Carlo methods are a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results [44].

We now present a Monte Carlo method that provides an efficient way to evaluate integrals like (4.23). This method is also described in for example [8].

The first step is to recognize that the desired probability (4.23) is the probability that a realization of a stochastic vector V is in a set D:

$$\begin{array}{rcl} P(\cdot\cdot) & = & \int_D f_V(V)dv \\ & = & P(V \in D). \end{array} \tag{4.24}$$

As V is a multivariate normal vector, we can write it as a linear combination of independent standard normal distributions. Assume V consists of n variables $v_1, \ldots, v_n$. Then $Z^n \sim N(0_n, I_n)$. Using the Linear Transformation (Theorem 4) we can derive:

$$V = \mu_V + LZ^n, \tag{4.25}$$

where μ_V is the expectation of V and L the lower Cholesky factorization of the covariance matrix Σ^V. Z^n is an $n-$vector of independent standard normal distributions. An extensive explanation of this procedure is given in [33]. Then we can derive:

$$\begin{array}{rcl} P(\cdot\cdot) & = & P(V \in D) \\ & = & P(Z^n \in D'), \end{array} \tag{4.26}$$

where

$$D' = \{z \in \mathbb{R}^n | \mu_V + Lz \in D\}. \tag{4.27}$$

We can rewrite Z^n to a vector of n independent uniform $[0, 1]$ random variables using the inverse of the standard normal distribution. Thus, U^n is an $n-$vector where $u_i = \Phi(z_i)$. Thus we continue the derivation:

$$\begin{array}{rcl} P(\cdot\cdot) & = & P(V \in D) \\ & = & P(U^n \in D''), \end{array} \tag{4.28}$$

where

$$D'' = \{u \in \mathsf{dom}(U^n) | \mu_V + L \begin{bmatrix} \Phi^{-1}(u_1) \\ \vdots \\ \Phi^{-1}(u_n) \end{bmatrix} \in D\}. \tag{4.29}$$

A draw U_{out} from a uniform distribution U has the same distribution function F as the uniform distribution itself. Thus, if we sample enough values for U^n we can construct an arbitrarily small confidence interval for the value of the desired probability. Concluding we have derived:

$$\begin{array}{rcll} P(\cdot\cdot) & = & \int_D f_V(V)dv & \\ & = & P(V \in D) & \\ & = & P(Z^n \in D') & \text{where } D' = \{z \in \mathsf{dom}(Z^n) | \mu_V + Lz \in D\} \\ & = & P(U^n \in D'') & \text{where } D'' = \{u \in \mathsf{dom}(U^n) | \mu_V + L \begin{bmatrix} \Phi^{-1}(u_1) \\ \vdots \\ \Phi^{-1}(u_n) \end{bmatrix} \in D\}. \end{array} \tag{4.30}$$

Performance measures for normal b and c

In this section we will combine the results from Section 4.1 and 4.2 to give an overview how the performance measures can be calculated. If possible an exact answer is provided, otherwise the parameters D, μ_V and Σ_V are specified to be applied in the Monte Carlo simulation as given in Section 4.3. All cases and results are summarized in Table 4.1.

The following four uncertainty models are studied:

- Deterministic b, normal c.
- Normal b, deterministic c
- Independent normal b, normal c
- Dependent normal b, normal c

The following four performance measures are evaluated:

- Feasibility.
- Optimality.
- Better alternative.
- Outliers.

We now combine these in 10 cases.

Case a: Feasibility for normal b. Assume b has a multivariate normal distribution with $b \sim N(\mu_b, \Sigma^b)$. Then we obtain from Theorem 10:

$$x_B \sim N\left(A_B^{-1}\mu_b, A_B^{-1}\Sigma^b(A_B^{-1})^T\right).$$

Application of Lemma 5 then yields:

$$P(x_{B1} \geq 0, \ldots, x_{Bm} \geq 0) = \int_0^\infty \cdots \int_0^\infty f_{x_B}(x_{B1}, x_{B2}, \ldots, x_{Bm}) dx_{B1} \cdots dx_{Bm}, \tag{4.31}$$

where

$$f_{x_B}(x_{B1}, x_{B2}, \ldots, x_{Bm}) = \frac{1}{(2\pi)^{m/2}|\Sigma^{x_B}|^{1/2}} \exp(-\frac{1}{2}(x_B - \mu_{x_B})^T \Sigma_{x_B}^{-1}(x_B - \mu_{x_B})). \tag{4.32}$$

We thus apply the Monte Carlo simulation with $V = x_B$, $\mu_V = \mu_{x_B}$, $\Sigma^V = \Sigma^{x_B}$ and

$$D = \{(x_{B1}, \ldots x_{Bn}) | x_{B1} \geq 0, \ldots, x_{Bn} \geq 0\}.$$

For each individual variable x_{Bi} we can obtain $P(x_{Bi} \geq 0)$ by using the standard normal distribution Z:

$$\begin{aligned} P(x_{Bi} \geq 0) &= P\left(\frac{x_{Bi} - \mu_i^{x_B}}{\sigma_i^{x_B}} \geq \frac{0 - \mu_i^{x_B}}{\sigma_i^{x_B}}\right) \\ &= P\left(Z \leq \frac{\mu_i^{x_B}}{\sigma_i^{x_B}}\right) \\ &= \Phi\left(\frac{\mu_i^{x_B}}{\sigma_i^{x_B}}\right). \end{aligned} \tag{4.33}$$

Thus, we can use (4.33) to investigate which variables are a bottleneck in the feasibility.

Case b: Optimality for normal c. Assume c has a multivariate normal distribution with $c \sim N(\mu_c, \Sigma^c)$. Consider Lemma 6. There is no simplification here so apply the Monte Carlo Simulation with parameters $V = c$, $\mu_V = \mu_c$, $\Sigma^V = \Sigma^c$,

$$D = \left\{ \begin{array}{l} (c_B^1, \ldots, c_B^m, c_N^1, \ldots c_N^{n-m})| \\ \quad c_N^1 \geq (A_B^{-1}A_N)_1 c_B, \ldots, c_N^{n-m} \geq (A_B^{-1}A_N)_{n-m} c_B \end{array} \right\}.$$

Case c: Better alternative for constant b, normal c. Assume c has a multivariate normal distribution with $c \sim N(\mu_c, \Sigma^c)$. Assume b is constant. Consider Lemma 8. As the values x of the corresponding bases are deterministic, the expression reduces to:

$$\begin{array}{rcl} P(y_1 \leq y_2) & = & P\left(\sum_{i \in B_1} c_i x_i^1 \leq \sum_{j \in B_2} c_j x_j^2 \right) \\ & = & P\left(\sum_{i \in B_1 \backslash B_2} c_i x_i^1 - \sum_{j \in B_2 \backslash B_1} c_j x_j^2 + \sum_{k \in B_1 \cap B_2} c_k (x_k^1 - x_k^2) \leq 0 \right). \end{array} \tag{4.34}$$

Note that the probability in this last expression is the probability that a sum of normal random variables is less or equal to zero. Thus, define the following vectors:

$$\begin{array}{rcl} \Delta x & = & (x_i^1 | i \in B_1 \backslash B_2,\ x_j^2 | j \in B_2 \backslash B_1,\ (x_k^1 - x_k^2) | k \in B_1 \cap B_2), \\ \Delta c & = & (c_i | i \in B_1 \backslash B_2,\ c_j | j \in B_2 \backslash B_1,\ c_k | k \in B_1 \cap B_2), \\ \Delta y & = & (\Delta c)^T \Delta x. \end{array} \tag{4.35}$$

The covariance matrix corresponding to Δc consists of the rows and columns i of the covariance matrix of c with $i \in (B_1 \cup B_2)$. We can then apply the linear transformation property to find the mean and variance of Δy. Finally, the desired probability can then be derived using the standard normal distribution:

$$P(y_1 \leq y_2) = P(\Delta y \leq 0) = \Phi\left(\frac{-\mu_{\Delta y}}{\sigma_{\Delta y}} \right).$$

Case d: Better alternative for normal b.
Assume b has a multivariate normal distribution with $b \sim N(\mu_b, \Sigma^b)$. Assume c is constant. Consider Lemma 8. There are two cases to be considered. The first is that feasibility is not considered to be important, the second is the probability conditioned on feasibility.

For the first, a similar procedure as in Case c can be applied: for normal b, the procedure is similar though more complicated. Although the different bases lead to different solutions, they are highly dependent as they are derived from the same vector b. Thus, define the following vectors:

$$\begin{array}{rcl} b_2 & = & \left[\begin{array}{c} b \\ b \end{array} \right], \\ \mu^{b_2} & = & \left[\begin{array}{c} \mu^b \\ \mu^b \end{array} \right], \\ \Sigma^{b_2} & = & \left[\begin{array}{cc} \Sigma^b & \Sigma^b \\ \Sigma^b & \Sigma^b \end{array} \right], \\ A_2 & = & \left[\begin{array}{cc} A_{B1}^{-1} & 0 \\ 0 & A_{B2}^{-1} \end{array} \right], \\ \Delta x & = & A_2 b_2, \\ \Delta c & = & \left[\begin{array}{c} c_{B1} \\ c_{B2} \end{array} \right]. \end{array} \tag{4.36}$$

Then we can derive with the linear transformation property:

$$\begin{array}{rcl} \mu^{\Delta x} & = & A_2\ \mu^{b_2}, \\ \Sigma^{\Delta x} & = & A_2\ \Sigma^{b_2} A_2^T, \\ \mu^{\Delta y} & = & (\Delta c)^T\ \mu^{\Delta x}, \\ \Sigma^{\Delta y} & = & (\Delta c)^T\ \Sigma^{\Delta x} \Delta c. \end{array} \tag{4.37}$$

Concluding, we have again a normal distribution for the difference in cost Δy. Thus, $y_1 - y_2$ has a normal distribution with parameters:

$$\mu_{y_1-y_2} = [1 \quad -1]\mu^{\Delta y},\ \ \sigma^2_{y_1-y_2} = [1 \quad -1]\Sigma^{\Delta y} \left[\begin{array}{c} 1 \\ -1 \end{array} \right].$$

Finally, the desired probability can then be derived using the standard normal distribution:

$$P(y_1 \leq y_2) = \Phi\left(\frac{-\mu_{y_1-y_2}}{\sigma_{y_1-y_2}} \right).$$

The probability $P(y_1 \leq y_2 | y_1,\ y_2 \text{ feasible})$ has to be calculated via Monte Carlo Simulation. The involved parameters are $V = x_B$, $\mu_V = \mu_{x_B}$, $\Sigma^V = \Sigma^{x_B}$ where the distribution of x_B follows from Theorem 10. Furthermore use

$$D = \left\{ \begin{array}{ll} x_i^{B1}, x_j^{B2} | & i \in B_1, j \in B_2,\ x_i^{B1} \geq 0,\ x_j^{B2} \geq 0, \\ & \sum_{i \in B_1} c_i x_i^{B1} \leq \sum_{j \in B_2} c_j x_j^{B2} \end{array} \right\}.$$

Case e: Better alternative for independent normal b, c. Assume b has a multivariate normal distribution with $b \sim N(\mu_b, \Sigma^b)$. Assume c has a multivariate normal distribution with $c \sim N(\mu_c, \Sigma^c)$. This case is studied in Theorem 11. As both y and $y_{|\text{feas}}$ have a general distribution, the desired probabilities from Lemma 8 have to be calculated via Monte Carlo Simulation. For both probabilities we have:

$$V = \left[\begin{array}{c} c_B \\ x_B \end{array} \right],\ \mu_V = \left[\begin{array}{c} \mu_{c_B} \\ \mu_{x_B} \end{array} \right],\ \Sigma^V = \left[\begin{array}{cc} \Sigma^{c_B} & 0 \\ 0 & \Sigma^{x_B} \end{array} \right].$$

The parameter D concerns the feasibility. For $P(y_1 \leq y_2)$ we have:

$$D = \left\{ \begin{array}{ll} x_i^{B1}, x_j^{B2}, c_k | & i \in B_1, j \in B_2, k \in (B_1 \cup B_2), \\ & \sum_{i \in B_1} c_i x_i^{B1} \leq \sum_{j \in B_2} c_j x_j^{B2} \end{array} \right\}.$$

For $P(y_1 \leq y_2 | y_1, y_2 \text{ feasible})$ we have:

$$D = \left\{ \begin{array}{ll} x_i^{B1}, x_j^{B2}, c_k | & i \in B_1, j \in B_2, k \in (B_1 \cup B_2),\ x_i^{B1} \geq 0, x_j^{B2} \geq 0, \\ & \sum_{i \in B_1} c_i x_i^{B1} \leq \sum_{j \in B_2} c_j x_j^{B2} \end{array} \right\}.$$

Case f: Better alternative for dependent normal b, c. The distribution for dependent normal b and c is studied in Theorem 12. The probabilities $P(y_1 \leq y_2)$ and $P(y_1 \leq y_2 | y_1, y_2 \text{ feasible})$ are calculated in the same way as in Case e. The only difference is the covariance matrix Σ^V:

$$\Sigma^V = \left[\begin{array}{cc} \Sigma^{c_B} & Cov(c_B, x_B) \\ Cov(x_B, c_B) & \Sigma^{x_B} \end{array} \right].$$

Case g: Outliers for normal c, constant b. In case only c has a normal distribution, we know that y has a normal distribution as well with parameters μ_y and σ_y^2. Then the probability that the cost y exceeds a value y' is given by:

$$\begin{array}{rcl} P(y \geq y') & = & P\left(Z \geq \dfrac{y' - \mu_y}{\sigma_y} \right) \\ & = & 1 - \Phi\left(\dfrac{y' - \mu_y}{\sigma_y} \right). \end{array} \tag{4.38}$$

Case h: Outliers for normal b, constant c.
There are two probabilities that need to be calculated: $P(y \geq y')$ and $P(y \geq y'|y \text{ feasible})$. The first is the same as in Case g. The second is established via Monte Carlo Simulation with the following parameters:

$$V = x_B,\ \mu_V = \mu_{x_B},\ \Sigma^V = \Sigma^{x_B},\ D = \left\{x_B | c_B^T x_B \geq y',\ x_B \geq 0\right\}.$$

Case i: Outliers for independent normal b, normal c.
There are two probabilities that need to be calculated: $P(y \geq y')$ and $P(y \geq y'|y \text{ feasible})$. Both are calculated using a Monte Carlo Simulation. The shared parameters are:

$$V = \begin{bmatrix} c_B \\ x_B \end{bmatrix},\ \mu_V = \begin{bmatrix} \mu_{c_B} \\ \mu_{x_B} \end{bmatrix},\ \Sigma^V = \begin{bmatrix} \Sigma^{c_B} & 0 \\ 0 & \Sigma^{x_B} \end{bmatrix}.$$

The domain for $P(y \geq y')$ is given by:

$$D = \left\{c_B,\ x_B | c_B^T x_B \geq y', \right\}.$$

The domain for $P(y \geq y'|y \text{ feasible})$ is given by:

$$D = \left\{c_B,\ x_B | c_B^T x_B \geq y',\ x_B \geq 0\right\}.$$

Case j: Outliers for dependent normal b, normal c. The procedure is the same as in Case i. The only difference is the dependence of x_B and c_B. Thus we obtain a different covariance matrix for V:

$$\Sigma^V = \begin{bmatrix} \Sigma^{c_B} & Cov(c_B, x_B) \\ Cov(x_B, c_B) & \Sigma^{x_B} \end{bmatrix}.$$

	Feasibility $P(x_B \geq 0)$ Reference: Lemma 5	Optimality $P(c_N \geq c_B A_B^{-1} A_N)$ Reference: Lemma 6	Better alternative $P(y_1 \leq y_2)$ Reference: Lemma 8	Outlier $P(y \geq y')$ Reference: Lemma 7
c normal: x constant $y \sim N(\mu_y, \sigma_y^2)$ Reference: Th. 9	Not relevant (x constant)	$P(c_N \geq c_B A_B^{-1} A_N)$: MC via f_{c_B} Reference: (b)	$P(y_1 \leq y_2) = \Phi(..)$ Reference: (c)	$P(y \geq y')$ $= 1 - \Phi\left(\frac{y' - \mu_y}{\sigma_y}\right)$ Reference: (g)
b normal: $x \sim N(\mu_{x_B}, \Sigma^{x_B})$ $y \sim N(\mu_y, \sigma_y^2)$ $y_{\|\text{feas}} \sim G$ Reference: Th. 10	$P(x_B \geq 0)$: MC via f_{x_B} $P(x_i \geq 0) = \Phi\left(\frac{\mu_i^{x_B}}{\sigma_i^{x_B}}\right)$ Reference: (a)	Not relevant (c constant)	$P(y_1 \leq y_2) = \Phi(..)$ $P(y_1 \leq y_2\|\text{feas})$: MC via f_{x_B} Reference: (d)	$P(y \geq y')$ $= 1 - \Phi\left(\frac{y' - \mu_y}{\sigma_y}\right)$ $P(y \geq y'\|\text{feas})$: MC via f_{x_B} Reference: (h)
b, c normal, independent: $x \sim N(\mu_{x_B}, \Sigma^{x_B})$ $y \sim G$ $y_{\|\text{feas}} \sim G$ Reference: Th. 11	$P(x_B \geq 0)$: MC via f_{x_B} $P(x_i \geq 0) = \Phi\left(\frac{\mu_i^{x_B}}{\sigma_i^{x_B}}\right)$ Reference: (a)	$P(c_N \geq c_B A_B^{-1} A_N)$: MC via f_{c_B} Reference: (b)	$P(y_1 \leq y_2)$: MC via f_{x_B}, f_{c_B} $P(y_1 \leq y_2\|\text{feas})$: MC via f_{x_B}, f_{c_B} Reference: (e)	$P(y \geq y')$: MC via f_{x_B}, f_{c_B} $P(y \geq y'\|\text{feas})$: MC via f_{x_B}, f_{c_B} Reference: (i)
b, c normal, dependent: $x \sim N(\mu_{x_B}, \Sigma^{x_B})$ $y \sim G$ $y_{\|\text{feas}} \sim G$ Reference: Th. 12	$P(x_B \geq 0)$: MC via f_{x_B} $P(x_i \geq 0) = \Phi\left(\frac{\mu_i^{x_B}}{\sigma_i^{x_B}}\right)$ Reference: (a)	$P(c_N \geq c_B A_B^{-1} A_N)$: MC via f_{c_B} Reference: (b)	$P(y_1 \leq y_2)$: MC via f_{x_B, c_B} $P(y_1 \leq y_2\|\text{feas})$: MC via f_{x_B, c_B} Reference: (f)	$P(y \geq y')$: MC via f_{x_B, c_B} $P(y \geq y'\|\text{feas})$: MC via f_{x_B, c_B} Reference: (j)

Table 4.1: A summary of Chapter 4. The rows represent the different distributions possible in an LP and the columns represent the several calculated probabilities. The cells summarize the calculation method and the letter in brackets refers to a further explanation in Section 4.3.

CHAPTER 5

IMPLEMENTATION AND APPLICATION

In this chapter we formulate a method to implement the derived mathematics in a real world case. It is presented in Section 5.1 and illustrated with a toy example in Section 5.2. It is applied to a real world case in Section 5.3 and a conclusion is given in Section 5.4.

5.1 A framework to analyze uncertainty

A flowchart is given in Figure 5.1 to show the full proposed method. It gives the order of the different steps and indicates what calculations can be done exact and where simulation is necessary. Simulation here refers to the method explained in Section 4.3. Details on the implementation are given in Appendix B. We now give a short description of the steps.

Step 1: Data gathering. In this step we collect data, formulate an LP and optimize the LP to obtain a basic solution based on the expected values.
Step 2: Uncertainty gathering. In this step we define which parameters in the LP are stochastic. We then estimate probability distributions for these parameters.
Step 3: Alternative basic solutions. In this step we determine alternatives for the solution obtained in Step 1. These need to be basic solutions and can be generated in several ways. An extensive study about generating alternative solutions is given in [7].
Step 4: Transfer uncertainty. This step is only necessary when parameters in A are uncertain. We transfer the uncertainty in this parameters to the vector b.
Step 5: Evaluate each alternative basis. First we determine which vectors are stochastic and whether they are dependent. Then we evaluate the probability distribution of x_B and determine whether feasibility can safely be assumed or not. We calculate the following performance measures using Table 4.1:

a) The probability that this alternative is feasible is α%. This question is only relevant if there is uncertainty in the constraint vector. Otherwise, $\alpha = 100\%$. This is calculated using Lemma 5.

b) The expected cost is μ_y. This can be calculated using the theorem corresponding theorem from Theorems 9 - 12.

c) The probability that this alternative gives the best cost of all possible solutions is β%. This is calculated using Lemma 6.

d) The probability that the cost exceeds the current cost is γ%. The current cost should be available from the data and can be used as value for y' in Lemma 7.

e) The probability that the cost will be at most 5% higher then expected is δ%. This can be calculated by using $y' = 1.05\mu_y$ in Lemma 7.

f) The probability that the cost in this alternative is less or equal to the cost in the Base Case or scenario derived in Step 1 is ϵ%. This can be calculated using Lemma 8.

Step 6: Determine a small set of bases. Using the results from Step 5, we now select a small number of bases for a further comparison. These can be alternatives that perform likewise.
Step 7: Compare alternative bases. Calculate the probability that a basis gives better cost then the other alternatives using Lemma 8.
Step 8: Report the results and choose a basis.

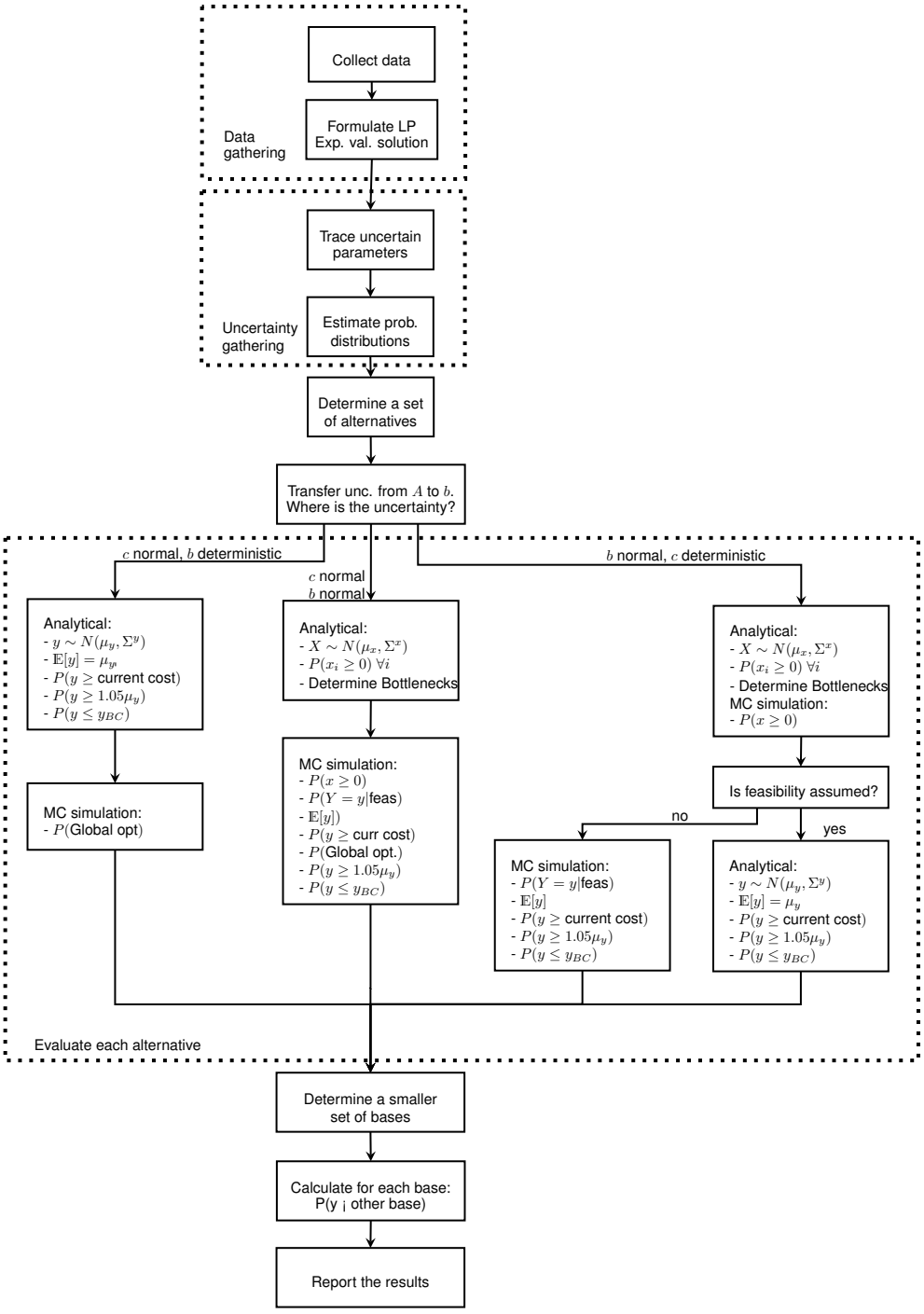


Figure 5.1: A summary of the proposed method.

5.2 A small factory example

In this section we will formulate and analyze a toy example to illustrate the results so far. The framework given in Section 5.1 will be used and results will be calculated using the theory from Chapter 4.

Step 1: Data gathering. Consider a small factory which produces a product that comes in three different types. Assume that the amount of products need not be integers. Each type has a couple of characteristics and a production cost. Suppose we want to minimize the total cost given an amount we have to produce:

$$\begin{array}{rcrrrll} \min y & = & 5x_1+ & 7x_2+ & 10x_3 & & \\ \text{s.t.} & & x_1+ & x_2+ & x_3 & = 1000 & \text{(Production constraint)} \\ & & 3x_1+ & 7x_2+ & 5x_3 & \leq 4500 & \text{(Materials constraint)} \\ & & 15x_1+ & 10x_2+ & 7x_3 & \leq 11000 & \text{(Pollution constraint)} \\ & & 2x_1+ & 3x_2+ & 5x_3 & \leq 4500 & \text{(Labour constraint)} \\ & & & & x_i & \geq 0 & \end{array} \tag{5.1}$$

As this problem has three inequality constraints, we can replace them by equality constraints using the slack variables x_4, x_5, x_6:

$$\begin{array}{rcrrrrrrrl} \min y & = & 5x_1+ & 7x_2+ & 10x_3 & & & & & \\ \text{s.t.} & & x_1+ & x_2+ & x_3 & & & & = & 1000 \\ & & 3x_1+ & 7x_2+ & 5x_3 & +x_4 & & & = & 4500 \\ & & 15x_1+ & 10x_2+ & 7x_3 & & +x_5 & & = & 11000 \\ & & 2x_1+ & 3x_2+ & 5x_3 & & & +x_6 & = & 4500 \\ & & & & & & & x_i & \geq & 0 \end{array} \tag{5.2}$$

Note that this problem does not have any redundancy in it, as no constraint is a linear combination of other constraints. This problem can then be solved using AIMMS. The optimal basis is unique and formed by the variables $x_1 = 432, x_2 = 182, x_3 = 386, x_6 = 1159$ and $B = \{1, 2, 3, 6\}$. Thus, the first three constraints are tight and the last is not. As we know that $x_4 = x_5 = 0$ in the basic solution, we have:

$$A_B = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 3 & 7 & 5 & 0 \\ 15 & 10 & 7 & 0 \\ 2 & 3 & 5 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 1000 \\ 4500 \\ 11000 \\ 4500 \end{bmatrix}, \quad c_B = \begin{bmatrix} 5 \\ 7 \\ 10 \\ 0 \end{bmatrix}, \quad x_B = \begin{bmatrix} 432 \\ 182 \\ 386 \\ 1159 \end{bmatrix}, y_B = 7294. \tag{5.3}$$

Step 2: Uncertainty gathering. We will now introduce the stochasticity. As an example, suppose that all values of b have a standard deviation of 100. Thus, the covariance matrix is a diagonal matrix with the value 10000 on the diagonal. Then we have:

$$\Sigma^b = \begin{bmatrix} 10000 & 0 & 0 & 0 \\ 0 & 10000 & 0 & 0 \\ 0 & 0 & 10000 & 0 \\ 0 & 0 & 0 & 10000 \end{bmatrix}.$$

Step 3: Alternative solutions. As this example is very small, specialized techniques to create other bases are not necessary. Instead, consider leaving out the production of product 2: $x_2 = 0$. Furthermore, assume that the pollution constraint is tight so $x_5 = 0$. That yields a basis $B = 1, 3, 4, 6$. The corresponding matrix A_B is invertible, so this is a valid basis.

Step 4: Transfer uncertainty. This step is not necessary, as we have not specified uncertainty in A.

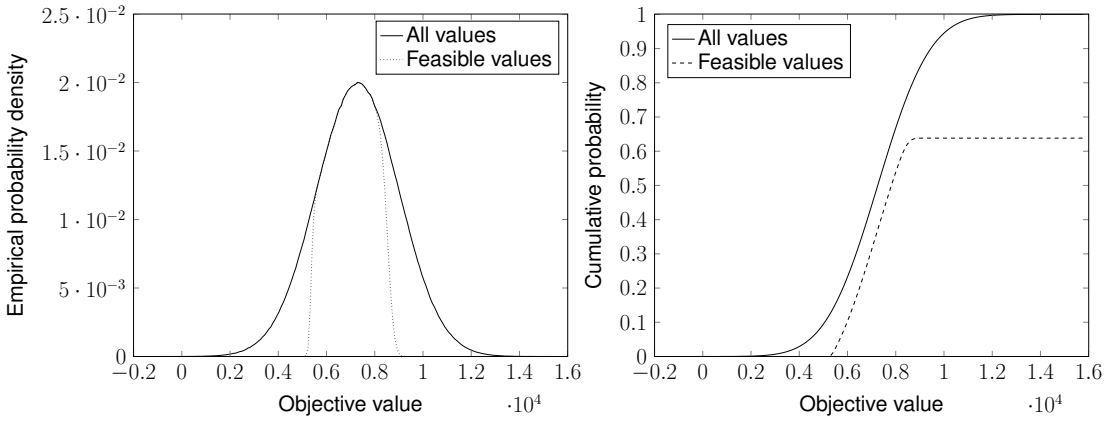


Figure 5.2: The objective value for stochastic b for the basis that is optimal for expected values of the data. The left picture shows the empirical probability density, the right picture the cumulative probabilities.

Step 5: Evaluate each alternative. We will first consider the basis obtained in Step 1. We can derive using Theorem 10:

$$A_B^{-1} = \begin{bmatrix} -0.0455 & -0.1364 & 0.0909 & 0 \\ -2.4545 & 0.3636 & 0.0909 & 0 \\ 3.4091 & -0.2273 & -0.1818 & 0 \\ -9.7727 & 0.3182 & 0.4545 & 1 \end{bmatrix}, \quad \Sigma^b = \begin{bmatrix} 10000 & 0 & 0 & 0 \\ 0 & 10000 & 0 & 0 \\ 0 & 0 & 10000 & 0 \\ 0 & 0 & 0 & 10000 \end{bmatrix},$$
$$\Sigma^{x_B} = A_B^{-1}\Sigma^b(A_B^{-1}) \begin{bmatrix} 289 & -1529 & 169 & -4463 \\ -1529 & 61653 & -84669 & 241446 \\ 169 & -84669 & 117066 & -334711 \\ -4463 & 241446 & -334711 & 968140 \end{bmatrix}, \quad \sigma_y^2 = c^T\Sigma^{x_B}c = 1716. \tag{5.4}$$

Note that although the values of b are independent random variables, the vector x_B is dependent. As Figure 5.1 indicates, we will first calculate the feasibility. In Table 4.1 we see that $P(x_i \geq 0)$ has an analytic expression and for $P(x_B \geq 0)$ we need an MC simulation. The procedure for this calculation is given as 'Case a' in Section 4.3 on Page 27. For individual values of x_B we can derive:

$$P(x_1 \geq 0) = \Phi\left(\frac{432}{\sqrt{289}}\right) = 1.00, \quad P(x_2 \geq 0) = 0.77, \quad P(x_3 \geq 0) = 0.87, \quad P(x_6 \geq 0) = 0.88.$$

As the values of x_B are dependent, we need a MC simulation to determine $P(x_B \geq 0)$. We use $V = x_B$, $\mu_V = \mu_{x_B}$, $\Sigma^V = \Sigma^{x_B}$ and $D = \{(x_{B1}, \ldots, x_{Bn}) | x_{bi} \geq 0 \ \forall i\}$. This gives the following result:

- The probability that this alternative is feasible is 64 %.

We now have to decide whether to assume feasibility or not. Recall the discussion in Lemma 7: for normally distributed x we have a strictly positive probability that x is not feasible. This probability, however, can be arbitrarily small. If this probability is extremely small, we can ignore this and obtain exact expressions. If this is not, we need a simulation. As $P(x_B \geq 0) = 64\%$, we cannot assume feasibility and thus calculate all remaining performance measures conditioned on feasibility. Only 64 % of all samples leads to a feasible solution, and these are all centered around the original objective value. This is presented in Figure 5.2.

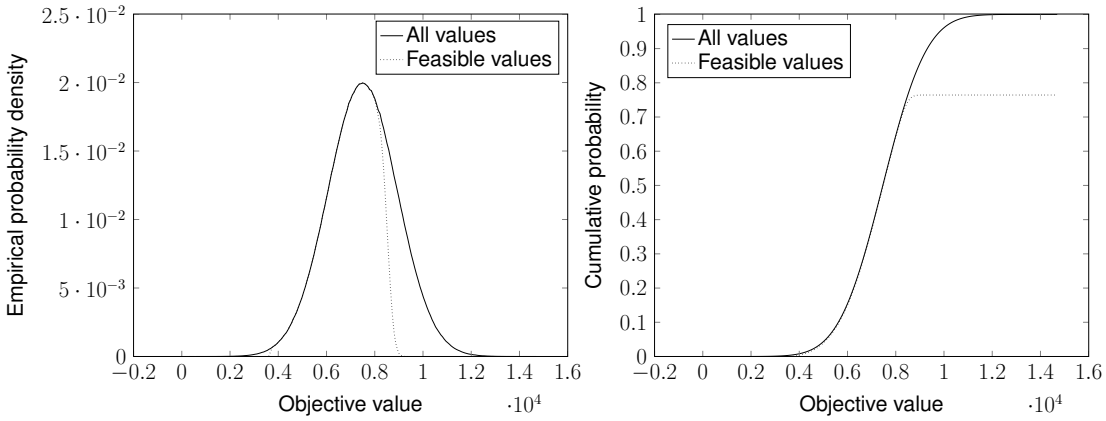


Figure 5.3: The objective value in a change in b for the second basis. The left picture shows the empirical probability density, the right picture the cumulative probabilities.

We can see in Figure 5.1 and in Table 4.1 that we thus need to use an MC simulation to find these answers. This yields:

- The average cost for feasible x_B is $\mu_y = 7048$.
- The probability that for feasible x_B the cost is higher then the current cost 7294 is 42 %.
- The probability that for feasible x_B the cost is 5% higher then the average cost 7048 is 38 %.

The last performance measure is to compute the probability that the cost in this alternative is less or equal to the cost in the expected values scenario. As this is the expected values scenario, it makes no sense calculating this measure.

In the same way as above we can compute the performance measures for basis $B = \{1, 3, 4, 6\}$. The individual feasibility probabilities are:

$$P(x_1 \geq 0) = \Phi\left(\frac{500}{\sqrt{7813}}\right) = 1.00, \quad P(x_2 \geq 0) = 0.996, \quad P(x_3 \geq 0) = 0.77, \quad P(x_6 \geq 0) = 0.90.$$

Furthermore we can calculate:

- The probability that x_B is feasible is 76 %. We thus cannot assume feasibility.
- The average cost for feasible x_B is $\mu_y = 6954$.
- The probability that for feasible x_B the cost is higher then the current cost 7294 is 43 %.
- The probability that for feasible x_B the cost is 5% higher then the average cost 6954 is 42 %.
- The probability that if both bases are feasible, the expected values basis gives better cost then this basis is 96%.

A plot of the probability distribution function of the objective value assuming feasibility is given in Figure 5.3.

Step 6: Determine a small set of bases. If we had performed the previous analysis for more then two bases, we would now select the most promising bases to compare. As we considered only two bases in this example, this step is not necessary.

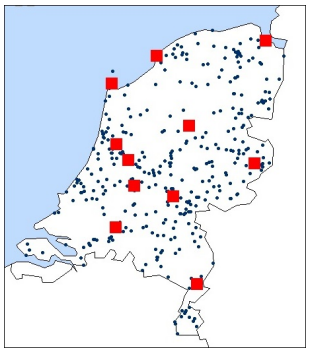


Figure 5.4: The map of locations corresponding to the oil test case. The squares correspond to depots and the dots correspond to customers.

Step 7: Compare alternatives. In this step we calculate the probability that a basis gives better cost then the other alternatives. In Step 5 we already compare the alternatives to the expected values case. As the bases we have left now are the expected values case and one alternative, we have no need for further calculations.

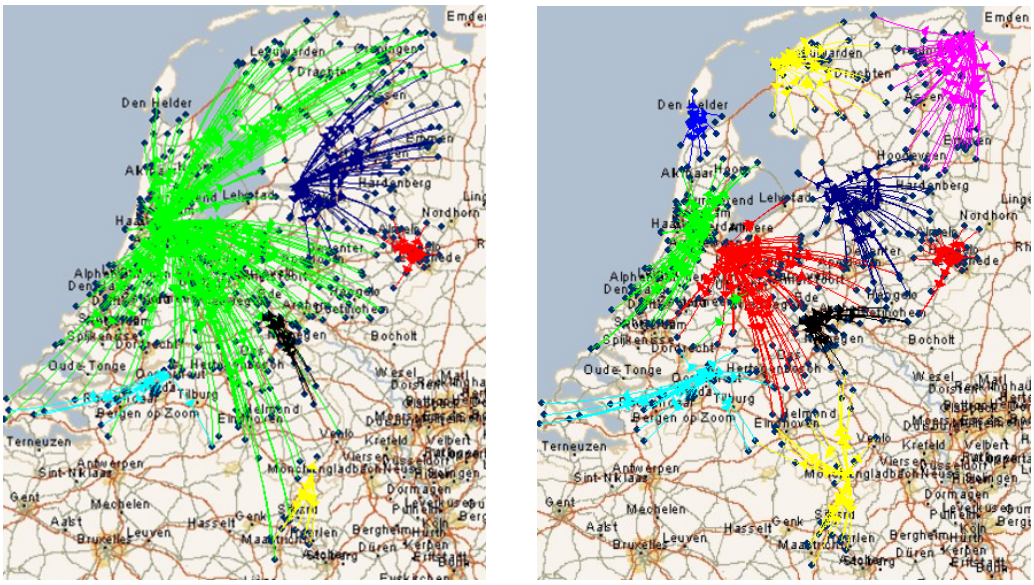
Step 8: Report the results and choose a basis. Consider the results given in Step 5. As we can see, the alternative basis is more often feasible then the expected values basis. Furthermore, this alternative basis gives better expected cost. However, for realizations of b that both bases are feasible, the expected values basis gives better costs in nearly all cases. At first sight, this is counterintuitive: the basis with lower average cost has nearly always higher costs then a basis with high average cost. The difference can be explained when Figures 5.2 and 5.3 are compared. In Figure 5.2 we see that the feasible values are clustered around the expectation of the normal distribution. High and low values correspond to non-feasible values for x_B. In Figure 5.3 we see a difference: only the high values for y correspond to non-feasible values.

We therefore conclude that the decision maker should make a trade-off here. If feasibility is more important, the alternative basis should be chosen. If lower costs are more important, the expected values basis should be chosen.

To make a better trade-off, we can further analyze the model and see if we can reduce the variance in the cost and variables. Consider the covariance matrix of x_B in Equation (5.4). Here we can see that the last variable has most variance, corresponding to the slack value of the labour constraint. Thus, we can repeat the above analysis for a lower variance in the labour constraint. If that produces results that are much better, then the conclusion would be that it is worthwhile to invest in more certainty there.

5.3 The oil test case.

Consider a supplier in the Netherlands. The company sells two products to approximately 350 customers and delivers these from eleven depots: five own depots and six depots that are rented from a third party. Their annual distribution costs are approximately 6,2 million euro. The main depot has the lowest cost, the other own depots are slightly more expensive and the commercial depots have the highest costs. The current costs should be analyzed and the future strategy should be determined: which depots should be used? Which customers should be served from what depot? The depots and customers are shown on a map in Figure 5.4. The company analyzed is a customer of ORTEC. As the procedure is already demonstrated in the previous section, we will now restrict ourselves to the details of this test case.



(a) Product 1

(b) Product 2

Figure 5.5: The assignment of customers to depots in the Base Case scenario, which is a model of the current situation. A line between a depot and a customer means that the customer is served from that depot.

Scenarios are created with different depot selections. Furthermore, scenarios are created by adjusting the delivery of products. Are the products delivered separately or is an investment made to transport them both in one truck? Should we always drop one or more full trucks of product or is it better to deliver only as much as is needed for the next five days? First the optimal depot selection will be studied. The delivery adjustments will then be applied to the optimal depot selection to find the best strategy. The scenarios are:

- Base Case: use own depots and all six commercial depots, current depot volumes as constraints (BC).
- Use only own depots (S1a).
- Use only the own main depot (S2a).
- Use own depots and all six commercial depots (S3a).
- Use own depots and three commercial depots (S4a).
- Use own main depot and all six commercial depots (S5a).
- The default distribution is separate distribution of Product 1 and Product 2 . The options are:
  - Separate distribution of Product 1 and Product 2 (a).
  - Combined distribution of Product 1 and Product 2 (b).
  - Maximal distribution for customers: drop full trucks (c).
  - Minimal distribution for customers: drop an amount that lasts at least five days (d).
  - Combined and minimal distribution (e).

There are five own depots, one of which is the main depot. This main depot can deliver to customers and to other depots. Furthermore, there are six commercial depots. In the Base Case, all these depots are used. The current situation or Base Case scenario is given in Figure 5.5.

Step 1: Data gathering. The current situation is modeled and validated in a Base Case. For each depot we have a flow constraint, which assigns the maximum flow of a depot. In the Base Case the flow constraints are assigned with the current values, to force the optimization model to chose a current utilization. In the later optimization, this maximum flow will have an infinite value. The corresponding LP has 9195 variables and 10006 constraints, including inequality constraints.

Step 2: Uncertainty gathering. There are various sources of uncertainty in this test case. Possibilities are the demand of customers, costs of truck drivers, fuel costs for the trucks, etcetera. We will now study the effect of uncertainty in demand. Assume that the demand for a product can vary with at most 5%. Furthermore assume a correlation coefficient $\rho = 0.7$. The demand per customer is a parameter in the constraint vector b. The first 762 constraints of the LP make sure all customers get at least their demand of product. The other values of b will be deterministic.

Step 3: Generate alternatives. In this oil test case, the strategic decisions what depots to open or close are taken outside of the optimization model. By assigning a maximum flow of zero to a depot, we can force a closure. Note that this is different from the approach as explained in Section 5.1. In this test case, we adjust values in the constraint vector b and use those to choose a different basis B. This is due to the modeling of this specific case. Ideally, we would not need to adjust values in b.

The two methods are however equivalent: consider Section 3.2, where we transfer uncertainty from b to A. We can use a similar technique to transfer parameters that are not uncertain, but that we want to change according to a scenario. Introduce two new variables x_{n+1} and x_{n+2}. Let b^1 be the vector corresponding to scenario 1 and b^2 the vector corresponding to scenario 2. The scenarios are e.g. whether a depot is available or not. Then introduce two new columns in A and one new row. Then define as new data A', b' and c':

$$A' = \begin{bmatrix} A & -b^1 & -b^2 \\ 0_n^T & 1 & 1 \end{bmatrix}, \quad b' = \begin{bmatrix} 0_m \\ 1 \end{bmatrix}, \quad c' = \begin{bmatrix} c \\ 0 \\ 0 \end{bmatrix}.$$

Note that to ensure correctness, we need that either x_{n+1} or x_{n+2} is a basic variable. If both are in the basis or none is, the result is not correct. A more straightforward approach is to introduce binary variables, which is suggested in Section 6.2.

Step 4: Transfer uncertainty. This step is not necessary, as there is no uncertainty in the matrix A of the LP.

Step 5: Evaluate each alternative basis. The uncertainty is all in the constraint vector b. As presented in Table 4.1, we can analytically derive the probability distribution of x and the probability that a single variable x_i is nonnegative. With the simulation presented in Section 4.3 we can find the probability that all variables are nonnegative and thus the solution is feasible. It turns out that except for the Base Case, all scenarios are approximately always feasible. The results are summarized in Table 5.1.

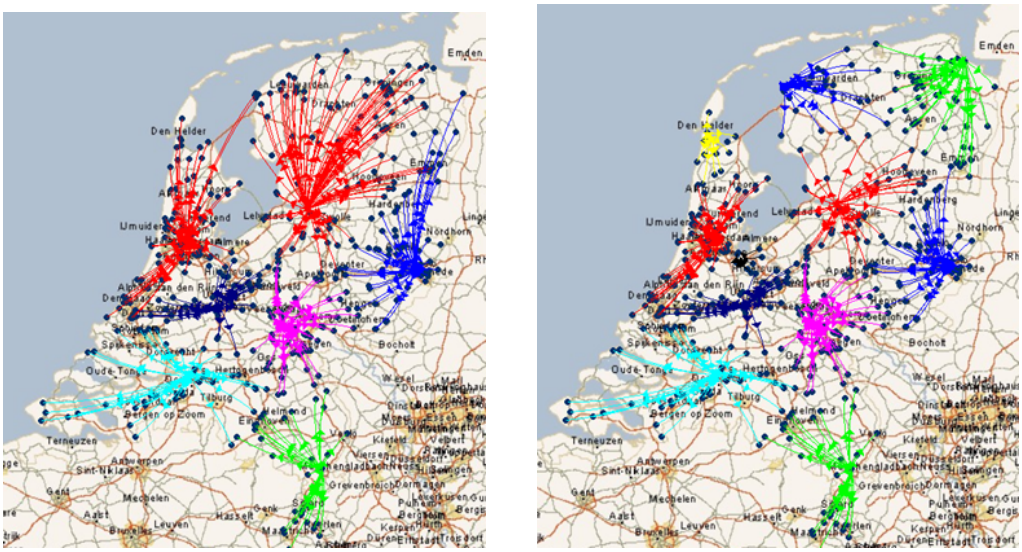
Step 6: Determine a small set of bases. Note the structure in the results: more open depots lead to lower costs. These depots are the same as in the Base Case, but the assignment of depots to customers was not optimal. Furthermore, the combined distribution of products is advisable. The assignment in Scenario 3 is shown in Figure 5.6. As we can see in Table 5.1, Scenario 3c and 3e have similar costs and score equally on the other performance measures. Therefore, we select Scenario 3c and 3e as the small set of bases.

Step 7: Compare alternative bases. We calculate the probability that the cost of S3c is less then the cost of S3e. This probability turns out to be zero: the cost of scenario S3e is under demand uncertainty always at least as good as the cost of scenario S3c.

Step 8: Report the results and choose a basis. The best basis is that of Scenario S3e: it always performs better or equal to other scenarios.

	(b) Expected distribution cost: $\mathbb{E}[y]$	(a) Feasible in % of cases: $\mathrm{P}(x \geq 0)$	(d) Worse then current cost in % of cases: $\mathrm{P}(y \geq \mathbb{E}[y_{BC}])$	(e) Cost increase of more then 5% in % of cases: $\mathrm{P}(y \geq 1.05\mathbb{E}[y])$	(f) Better cost then BC in % of cases: $\mathrm{P}(y \leq y_{BC})$
BC	€6.191.663	2.55 %	50 %	5.42 %	50 %
S1a	€7.849.883	100 %	100 %	1.75 %	0 %
S2a	€8.636.681	100 %	100 %	1.75 %	0 %
S3a	€5.209.702	100 %	0 %	1.75 %	100 %
S3b	€4.135.935	100 %	0 %	1.75 %	100 %
S3c	€3.908.088	100 %	0 %	1.75 %	100 %
S3d	€4.936.354	100 %	0 %	1.75 %	100 %
S3e	€3.892.095	100 %	0 %	1.75 %	100 %
S4a	€5.858.364	100 %	0.8 %	1.75 %	100 %
S5a	€5.465.244	100 %	0 %	1.75 %	100 %

Table 5.1: The results for the oil test case.



(a) Product 1 (b) Product 2

Figure 5.6: The assignment of customers to depots in the third scenario. Compare to Figure 5.5 to see that the routes are shorter then in the Base Case scenario. Furthermore, the Amsterdam depot is used less.

Analysis of results As not all results in Table 5.1 are intuitive, we will analyze and motivate them in this section. The feasibility is low in the Base Case, but 100 % in all other scenarios. If we analyze the results, we can see that each customer is served through one depot. The corresponding variables have therefore the same average as the demand values or have value zero. As the demand varies at most 5 %, the corresponding values of x vary with only 5% and therefore have a very low probability of infeasibility. The other variables are related to these: they represent the volumes of depots or some related costs. But as the amounts that have to be distributed never get negative, the related variables neither get negative. This does not hold for the Base Case scenario, as we assigned maximal values to the depots. Infeasibility corresponds in that case to a depot that has not enough capacity and the corresponding slack variables become negative.

Another striking result is the percentage of realizations that has a cost increase of more than 5%. This is, except for the Base Case, equal for all cases. The reason for that is found in Theorem 10. As the standard deviations of the stochastic variables in b are all defined as a percentage of the average value for b, we have that the expectation of y divided by the standard deviation of y form a constant fraction. As in each scenario we calculate $\Phi(\frac{\mu_y}{\sigma_y})$, we thus find that for a constant fraction we have a constant probability. In this Base Case this fraction has a different value due to the extra constraints. This is different in the Base Case, as there we cannot assume feasibility and thus we cannot use the properties of the normal distribution.

We can explain the result of the last column with the following argument. As the demand increases, the cost of all compared scenarios will rise. Although one may have a steeper increase in cost then the other, only scenarios that have a very similar expected cost are likely to show a different result. Only in extreme cases, one scenario can outperform the other. As the variance of variables is low, will rarely happen.

5.4 Conclusions

Recall the questions that were stated in the introduction of this work:

> 1. Suppose we have chosen a solution that has optimal cost. Can we be sure that costs will not increase too much when data changes?

We see that in the small example, the costs were very sensitive to data changes. In the oil test case, however, costs are relatively stable and do not increase much.

> 2. As data changes, it might be possible that the chosen solution does not remain feasible. For example, a factory has to produce more than its capacity. How sure are we that the chosen solution stays feasible under a change of data?

In the small example we see that different bases give a different probability of feasibility. A further analysis of the example is there necessary to investigate if there are other bases that give a better feasibility. In the oil example, the chosen solution is robust for changes and it is certain that the chosen solution stays feasible.

> 3. Suppose we have chosen a solution that is optimal using expected values of data.. Can we give a guarantee that it will stay optimal when data changes? Thus, that although costs may rise, the costs are still lower than that of alternative solutions.

In both examples we see that the basis with lowest cost in the deterministic LP also gives the lowest cost for other realizations of data.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

We present the conclusions of this research project in Section 6.1. In Section 6.2 possible extensions and generalizations are discussed.

6.1 Conclusions

The trigger for this research is the effect of data uncertainty in Linear Programming. We consider a Linear Program with multivariate normal distributed costs and constraint vectors. Furthermore we use the basic solution of the LP and not the exact values of all parameters. The research questions are given in Section 1.4.

First, we investigate which methods are available in the literature: an extensive literature review is done in Chapter 2. Several common methods to deal with uncertainty in LP's are described and their properties are discussed in Section 2.1. There are basically two kinds of methods:

- Methods that model the LP with some risk measure for uncertainty, rewrite it to a deterministic equivalent and then optimize.
- Methods that optimize the LP as if no uncertainty exists, and then evaluate the effect of uncertainty on the obtained solution.

Many of these methods focus on one optimal solution, either by analyzing a deterministic optimal solution or by optimizing over the uncertainty. However, we prefer to have some alternative solutions as not all aspects of the problem might be modeled or the exact uncertainty model is not known. We therefore conclude the chapter with a description of the properties required in this project.

Second, we derive expressions for performance measures for alternative solutions. These are feasibility, the probability that a basis is optimal, the probability on outliers and the a comparison of costs of two bases. To do so, we list and motivate assumptions in Chapter 3. We consider Linear Programs, basic solutions, and normal distributions in constraint vector b and cost vector c. We then state exactly which probabilities we want to derive. These probabilities are given in Lemmas 5 - 8 in Chapter 4. Furthermore, the distributions for the basic solution x_B and the cost y are derived in Theorems 9 - 12. We see that we can always calculate the required probabilities using a Monte Carlo simulation and describe how that can be done in each case. Furthermore, we derive that in some special cases the probability can easily be calculated using only some matrix-vector calculations and the standard normal distribution. This is summarized in Table 4.1.

Third, we develop a method to systematically evaluate of a Linear Program for varying costs and constraint values in Chapter 5. This method consists of the following steps. First we formulate an LP with probability distributions. Then alternative bases are generated and evaluated. After that, they are compared and if necessary extra calculations are done. In this way, we make a systematic comparison of the effects of uncertainty in several alternatives. This method is summarized in Figure 5.1.

Finally, we test the derived theory and method on a real-world case in Section 5.3 and 5.4. We see in the results for both a small example and a real-world case that we are able to answer the questions that have been stated in the introduction. We can make statements about the feasibility, optimality and outliers of a solution to a Linear Program. We therefore suggest that this method is further analyzed and implemented.

6.2 Generalization and possibilities for future research

During this research several discussions have been held on possible extensions or generalizations of this work. In Chapter 3 we state four assumptions, which are reconsidered in this section.

LP assumption. The approach is based on a Linear Program, thus all constraints are linear equalities and all variables continuous non-negative variables. This is the standard Linear Program as given in 1.1. The approach is not restricted to only this LP, as other types of LP's can be rewritten to the standard:

- Inequality constraints: these can be rewritten to equality by adding a slack variable.
- Variables that can be negative: replace x by $x^+ - x^-$, which both have to nonnegative.
- Variables with a range: add a constraint to ensure $x \leq a$.

Recall from Section 3.1 that we are mainly interested in the strategic and tactical decisions resulting from a solution. The exact amount that is transported from a depot to a customer is not that important, it is more important to know which customer is served from what depot and where full capacities are used and where slack is available. Now consider a binary variable. This variable often represents a yes or no decision: open a location or not, do an investment or not, use a facility or not. Thus, we can fix the variable to zero or one and then consider it as a constant. This can be done by either removing the variable or by adding an extra constraint to ensure its value. The same is possible for problems with integer variables by fixing the variable at the obtained value.

Extension to non-linear programs has not been considered, as the linearity of the problem is extensively used by the normal distribution and the matrix-notation. Thus, it is not likely that a similar approach will work for non-linear optimization problems.

A basic solution. In the oil test case we occasionally encountered variables that had been given a basic status in CPLEX, but had value zero and gave a matrix that was not of full rank. This could then be resolved by removing these variables from the basis. It would, however, be better to use one definition of a basis and hold on to that. That rises the question whether all reasonable alternatives can be written as a basis solution. We know that the there is always a basic solution that is optimal. What if we consider a solution that is not optimal and neither basic?

Normality assumption. The normal distribution has been used in this thesis as many real world data can be described by a normal distribution. Furthermore, linear combinations of normal distributions form again a normal distribution. A possible generalization is to consider phase-type distributions: under certain conditions, a sum of them will again be a phase-type distribution. Uniform distributions, however, involve difficult convolutions which makes it difficult to derive analytic expressions for the performance measures.

Uncertainty in b and c. In Section 3.2 we suggested inverting the matrix A to transfer its uncertainty to b. In the same section, we also mentioned the difficulty of this: the inverse of a stochastic matrix is a complex matter. So what if we do not use the inverse but consider $P(Ax = b)$ directly. Then if A has some nice properties, we may be able to derive properties of x directly from $P(Ax = b)$. Another possibility is to not give each element of A a probability distribution, but to consider the eigenvalues of A. This will simplify calculations as we then have probability distribution of a vector instead of a matrix. However, the interpretation of eigenvalues in LP's then has to be considered.

APPENDIX A

THE DYER-FRIEZE-MCDIARMID BOUND

In [13], a bound on the optimal value of a Linear Program with random cost coefficients is given. In contrast to the previous sections, this is not based on one basis on which then the uncertainty is applied. It assumes that for every realization of the random variables an optimal solution exists. Thus, each 'optimal' solution is scenario-specific. The DFM-bound corresponds to the probability distribution of the best possible cost of all these scenarios. The thought behind the proof is given in Section 2.6 and the more interested reader is referred to the original article [13] and a more lengthy explanation with examples in [40].

Thus, at first this might seem counter-intuitive: why construct a probability distribution over all costs if these costs assume that the realization is known in advance. However, it might provide a valuable bound on the question how good we could have done if we will get to know the realization. If the DFM-bound is sharp, then this can be used to estimate a cost of uncertainty – it can give an estimate of the extra costs that arise due to the choice we have to make before a realization is known. In Figure A.1 a graph is given to explain this further.

Now we will investigate the practical value of the DFM-bound. This is not meant to be a complete overview of the properties, but only a couple of examples will be considered. The first example will be the one as given in Section 5.2, this will then be extended to more variables. As the bound is mainly used in applications that can be formulated as an assignment problem, we will finally consider that problem.

A.1 The small factory example

Consider the following example to investigate the tightness of the bound. This is the example that is discussed in more detail in Section 5.2.

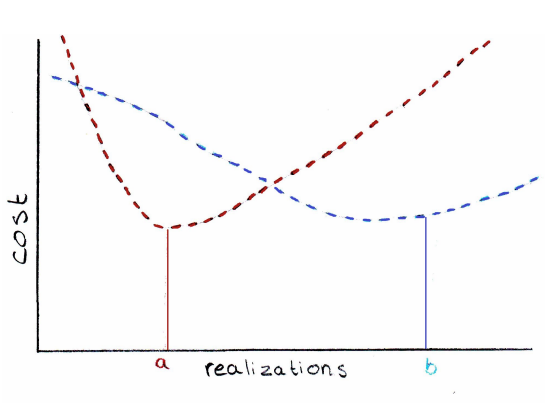
$$\begin{array}{rrrrrrrrrr} \min y & = & c_1x_1+ & c_2x_2+ & c_3x_3 & & & & & \\ \text{s.t.} & & x_1+ & x_2+ & x_3 & & & & = & 1000 \\ & & 3x_1+ & 7x_2+ & 5x_3 & +x_4 & & & = & 4500 \\ & & 15x_1+ & 10x_2+ & 7x_3 & & +x_5 & & = & 11000 \\ & & 2x_1+ & 3x_2+ & 5x_3 & & & +x_6 & = & 4500 \\ & & & & & & & x_i & \geq & 0 \end{array} \tag{A.1}$$

Now consider a uniform distribution on the coefficients: $c_j \sim U(0,1)$ for $j = 1,2,3$. For the slack variables $j = 4,5,6$ we have $c_j = 0$. For the uniform distribution, we have $\beta = 1/2$. A feasible solution for (A.1) would be the following:

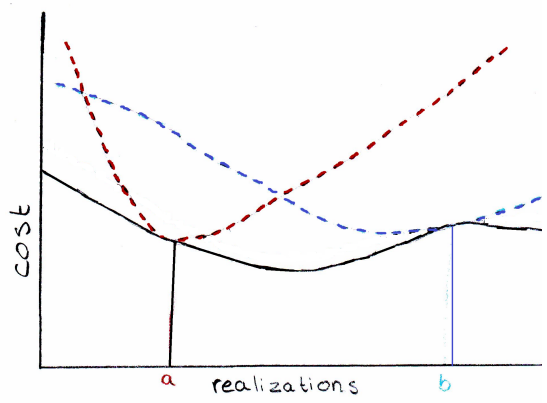
$$x'_1 = 400, x'_2 = 100, x'_3 = 500, x'_4 = 100, x'_5 = 500, x'_6 = 900.$$

This solution would yield an expected objective value of $400/2 + 100/2 + 500/2 = 500$. Note that any other feasible solution will also give this expected value:

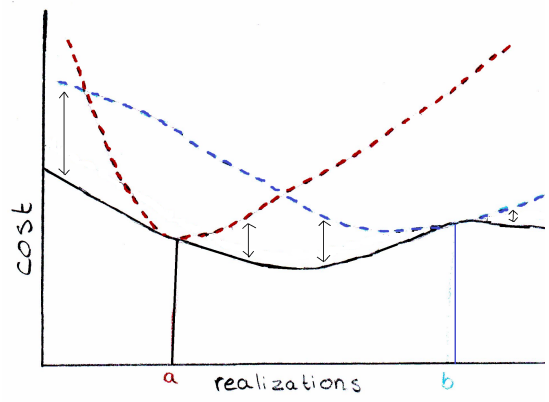
$$\begin{array}{rcl} E[y] & = & E[c_1x_1 + \cdots + c_nx_n] \\ & = & x_1E[c_1] + x_2E[c_2] + x_3E[c_3] + x_4E[c_4] + x_5E[c_5] + x_6E[c_6] \\ & = & \frac{x_1}{2} + \frac{x_2}{2} + \frac{x_3}{2} \\ & = & \frac{1}{2}(x_1 + x_2 + x_3) \\ & = & 500 \end{array} \tag{A.2}$$



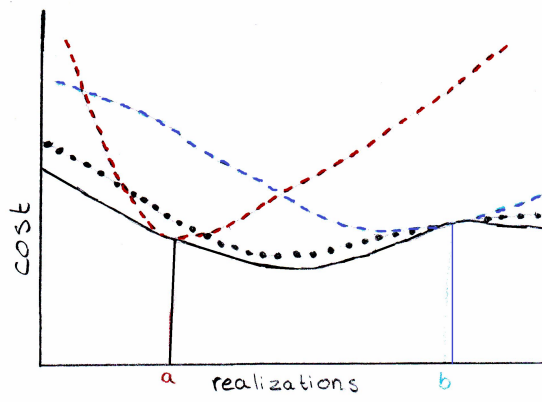
(a) Suppose we optimize for realization a (brown) or realization b (blue). Then these give optimal values for a or for b, but we do not know their quality at other points.



(b) If we could optimize for all other possible realizations as well, the optimal points form the solid line.



(c) Then we can calculate how far from optimal a solution is at a given realization, as indicated by the arrows. An expression for the solid line is not known.



(d) Suppose the dotted line is a bound for the solid line. Thus, if we have a good bound, we can compare solutions to that bound or use the bound to find a robust solution.

Figure A.1: The DFM bound. Can we use the bound to find a solution that is not optimized for one realization, but near-optimal for all realizations? Can we estimate how close a solution is to optimal by calculating the difference between the solution and the bound?

Now use this particular solution to obtain a value for the bound (2.2). As the expectation of the costs of the slack variables is zero, we obtain:

$$\begin{aligned} E[y^*]/2 &\leq \max_{S:|S|=m} \sum_{j \in S} x'_j E[c_j] \\ &= x_1 E[c_1] + x_2 E[c_2] + x_3 E[c_3] \\ &= 500 \\ E[y^*] &\leq 1000 \end{aligned} \tag{A.3}$$

Whatever solution x' is used in this expression for the case with $c_j \sim U(0,1)$, we will always yield this value of 1000: because $x_1 + x_2 + x_3 = 1000$ and the cost coefficients have the same distribution. Thus, there is a factor 2 difference between the bound on the optimal value and the actual optimal value – which is in this case exactly the value $1/\beta$.

The same happens with other uniform values for the cost coefficients. Suppose we have that $c_1 \sim U(0,10), c_2 \sim U(0,14)$ and $c_3 \sim U(0,20)$. The above mentioned feasible solution x' would then give in (2.2):

$$E[y^*] \leq 2 \cdot (400 \cdot 5 + 100 \cdot 7 + 500 \cdot 10) = 15400.$$

However, we can adjust this x to the basic solution $x_1 = x_3 = x_4 = 500$, $x_6 = 100$. This will give the following expectation for the objective value:

$$E[y] = 5 \cdot 500 + 10 \cdot 500 = 7500$$

This value of x is still not optimal, but it illustrates that the bound is not useful for a uniform distribution and a small number of variables.

A.2 The number of variables in the DFM-bound

In the proof of the bound a conditioning argument is used to select the variables with the lowest impact on the objective value. In the above example, this set is fixed as the number of variables is comparable to the number of constraints. If we increase the number of variables, there is more choice possible and therefore the optimal value will be better and the DFM-bound might perform better as well.

Consider a general production problem: an amount of 1000 products has to be produced. This is bound to constraints, which have the following shape for all constraints i:

$$\sum_j a_{ij} x_j \leq b_i$$

In this case, we will consider the case that $a_{ij} \in [0,10]$ and $b = 7000$. The cost coefficients c_j are independent random variables with a uniform $[0,1]$ distribution. First we will construct an example with four constraints, then one with seven constraints and finally one with 200 constraints.

A.2.1 DFM bound with four constraints

We will now give an example with four constraints. The first constraint is that the sum of all variables should be 1000. The second, third and fourth constraint are $\sum_j a_{ij} x_j \leq 7000$ with $a_{ij} \in [0,10]$. The number of variables used will range from one to fifty. In addition, there will be three slack variables due to the inequality constraints. The matrix A will be created once, then the columns corresponding to the number of variables will be selected for use. As an example, for three variables the problem can look like:

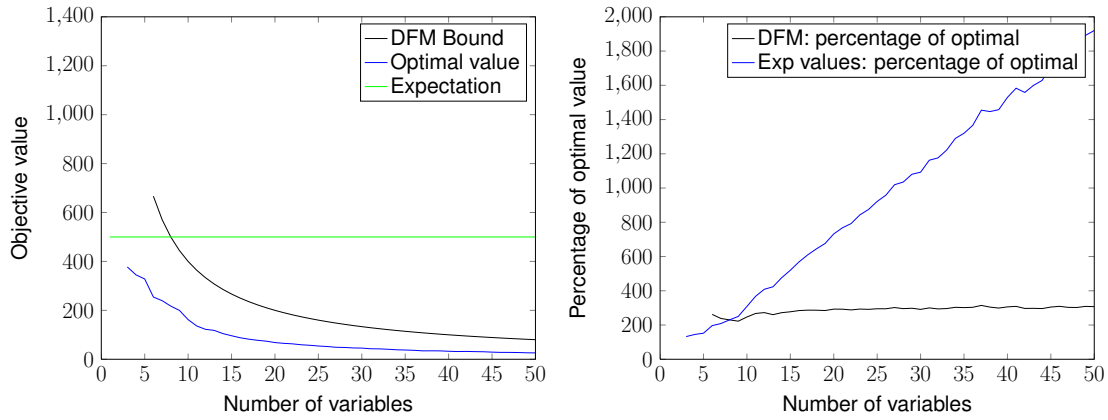


Figure A.2: The DFM bound and the average optimal value for an example with four constraints. Furthermore, the expected optimal value when the expectation of the cost coefficients is used is drawn. The right figure shows the strength of methods used: it gives the bounds as percentage of the optimal value.

$$\begin{array}{rrrrrrrrrrr} \min y & = & c_1x_1+ & c_2x_2+ & c_3x_3 & & & & & & \\ \text{s.t.} & & x_1+ & x_2+ & x_3 & & & & & = & 1000 \\ & & 6.92x_1+ & 7.04x_2+ & 1.98x_3 & +x_4 & & & & = & 7000 \\ & & 2.63x_1+ & 0.54x_2+ & 6.92x_3 & & +x_5 & & & = & 7000 \\ & & 6.03x_1+ & 5.49x_2+ & 9.77x_3 & & & +x_6 & & = & 7000 \\ & & & & & & & & x_i & \geq & 0 \end{array} \tag{A.4}$$

For this instance, the DFM-bound is calculated. After that, realizations of c are generated and for each of the realizations the optimal value is calculated. Then, the average optimal value can be computed. If the DFM-bound would be perfect, the average optimal value and the DFM-bound will have the same value.

After that, a variable will be added to problem (A.4). To do so, a column will be inserted:

$$\begin{array}{rrrrrrrrrrr} \min y & = & c_1x_1+ & c_2x_2+ & c_3x_3+ & c_4x_4 & & & & & \\ \text{s.t.} & & x_1+ & x_2+ & x_3+ & x_4 & & & & = & 1000 \\ & & 6.92x_1+ & 7.04x_2+ & 1.98x_3+ & 8.15x_4 & +x_5 & & & = & 7000 \\ & & 2.63x_1+ & 0.54x_2+ & 6.92x_3+ & 6.63x_4 & & +x_6 & & = & 7000 \\ & & 6.03x_1+ & 5.49x_2+ & 9.77x_3+ & 5.69x_4 & & & +x_7 & = & 7000 \\ & & & & & & & & x_i & \geq & 0 \end{array} \tag{A.5}$$

Again, the DFM bound will be calculated and samples of c will be generated to calculate the average or expected optimal value. This procedure is repeated until there are 50 variables. The results are presented in Figure A.2.

As we can see in Figure A.2, the value of the DFM-bound and the optimal value are decreasing in the number of variables. This can be explained by the freedom introduced by the extra variables. As there are more variables, there are with higher probability variables that yield a feasible solution with low cost coefficients. The DFM-bound uses that knowledge, but appears to be a constant factor larger then the actual expected optimal value. In this particular example, this is around a factor 3.

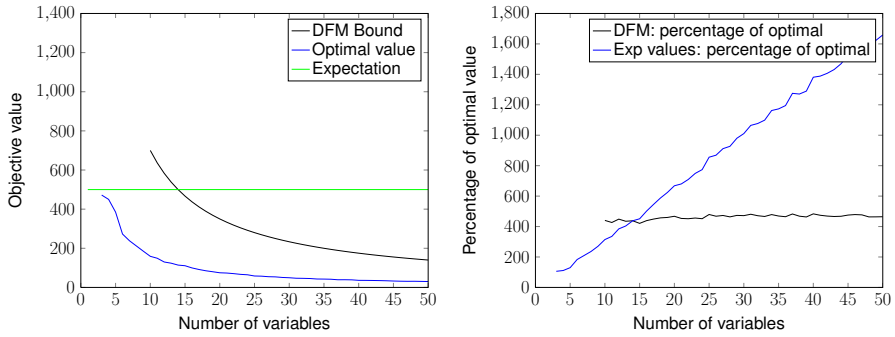


Figure A.3: The DFM bound and the average optimal value for an example with seven constraints. Furthermore, the expected optimal value when the expectation of the cost coefficients is used is drawn. The right figure shows the strength of methods used: it gives the bounds as percentage of the optimal value.

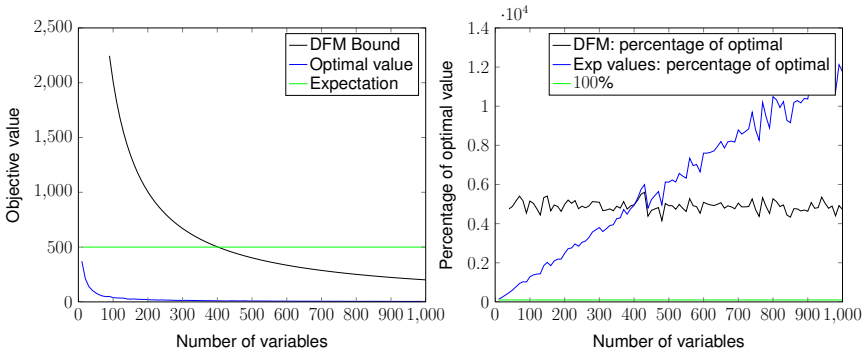


Figure A.4: The DFM bound and the average optimal value for an example with 200 constraints. Furthermore, the expected optimal value when the expectation of the cost coefficients is used is drawn. The right figure shows the strength of methods used: it gives the bounds as percentage of the optimal value.

A.2.2 DFM-bound with seven constraints

Consider a similar example, now with seven variables. The results are shown in Figure A.3.

The results are similar to the example with four constraints. The main difference is in the strength of the DFM-bound: it is always at least four times the expected optimal value. Only from fifteen variables on, it is better then using the deterministic equivalent of the problem.

A.2.3 DFM-bound with 200 constraints

As these are only small examples, the question arises what the sharpness of the DFM-bound is for larger problems. Therefore, a problem with 200 constraints is also constructed in the same way as the above two problems. The results are shown in Figure A.4.

In this example with 200 constraints the DFM-bound seems to be useless: it is always at least a factor 40 more then the expected optimal value.

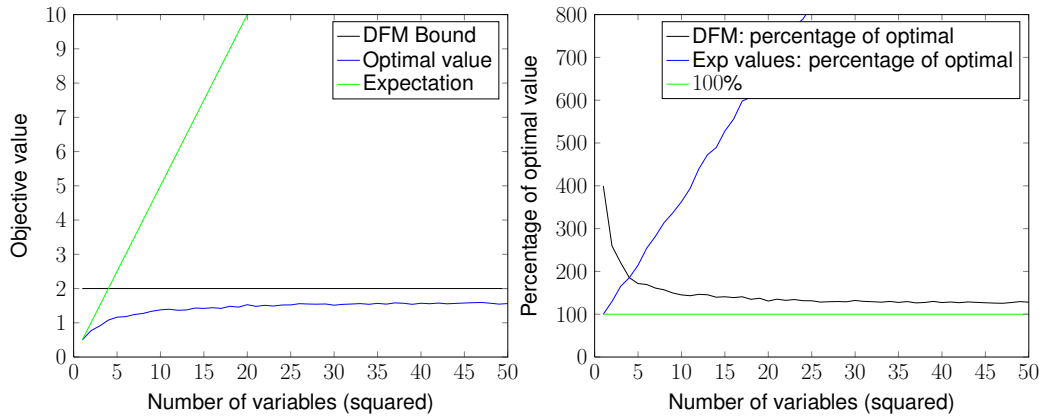


Figure A.5: The DFM bound and the average optimal value for the assignment problem. Furthermore, the expected optimal value when the expectation of the cost coefficients is used is drawn. The right figure shows the strength of methods used: it gives the bounds as percentage of the optimal value.

A.3 The Assignment Problem

In the previous examples, the DFM-bound turned out to perform badly. At the other hand, this is an example with a very dense A matrix. In more realistic problems, this matrix might be more sparse. This is because most constraints will not involve all variable, but only a few of them. However, we cannot give a general example which clearly illustrates at which instances the DFM-bound gives sharp results or when it does not. Neither can we give a proof that indicates the performance of the bound.

The DFM-bound was originally used to give a bound on the optimal value for the assignment problem. Other work that refers to [13] is also related to this assignment problem. Therefore, we can assume that the best results are gained in the assignment problem. We will now investigate the value of the DFM-bound in that problem. Thus, we want to find the expected optimal value for the following problem:

$$\begin{array}{lllll} \min y & = & \sum_{i,j} c_{ij}x_{ij} & & \\ \text{s.t.} & & \sum_i x_{ij} & = \; 1 & \forall j \\ & & \sum_j x_{ij} & = \; 1 & \forall j \\ & & x_{ij} & \geq \; 0 & \forall i,j \end{array} \tag{A.6}$$

Assume that c_{ij} are all independent random variables with a uniform distribution on $[0,1]$. As the cardinality of i and j should be equal, the number of variables will be $1^2, 2^2, 3^2, \ldots$. The results are presented in Figure A.5.

As we can see, the DFM bound is much better in the assignment problem. However, the bound still gives a 50% overestimation of the optimal costs when $50^2 = 2500$ variables are involved.

Concluding, it is not possible to make general statements on the quality of the DFM-bound based only on the previous examples. However, the results do not give an indication that this bound will give a useful estimation of costs in a real-world problem as 5% margins are already significant there. Thus, [13] might be of great theoretical value, it will not be incorporated in method of this work.

APPENDIX B

IMPLEMENTATION DETAILS

The ORTEC modeling tool BOSS is implemented in AIMMS. Although this is a very powerful program to model and optimize mathematical programs, it has less possibilities for simulations and matrix manipulations. The computation of the performance characteristics of a basis is this implemented in MATLAB. There are six main steps:

1. Formulate and solve a Linear Program.
2. Read the files produced by AIMMS into MATLAB.
3. Determine the probability distributions.
4. Remove redundancy from the LP.
5. Construct a basis matrix.
6. Calculate the performance measures.

These six steps will now be discussed in some more detail, if necessary pseudo code is given. As the implemented code is lengthy it is not included in this report but available on request.

B.1 Formulate and solve a Linear Program

Suppose we have a distribution problem with given data. The first step is to model this as a Linear Program. ORTEC has an application for this: BOSS, which is built in AIMMS. It is a tool for supply chain optimization studies and decision support at a strategic and tactical level. It calculates the optimal supply chain configuration within the allowed solution space. All data can be entered in a Microsoft Access database: information on costs, on demands, on locations, restrictions and many more. Several objectives can be chosen: minimize costs, maximize profit, minimize waste or minimize shortages. The solver CPLEX can produce reports on the mathematical LP formulation and the corresponding optimal solution.

B.2 Read the files produced by AIMMS into MATLAB.

During the optimization in AIMMS, the solver CPLEX can produce three files on the model. The first file is a .LP file, which gives the LP that is being solved in the same presentation as on paper. The second is a .MPS file, which gives the same LP but in a format that is easier to read into another program. The third is a .SOL file, which gives the solution characteristics. The .SOL file and the .MPS file are used to read in the LP and the corresponding solution.

The .MPS file. The MPS file starts with a declaration of constraints. This is marked by the line "ROWS". Each next line then contains a letter and a name. The letter (N, G, E, L) denotes the kind of constraint (Objective function, $\geq$, $=$, $\leq$). The name gives the constraint name, usually a "'c" followed by a number.

The next section starts with the line "COLUMNS". Each next line then contains a variable name, a constraint name and a value. These fill the constraint matrix A and objective function c. Then a line

"RHS" marks the start for the constraint coefficients. Each line starts with "RHS", then followed by a constraint number and a value. These form the vector b. The default declaration for a variable is ≥ 0. After the line "BOUNDS", integers and other ranges are declared.

The .SOL file. After some general solution characteristics, the line "<linearConstraints>" marks the start of the constraint information. Each rule gives the constraint number, the status basic or not, the slack value and the dual value. We store the constraint number, the status and the slack value. The end of this section is marked by the line "</linearConstraints>"

The start of the information on variables is marked by the line "<variables>". As with the constraints, each rule gives the variable number, the status, the value and reduced costs. We store the variable number, the status and the value. The end of this section is marked by the line "</variables>".

B.3 Determine the probability distributions.

In the previous step, we have produced matrices and vectors giving the LP and the solution values. Now the information on uncertainty should be added. The content of this function is strongly dependent on the uncertainty implemented. The general version implemented assumes a percentage of change in a coefficient and a standard correlation coefficient. The coefficient or constraint numbers used in the LP indicate which parameters are variable. Assuming the parameter value of the LP is also the mean of the corresponding distribution, this is enough to specify the distributions. Take a change percentage of $\alpha\%$: 95% of realizations will be not more then $\alpha\%$ different from the mean. Furthermore assume a value ρ for the correlation coefficient. Then we can derive:

$$\begin{aligned} \sigma_i &= \tfrac{1}{2}\alpha\mu_i \\ \Sigma &= \rho\sigma\sigma^T + I\tfrac{1-\rho}{\rho}\sigma^2 \end{aligned}$$

Note that the correlation coefficient should be one at the diagonal entries. This is corrected for in the second term.

Find the uncertainty. The modeler will in general not formulate an LP constraint by constraint, but will give a structure of the problem in Access and collect the necessary data. Furthermore, it can be difficult to give a probability distribution of the uncertain data. This is because data is usually uncertain because of a lack of information. Thus, we need to give some tools to translate statements like 'the price of wood will be around 10' into a normal distribution of a cost coefficient somewhere in the LP.

The place of uncertainty can be traced using the terminology in Access and AIMMS. In the modeling section in AIMMS, a search can be performed using CTRL+F to see in what constraints or objective functions this appears. After that, the 'Math Program Inspector' in AIMMS can be used to locate the exact constraint numbers and variables where this parameter appears.

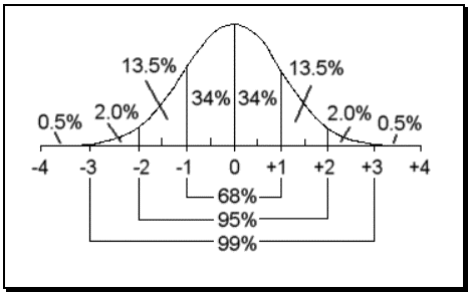


Figure B.1: The percentiles for a normal distribution.

Estimate the distribution of a parameter. The distribution of the uncertain parameter can then be estimated using the percentiles of a normal distribution, as given in Figure B.1. The expectation and variance of a parameter can be estimated if one of the following questions can be answered:

- Approximately two-third of realizations of the parameter α will be in the interval $[a, b]$. Thus, we have $E[\alpha] = \frac{a+b}{2}$ and $Var[\alpha] = \frac{(b-a)^2}{4}$.
- Unless in exceptional events, the parameter α will be in the interval $[c, d]$. This corresponds to a probability of 95% and gives $E[\alpha] = \frac{c+d}{2}$ and $Var[\alpha] = \frac{(d-c)^2}{16}$.

It would be more accurate if the distribution could be derived from historical data. However, this data will often not be available. In that case, the above estimates can be used.

Estimate the dependency between parameters. The covariance matrix of parameters can be estimated by giving correlation coefficients. Many parameters will be somehow be related: if the demand for gas of one customer will rise, it is likely that the demand of another customer will also rise. The correlation coefficient is a number in the interval $[-1, 1]$. A value zero means that a change in the one parameter will give no information about a change in the other parameter. A value one means that if we know the change in the one parameter, we can derive the change in the other parameter. It furthermore means that the sign of the change is equal: if the one parameter has a positive change, the other will have a positive change as well.

Guidelines for the interpretation of the correlation coefficient are:

- A value of 0: knowing one parameter gives no extra information on another parameter.
- A value of 0.7: There is a relationship between the two parameters, but knowing one parameter is not enough to carefully predict the other. Knowledge of the first parameter reduces the error of the prediction of the second parameter with 30%.
- A value of 0.9: There is a strong relationship between the two parameters. Knowledge of the first parameter reduces the error of the prediction of the second parameter with 56%.
- A value of 1: Knowledge of the one parameter gives enough information to precisely predict the value of the second parameter.

This correlation coefficient can be chosen for a group of parameters, for example 'the demand of customers'. Then we can assume each pair of customers has the same correlation coefficient. The covariance between parameter x and y can then be calculated using the correlation coefficient ρ_{xy}:

$$Cov(x, y) = \rho_{xy}\sigma_x\sigma_y.$$

B.4 Remove redundancy from the LP.

The modeler formulates the LP based on constraint rules: 'A truck cannot carry more load then its capacity', 'A flow should be nonnegative'. Furthermore, all variables have an assigned range. This leads to redundancy: There is a constraint that a flow variable should be nonnegative and the constraint declaration makes sure it is nonnegative. Thus, we can reduce the complexity of calculations by removing this redundancy. For each row in A, we check how many variables are involved. We can remove a constraint if it satisfies the following conditions:

- There is only one variable involved.
- The constraint is of type ≥ 0.
- There is no variation assigned to that constraint.

Furthermore, a check is performed to make sure the matrix is of full rank.

Rewrite to a standard LP. As our calculations are performed for a standard LP, constraints with a $\geq$ or $\leq$ sign have to be rewritten into equality constraints. This can be done by adding a slack variable and if necessary change signs.

B.5 Construct a basis matrix.

In this function, the columns of A, c and the covariance matrix of c are sorted into basic and non-basic variables. The matrix A_B is then inverted. The performance measures with an analytic expression can now be calculated. For the others, a Monte Carlo Sampling technique will be used.

B.6 Calculate performance measures.

A property of the multivariate normal distribution is that the integrals involved cannot be evaluated analytically, but that numerical techniques should be involved in the computation. A single normal distribution can easily be rewritten as a standard normal distribution and numerical values for that are widely available. For small numbers of dependent variables, integration schemes are available in the literature. However if the number of dependent variables grows, more and more computational effort is needed to compute the values within reasonable error bounds.

An alternative is Monte Carlo simulation. If we can create samples of the values of the random variables and then compute x and y, we can count how often certain values are observed and estimate the distributions and probabilities based on those. As we are mainly interested in the solution x and objective value y, we will focus on those. Recall that if $b \sim N(\mu^b, \Sigma^b)$, then due to the linear property we have that $x \sim N(\mu^x, \Sigma^x)$. Furthermore, as B is invertible there is a one-to-one correspondence between x and b. Thus, instead of sampling values from the distribution of b and then computing x, we can just as well sample from the distribution of x.

We need to take into account the dependence of the different elements of x. However, we can exploit the properties of the multivariate normal distribution of x to obtain such samples easily. A method for this is given in [33]. One important property is the linear transformation property, which we have also used to derive the relation between b and x. For a normal vector x and full-rank matrix C we have:

$$x \sim N(\mu_x, \Sigma_x) \Rightarrow Cx \sim N(C\mu_x, C\Sigma_x C^T) \tag{B.1}$$

Thus, if we know that $Z \sim N(0, I)$ and $x = \mu_x + CZ$, then $x \sim N(\mu_x, CC^T)$. Thus, if we can find a matrix C such that $CC^T = \Sigma_x$, then we can sample independent values from the standard normal distribution and use those to construct a corresponding sample of x. In the same way, we can construct a sample of c. Then the value $y = c^T x$ can be computed and the values of x and y can be stored.

The matrix C can be composed using a Cholesky factorization. This algorithm provides a lower triangular matrix such that $CC^T = \Sigma$. Such a factorization only exists if the matrix Σ is symmetric positive definite. Thus, if not all values of c or x are stochastic, we should leave them out of this part of the analysis. Furthermore, the covariance matrix has to be of full rank. Even if all variables are stochastic, there can be a strong dependence amongst them. Consider for example the case that one of the coefficients of b is stochastic, then this can influence several values of x. Conversely, if one value of x changes this can have an effect on several values of b. Thus to be sure the covariance matrix is indeed positive definite, it suffices to add a small value ϵ to the diagonal elements.

In short, the algorithm works as following:

```
1. Input: the mean and covariance of c and x, the variance of y.
2. Determine which of the elements of c and x are stochastic and proceed
   with those.
3. Compute the Cholesky factorizations C_x and C_c.
4. Determine the relevant intervals of c, x and y to categorize the
   outcomes in frequency vectors.
5. For each iteration perform the following actions:
   5a. Draw random samples z1 and z2 of standardnormal distributions
       (using randn in matlab).
   5b. Calculate x=mu_x + C_x*z1 and c=mu_c + C_c *z2. Store the values
       in the frequency vectors.
   5c. Determine and store the feasibility of x.
   5d. Calculate and store the objective value y.
6. Calculate the cumulative probabilities.
7. Repeat the steps 5 and 6 to be able to make good confidence intervals.
```

APPENDIX C

LIST OF USED SYMBOLS

Symbol	Description	Properties
A	Data matrix	m by n matrix
A_B	Basic data matrix	m by m matrix
A_N	Non-basic data matrix	m by $n-m$ matrix
B	Index set of basic variables	Set with cardinality m
b	Constraint vector	m-vector
c	Cost vector	n-vector
c_B	Basic cost vector	m-vector
c_N	Non-basic cost vector	$n-m$-vector
D	Domain	-
L	Lower Cholesky factorization	Matrix
m	Number of constraints	Positive integer
N	Index set of non-basic variables	Set with cardinality $n-m$
n	Number of variables	Positive integer
V	Set of variables	Set
x	Decision variables or solution	n-vector
x_B	Basic decision variables	m-vector
x_N	Non-basic decision variables	$n-m$-vector
x^*	Optimal solution	n-vector
y	Objective value	Real number
y^*	Optimal objective value	Real number
y_B	Objective value of basis B	Real number
y_{DFM}	Opt. obj. val. if x can be adapted at every instance of time	Real number
μ_b	Expected value of constraints	n-vector
μ_c	Expected value of costs	n-vector
μ_V	Expected value of variables V	Vector
μ_{x_B}	Expected value of variables	m-vector
μ_y	Expected value of objective value	Real number
$\Phi()$	Standard normal distribution	Distribution function
Σ^b	Covariance of constraints	$m \times m$-matrix
Σ^c	Covariance of costs	$n \times n$-matrix
Σ^V	Covariance of variable set	Matrix
Σ^{x_B}	Covariance of variables	$m \times m$-matrix
$\sigma_y 2$	Variance of objective value	Real number

Table C.1: Symbols used in this thesis.

BIBLIOGRAPHY

[1] K. E. Avrachenkov. Analytic perturbation theory and its applications, 1999.

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

[3] B. Bereanu. The continuity of the optimum in parametric programming and applications to stochastic programming. *Journal of Optimization Theory and Applications*, 18:319–333, 1976.

[4] D. Bertsimas, D. B. Brown, and C. Caramanis. Theory and applications of robust optimization. *SIAM Rev.*, 53(3):464–501, Aug. 2011.

[5] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming*, 98:49–71, 2003. 10.1007/s10107-003-0396-4.

[6] S. P. Bradley, A. C. Hax, and T. L. Magnati. *Applied Mathematical Programming*. Addison-Wesley, 1977.

[7] R. Busschers. Generating alternative solutions within the supply chain optimization model boss: a mathematical programming approach, 2012.

[8] L. Cappellari. Calculation of multivariate normal probabilities by simulation, with applications to maximum simulated likelihood estimation. *Stata Journal*, 6(2):156–189(34), 2006.

[9] C. C. Carøe and R. Schultz. Dual decomposition in stochastic integer programming. *Operations Research Letters*, 24:37–45, 1997.

[10] P. Carvalho, L. Ferreira, F. Lobo, and L. Barruncho. Optimal distribution network expansion planning under uncertainty by evolutionary decision convergence. *International Journal of Electrical Power & amp; Energy Systems*, 20(2):125 – 129, 1998.

[11] F. T. Chan, S. Chung, and K. Choy. Optimization of order fulfillment in distribution network problems. *Journal of Intelligent Manufacturing*, 17(3):307–319, 2006.

[12] G. B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1965.

[13] M. E. Dyer, A. M. Frieze, and C. J. H. McDiarmid. On linear programs with random costs. *Math. Program.*, 35(1):3–16, May 1986.

[14] Y. Fujikoshi, V. V. Ulyanov, and R. Shimizu. L1-norm error bounds for asymptotic expansions of multivariate scale mixtures and their applications to hotelling's generalized t02. *J. Multivar. Anal.*, 96(1):1–19, Sept. 2005.

[15] J. Ge, Y. Xia, and C. Nadungodage. Unn: A neural network for uncertain data classification. In M. Zaki, J. Yu, B. Ravindran, and V. Pudi, editors, *Advances in Knowledge Discovery and Data Mining*, volume 6118 of *Lecture Notes in Computer Science*, pages 449–460. Springer Berlin Heidelberg, 2010.

[16] W. Klein Haneveld and M. vd Vlerk. *Stochastic Programming*. University of Groningen, 2010.

[17] A. J. Kleywegt and A. Shapiro. The sample average approximation method for stochastic discrete optimization. Stochastic Programming E-Print Series. Institut für Mathematik, 1999. published; Society for Industrial and Applied Mathematics - SIAM; Philadelphia, Pa.; SIAM Journal on Optimization; 12; 2.2000.

[18] Z. Li and M. Ierapetritou. Process scheduling under uncertainty: Review and challenges. *Computers and Chemical Engineering*, 32(4–5):715 – 727, 2008.

[19] K. Lieckens and N. Vandaele. Reverse logistics network design with stochastic lead times. *Computers & OR*, pages 395–416, 2007.

[20] D. Lu and W. V. Li. A note on multivariate Gaussian estimates. *Journal of Mathematical Analysis and Applications*, 354:704 – 707, 2009.

[21] C. Luo and G. Rong. A strategy for the integration of production planning and scheduling in refineries under uncertainty. *Chinese Journal of Chemical Engineering*, 17(1):113 – 127, 2009.

[22] F. D. Mele, G. Guillén, A. Espuña, and L. Puigjaner. An agent-based approach for supply chain retrofitting under uncertainty. *Computers and Chemical Engineering*, 31(5–6):722 – 735, 2007. Selected Papers from the 15th European Symposium on Computer Aided Process Engineering held in Barcelona, Spain, May 29-June 1, 2005.

[23] M. Minoux. Robust linear programming with right-hand-side uncertainty, duality and applications. In C. A. Floudas and P. M. Pardalos, editors, *Encyclopedia of Optimization*, pages 3317–3327. Springer, 2009.

[24] K. Mosler and P. Bazovkin. Stochastic linear programming with a distortion risk constraint. Discussion Papers in Statistics and Econometrics 6/11, University of Cologne, Department for Economic and Social Statistics, 2011.

[25] C. Muscas, F. Pilo, G. Pisano, and S. Sulis. Considering the uncertainty on the network parameters in the optimal planning of measurement systems for distribution state estimation. In *Instrumentation and Measurement Technology Conference Proceedings, 2007. IMTC 2007. IEEE*, pages 1 –6, may 2007.

[26] C. Muscas, F. Pilo, G. Pisano, and S. Sulis. Optimal allocation of multichannel measurement devices for distribution state estimation. *Instrumentation and Measurement, IEEE Transactions on*, 58(6):1929 –1937, june 2009.

[27] K. Natarajan, M. Song, and C.-P. Teo. Persistency model and its applications in choice modeling. *Management Science*, 55(3):453–469, 2009.

[28] K. Natarajan, C.-P. Teo, and Z. Zheng. Distribution of the optimal value of a stochastic mixed zero-one linear optimization problem under objective uncertainty, 2011. http://www.optimization-online.org/DB_HTML/2011/06/3069.html.

[29] D. Peidro, J. Mula, R. Poler, and F.-C. Lario. Quantitative models for supply chain planning under uncertainty: a review. *The International Journal of Advanced Manufacturing Technology*, 43:400–420, 2009. 10.1007/s00170-008-1715-y.

[30] M. Pishvaee and S. Torabi. A possibilistic programming approach for closed-loop supply chain network design under uncertainty. *Fuzzy Sets and Systems*, 161(20):2668 – 2683, 2010.

[31] M. S. Pishvaee, M. Rabbani, and S. A. Torabi. A robust optimization approach to closed-loop supply chain network design under uncertainty. *Applied Mathematical Modelling*, 35(2):637 – 649, 2011.

[32] A. Prèkopa. On the probability distribution of the optimum of a random linear program. *SIAM Journal on Control*, 4(1):211–222, 1966.

[33] L. Reichel. Math 60093 monte carlo modeling: Generating multivariate normal.

[34] V. S. Rodrigues, D. Stantchev, A. Potter, M. Naim, and A. Whiteing. On assessing the sensitivity to uncertainty in distribution network design. *International Journal of Physical Distribution & Logistics Management*, 38(5):388 – 411, 2008.

[35] M. I. G. Salema, A. P. Barbosa-Povoa, and A. Q. Novais. An optimization model for the design of a capacitated multi-product reverse logistics network with uncertainty. *European Journal of Operational Research*, 179(3):1063 – 1077, 2007.

[36] T. Santoso, S. Ahmed, M. Goetschalckx, and A. Shapiro. A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research*, 167(1):96 – 115, 2005.

[37] M. J. Schervish. Multivariate normal probabilities with error bound. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 33(1):81 – 94, 1984.

[38] P. Schütz, A. Tomasgard, and S. Ahmed. Supply chain design under uncertainty using sample average approximation and dual decomposition. *European Journal of Operational Research*, 199(2):409 – 419, 2009.

[39] M. Skok, S. Krajcar, and D. Skrlec. Dynamic planning of medium voltage open-loop distribution networks under uncertainty. In *Intelligent Systems Application to Power Systems, 2005. Proceedings of the 13th International Conference on*, page 6 pp., nov. 2005.

[40] J. Steele. *Probability Theory and Combinatorial Optimization*. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics, 1987.

[41] M. Taboga. Lectures on probability and statistics. http://www.statlect.com, 2010.

[42] G. Tsakiris and M. Spoliotis. Fuzzy linear programming for problems of water allocation under uncertainty. *European Water*, 7/8, 2004.

[43] D. Wang, L. Ochoa, and G. Harrison. Expansion planning of distribution networks considering uncertainties. In *Universities Power Engineering Conference (UPEC), 2009 Proceedings of the 44th International*, pages 1 –5, sept. 2009.

[44] Wikipedia. Monte carlo method. *http://en.wikipedia.org/wiki/Monte_Carlo_method*, June 2012.

[45] W. Winston and J. Goldberg. *Operations Research: Applications and Algorithms*. Thomson Brooks/Cole, 2004.