Paper Business Mathematics and Informatics

Stochastic Programming in Health Care Planning

Student name: J.M. de Rue
Student number: 1163493

Supervisor: S. Bhulai

Preface

This BMI paper is the last phase of the study Business Mathematics and Informatics at the VU University in Amsterdam. This paper has the goal to use the knowledge gained from the study to address a problem, formulate this in a clear way, and to provide a solution.

Operations Research is a widely applied mathematical technique. It is used for many different business problems. However, Operations Research has its shortcomings. In this paper, I will give an example of these shortcomings and describe several approaches to solve these shortcomings. I will apply one of these approaches to an example in which the major shortcoming is solved by recourse models.

I would hereby like to thank Sandjai Bhulai. Without his time, help, and catching enthusiasm, this paper would not have been possible. Furthermore, I would like to thank Maarten van der Vlerk for sending me his lecture notes on Stochastic Programming. These lecture notes enabled me to learn a great deal on the subject of Stochastic Programming.

Jaap de Rue
Haarlem, August 2007

Management summary

Linear programming is one of the many specializations of Operations Research and it is one of the most applied mathematical techniques. The many different algorithms of linear programming can be used for the optimization of a wide range of problems. However, there are some drawbacks to using these algorithms. The major drawback is that all coefficients (i.e., the quantity of resources, time, and distance) need to be deterministic and known in advance. However, in many practical situations one can not be certain of the true value of these coefficients. These shortcomings can be solved by stochastic linear programming, which uses random variables to approach the values of these coefficients.

This paper describes several approaches that can be applied when the model contains uncertainty on some of the parameters. In particular, recourse structures are seen as the most important class within Stochastic Programming. This paper therefore concentrates on the application of recourse structures. An example is given of a model which contains uncertainty on some of the parameters. Adding a recourse structure to this model shows that this leads to a model in which the risk is explicitly taken care of.

Operations Research is already one of the mostly applied mathematical techniques. Combining Stochastics and Mathematical Programming increases the applicability enormously and may thus account for an even greater number of application areas.

Table of contents

1 Introduction

Operations research, operational research, or simply OR, is an interdisciplinary science which uses scientific methods like mathematical modelling, statistics, and algorithms to help with decision making in complex real-world problems which are concerned with coordination and execution of the operations within an organization (Wikipedia.org). It is one form of applied mathematics that is being used more and more often.

Linear programming is one of the many methods of Operations Research. It is one form of applied mathematics that is being used more and more often, especially after the introduction of the personal computer. However, linear programming has some shortcomings. One of the biggest shortcomings is that the model builder needs to provide numerical values for each of the coefficients, for instance, the run times of a production plant or the number of different resources. But, in most cases the model builder can not be completely certain about the true values of these coefficients.

Stochastic linear programming is a framework for modelling optimization problems that take into account this uncertainty. Whereas combinatorial optimization problems are formulated with deterministic known parameters, real world problems almost invariably include some unknown parameters. Stochastic linear programming interprets the unknown parameters as realizations of random variables. Hereby, they use the framework for the quantitative analysis of uncertainty that is provided by probability theory.

In this thesis, the most important class of stochastic linear programming will be given and an example will be given where this concept will be used to take into account the uncertainty of the values of some of the parameters of the linear programming model.

2 Operations Research

The field of operations research arose during World War II, during which the team of Patrick Blackett made numerous crucial analyses to aid the war effort. An example of one of these analyses is the number of boats in a convoy. The idea of ships travelling in convoys was introduced by Britain, this way warships could be used to accompany and protect merchant ships. Since these convoys travel at the speed of the slowest member, the question arose whether convoys should be large or small. Since a small convoy will be faster, it will be harder to detect by a U-boat. Large convoys may be slower, but they can deploy more warships against such an attack.

Another example of the work of Blackett's team is given by a report they analysed. This report was carried out by RAF Bomber Command, for which they inspected all bombers returning from bombing raids in Germany. This report contained a note of all the damages on the plane inflicted by Germany's air defence. The RAF concluded from this report that the bombers should be armoured more heavily on the damaged areas. However, they made the remarkable conclusion that the armour should be placed on the areas that were not damaged. Since only the planes that came back from Germany were included in the survey, his team reasoned that the untouched areas were apparently the most vital areas.

After the war, people realized that the planning methods that were developed for the military could also be used in the profit- and non-profit sector. The study after operations research flourished after the war and not long after that the famous simplex-method was invented by George Bernard Dantzig that still is of great practical use nowadays.

Dantzig's original example of finding the best assignment of 70 people to 70 jobs still explains its success. The computing power required to scan all the permutations to select the best assignment is vast and impossible. He observed that it takes only a moment to find the optimum solution using the simplex method, which is effectively observing that a solution exists in the extreme points of the polygon described by the equations formed from the given constraints.

Two other founders are John van Neumann, who developed the theory of duality in the same year, and Leonid Kantorovich, a Russian mathematician who used similar techniques in economics before Dantzig and won the Nobel Prize in 1975 in economics. The linear programming problem was first shown to be solvable in polynomial time by Leonid Khachiyan in 1979, but a larger major theoretical and practical breakthrough in the field came in 1984 when Narendra Karmakar introduced a new interior point method for solving linear programming problems.

The possibilities of applying linear programming only became bigger and bigger after the introduction of the personal computer and nowadays it is one of the most applied mathematical techniques.

2.1 Linear Programming

In mathematics, linear programming involves the optimization of a linear objective function that is subject to (multiple) linear equality and inequality constraints. The goal is to find a feasible point for which the objective function assumes the smallest or largest value. Such points may not exist, but if they do, searching through the vertices of the model is guaranteed to find at least one of them.

The standard (canonical) form of a linear program is the following:

$$\begin{aligned} &\text{Maximize} && c^T x \\ &\text{Subject to} && Ax = b \\ & && x \geq 0, \end{aligned}$$

where x represents the vector of variables, where c and b are vectors of coefficients, and A is a matrix of coefficients. The expression $c^T x$ to be maximized (or minimized) is called the objective function. The equations $Ax = b$ are the constraints which specify a convex polyhedron over which the objective function is to be optimized. Linear programming can be applied to various fields of study. Most extensively it is used in business and economic situations, but can also be utilized for some engineering problems. Some industries that use linear programming models include transportation, energy, telecommunications, and manufacturing. It has proved useful in modelling diverse types of problems in planning, routing, scheduling, assignment, and design.

There are many practical problems in operations research that can be expressed as linear programming problems, of which certain cases are considered important enough to have generated much research on specialized algorithms for their solution. Linear programming is, for example, heavily used in microeconomics and business management where it is used to minimize costs or maximize the income. Here one could think of, among others, inventory management, resource allocation for human and machine resources, and planning. Next to that, many of the central concepts of optimization theory are based on ideas from linear programming. It can thus be said that linear programming is an important field of optimization.

2.2 Techniques

This paragraph will give some of the most frequently used techniques to solve linear programming problems.

2.2.1 Simplex method

The simplex method is a technique that solves a linear programming problem in a finite number of steps, or it determines the insolvability of the problem. The name originates from the fact that a simplex (a convex hull of *n + 1* points) is formed by the equations of the problem.

Given the following linear programming problem:

$$\begin{array}{ll} \text{Maximize} & c^T x \\ \text{Subject to} & Ax = b \\ & x \geq 0, \end{array} \qquad (1)$$

with *n* variables and *m* restrictions and for which holds that *m*<*n*. In the simplex algorithm (an iterative procedure) bases $\{B\}$ are formed with corresponding basis solutions $\{x\}$. A basis *B* is a part of the matrix *A* consisting of *m* linear independent columns of the matrix *A*. The remaining columns of *A* form the matrix *R*. The basis solution *x* for $j = 1,2,....,n$ is then defined by:

$$x_j = \left(B^{-1}b\right)_j, \qquad \text{for } j \in B,$$
$$x_j = 0, \qquad \text{for } j \in R.$$

The reduced costs are defined by:

$$\hat{c} = \left(c_B^T B^{-1} A\right)^T - c \,,$$

and the corresponding criterion value of a solution $x = \left(x_B, x_R\right)$ given a basis B is:

$$z(x) = c_B^T B^{-1} b - \sum_{j \in R} \hat{c}_j x_j \ .$$

The linear programming problem (1) is called the primary problem and the dual problem is defined as follows:

Minimize $\quad b^T y$
Subject to $\quad A^T y \geq c.$

In the primary problem it is taken care of that for every iteration it holds that x is allowed for the primary problem, i.e., $x_j \geq 0 \ \ \forall j$ and a new iteration is started until x is also feasible for the dual problem, i.e., $\hat{c}_j \geq 0 \ \ \forall j$.

If $\exists j \in R : \hat{c}_j < 0$, a non-basis column $a_s \in R$ is chosen with negative reduced costs ($\hat{c}_s < 0$) to be entered into the basis B. A ratio-test has to point out which column a_l has to leave the basis B, with:

$$l = \arg\min_{j \in B} \left\{ \frac{\left(B^{-1} b\right)_j}{\left(B^{-1} a_s\right)_j} : \left(B^{-1} a_s\right)_j > 0 \right\}.$$

This test ensures that for all new basis variables it holds that $x_j^{new} \geq 0$.

In the dual problem it is taken care of that in every iteration x is feasible for the dual problem, i.e., $\hat{c}_j \geq 0 \ \ \forall j$ and a new iteration is started until x is also feasible for the primary problem, i.e., $x_j \geq 0 \ \ \forall j$.

If $\exists j \in B : x_j < 0$, a basis column $a_l \in B$ with a negative value $\left(x_l = \left(B^{-1} b\right)_l < 0\right)$ is chosen to leave the basis. A ratio-test has to point out which column a_s enters the basis B, with:

$$s = \arg\min_{j \in R} \left\{ \frac{\hat{c}_j}{\left(B^{-1}a_j\right)_l} : \left(B^{-1}a_j\right)_l < 0 \right\}.$$

This test ensures that for all new reduced costs it holds that $\hat{c}_j^{new} \geq 0$, where $\hat{c}_j^{new}$ is defined by:

$$\hat{c}_j^{new} = \hat{c}_j - \frac{\left(B^{-1}a_j\right)_l}{\left(B^{-1}a_s\right)_l}\hat{c}_s, \qquad \text{for } j = 1,2,\ldots,n.$$

If a basis solution corresponding to a base of the primary problem is both feasible for the primary problem and for the dual problem, then this solution is optimal.

However, in some cases the matrix *A* can become so large that it is almost impossible to solve because of enormous calculating times. A technique that can be used when matrix *A* becomes too large is column generation. This technique is described in the next sub-paragraph.

2.2.2 Column generation

Column generation can most easily be explained by considering a cutting stock problem. This problem is defined as a company producing iron bars of a given length *L*. Each week the company receives orders of customers, so the company has to produce b_i bars of length $l_i \left(i = 1,\ldots.,m\right)$ which have to be cut of the large bars. The question then is: How to cut these bars in such a way that as less iron bars as possible are used? This problem can be solved by applying the technique of column generation.

This technique is based on taking a part of the total collection of cutting patterns, while the rest of these cutting patterns is (not yet) taken into account. The linear programming problem for the chosen part of the collection is solved and studied for optimality for the linear programming problem for all cutting patterns. If this is not the case, one or more cutting patterns are added to the collection that is used. This iteration is repeated until the linear programming problem for the total collection of cutting patterns is optimal.

To describe this technique in mathematical terms, suppose we have the following linear programming problem:

$$\begin{aligned}&\text{Maximize} && c^T x\\ &\text{Subject to} && Ax = b\\ & && x \geq 0,\end{aligned}$$

with n variables and m restrictions for which holds that $m<n$. Now, let $N_1 \subset \{1,....,n\}; k=1$; and the matrix $D = (A_i : i \in N_k)$ be a part of the matrix A for which holds that $rank(D)=m$, i.e., an optimal solution is possible. The vector c_{N_k} and x_{N_k} are formed in a similar way. Then, the limited linear programming problem is defined by:

$$\begin{aligned}&\text{Maximize} && c_{N_k}^T x_{N_k}\\ &\text{Subject to} && Dx_{N_k} = b\\ & && x_{N_k} \geq 0.\end{aligned}$$

This limited linear programming problem is solved with the simplex method. If an optimal solution is found for this limited linear programming problem, this solution is also optimal for the original problem if all reduced costs are non-negative, which are defined as follows:

$$\hat{c}_j = y^T A_j - c_j, \qquad \text{for } j = 1,....,n.$$

If the solution is not optimal for the original linear programming problem, the matrix D is changed, i.e., N_k is replaced with $N_{k+1}; k = k+1$ and the whole process is repeated.

2.3 *Drawbacks*

It is obvious that linear programming is very popular, since so many practical problems can be modelled as a linear program, or at least an approximation can be made. When a problem is modelled as a linear program, powerful software is available to solve these problems. However, the linear programming approach does have some drawbacks.

One of these drawbacks is that numerical values have to be provided for each of the coefficients. But, in most situations one can not be completely certain about the true values of these coefficients. There are some approaches to take this uncertainty into account when building the model. For instance, one could use ranges or solve the linear programming model for every possible realization of the coefficients.

However, if the optimal solution depends heavily on the value of, some of, these coefficients, this uncertainty will have to be taken into account in a more fundamental way. Stochastic programming is characterized by interpreting the uncertain coefficients as realizations of random variables. Hereby the framework for the quantitative analysis of uncertainty provided by probability theory is used.

3 Stochastic Programming

Stochastic programming is a framework for modelling optimization problems that involve uncertainty (stoprog.org). Many practical problems can be modelled with linear programming, in case these problems include known parameters. But, real-world problems almost invariably include some parameters of which the true values are unknown. Stochastic programming interprets these unknown parameters by taking advantage of the fact that probability distributions governing the data are known or can be estimated.

When a linear programming problem has been completely specified with the exception of some parameters, which are assumed to be random variables with a known joint distribution, two types of models are considered in stochastic programming. These two types of models are the wait-and-see model and the here-and-now model.

In the first case, the model builder is assumed to be able to wait for the realization of the random parameters. In the second case, the model builder will have to decide upon the value of these parameters without knowledge of the realizations. So, in the here-and-now model some parameters are undetermined, which makes the feasibility and/or optimality of the solution useless. Because of this, additional specification is needed to deal with questions of risk and risk aversion.

The following paragraph will describe some approaches to deal with these questions of risk and risk aversion.

3.1 Approaches

Given is the following linear programming problem with random parameters in the constraints:

$$\begin{array}{ll} \text{Minimize} & c^T x \\ & Ax \sim b \\ \text{Subject to} & Tx \sim h \qquad\qquad (2), \\ & x \geq 0. \end{array}$$

where the relational symbol ~ denotes =, $\geq$, or $\leq$.

Then assume that the real value of (T, h) is not known, i.e., it is not known which instance of the model occurs. Furthermore, assume that the uncertainty is expressed by a probability distribution, e.g., so-called scenarios:

$$\Pr\{(T,h) = (T^s, h^s)\} = p_s, \qquad s = 1,2,....,S.$$

In addition to this, assume that the probability distribution is known, e.g., by data, or experts, and that a deterministic linear program is a degenerate case.

By stochastic linear programming it is possible to decide on x here and now, without knowing the real value of (T, h), but only by knowing its probability distribution. This is done by interpreting $Tx \geq h$ as a goal constraint, which is to be specified more precisely.

There are several approaches which can be taken:

- **Fat solution**

  The idea is to replace $Tx \geq h$ by $T^s x \geq h^s$, $s = 1,2,....,S$. This constraint is to be satisfied in all scenarios. The advantage this approach has is that the problem is deterministic again. But on the other hand, it is very conservative and expensive. In addition to this, it results in a lot of constraints, which is the reason this approach is called the fat solution.

- **Expected value**

  The idea is to replace $Tx \geq h$ by $\bar{T} x \geq \bar{h}$, with $\bar{T} = \sum_s p_s T^s$ and $\bar{h} = \sum_s p_s h^s$. The advantage is again that this results in a deterministic linear program. But, the risk is not addressed in this model. The constraint $T^s x \geq h^s$ only holds for some of the scenarios. This could be partly solved by using more conservative values for $\bar{T}$ and $\bar{h}$, or by applying sensitivity analysis. But, this would still lead to a poor model of a decision under uncertainty.

- **Scenario analysis**

  The idea in scenario analysis is that for every scenario (T^s, h^s), $s = 1,2,....,S$ the following linear programming problem is solved:

  Minimize $\qquad c^T x$

$$\text{Subject to} \quad \begin{array}{l} Ax \sim b \\ T^s x \sim h^s \\ x \geq 0 \end{array}.$$

The solutions for the corresponding scenarios are given by x_s. An overall solution is then found by examining the different scenario solutions, i.e., what solutions are there and what are the probabilities p_s of the corresponding scenarios occurring. This approach is an improvement over the expected value approach and the advantage is that each scenario problem is a linear programming problem. The disadvantage, however, is that flexible solutions do not show up.

- **Chance constraint**

  With chance constraints, $Tx \geq h$ is replaced by $\Pr(Tx \geq h) \geq \alpha$ for some prescribed reliability level $\alpha \in \left(1/2, 1\right)$, where α is to be determined by the problem owner. The advantage this approach has is that the risk is explicitly taken care of, i.e., risk $:= \Pr\{Tx \ngeq h\}$. So the maximum accepted risk is $1-\alpha$. The disadvantage this approach has is that a discrete probability distribution leads to a mixed-integer linear programming problem.

The final approach is called recourse actions. Since recourse actions are seen as the most important class of stochastic programming, this concept will be discussed in more detail in the following paragraph.

3.2 Recourse models

Given is the following linear program with random parameters in the constraints:

$$\begin{array}{ll} \text{Minimize} & c^T x \\ & Ax \sim b \\ \text{Subject to} & T(\omega)x \sim h(\omega) \\ & x \geq 0. \end{array} \qquad (3).$$

Here $Ax \sim b$ represents m_1 inequality constraints. The m random inequality constraints are represented by $T(\omega)x \sim h(\omega)$, where $T(\omega)$ is an $m \times n$ matrix, and $h(\omega)$ is an $m \times 1$ vector. Both the matrix and the vector are dependent on a random vector $\omega \in \Re^r$. It is possible that the whole of *T* and *h* is random, but more often than not only a restricted number of matrix and vector elements is random. We assume that the joint distribution of the random vector ω is known.

Now consider the linear programming problem given by (3) and suppose this is a so-called here-and-now problem. This problem needs a decision on *x* before the real value of ω is known, i.e., only the joint distribution of ω on Ω is known. Moreover, in recourse models the random constraints of (3) are reformulated as soft constraints, i.e., violation of the constraints is accepted but not at any price. To describe how violated constraints are dealt with a *second-stage* linear programming model is introduced. This second stage model contains second stage variables $y \in \Re^p$. The name recourse can now be explained by the second stage, as its decisions are made after observations of the value of ω.

A recourse structure is formally specified by a triple (Y, q, W), which is defined as follows:

> $Y = \{y \in \Re^p : y \geq 0\}$, which describes the feasible set of recourse actions *y*,
> *q* is a $1 \times p$ vector of unit recourse costs,
> *W* is an $m \times p$ matrix, the recourse (technology) matrix.

When the recourse structure is applied to the linear programming problem (3), the following decision problem can be defined:

Minimize	$c^T x + \mathrm{E}_\omega\left[\min q^T y\right]$				
	Ax			$\sim b$	first stage constraints
Subject to	$T(\omega)x$	$+$	Wy	$\sim h(\omega),\ \forall \omega \in \Omega$	second stage constraints
	$x \in X$		$y \in Y$		
	$\uparrow$		$\uparrow$		
	first-stage decisions		second-stage decisions		

This representation makes clear how the introduction of the recourse structure is based on a relaxation of the constraints $T(\omega) \sim h(\omega)$.

3.3 Modelling aspects

Given a linear programming problem of which some parameters are unknown, the question is how to transform this problem into a well-specified linear programming problem with a recourse structure. There are three possibilities to insert the recourse structure (Y, q, W) into the linear programming problem.

- The first possibility is not to model recourse actions. Instead, deviations from the goals are penalized. This means that the recourse variables only represent surpluses or shortages with respect to the goals. These recourse variables are then used in the objective function to penalize these surpluses or shortages.

- The second possibility is to introduce recourse variables which represent corrective actions. These corrective actions are to be taken after realizations of ω if the goals are not reached.

- The third possibility is to split the vector of decision variables in the original linear programming problem in two parts. One set has to be determined before obtaining ω, and the remaining set of variables may depend on the value of ω. The constraints are also split into two parts. The first part of the constraints does not include ω, and the second part of the constraints does.

In the next chapter an example will be given of a linear program which is transformed into a well-specified linear programming problem with a recourse structure.

4 The model

In this chapter an example will be worked out in which the theory of the previous chapters will be applied. A linear programming model will be built to schedule operations with a given duration over multiple days and operating rooms in such a way that as much of the available time will be used for these operations. First a simple model will be built, after which the model will be expanded to deal with multiple operating rooms.

Operations almost always take more or less time than anticipated beforehand by the model builder. Therefore, recourse actions will be applied to deal with a random surplus or shortage of the duration of these operations.

For this first model, numerous assumptions have to be made regarding the values of the parameters such as the duration of an operation, the number of hours the operating staff works on one day, and more of such assumptions will be made along the path of building the model.

4.1 Modelling

The problem that has to be modelled is how to plan operations such that the staff and equipment are being scheduled most efficiently. In other words, if the staff and equipment are available for eight hours per day, operations have to be planned such that the staff and equipment are scheduled to be occupied during these eight hours.

First the decision variable is defined:

$x_{ij} =$ starting time of operation *j* on day *i*.

The assumption is thus made that the staff is expected to work eight hours per day. Then assume that there are *n* operations, that are being planned over *m* days, whereby operation *j* has a duration of d_j hours. The decision variable is equal to zero if operation *j* is not scheduled on day *i* and it is equal to or greater than $\alpha > 0$ if operation *j* is scheduled on day *i*. Hereby, one could think of a starting time for the staff of eight o'clock in the morning, which would result in $\alpha = 8$.

First, the assumption is made that only one operating room is available to the hospital. This assumption keeps the model somewhat simpler, but will eventually be dropped later on. This assumption means that only one operation at the time can be carried out. To make sure that an operation *j* that is scheduled for day *i* is not scheduled again on a different day, a dummy-variable is needed. This dummy-variable z_{ij} is a 0-1 variable, which means it will only have the values 1, if operation *j* is scheduled on day *i*, and the value 0 in all other cases. Now, the following constraints can be formulated:

$$x_{ij} \leq Nz_{ij}, \qquad \forall i, j,$$
$$x_{ij} \geq \alpha z_{ij}, \qquad \forall i, j,$$
$$\sum_{i=1}^{m} z_{ij} = 1, \qquad \forall j,$$

whereby *N* has a sufficiently large value, such that it always holds that $x_{ij} \leq N$ $\forall i, j$ and where α represents the starting time of the operating staff.

So, if operation *j* is to be scheduled on day *i*, the variable z_{ij} will be assigned the value of 1, which will result in $x_{ij} \geq 8$ and $x_{ij} \leq N$. If operation *j* is not to be scheduled on day *i*, the variable z_{ij} will be assigned the value of 0, which will result in $x_{ij} \geq 0$ and $x_{ij} \leq 0$.

Next to this, it has to hold that for every pair of operations *j* and *k*, $x_{ij} + d_j \leq x_{ik}$, or $x_{ik} + d_k \leq x_{ij}$ $\forall i, j$ and $k > j$. In other words, if operation *j* is scheduled to start before operation *k*, the starting time of operation *k* has to be greater or equal to the starting time of operation *j* plus the duration of this operation so that operation *k* does not start until operation *j* is finished. In addition to this, this also holds for the case when operation *k* is scheduled to start before operation *j* respectively.

To make this work, another dummy-variable y_{jk} is needed, which again is a 0-1 variable. This dummy-variable has the value 1 if operation *j* is scheduled to start before operation *k*, and has the value 0 in all other cases. This results in the following constraints:

$$x_{ij} + d_j \leq x_{ik} + M(1 - y_{jk}), \qquad \forall i, j \text{ and } k > j,$$
$$x_{ik} + d_k \leq x_{ij} + My_{jk}, \qquad \forall i, j \text{ and } k > j,$$

where *M* has a sufficiently large value so that $x_{ij} + d_j \leq M$ holds $\forall i, j$.

So, if operation *j* is scheduled to start before operation *k*, the variable y_{jk} will be assigned the value of 1, which would result in $x_{ij} + d_j \leq x_{ik}$ and $x_{ik} + d_k \leq x_{ij} + M$. If operation *j* is not scheduled to start before operation *k*, the variable y_{jk} will be assigned the value of 0, which would result in $x_{ik} + d_k \leq x_{ij}$ and $x_{ij} + d_j \leq x_{ik} + M$.

As is assumed before, the operating staff and equipment is available for eight hours per day. So, given the duration of the operations, some additional constraints are needed to make sure no operations are scheduled with a cumulative duration of over eight hours. To take this into account in the model, the previously declared variable z_{ij} will be used. This variable will be assigned the value of 1 when operation *j* is scheduled on day *i* and the value of 0 in all other cases. So, when this variable is multiplied with the duration of the corresponding operations, the total duration of all operations scheduled on day *i* will be known. This leads to the following constraint:

$$\sum_{j=1}^{n} z_{ij} d_j \leq 8, \ \forall i.$$

This constraint will make sure that the total duration of all operations scheduled on day *i* will be eight hours at maximum.

What remains to be done is the objective function. To schedule the operations as efficient as possible, the difference $8 - z_{ij} d_j$ needs to be minimized. This is the difference between the total duration of all operations planned on day *i* and the total available time (eight hours) of the operating staff and equipment. So, to optimize the schedule, the maximum difference has to be found. In other words, the biggest difference needs to be as small as possible. The biggest difference is given by the next expression:

$$\max\left(8 - \sum_{j=1}^{n} z_{1j} d_j, 8 - \sum_{j=1}^{n} z_{2j} d_j, \ldots, 8 - \sum_{j=1}^{n} z_{mj} d_j\right).$$

To minimize this maximum difference, an additional variable *q* is needed, such that:

$$q = \max\left(8 - \sum_{j=1}^{n} z_{1j} d_j, 8 - \sum_{j=1}^{n} z_{2j} d_j, \ldots, 8 - \sum_{j=1}^{n} z_{mj} d_j\right).$$

Now, the linear programming problem becomes:

Minimize q

Subject to

$$
\begin{aligned}
& q \geq 8 - \sum_{j=1}^{n} z_{ij} d_j \text{, for } i = 1, \ldots, m \\
& x_{ij} \leq N z_{ij}, && \forall i, j, \\
& x_{ij} \geq \alpha z_{ij}, && \forall i, j, \\
& \sum_{i=1}^{m} z_{ij} = 1, && \forall j, \\
& x_{ij} + d_j \leq x_{ik} + M(1 - y_{jk}), && \forall i, j \text{ and } k > j, \\
& x_{ik} + d_k \leq x_{ij} + M y_{jk}, && \forall i, j \text{ and } k > j, \\
& \sum_{j=1}^{n} z_{ij} d_j \leq 8, && \forall i, \\
& x_{ij} \geq 0, && \forall i, j, \\
& y_{jk} \in \{0,1\}, && \forall j, k, \\
& z_{ij} \in \{0,1\}, && \forall i, j.
\end{aligned}
$$

Suppose the model needs to be extended in order to be able to schedule operations in multiple operating rooms. Then, the decision variable needs to be adjusted first so it shows in which operating room which operation is scheduled on which day. The decision variable is defined as follows:

x_{ijl} = starting time of operation *j* on day *i* at operating room/location *l*.

Second, additional constraints are needed to prevent one operation being scheduled in more than one operating room. To prevent this from happening, the dummy-variable z_{ij} is used again, but with an additional index. The dummy-variable z_{ijl} will have the value of 1 when operation *j* is scheduled at operating room/location *l* on day *i*, and will have the value of 0 in all other cases. The additional constraints that are needed then become the following:

$$
\begin{aligned}
& x_{ijl} \leq O z_{ijl}, && \forall i, j, l, \\
& x_{ijl} \geq \alpha z_{ijl}, && \forall i, j, l,
\end{aligned}
$$

$$\sum_{i=1}^{m}\sum_{l=1}^{s} z_{ijl} = 1, \quad \forall j,$$

where O has a sufficiently large value so that $x_{ijl} \leq O \;\; \forall i, j, l$ always holds.

The new linear programming problem then becomes:

$$
\begin{array}{lll}
\text{Minimize} & q & \\
\text{Subject to} & q \geq 8 - \sum_{j=1}^{n} z_{ijl} d_j, & \forall i, l, \\
 & x_{ijl} \leq O z_{ijl}, & \forall i, j, l, \\
 & x_{ijl} \geq \alpha z_{ijl}, & \forall i, j, l, \\
 & \sum_{i=1}^{m}\sum_{l=1}^{s} z_{ijl} = 1, & \forall j, \\
 & x_{ijl} + d_j \leq x_{ikl} + M(1 - y_{jkl}), & \forall i, j, l \text{ and } k > j, \\
 & x_{ikl} + d_k \leq x_{ijl} + M y_{jkl}, & \forall i, j, l \text{ and } k > j, \\
 & \sum_{j=1}^{n} z_{ijl} d_j \leq 8, & \forall i, l, \\
 & x_{ijl} \geq 0, & \forall i, j, l, \\
 & y_{jkl} \in \{0,1\}, & \forall j, k, l, \\
 & z_{ijl} \in \{0,1\}, & \forall i, j, l.
\end{array}
$$

4.2 Extending the model

The model as built in the last paragraph will optimize a schedule such that personnel and equipment are planned as efficiently as possible. This model and resulting schedule are heavily dependent on the parameters, among others the duration of the operations. These durations will have to be known upon building the model. The drawback this has is that this is often not the case. Operations often take more, sometimes less, time than anticipated beforehand.

To be able to deal with the surpluses or shortages in the duration of the operations, and thus the random factor in these parameters, the model will be expanded with a recourse structure.

First, the following additional variables are needed:

μ_j^- = planned decrease in duration of operation *j*,

μ_j^+ = planned increase in duration of operation *j*,

ξ_j = random demand of extra time for operation *j*,

c_j^+ = unit cost for the increase in duration of operation *j*,

c_j^- = unit cost for the decrease in duration of operation *j*.

If an operation runs over time it means that personnel will have to work overtime. Obviously, this leads to additional costs. If an operation takes less time than anticipated, personnel is scheduled less efficiently than would have been possible afterwards. This will also lead to additional costs.

Taking into account the aforementioned additional variables, the objective function needs to be adapted. The adapted objective function is as follows:

$$\text{Minimize} \qquad c^+\mu^+ + c^-\mu^- .$$

Next to the objective function, the following constraint also needs to be adapted:

$$\sum_{j=1}^{n} z_{ijl} d_j + \xi_j = 8 - \mu_j^- + \mu_j^+ , \qquad \forall i, l .$$

Notice that this constraint is a second stage constraint, as described in paragraph 3.2. Lastly, the following constraints need to be added to the model:

$$\mu_j^- \le d_j , \qquad \forall j ,$$
$$\mu_j^+, \mu_j^- \ge 0 , \qquad \forall j .$$

These two constraints give an upper – and a lower bound to the increase and decrease in duration of operation *j*.

Suppose that for the operations hold that the random demand of extra time is assumed to be normally distributed. Then for every operation, the mean and standard deviation of the random deviation ξ_j per operation has to be given. In addition to this, the different costs for an increase or decrease in duration need to be specified.

The linear programming problem with the recourse structure then is as follows:

Minimize $c^{+}\mu^{+} + c^{-}\mu^{-}$

Subject to

$$\sum_{j=1}^{n} z_{ijl} d_j + \xi_j = 8 - \mu_j^{-} + \mu_j^{+}, \qquad \forall i, l,$$
$$x_{ijl} \leq O z_{ijl}, \qquad \forall i, j, l,$$
$$x_{ijl} \geq \alpha z_{ijl}, \qquad \forall i, j, l,$$
$$\sum_{i=1}^{m} \sum_{l=1}^{s} z_{ijl} = 1, \qquad \forall j,$$
$$x_{ijl} + d_j \leq x_{ikl} + M(1 - y_{jkl}), \qquad \forall i, j, l \text{ and } k > j,$$
$$x_{ikl} + d_k \leq x_{ijl} + M y_{jkl}, \qquad \forall i, j, l \text{ and } k > j,$$
$$x_{ijl} \geq 0, \qquad \forall i, j, l,$$
$$y_{jk} \in \{0,1\}, \qquad \forall j, k,$$
$$z_{ijl} \in \{0,1\}, \qquad \forall i, j, l,$$
$$\mu_j^{-} \leq d_j, \qquad \forall j,$$
$$\mu_j^{+}, \mu_j^{-} \geq 0, \qquad \forall j.$$

5 Conclusions

Linear programming is one of the many specializations of Operations Research and is one of the most applied mathematical methods. The many different algorithms of linear programming can be used for the optimization of a wide range of problems.

The example that is worked out in this paper showed that there are drawbacks to Linear Programming. If there are parameters of which the values are uncertain, the measure of optimality is questionable. There are several approaches to take this uncertainty into account in building the model. However, it shows that these approaches lead to models in which the risk is not taken into account, or to models that are very restrictive.

We have shown that adding a recourse structure to a well-defined linear programming problem is relatively easy to do. Furthermore, adding a recourse structure leads to a model in which the risk is explicitly taken care of. In addition to this, it leads to a large scale Linear Programming problem. However, it does have one disadvantage. The disadvantage is that the model may become too large to solve.

6 Bibliography

H. Tijms
Operationele analyse, 2004

H. Tijms, A.A.N. Ridder
Mathematische Programmering, 2002

W.K. Klein Haneveld, M.H. van der Vlerk
Stochastic Programming, 2006

Wikipedia
http://www.wikipedia.org

Stoprog
http://stoprog.org

Google
http://www.google.com

Mathworld
http://www.mathworld.com