UNIVERSITY OF SZEGED
Faculty of Science and Informatics
Department of Computer Algorithms and Artificial Intelligence

PhD School in Computer Science

Decision models and soft optimization

Thesis of PhD Dissertation

Nándor Vincze

Supervisor

Dr. József Dombi

Szeged, 2018

Preface

In expected utility theory, the violations of the axioms and the underlying principles, were generated by certain experimental conditions and framing procedures. In expected utility theories the failure of the transitivity axiom have not been ignored. Fishburn introduced the 'non-transitive measurable utility' theory [35]. In this utility theory, the preferences on pairs of lotteries or risky decisions are represented by the positive part of the skew-symmetric bilinear (SSB) functional [36] [37]. Fishburn supposed that the value of the SSB functional is given with the univariate real-valued function. Fishburn gave five lotteries as example, and presented a univariate function with which we obtain a preference cycle. Our purpose in the Chapter 1 is to give a generalization of the cyclicity given by Fishburn, and to give a class of the univariate function for the characterization of the generalized cyclical preferences.
In the Chapter 2 we examine the lexicographic decision method. In the last decades the decision making models have been widely applied in the practice. After the The Neumann-Morgenstern model, the Fishburn SSB utility theory, the ELECTRE, PROMETHEE and the AHP models by Saaty were the most important models developed. This applications generate the requirement for a general representation for this decision making models. Dombi [13] [14] gave a framework for the utility based and outranking decision methods, using a general preference function. Our main goal in the Chapter 2 is to give a numerical representation of the lexicographic decision method, which is applicable into this framework. For the representation we use a general preference function and a threshold function.
There are many mathematical methods in multicriteria decision theory for solving problems in practice. There are lots of decision problems, where we can solve the decision problem, and apply several decision models succesively in parallel. The various decision making models and the general decision frameworks, uses the different parameters and parameter values. It is neccessary to give the exact meaning of these parameters, and the parameters should be easily handled by users. It is important to find the right parameter values in these models. In the lexicographic decision method the criteria are linearly ordered. Our purpose in the Chapter 3 is to give algorithm to learn the importance of the criteria in the lexicographic decision method, and we examine the conditions of the learning.

Friedler et al.[40] [41] introduced a new process network methodology for solving chemical engineering problems in practice. This is a successfully widely adapted method for solving the problems such as the routing and scheduling of evacuees, facing a life-threatening situation, or solving problems of workflow management. The critical path method (CPM) in project management is an algorithmic approach for scheduling a set of activities. The purpose of the Chapter 4 is to give a process network representation of the map the CPM problems. Here we mapping CPM to a process network. In the process network structure we examine the extension of the CPM problem with alternative of activities. Our aim is to provide a mathematical optimization model for solving the extended CPM problem in the process network.
The Critical Path Method(CPM) today is widely used for the time scheduling of projects, in various areas with practical applications. In the projects, in many cases we cannot determine exactly the time parameters. In the CPM method many models have been developed to handle the uncertainty time parameters. The fuzzy theory is the most successful model for solving this problem. The purpose of the Chapter 5 is to give a new concept of fuzzy linear optimization in order to solve the alternative extended CPM problem in a process network.

Acknowledgement

I would like to thank Dr. József Dombi for his guidance, advice and remarks and for support.

Contents

List of Tables

List of Figures

Chapter 1

Universal characterization of non-transitive preferences

1.1 Introduction

The theories of preference comparisons under risk and under uncertainty have been widely adapted over the past thirty five years. During this period there has been a growing awareness that human reasoned judgments often violate the basic assumptions of expected utility. An important task for normative theory is to decide which violations of the von Neumann-Morgenstern axioms are experimental artifacts and which violations constitute fundamental rejections of the axioms by intelligent people. Many generalizations of the expected utility theories have been proposed. Systematic failures of the independence axiom or expectation principle have received special attention, but independence failures and intransitivities have not been ignored. Fishburn in [35] introduced the 'non-transitive measurable utility' in which preferences are represented by the positive part of the skew-symmetric bilinear (SSB) functional on pairs of lotteries or risky decisions. On the other hand, the SSB utility theory preserves the continuity, convexity and monotonicity. Its representation for preference between risky prospects uses an SSB form ϕ on PXP (where P denote the convex set of probability distributions) rather than the bilinear form ϕ on P. This apparently modest generalization of linear utility has far-reaching consequences [36] [37]. In the first place, it accommodates phenomena (such as preference cycles, the preference reversal phenomenon and systematic failures of independence) that violate the axioms of linear theory and severely challenge its claim as a viable normative theory. Second, and despite the possibility of cyclic preferences, it supports a theory of choice by maximally preferred lotteries that guarantees such lotteries for all finitely based situations and ensures their independence from inferior infeasible alternatives. Third, it adapts well to areas where linear utility has been applied. The SSB functional $\phi(x, y)$ measures the preference intensity between alternatives

x, y and $\phi(x, y) > 0$ if and only if $x > y$. Fishburn supposed $\phi(x, y) = h(x-y),\ x \geq y$, where $h(x-y)$, is a univariate real-valued function, such that the properties of the preference can be characterized by the properties of $h(x-y)$. Fishburn showed that with a particular choice of $h(x-y)$, we obtain cyclical preference. The question arises of how we can characterize the cyclicity of non-transitive preferences. Fishburn in [36] also supposed that the elements of P are two-valued. Such elements of P are typical examples of lotteries: to win $\$m$ with probability α and $\$0$ with probability $1-\alpha$, denoted by $[m, \alpha]$. Throughout this paper, X will denote a non-empty set of potential decision outcomes and P will denote a convex set of probability distributions on X. Because the elements of X are the sums which we can win in the game, further let $X = \{0, 1, ..., n\}$ for any positive integer n. A preference cycle in SSB utility theory for $[m_1, p_1], [m_2, p_2], ..., [m_n, p_n]$ lotteries from P is:

$$[m_1, p_1] > [m_2, p_2] > ... > [m_n, p_n] > [m_1, p_1]$$

Fishburn gave an example with five lotteries, and presented a function $h(x-y)$ with which we obtain a preference cycle:

$$[6, 0.9] > [7, 0, 8] > [8, 0, 72] > [9, 0, 66] > [10, 0, 61] > [6, 0.9].$$

In this chapter we give a generalization of this cyclicity. We will define the k-cyclicity of any positive integer k. The preference is k-cyclic on the lotteries if

$$[m, p(m)] > [m+1, p(m+1)] > ... > [m+k, p(m+k)] > [m, p(m)]$$

$$[m, p(m)] > [m+1, p(m+1)] > ... > [m+2k, p(m+2k)] > [m, p(m)]$$

. . .

$$[m, p(m)] > [m+1, p(m+1)] > ... > [m+lk, p(m+lk)] > [m, p(m)]$$

for $lk < n-j$. Our main result is that one class of k-cyclic preference function is given by the theory of finite difference equations, from the solution of the linear homogeneous second-order difference equation

$$g(m+2) - sg(m+1) + g(m) = 0$$

for the special parameter s. In our case the explicit form of $h(x-y)$ is:

$$h(m) = u(m)g(m)$$

where $g(m)$ is a solution of difference equation, positive on $\{1, 2, ..., n\}$. All solutions are for any non-zero real number $g(1) \neq 0$ and for the parameter s, defined by $s = \dfrac{g(2)}{g(1)}$.

$$g(m)=\begin{cases}\frac{g(1)}{\sqrt{s^2-4}}\left(\left(\frac{s}{2}+\frac{\sqrt{s^2-4}}{2})^m-(\frac{s}{2}-\frac{\sqrt{s^2-4}}{2}\right)^m\right) & if \quad \mid s \mid > 2\\ g(1)m & if \quad s=2\\ (-1)^m g(1)m & if \quad s=-2\\ \frac{g(1)}{sin\left(arccos\frac{s}{2}\right)}sin\left(m\cdot \arccos\frac{s}{2}\right) & if \quad \mid 2 \mid < 2\end{cases} \tag{1.1}$$

and for the function $u(m): N \to R_+$ the following inequality system holds:

$$\begin{aligned} v(i) &< v(m)v(m+i) & if \quad & i \neq rk \\ v(rk) &< v(m)v(m+rk) & if \quad & r \in N, r \leq \left[\frac{N}{2k}\right] \end{aligned} \tag{1.2}$$

where $j \leq m, m+i, rk \leq n$ for fixed positive integer k. Here we give convex and concave k-cyclic preference functions. There is no linear preference function, but we can get a k-cyclic preference function which differs from the linear solution (1.2) by as little as we want. Therefore we introduce ε-linearity as the measure of the difference between $h(m)$ and $g(m)$. We show that an ε-linear k-cyclic solution exists. That is neither concave nor convex. Finally, examples are presented of k-cyclic preferences which have a convex, concave and 0.4-linear form. The following results can be found in the article [21].

1.2 Preliminary definitions

Throughout, X will denote a non-empty set of potential decision outcomes, and P will denote a convex set of probability distributions on X. Here, $X = \{1, 2, ..., n\}$ for positive integer n, and each $p \in P$ is a two-valued function from X into $[0, 1]$ that has $p(m) > 0$ for only one $m \in X$, that takes the value $1 - p(m)$ on 0 and for every $m \in X$ there exists only one $p \in P$ which has a positive value on m.

Definition 1.1 *A set P of probability distributions on X is convex if $\lambda p + (1-\lambda)q$, which takes on the value $\lambda p(x) + (1-\lambda)q(x)$ for each $x \in X$, and is in P whenever $p, q \in P$ and $0 \leq \lambda \leq 1$.*

Definition 1.2 *A real-valued function φ on XxX is skew symmetric if*

$$\varphi(a,b) = -\varphi(b,a) \quad for\ all\ a,b \in X$$

Definition 1.3 *Φ is a bilinear functional on PxP if it is real-valued and linear separately in each argument, such that*

$$\phi(\lambda p + (1-\lambda)q, r) = \lambda\phi(p,r) + (1-\lambda)\phi(q,r)$$

$$\phi(p, \lambda q + (1-\lambda)r) = \lambda\phi(p,q) + (1-\lambda)\phi(p,r)$$

for every $p, q, r \in P$ *and* $0 \leq \lambda \leq 1$.

Now consider the risky prospect $p \in P$ vs. outcome $y \in Y$. Since $\phi(x, y)$ measures the person's preference intensity for x over y for each x that has $p(x) > 0$, his expected intensity for p vs. y, written as $\phi(p, y)$, is given by the following:

Definition 1.4 *The preference between* $p \in P$ *and* $y \in X$ *according to Fishburn [36] [38] is*

$$\sum_{x \in X} p(x)\phi(x, y)$$

so that

$$\begin{array}{lll} p > y & if & \phi(p, y) > 0 \\ p < y & if & \phi(p, y) < 0 \\ p \sim y & if & \phi(p, y) = 0 \end{array}$$

Finally, consider lottery p vs. lottery q. Since $\phi(p, y)$ is the person's expected intensity for p vs. y for each y that has $q(y) > 0$, his overall expected intensity for p vs. q, written as $\phi(p, q)$, is:

Definition 1.5 *The preference between* p *and* q *if* $p, q \in P$ *is*

$$\sum_{x \in X} \sum_{y \in X} p(x)p(y)\phi(x, y)$$

We consider the case $\phi(x, y) = h(x - y), \quad x > y$, while $h(x - y)$ is a positive increasing real-valued function Fishburn in [36] [38] called the preference function.

1.3 Example of preference intransitivity

Fishburn (1984a, 1988) gives an example of non-transitivity. When $[x, \alpha]$ represents a lottery that pays x with probability α and nothing otherwise, a number of people exhibit the cyclic pattern:

$$[6, 0.9] > [7, 0, 8] > [8, 0, 72] > [9, 0, 66] > [10, 0, 61] > [6, 0.9]$$

In decision theory, this is known as a 4-cycle; see [56]. Fishburn in [36] shows how this preference cycle can be accounted for by SSB representation even when $\phi(x, y)$ depends only on the difference between x and y. If $h(x - y)$ is given by

$$h(1) = 0.75, h(4) = 4.6, h(6) = 6.2, h(7) = 6.8, h(8) = 7.5, h(9) = 7.8, h(10) = 8$$

then it can represent the preference cycle. From the definition:

$$\phi([6, 0.9], [7, 0.8]) = (0.9) \cdot (0.8)\phi(6, 7) +$$

$$+(0.9)\cdot(1-0.8)\phi(6,0)+(1-0.9)(0.8)\phi(0,7)=$$

$$=-(0.72)h(1)+(0.18)h(6)-(0.8)h(7)=$$

$$=-(0.72)\cdot(0.75)+(0.18)\cdot(6.2)-(0.8)\cdot(6.8)=0.032$$

$$\phi([7,0.8],[8,0.72])=0.112,\phi([8,0.72],[9,0.66])=0.038$$

$$\phi([9,0.66],[10,0.61])=0.046,\phi([10,0.61],[6,0.9])=0.837$$

1.4 Connection between preference and functional inequalities

The question arises of how we can characterize the cyclicity of non-transitive preferences. If $\phi(x,y)=h(x-y)$, then the properties of the preference relation [7] [30] give conditions for the preference function $h(x-y)$. For example, if $h(x-y)$ is the preference function of a transitive preference, then:

$$if \quad h(x-y)>0 \quad and \quad h(y-z)>0 \quad then \quad h(x-z)>0, \qquad x,y,z\in X.$$

Therefore, the properties of the preference can be characterized with the properties of the function $h(x-y)$. Accordingly, we consider the function $h(x-y)$. First, we introduce the function $p(m)$. Since every $p\in P$ has a positive value on one and only one element of X and for every $m\in X$ there exists only one $p\in P$ which has a positive value on X, we can consider these positive values as the range of the univariate positive real-valued function $p(m)$. It is presumed here that $p(m)$ is decreasing. Now we introduce the k-cyclicity as the generalization of the cyclicity of the SSB utility theory.

Definition 1.6 *Let $X\in\{j,j+1,...,n\}$ for positive integers j,n. The preference is then k-cyclic on X, for fixed $k\in N$, if*

$$[m,p(m)]<[m+rk,p(m+rk)]$$

and

$$[m,p(m)]>[m+i,p(m+i)] \quad if \quad i\neq rk$$

for every $m,r,i\in N$ for which $j\leq m, m+rk, m+i\leq n, k<n-j$.

For $k=1$, this obviously gives the transitivity of the preference. Here the cyclicity of preference means the k-cyclicity of this preference for any positive integer k. The above question now arises in another form: Is there a positive, increasing, real-valued function $h(x-y)$ such that the preference is k-cyclic on the interval $[j,n]$?

The condition of k-cyclicity with functional $\phi(x,y)$ is:

$$\phi[m,p(m)],[m+i,p(m+i)]>0 \quad if \quad i\neq rk$$

$$\phi[m, p(m)], [m+i, p(m+i)] < 0 \quad if \quad i = rk$$

$j < i, r, k, m \leq n$.

According to Definition 1.5:

$$p(m)p(m+i)\phi(m, m+i) + p(m)(1-p(m+i))\phi(m, 0)+$$

$$+(1-p(m))p(m+i)\phi(0, m+i) > 0 \quad if \quad i \neq rk$$

$$p(m)p(m+i)\phi(m, m+i) + p(m)(1-p(m+i))\phi(m, 0)+$$

$$+(1-p(m))p(m+i)\phi(0, m+i) < 0 \quad if \quad i = rk$$

Since $p(m)$ is positive, we can divide by $p(m)p(m+i)$:

$$-h(i) + \frac{1-p(m+i)}{p(m+i)}h(m) - \frac{1-p(m)}{p(m)}h(m+i) > 0 \quad if \quad i \neq rk$$

$$-h(i) + \frac{1-p(m+i)}{p(m+i)}h(m) - \frac{1-p(m)}{p(m)}h(m+i) < 0 \quad if \quad i = rk$$

Let us denote $\frac{1-p(m)}{p(m)}$ by $f(m)$:

$$\begin{aligned} f(m+i)h(m) - h(m+i)f(m) > h(i) \quad if \quad i \neq rk \\ f(m+i)h(m) - h(m+i)f(m) < h(i) \quad if \quad i = rk \end{aligned} \tag{1.3}$$

$j \leq m, m+i, rk \leq n$

This is a functional inequality system. It can be seen that the existence of the positive real-valued increasing solutions $h(m)$ and $f(m)$ of (1.2) on the interval $[j, n]$ is equivalent to the existence of k-cyclicity.

1.5 Characterization of non-transitivity with preference functions

Lemma 1.5.1 *We can get a solution $h(m)$ of (1.2) in the form $h(m) = v(m)g(m)$, where g(m) is the solution of the functional equation*

$$F(m+i)g(m) - g(m+i)F(m) = g(i) \tag{1.4}$$

for

$$F(m) = \frac{f(m)}{v(m)} \qquad 1 \leq m, i, m+i \leq n$$

and $g(m)>0$ *is positive on* $m\in\{1,2,...,n\}$ *and* $v(m)$ *is positive and satisfies:*

$$v(i)<v(m)v(m+i) \quad if \quad i\neq rkv(i)>v(m)v(m+i \quad if \quad i=rk \tag{1.5}$$

where $r\in N$, $r\leq\left[\dfrac{n}{2k}\right]$, $j\leq m,m+i,rk\leq n$

Proof:

Let $v(m)$ be as in (1.4), (see Lemma 1.5.2) and let $F(m)=\dfrac{f(m)}{v(m)}$. To solve (1.3), let $u(m)=\dfrac{F(m)}{g(m)}$. The function $u(m)$ exists, because $g(m)>0$ for every $m\in\{1,2,...,n\}$. Substituting $F(m)$ by $u(m)g(m)$ in the form (1.3):

$$g(m)g(m+i)[u(m+i)-u(m)]=g(i)$$

. Since $g(m)>0$ for every $m\in 1,..,n$:

$$u(m+i)-u(m)=\frac{g(i)}{g(m)g(m+i)} \tag{1.6}$$

where $1\leq m,i,m+i\leq n$. Then multiply the right-hand side of (1.5) by $\dfrac{v(i)}{v(m)v(m+i)}$:

$$u(m+i)-u(m)>\frac{g(i)}{g(m)g(m+i)}\frac{v(i)}{v(m)v(m+i)} \quad if \quad i\neq rk$$

$$u(m+i)-u(m)<\frac{g(i)}{g(m)g(m+i)}\frac{v(i)}{v(m)v(m+i)} \quad if \quad i=rk$$

where $r\in N,r\leq\left[\dfrac{n}{2k}\right],j\leq m,m+i,rk\leq n$. Multiply by $v(m)v(m+i)g(m)g(m+i)$:

$$v(m+i)u(m+i)g(m+i)v(m)g(m)-$$

$$-v(m)u(m)g(m)v(m+i)g(m+i)>v(i)g(i) \quad if \quad i\neq rk$$

$$v(m+i)u(m+i)g(m+i)v(m)g(m)-$$

$$-v(m)u(m)g(m)v(m+i)g(m+i)<v(i)g(i) \quad if \quad i=rk$$

where $r\in N,r\leq\left[\dfrac{n}{2k}\right],j\leq m,m+i,rk\leq n$.

Using $F(m)=u(m)g(m), \quad f(m)=v(m)F(m) \quad and \quad h(m)=v(m)g(m)$ we get:

$$f(m+i)h(m)-h(m+i)f(m)>h(i) \quad if \quad i\neq rk$$

$$f(m+i)h(m)-h(m+i)f(m)>h(i) \quad if \quad i=rk$$

where $j \leq m, m+i, rk \leq n$ ■

Lemma 1.5.2 *There exists a function $v(m) : N \longrightarrow R^+$ such that, for every $k, n \in N$, if $2k \leq n$, there exists a $j \in N$ such that (1.4) holds on the interval $[j, n]$.*

Proof: Let us take the following part of the inequality system (1.4):

$$\begin{array}{ll} (a) & v(rk) > v(rk)v(2rk) \\ (b) & v(2rk-h) < v(h)v(2rk) \\ (c) & v(h) < v(2rk-h)v(2rk) \end{array}$$

for any $h \neq rk, r, k, h \in N, 1 \leq 2rk, h \leq n$.
Then, from (a) we get $v(2rk) < 1$, while from (b) and (c), after multiplication of (c) by $v(2rk)$:

$$v(2rk-h) < v(h)v(2rk) < v(2rk-h)v^2(2rk)$$

Since $v(m)$ is positive, $v(2rk) > 1$ is in contradiction with $v(2rk) < 1$. Accordingly, let $l = max\{r \mid 2rk \leq n\}$ and then let $j = lk + 1$. It is easy to see that this is a minimal such value that system (1.4) does not have the above-mentioned contradiction on $[j, n]$, and we have a function $v(m)$ which satisfies the inequality system (1.4) on $[j, n]$. Let $v(k) = ... = v(rk) = d$, $v((l+1)k) = q$, and $v(i) = t$ for all other i, where $d, t, q \in R^+$ and d, t, q are such that $q > t$ and $d > qt$. This $v(m)$ gives the solution of inequality system (1.4). ■
To solve (1.3) in Lemma 1.5.1 , we get $F(m)$ in the form $F(m) = u(m)g(m)$, so

$$u(m+i) - u(m) = \frac{g(i)}{g(m)g(m+i)}$$

for $1 \leq m, i, m+i \leq n$. And

$$u(m+i) - u(m) = (u(m+i) - u(m+i-1)) + ... + (u(m+1) - u(m))$$

then $g(m)$ is a solution of (1.5) if and only if it is a solution of

$$\frac{g(1)}{g(m)g(m+1)} + ... + \frac{g(1)}{g(m+i-1)g(m+i)} = \frac{g(i)}{g(m)g(m+i)} \tag{1.7}$$

for $1 \leq m, i, m+i \leq n$. Hence $g(m)$ is a solution of (1.3) if and only if $g(m)$ is a solution of (1.6).

Lemma 1.5.3 *Let n be a positive integer with $n > 1$ and let the function $g(m) : \{1, 2, ..., n\} \to R^+$ be such that $g(m) \neq 0$ for every $m \in l, ..., n$. If (1.6) holds for $i = 2$ and for every m for which $1 \leq m \leq n-2$, then (1.3) holds for every i, m where $l \leq m, i, m+i \leq n$.*

Proof: We shall prove that from (1.6):

$$\frac{g(1)}{g(m)g(m+1)} + ... + \frac{g(1)}{g(m+i)g(m+i+1)} = \frac{g(i+1)}{g(m)g(m+i+1)} \tag{1.8}$$

Adding $\dfrac{g(1)}{g(m+i)g(m+i+1)}$ to (1.6). So on the left-hand side of (6), we have the left-hand side of (1.7).

Accordingly we must prove that the right-hand sides are equivalent, i.e.

$$\frac{g(i)}{g(m)g(m+i)} + \frac{g(1)}{g(m+i)g(m+i+1)} = \frac{g(i+1)}{g(m)g(m+i+1)}$$

This is equivalent to

$$g(1)g(m) = g(i+1)g(i+m) - g(i)g(i+m+1) \tag{1.9}$$

We suppose that (1.6) holds for $i = 2$ and for $1 \leq m \leq n-2$, i.e.

$$\frac{g(1)}{g(m)g(m+1)} + \frac{g(1)}{g(m+1)g(m+2)} = \frac{g(2)}{g(m)g(m+2)}$$

which is equivalent to

$$g(1)g(m) = g(2)g(m+1) - g(1)g(m+2) \tag{1.10}$$

i.e.

$$g(m+2) = sg(m+1) - g(m) \tag{1.11}$$

for $s := \dfrac{g(2)}{g(1)}$

Adding $sg(2)g(m+2) - sg(2)g(m+2)$ to both sides in (1.9):

$$g(1)g(m) = sg(2)g(m+2) - g(1)g(m+2) - [sg(2)g(m+2) - g(2)g(m+1)]. \tag{1.12}$$

Using (1.10):

$$sg(2)g(m+2) - g(1)g(m+2) = [sg(2) - g(1)]g(m+2) = g(3)g(m+2).$$

and

$$sg(2)g(m+2) - g(2)g(m+1) = g(2)[sg(m+2) - g(m+1)] = g(2)g(m+3).$$

Substituting this into (1.ll), we get

$$g(1)g(m) = g(3)g(m+2) - g(2)g(m+3) \tag{1.13}$$

Thus, every argument on the right-hand side of (1.9) is increased by 1 in (1.12). Equations (1.9) and (1.12) are special cases of the form

$$g(1)g(m) = g(k)g(k+m-1) - g(k-1)g(k+m) \tag{1.14}$$

$k \in \{1, 2, ..., n-1\}$ for $k = 1$ and for $k = 2$.

Adding $sg(k)g(k+m) - sg(k)g(k+m)$ to the right-hand side of (1.13), we get

$$g(1)g(m) = sg(k)g(k+m) - g(k-1)g(k+m) - [sg(k)g(k+m) - g(k)g(k+m-1)]. \quad (1.15)$$

Using (1.10):

$$sg(k)g(k+m) - g(k-1)g(k+m) = [sg(k) - g(k-1)]g(k+m) = g(k+1)g(k+m)$$

and

$$sg(k)g(k+m) - g(k)g(k+m-1) = g(k)g(k+m+1)$$

Substituting this result into (1.14), we get

$$g(1)g(m) = g(k+1)g(k+m) - g(k)g(k+m+1)$$

Here, every argument on the right-hand side is increased by 1 relative to (13). In $i-1$ steps from (1.9) we get (1.8):

$$g(1)g(m) = g(i+1)g(i+m) - g(i)g(i+m-1)$$

This means that if (1.6) holds for $i = 2$ and for $m \in \{1, 2, ..., n-2\}$ then (1.6) holds for every i, m for which $1 \leq m, i, m+i \leq n$. Thus, the solutions of (1.6) and (1.10) are the same functions. ■

Lemma 1.5.4 *Let n be a positive integer with $n > 1$ and let the functions $f(m), g(m) : \{1, 2, ..., n\} \rightarrow R$ be such that $g(m) \neq 0$ for every $m \in \{1, ..., n\}$. Then, $f(m)$ and $g(m)$ are solutions of (1.3) if and only if $g(m)$ has the same form as in (1.2).*

Proof. From Lemma 1.5.3 we get that the solutions of (1.6) are the same as the solutions of (1.10). In another form, (1.10) is

$$g(m+2) - sg(m+1) + g(m) = 0 \quad (1.16)$$

which is a second-order homogeneous linear difference equation. The solutions of (15) with the parameter s are as in (1.1); see [1] [70]. ■

Stability of the solutions. The theory of finite difference equations gives the following theorem:

Theorem 1.1 *(Stoer and Bulirsch, [82]). The stability of the solution $g(m)$ of the homogeneous linear difference equation,*

$$g(n+j) + a_{j-1}g(n+j-1) + ... + a_0 g(n) = 0$$

is equivalent to the condition that the roots of the characteristic equation,

$$Q(z) = z^j + a_{j-1}z^{j-1} + ... + a_0$$

are in the unit circle and there are no multiple roots on the line of the unit circle.

Therefore, if $\mid s \mid> 2$ then the solution is not stable because,

$$\frac{s+\sqrt{s^2-4}}{2} > 1 \qquad if \quad s > 2$$
$$\frac{s+\sqrt{s^2-4}}{2} < -1 \quad if \quad s < -2$$

and thus these roots are not in the unit circle. If $\mid s \mid= 2$, then the root is on the line of the unit circle and is a 2-fold root. In the case $\mid s \mid< 2$, both roots are on the line of the unit circle and are not multiple roots. Therefore, the solution is stable. In our theorem we give one class of the k-cyclic preference functions and we show convex and concave preference functions for each k. No linear preference function exists. However, we can obtain a k-cyclic preference function which differs from the linear solution $g(m)$ of (1.3) as we want. Therefore, we introduce ε-linearity as the measure of the difference between $h(m)$ and $g(m)$ and generally between an arbitrary function $c(m)$ and the linear function $am+b$.

Definition 1.7 *Let ε be any positive real number. The function $c(m) : \{1, 2, ..., n\} \to R$ is ε-linear if there exists a linear function $am + b$ such that*

$$\mid c(m) - (am + b) \mid< \varepsilon$$

holds for every m.

Our main results may be summarized in the following theorem:

Theorem 1.2 *For every $k, n \in N$, where $2k \leq N$, and for every $\varepsilon > 0$, there exists $j \in N$, such that the preference $\succ$ is k-cyclic on the interval $[j, n]$ and there exist k-cyclic preference function $h(m)$ for every k in the form $h(m) = g(m)v(m)$, where $g(m)$ is the solution of (1.3) so that $g(m)$ is positive on $\{1, 2, ..., n\}$ and $v(m)$ is a solution of (1.4) and $h(m)$ may be convex, concave and ε-linear.*

Proof The theorem states that there exist strictly monotone increasing positive solutions of the functional inequality system (1.2):

$$f(m+i)h(m) - h(m+i)f(m) > h(i) \quad if \quad i \neq rk$$
$$f(m+i)h(m) - h(m+i)f(m) > h(i) \quad if \quad i = rk$$

for every k, in the form $h(m) = v(m)g(m)$, where $g(m)$, $v(m)$ as in theorem, and $h(m)$ may be convex, concave and ε-linear. We can get the form of $h(m)$ from Lemma 1.5.1 , Lemma 1.5.3 gives the solution of the second-order homogeneous linear difference equation (1.15). We seek $h(m)$ in concave, convex and ε-linear form.

If $s > 2$, and $g(1) > 0$, then $g(m)$ is positive and monotone increasing. Let now

$$\varepsilon_0 = min\{g(m) - g(m-1) \mid 2 \leq m \leq n\}$$

and let t, q, d be as given in Lemma 1.5.2. It is easy to see that if $s(m)$ is a discrete convex (concave) function, there exists a positive d, for which, if $\mid w(m) - s(m) \mid \leq \varepsilon$ for any function $w(m)$, then $w(m)$ is convex (concave). Let

$$\nu < min\{\varepsilon, d\} \quad and \quad 1 \leq t, q, d \leq 1 + \nu$$

and let j be as given in Lemma 1.5.2. Hence, on the interval $[j, n]$, $h(m) = v(m)g(m)$ for $s > 2$ will be a strictly monotone increasing positive convex function.
For $s = 2$ let

$$b = min\{\varepsilon/g(m) \mid 1 \leq m \leq n\} \quad and \quad \nu < min\{\varepsilon_0, b\}$$

as in Lemma 1.5.2. Then,

$$\mid h(m) - g(m) \mid = \mid v(m)g(m) - g(m) \mid < \mid (1+\nu)g(m) - g(m) \mid = g(m) < \varepsilon$$

so $h(m)$ is the ε-linear solution of (1.2).
If $0 < s < 2$, then the period of the solution depends on the parameter s. Thus the magnitude of the period will be changed for any $n \in N$. Accordingly, we can take a monotonously increasing positive quarter-period of $g(m)$ on $[l, n]$. Let n, t, q, d, j be as in Lemma 1.5.2 and Lemma 1.5.3, then $h(m) = v(m)g(m)$ gives a strictly monotone increasing concave solution of (1.2). ■

1.6 Example of 3-cyclic preference function

We present an example of a 3-cyclic preference function on the interval $[4, 11]$ with 0.4-linear function. Let the elements of a lottery be

$$[4, 0.24], [5, 0.19], [6, 0.16], [7, 0.14], [8, 0.12], [9, 0.11], [10, 0.10], [11, 0.09]$$

Let the parameter $s = 2$ and let the parameters $d = 1.098, q = 1.065, t = 1.03$. For the preference function, we have

$$\begin{array}{llll} h(1) = 1.03 & h(2) = 2.06 & h(3) = 3.29 & h(4) = 4.12 \\ h(5) = 5.15 & h(6) = 6.39 & h(7) = 7.21 & h(8) = 8.24 \\ h(9) = 9.27 & h(10) = 10.30 & h(11) = 11.33 & \end{array}$$

Hence, we have

$$\phi([4, 0.24], [5, 0.19]) =$$

$$= -p(4)p(5)h(1) + p(4)(1 - p(5))h(4) - (1 - p(4))p(5)h(5) =$$

$$= (-0.24) \cdot (0.19) \cdot (1.03) + (0.24) \cdot (0.81) \cdot (4.12) -$$

$$-(0.76) \cdot (0.19) \cdot (5.15) = 0.03$$

which means that $[4, 0.24] > [5, 0.19]$.
In the same way, we have

$$\begin{array}{cc} \phi([4,0.24],[6,0.16]) = 0.13 & \phi([4,0.24],[7,0.14]) = -0.11 \\ \phi([4,0.24],[8,0.12]) = 0.12 & \phi([4,0.24],[9,0.11]) = 0.15 \\ \phi([4,0.24],[10,0.1]) = -0.02 & \phi([4,0.24],[11,0.09]) = 0.21 \end{array}$$

which shows the 3-cyclicity for $[4, 0.24]$. For $[5, 0.19]$ we have

$$\begin{array}{cc} \phi([5,0.19],[6,0.16]) = 0.06 & \phi([5,0.19],[7,0.14]) = 0.06 \\ \phi([5,0.19],[8,0.12]) = -0.11 & \phi([5,0.19],[9,0.11]) = 0.12 \\ \phi([5,0.19],[10,0.1]) = 0.15 & \phi([5,0.19],[11,0.09]) = -0.021 \end{array}$$

which also shows the 3-cyclicity for [5,0.19].
In this way, we find that the rule of 3-cyclicity holds for every lottery on the interval $[4, 11]$, which means that the preference relation is 3-cyclic on this interval. We can see this example in figure 1.1 .

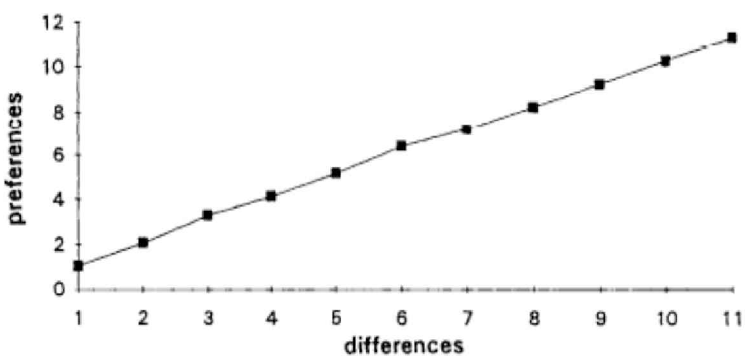


Figure 1.1: The 3-cyclic 0.4-linear preference function for s=2

In figures 1.2 and 1.3 are two 3-cyclic preference functions: in case 1 it is a convex one, with parameter s = 2.14, and in case 2 it is a concave one, with parameter s = 1.985.

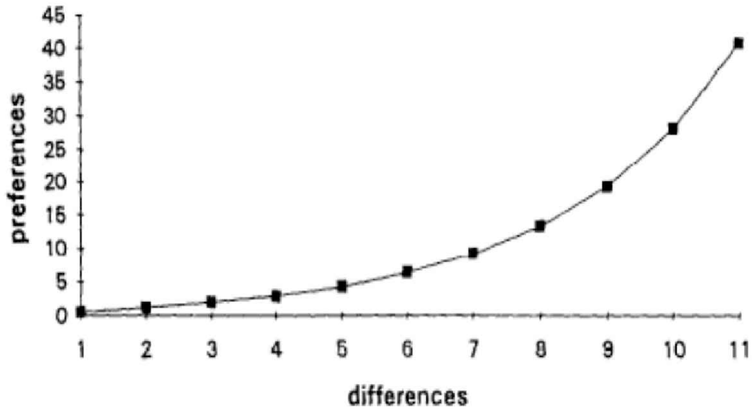


Figure 1.2: The 3-cyclic convex preference function for s = 2.14

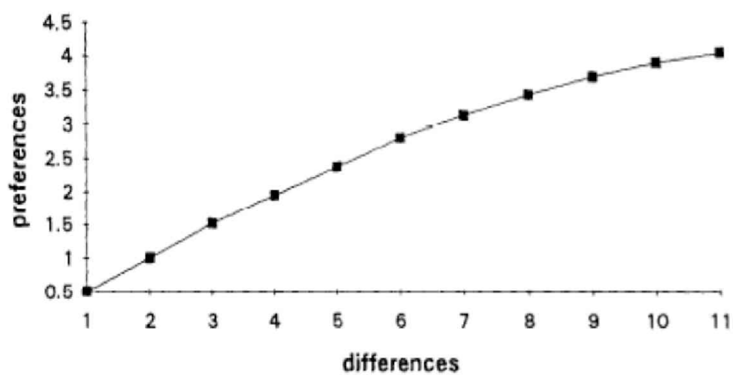


Figure 1.3: The 3-cyclic concave preference function for s = 1.985

Chapter 2

Lexicographic decision function construction

2.1 Introduction

Several methods were developed in decision theory. In the beginning Condorcet, Cramer and Bernoulli models were developed. Afterwards, Neumann and Mogenstern developed their axiomatized model and utility function representations. This had the greatest impact on the advance of the decision theory modeling in beginnings. Other important additive or non-transitive models follow the Neumann-Morgenstern model, as the Fishburn SSB model [38], the outranking methods, the ELECTRE, the PROMETHEE or the AHP model from Saaty,[73] [84]. In practical application the requirement of the general framework appears for this models. We can parametrize the general model, so that the different decision making models we can get from general framework, using the different parameter values. There are lots of decison situations, when we can choose the range of the alternatives, applying several decision model succesively in parallel. In such situations this general framework model seems important. The framework is given by Dombi, see in [14] [13]. In this chapter, our goal is to give a numerical representation of the lexicographic decision model, which is applicable to this framework. This cause the complexity of the representation for the lexicographic decision method. Let $a = (x_1, x_2, ..., x_n)$ and $b = (y_1, y_2, ..., y_n)$ alternatives be given with utility values x_k, y_k and let $w_1, w_2, ..., w_n$ be weigths. Let us be the preference:

$$p(a,b) = \sum_{i=1}^{n} w_i \tau_i(p_i(x_i, y_i))$$

with the preference function

$$p(x,y) = \tfrac{y-x+1}{2}$$

and $\tau_i : [0,1] \to [0,1]$ univariate monotone function. In the first part of his study Dombi[13] proved that in the case of $\tau_i(x) = x$ we get the weighted average of the preferences. On the other hand the decision based on the weighted average is interchangeable with the decision based on the preferences. It follows from this that this model contains the utility type model. Dombi[13] proved the following:

Theorem 2.1 *For the p^{EL} and p^{PR} preference functions of the ELECTRE and the PROMETHE methods there exists univariate $\tau_i^E L$ and $\tau_i^P R$ functions that:*

$$p^{EL}(a,b) = \sum_{i=1}^{n} w_i \tau_i^{EL}(p_i(x_i, y_i))$$

$$p^{PR}(a,b) = \sum_{i=1}^{n} w_i \tau_i^{PR}(p_i(x_i, y_i))$$

and in the case of linear functions $\tau^{EL}(x)$ and $\tau^{PR}(x)$:

$$\tau^{EL}(x) = \begin{cases} 0 & ha \quad x \le p_i \\ \frac{x-p_i}{q_i-p_i} & ha \quad p_i < x < q_i \\ 1 & ha \quad q_i \le x \end{cases}$$

when $0 \le p_i \le q_i \le \frac{1}{2}$, and

$$\tau^{PR}(x) = \begin{cases} 0 & ha \quad x \le p_i \\ \frac{x-p_i}{q_i-p_i} & ha \quad p_i < x < q_i \\ 1 & ha \quad q_i \le x \end{cases}$$

where $\frac{1}{2} \le p_i \le q_i \le 1$.

The question arises whether the lexicoraphic decison method can be inserted into this general methodology? Here we show that in the case of the special choice of the w_i weights and the $\tau(x)$ function, the above mentioned general methodology contains the lexicographic decision method. Because of the non-compensatory property of the lexicographic decison method we can apply the non-continuous $\tau(x)$ function.

$$\tau(\sum_{i=1}^{n} w_i \tau_i(p_i(x_i, y_i)))$$

We construct our method, that it can apply for the processing the on-line data. In the on-line case we work without the knowledge of the the exact values of the x_i, y_i where the difference between them could be anything. Then we not calculate in the method with the knowledge of the minimal difference of the on-line coming data. So we can work with weigths are given at te beginning of the process. In the above-mentioned general model with the modifying of the $\tau(x)$ we can reach that:

$$\tau(x) = \begin{cases} 0 & ha \quad 0 \le x < \frac{1}{2} - \delta \\ \frac{1}{2} & ha \quad \frac{1}{2} - \delta \le x \le \frac{1}{2} + \delta \\ 1 & ha \quad \frac{1}{2} + \delta < x \le 1 \end{cases}$$

In this section we will describe the concept of the lexicographic method and we will use the terminology of P.C. Fishburn, see [31]. The general concept of a finite lexicographic order - following the terminology Fishburn [31] - involves a set $I = \{1, 2, ..., n\}$ and an order relation $\prec_i$ on a nonempty set X_i for each $i \in I$. We let $\sim_i$ denote the symmetric complement of $\prec_i$ so that $x_i \sim_i y_i$ if and only if ($x_i \prec_i y_i$ or $y_i \prec_i x_i$) does not hold. With $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$, y precedes x lexicographically under the natural order $<$ on I and with respect to the $\prec_i$ or $x <^L y$ in short, iff $\{i : i \in I \; and \; (x_i \prec_i y_i \; or \; y_i \prec_i x_i)\}$ is nonempty, and $x_i \prec_i y_i$ for the first (smallest) i in this set. For this reason a lexicographic order $<^L$is also referred to as an order by first difference.
During the lexicographic aggregation of the orders, it is important to consider the properties of the aggregation of the orders [81]. The lexicographic aggregation of the linear orders is linear order and the lexicographic aggregation of the weak orders is of weak order. The lexicographic aggregation of the partial order it is not necessarily partial order, the aggregation may contain cycle [31]. An example of a lexicographic order arises from the alphabetical order of words in a dictionary or lexicon. To show this let $I = \{1, 2, ..., n\}$, let $X_i = A = \{\emptyset, a, b, ..., z\}$ with $\emptyset \prec_i a \prec_i b \prec_i ... \prec_i z$ for each i , take n as large as the longest listed word , and let the English word $\alpha_1\alpha_2...\alpha_m$ with $m \leq n$ correspond to $(\alpha_1, \alpha_2, ..., \alpha_m, \emptyset, ..., \emptyset)$ in A^n. Then $<^L$on the subset of A^n which corresponds to the "legitimate " words orders these words in their natural alphabetical order. For example, "as" precedes "ask" since $(a, s, \emptyset, ..., \emptyset) <^L (a, s, k, \emptyset, ..., \emptyset)$ which means $a \sim_1 a, \; s \sim_2 s, \; \emptyset \prec_3 k$.
In multicriteria decision making the idea of the lexicographic decision consists of a hierarchy or ordered set of attributes or criteria. Decision alternatives are examined initially on the basis of the first or most important criterion, If more than one alternative is "best" or "satisfactory" on this basis, then these are compared using the second most important criterion and so forth. The principle of order by first difference says that one alternative is "better" than another iff the first is "better" than the second on the most important criterion on which they differ.
So let x and y be two alternatives (actions) and $c_1, c_2, ..., c_n$ be different criteria, x_i and y_i are the utilities (evaluations) of x and y. We identify x and y with their evaluation vector $x = (x_1, x_2, ..., x_n)$, $y = (y_1, y_2, ..., y_n)$. Then $\prec_i$ is the order relation according to c_i on the set of alternatives.
We say that $x_i \sim_i y_i$ iff the alternatives x and y are indifferent according to the c_i, and we say that $x <^L y$ iff $x_i \sim_i y_i$ for $i \in \{1, 2, ..., k-1\}$ where $0 \leq k-1 \leq n-1$ and $x_k \prec_k y_k$ in other words the alternative y is preferred to the alternative x , according to the criteria c_k. The lexicographic decision method is a well adaptable method. The lexicographic decision rule we can apply very effectively, for example to evaluate the competition, [75], or for voting systems to avoid the tie. We suppose that the set of alternatives does not contains two lexicographically equal alternatives, i.e. on arbitrary A set of alternatives E denote the equivalence relation by which $a_j E a_k$ satisfies if a_j and a_k are lexicographically equal, then we regard A/E factor set as the set of alternatives . It can arrange data of arbitrary scales, and it is suitable to evaluate a set of considerable alternatives. This method does not require the weight of criteria and in spite of its simlicity always arranges the

alternatives [75]. Some decision procedures have lexicographic decision rules to prevent ties,[84]. Sequential screening procedures illustrate another common application of the lexicographic idea. Candidates or alternatives are first screened under a given criterion (perhaps with the use of a test or an interview) and separated into "rejects" and "others". In terms of $\prec_1$ of the set of candidates, $x \sim y$ whenever both x and y are "rejects" or "others", with $x \prec_1 y$ when x is a "reject" and y is an "other". The "others" are then screened further by the second criterion or test and sorted into two groups. Of course the "rejects" from the first stage may not be tested for the second stage, but this is of no importance from the viewpoint of the lexicographic rule except from the standpoint of efficiency that it may promote. This process may continue through several more stages, perhaps including a ranking of all candidates who survive to the last stage. Another aspect of using the lexicographic decision method is to avoid the intransitivity of preference. If $\prec_i$ is a weak order for every i then $<^L$ is a weak order, if $\prec_i$ is a linear order for every i then $<^L$ is also a linear order, but when $\prec_i$ is a partial order for every i it does not follow that $<^L$ is a partial order, even if $<^L$ includes cycles. If $\prec_i \neq \emptyset$ then the lexicographic aggeregation preserves transitivity, [31], [81]. About a general concept of the preference cycles and its representation, see [21]. In the evaluation of alternatives, according to the c_k criteria the values x_k, and y_k would be numerical values or categories. In the case of categories the lexicographic decision may be characterized by weighted criteria. We will prove that there exists a weighted representation of lexicographic decision method on the real numbers. This yields a universal form and PROMETHEE, ELECTRE and utility are special cases of it, see [13]. It is should be mentioned that the solution of many MCDM problems requires the application of two or three decision methods. For example when the groups of criteria need different aggregation procedures. In our model we can give different decision making methods by changing the parameters. We construct a weighted method to get the decision function of the lexicographic decision method. We choose the weights in such a way, that a range of alternatives by c_k criteria can not be changed by $c_{k+1}, c_{k+2}, ..., c_n$ criteria. Finally we compare the conditions of the lexicographic decision method and the Arrow impossibility theorem. Our motivation is that the above-mentioned non-compensatory property arrange the criteria in terms of their importance and hence is the dictator in this decision model. So dictatorship is an essential precept in this method. Here we suppose that among the alternatives there are no two lexicographically equal. The following results can be found in the articles [22] and [23].

2.2 The construction of the lexicographic decision function.

The lexicographic decision method is a seldom occuring theme in publications. For its numerical representation we could not find a solution. It may follow from the negative results in its logic, for example the lexicographic order of the plane:

Theorem 2.2 *There exist no continuous function $f(x, y)$ such that:*

$$(x, y) <^L (v, z) \text{ iff } f(x, y) < f(v, z).$$

Proof: Let the values x, x_1, x_2, y_1, y_2 be such that $x_1 < x < x_2$ and $y_1 < y_2$. We will suppose that there exists continuous $f(x, y)$ function, for which:

$$(x, y) <^L (v, z) \text{ iff } f(x, y) < f(v, z).$$

Then for the above values it is true, that:

$$(x, y_2) <^L (x_2, y_1) <^L (x_2, y_2) \text{ iff } f(x, y_2) <^L f(x_2, y_1) <^L f(x_2, y_2).$$

Because $f(x, y)$ is continuous, it is continuous at the point (x_2, y_2).
Let ε be an arbitrarily fixed positive value such that

$$\varepsilon < f(x_2, y_2) - f(x_2, y_1).$$

Then there exists a value δ, such that:

$$\text{if } |(x_2, y_2) - (x, y_2)| < \delta \text{ then } f(x_2, y_2) - f(x, y_2) < \varepsilon.$$

But

$$f(x_2, y_2) - f(x, y_2) > f(x_2, y_2) - f(x_2, y_1) > \varepsilon$$

which contradicts the fact that $f(x, y)$ is continuous. ■

2.2.1 The preference and the modifier functions

.

In the introduction we presented the lexicographical decision concept. Now we construct a lexicographical decision function. For the construction we will use a general preference function $p(x, y)$ and a $\tau(x)$ modifier, (or threshold) function, which are the following, according to Dombi [13]:

$$p(x, y) = (y - x + 1)/2$$

$$\tau(x) = \begin{cases} 0 & if \quad 0 \le x < 1/2 \\ 1/2 & if \quad x = 1/2 \\ 1 & if \quad 1/2 < x \le 1 \end{cases}$$

Let $A = \{a_1, a_2, \cdots, a_m\}$ be the set of alternatives. Let $C = \{c_1, c_2, \cdots, c_n\}$ be the set of the criteria, ordered by importance. Let x_{ij} denote the evaluation (utility) of c_j criteria in the case of choosing a_i as an alternative , $0 \le x_{ij} \le 1$. The decision situation can be described by the following decision matrix:

	c_1	c_2	$\ldots$	c_n
a_1	x_{11}	x_{12}	$\ldots$	x_{1n}
a_2	x_{21}	x_{22}	$\ldots$	x_{2n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
a_m	x_{m1}	x_{m2}	$\ldots$	x_{mn}

Properties of the preference function

Let $p(x,y) = P(y-x)$ consider as the function of $y-x$, and let $0 \leq x, y \leq 1$. Then $y-x \in [-1,1]$. We get that:

$$sign(y-x) = \begin{cases} -1 & if \quad 0 \leq p(x,y) < 1/2 \\ 0 & if \quad p(x,y) = 1/2 \\ 1 & if \quad 1/2 < p(x,y) \leq 1 \end{cases}$$

Then

$$P(sign(y-x)) = \begin{cases} 0 & if \quad 0 \leq p(x,y) < 1/2 \\ 1/2 & if \quad p(x,y) = 1/2 \\ 1 & if \quad 1/2 < p(x,y) \leq 1 \end{cases}$$

As decsribed in the introduction we identify the alternatives with its evaluation n-tuples, so we let

$$a_i = (x_{i1}, x_{i2}, ..., x_{in}) \text{ and } a_j = (x_{j1}, x_{j2}, ..., x_{jn}).$$

To order the alternatives a_i,and a_j with respect to criteria c_k, we set $x = x_{ik}$ and $y = x_{jk}$ in the preference function $p(x,y)$.

The composition of the preference and the modifier function.

Definition 2.1 *We can define for every (a_i, a_j) pair the $p^*(a_i, a_j)$ preference n-tuple in the following manner. Let*

$$p^*(a_i, a_j) = (\varepsilon_{ij}^1, \varepsilon_{ij}^2, \ldots, \varepsilon_{ij}^n) \textit{ for } \varepsilon_{ij}^k = \tau(p(x_{ik}, x_{jk})).$$

Then

$$\tau(p(x_{ik}, x_{jk})) = \begin{cases} 0 & if \quad x_{ik} > x_{jk} \\ 1/2 & if \quad x_{ik} = x_{jk} \\ 1 & if \quad x_{ik} < x_{jk} \end{cases}$$

The indicators ε_{ij}^k can be considered as the elements of a pairwise comparison matrix with respect to the c_k criterion.

c_k	a_1	a_2	$\ldots$	a_m
a_1	ε_{11}^k	ε_{12}^k	$\ldots$	ε_{1m}^k
a_2	ε_{21}^k	ε_{22}^k	$\ldots$	ε_{2m}^k
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
a_m	ε_{m1}^k	ε_{m2}^k	$\ldots$	ε_{mm}^k

All the elements in the main diagonal equal 0.5 . As mentioned before, we suppose, that among the alternatives there are no two lexicographically are equal, so for each pair (a_i, a_j) $a_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $a_j = (x_{j1}, x_{j2}, \ldots, x_{jn})$, there exist a k_1 and k_2 such that $x_{ik_1} < x_{jk_1}$ and $x_{jk_2} < x_{ik_2}$.

2.2.2 The lexicographic decision function

The main result of this study is the following theorem:

Theorem 2.3 *Let $A = \{a_1, a_2, \cdots, a_m\}$ be the set of alternatives. Let $C = \{c_1, c_2, \cdots, c_n\}$ be the set of criteria, ordered by importance. Let x_{ij} denote the evaluation (utility) of criterion c_j in the case of choosing a_i as an alternative, $0 \leq x_{ij} \leq 1$. The decision situation can be described by the decision matrix:*

	c_1	c_2	$\ldots$	c_n
a_1	x_{11}	x_{12}	$\ldots$	x_{1n}
a_2	x_{21}	x_{22}	$\ldots$	x_{2n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
a_m	x_{m1}	x_{m2}	$\ldots$	x_{mn}

Let $p(x, y)$ be the preference function and $\tau(x)$ be the modifier, (or threshold) function as defined earlier.
Then there exists weights w_k , $k = 1, 2, ..., n$ such that the real numbers:

$$l_i = \frac{1}{m} \sum_{j=1}^{m} \tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk}))),\ i = 1, 2, ..., m$$

satisfy

$$l_i < l_j \text{ if and only if } a_i >^L a_j.$$

So we can construct the lexicographic decision function with the help of a weighting system. This function is non compensatory. This we give in the following. Next we give the weighting system. Let the weight of c_i criterion be:

$$w_i = 1/2^i + 1/(n2^n)$$

It can be verified, that:

$$\sum_{k=1}^{n} w_k = 1.$$

The lexicographic decision function may be constructed using the following function composition:

$$\tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk}))) = \begin{cases} 0 & if \quad a_i >^L a_j \\ 1 & if \quad a_i <^L a_j \end{cases}$$

Since $\varepsilon_{ij}^k = \tau(p(x_{ik}, x_{jk}))$, we denote

$$\varepsilon_{ij} = \tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk}))).$$

The following matrix gives the pairwise comparison matrix with respect to the weighted system of criteria $(c_1, w_1; c_2, w_2; ...; c_n, w_n)$

(C, w)	a_1	a_2	$\dots$	a_m
a_1	ε_{11}	ε_{12}	$\dots$	ε_{1m}
a_2	ε_{21}	ε_{22}	$\dots$	ε_{2m}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
a_m	ε_{m1}	ε_{m2}	$\dots$	ε_{mm}

All the elements in the main diagonal always equal 0.5 . Normalizing the lexicographic decision function we get real l_i in the interval [0,1].

$$l_i = \frac{1}{m}\sum_{j=1}^{m}\tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk}))),\ i = 1, 2, ..., m$$

so that:

$$l_i < l_j \text{ iff } a_i >^L a_j.$$

This sequence of real numbers is constructed in such a way, that for alternative a_i we aggregate the preferences between a_i and a_j for $j = 1, 2, ..., i-1, i+1, ..., m$.
This is the main idea behind the global preference construction of the PROMETHEE method. To prove the correctness of the construction, first we prove the correctness of the weighting.

Lemma 2.2.1 *Let $\varepsilon_{ij}^k = \tau(p(x_{ik}, x_{jk})$ as defined earlier. Then the following statements are true:*
(1) $\min\limits_{i,j}\sum_{k=1}^{n} w_k \varepsilon_{ij}^k = 1/2 + 1/(n2^n)$ *if* $a_i <^L a_j$ *, and it is minimal if*

$$(\varepsilon_{ij}^1, \varepsilon_{ij}^2, \dots, \varepsilon_{ij}^n) = (1, 0, \dots, 0)\ .$$

(2) $\max\limits_{i,j}\sum_{k=1}^{n} w_k \varepsilon_{ij}^k = 1/2 - 1/(n2^n)$ *if* $a_i >^L a_j$ *, and it is maximal if*

$$(\varepsilon_{ij}^1, \varepsilon_{ij}^2, \dots, \varepsilon_{ij}^n) = (0, 1, \dots, 1)\ .$$

Proof: (1) If $a_i <^L a_j$ and $\varepsilon_{ij}^k = \tau(p(x_{ik}, x_{jk}))$ then a preference n-tuple

$$(\varepsilon_{ij}^1, \varepsilon_{ij}^2, \dots, \varepsilon_{ij}^n) = (1/2, 1/2, ..., 1/2, 1, \varepsilon_{ij}^{t+2}, ..., \varepsilon_{ij}^n) \text{ for } 0 \le t < n$$

has minimal non-zero element, if $\varepsilon_{ij}^{t+2} = \varepsilon_{ij}^{t+3} = ... = \varepsilon_{ij}^n = 0$.
In this case:

$$\sum_{k=1}^{n} w_k \varepsilon_{ij}^k = 1/2 + (t/2 + 1)[1/(n2^n)]$$

It is minimal if $t = 0$. Then $(\varepsilon_{ij}^1, \varepsilon_{ij}^2, \dots, \varepsilon_{ij}^n) = (1, 0, \dots, 0)$ and the minimum is equal to $1/2+1/(n2^n)$.

(2) If $a_i >^L a_j$ then a preference n-tuple

$$(\varepsilon_{ij}^1, \varepsilon_{ij}^2, \ldots, \varepsilon_{ij}^n) = (1/2, 1/2, ..., 1/2, 0, \varepsilon_{ij}^{t+2}, ..., \varepsilon_{ij}^n) \text{ for } 0 \leq k < n$$

has minimal zero element, if $\varepsilon_{ij}^{t+2} = \varepsilon_{ij}^{t+3} = ... = \varepsilon_{ij}^n = 1$.
Then

$$\sum_{k=1}^{n} w_k \varepsilon_{ij}^k = 1/2 - (t/2+1)[1/(n2^n)]$$

This is maximal, if $t = 0$ and the maximum is $1/2 - 1/(n2^n)$. Then

$$(\varepsilon_{ij}^1, \varepsilon_{ij}^2, \ldots, \varepsilon_{ij}^n) = (0, 1, \ldots, 1) \ . \ \blacksquare$$

Proof(of Theorem 2.3):
By Lemma 2.2.1 we get for the weighted sum that:

$$\begin{aligned} 0 \leq \sum_{k=1}^{n} w_k \varepsilon_{ij}^k < 1/2 \quad if \quad a_i >^L a_j \\ 1/2 < \sum_{k=1}^{n} w_k \varepsilon_{ij}^k \leq 1 \quad if \quad a_i <^L a_j \end{aligned}$$

Applying the modifier (or threshold) function $\tau(x)$ for this weighted sum, we obtain:

$$\tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk}))) = \begin{cases} 0 & if \quad a_i >^L a_j \\ 1 & if \quad a_i <^L a_j \end{cases}$$

So $\tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk})))$ gives the lexicographic preference ordering among alternatives. Hence with this construction we get a decision function. Then we get:

$$\sum_{j=1}^{m} \tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk}))) = \left|\{a_j : a_i <^L a_j\}\right| + \tfrac{1}{2}.$$

To transform this number to the [0,1] interval, we get the real values:

$$l_i = (1/m) \sum_{j=1}^{m} \tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk})))$$

for which

$$l_i < l_j \text{ if and only if } a_i >^L a_j \ . \ \blacksquare$$

2.2.3 The lexicographic decision method as the limit of decision methods

Using the above-mentioned $\tau(x)$ threshold function, we can construct the lexicographic decision function. This form is the general form of decision functions (PROMETHEE, ELECTRE and utility). In this formulation the form of the general modifier (threshold) function is see Dombi [13]:

$$\tau_{p_1 p_2}(x) = \begin{cases} 0 & if \quad 0 \leq x < p_1 \\ (x-p_1)/(p_2-p_1) & if \quad p_1 \leq x \leq p_2 \\ 1 & if \quad p_2 < x \leq 1 \end{cases}$$

This function is linear in the interval $[p_1, p_2]$.
Taking the limit of this function we get:

$$\lim_{p_1 \to \frac{1}{2}, p_2 \to \frac{1}{2}} \tau_{p_1 p_2}(x) = \tau(x).$$

So we get the lexicographic decision method as the limit of decision methods. These methods may be compensatory or non compensatory.

2.3 The main properties of the lexicographic decision function

2.3.1 Decisivity

A decision function

$$F:\ E \to \left\{0, \frac{1}{2}, 1\right\}$$

is decisive, if

$$\varepsilon^{ij} \neq (\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}) \Rightarrow F(\varepsilon^{ij}) \neq \frac{1}{2}$$

It is weak decisive, if:

$$\left\{\varepsilon^{ij} : F(\varepsilon^{ij} = \frac{1}{2})\right\} = (\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2})$$

It is strong decisive, if:

$$\left\{\varepsilon^{ij} : F(\varepsilon^{ij} = \frac{1}{2})\right\} = \emptyset$$

2.3.2 Neutrality

A decision function is neutral if:

$$f(1-\varepsilon_1^{ij}, 1-\varepsilon_2^{ij}, \ldots, 1-\varepsilon_n^{ij}) = 1 - f(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij})$$

i.e. if the preference between two alternatives change to its complement then the value of the decision function(i.e. the aggregated preference) change to its complement.

2.3.3 Montonicity (Positive response)

A decision function is monotone if:

$$\varepsilon^{ij} \geq^L \varepsilon^{kl} \quad then \quad F(\varepsilon^{ij}) \geq F(\varepsilon^{kl})$$

2.3.4 Weak Pareto optimality (enforcement of the concordant opinion)

A decision function perform the weak Pareto optimality if every (a_i, a_j) pair of alternatives satisfies

$$\forall k\ \varepsilon_k^{ij} = 1 \Longrightarrow F(\varepsilon_k^{ij}) = 1 \quad or \quad \forall k\ \varepsilon_k^{ij} = 0 \Longrightarrow F(\varepsilon_k^{ij}) = 0$$

2.3.5 Strong Pareto optimality

A decision function perform the strong Pareto optimality if:

$$\forall k:\ \varepsilon_k^{ij} \in \left\{\frac{1}{2}, 1\right\} \quad and \quad \exists l:\ \varepsilon_l^{ij} = 1 \Rightarrow F(\varepsilon^{ij}) = 1$$

$$and \quad if \quad \forall k:\ \varepsilon_k^{ij} = \frac{1}{2} \Rightarrow F(\varepsilon^{ij}) = \frac{1}{2}$$

Lemma 2.3.1 *The lexicographic decision function satisfies the following conditions:*

1.Strong decisivity

2.Neutrality

3.Monotonicity

4.Weak Pareto optimality

5.Strong Pareto optimality

Proof:

1. Decisivity We suppose that if the set of alternatives is Pareto optimal then for every a_i, a_j pair of alternatives there exist an integer k that:

$$x_{ik} \neq x_{jk} \quad so \quad \left\{\varepsilon^{ij} : F(\varepsilon^{ij} = \frac{1}{2})\right\} = \emptyset$$

which gives the strong decisivity property of the lexicographic decision function.

2. Neutrality This property means that for a decision function the preference between two alternatives change to its complement in the evaluation by every criteria, then the preference between two alternatives changes to its complement. Now we will prove that by the rule of the lexicographic decision the lexicographic decision function satisfies the neutrality property. That is for

$$\varepsilon_k^{ij} = \tau(p(x_{ik}, x_{jk}))$$

it is true that

$$\tau(\sum_{k=1}^{n} w_k \varepsilon_k^{ij}) = 1 - \tau(\sum_{k=1}^{n} w_k (1 - \varepsilon_k^{ij}))$$

Because

$$\sum_{k=1}^{n} w_k(1-\varepsilon_k^{ij}) = \sum_{k=1}^{n} w_k - \sum_{k=1}^{n} w_k\varepsilon_k^{ij} = 1 - \sum_{k=1}^{n} w_k\varepsilon_k^{ij}$$

by the definition of $\tau(x)$ it satisfies the functional equation:

$$\tau(\alpha) = 1 - \tau(1-\alpha)$$

so the lexicographic decision function satisfies the neutrality property.

3. Monotonicity (positive response)

Let be the ε^{ij} the n-tuple of the preferences corresponding to the alternatives a_i and a_j, and let ε^{kl} be the n-tuple of preferences corresponding to the alternatives a_k and a_l.

$$And if \quad a_i >^L a_j \Rightarrow F(\varepsilon^{ij}) = 1$$

Then for arbitrary ε^{kl}

$$F(\varepsilon^{ij}) \geq F(\varepsilon^{kl})$$

$$If \quad a_i <^L a_j \Rightarrow F(\varepsilon^{ij}) = 0$$

Then

$$\varepsilon^{ij} = (\underbrace{\frac{1}{2}, ..., \frac{1}{2}}_{k}, 0, ...) \quad for \quad 0 \leq k < n$$

So if $\varepsilon^{ij} \geq^L \varepsilon^{kl}$ then it is true that for ε^{kl} the value of a preference in ε^{ij} is equal to $\frac{1}{2}$ by a criteria, then the value of a preference by same criteria in ε^{kl} is equal to 0, or

$$\varepsilon^{kl} = (\underbrace{\frac{1}{2}, ..., \frac{1}{2}}_{k}, 0, ...) \quad for \quad 0 \leq k < n$$

Then in the both cases $F(\varepsilon^{kl}) = 0$, so the monotonicity property is satisfied.

4. Weak Pareto optimality

From the definition of the modifier function it follows that:

$$\tau(\sum_{k=1}^{n} w_k \cdot 1) = \tau(\sum_{k=1}^{n} w_k) = \tau(1) = 1$$

then the second part of the weak Pareto optimality satisfied for the lexicographic decision function. That is

$$\tau(\sum_{k=1}^{n} w_k \cdot \frac{1}{2}) = \tau(\frac{1}{2} \cdot \sum_{k=1}^{n} w_k) = \tau(\frac{1}{2}) = \frac{1}{2}$$

because we supposed that is there no two lexicographically equal alternative, the weak Pareto optimality is satisfied.

5. Strong Pareto optimality

The condition of the Strong Pareto optimality is satisfied, if

$$\forall k:\ \varepsilon_k^{ij} \in \left\{\frac{1}{2}, 1\right\} \quad and \quad \exists l:\ \varepsilon_l^{ij} = 1 \Rightarrow F(\varepsilon^{ij}) = 1$$

For the second part of the condition is true the same as we mentioned above. That is,

$$\forall k:\ \varepsilon_k^{ij} = \frac{1}{2} \Rightarrow F(\varepsilon^{ij}) = \frac{1}{2}$$

is satisfied, because we supposed that there are no two lexicographically equal alternatives. Hence

$$\forall k:\ \varepsilon_k^{ij} = \frac{1}{2}$$

the lexicographic decision function is strong Pareto optimal.■

2.4 The lexicographic decision method, and the Arrow paradox

As mentioned earlier, the concept of arranging the criteria according to their importance and the lexicographic decision method are dictatorial. Because of this, there may connections between the lexicographic rule and Arrow's impossibility theorem. But the conditions of Arrow's impossibility theorem are applied to the voting situation, and so the lexicographic decision situation should be applied to a voting situation.

Let the evaluation of alternatives with respect to criterion c_i be $x_{1i}, x_{2i}, ..., x_{ni}$. Let their order be $x_{1i}^* < x_{2i}^* < ... < x_{ni}^*$, and set $x_{ki}^* = \frac{k}{n}$, so we simply get an ordering on alternatives by c_i. To transform the voting situation to multicriteria decision situation we map the individuals to criteria. In this section the profile is a weak order on the alternatives based on a criteria (or individual). The social welfare function is a decision function which aggregates the criterion (or individual) ordering. Let R be the set of all possible weak orders on the set of alternatives. We say that an individual is a dictator if its preferences become automatically social prefrences.

The axioms and conditions of the Arrow paradox are the following, see Hwang, Lin [11]:

Axiom I.(The preference relation is strongly complete) For all a_i and a_j either a_i 'is preferred or indifferent to' a_j or a_j'is preferred or indifferent to' a_i

Axiom II. (The preference relation is transitive) For all a_i and a_j and a_k: a_i'is preferred or indifferent to' a_j and a_j 'is preferred or indifferent to' a_k imply a_i 'is preferred or indifferent to' a_k .

Condition 1. *Universal domain*

The social welfare function(decision function) f is defined for all possible profiles of individual(criteria)

Condition 2. *The weak Pareto concept*

If $a_k, a_l \in A$ and $a_k \prec_i a_l$ for $i = 1, 2, ..., n$ then $a_k <^L a_l$.

Condition 3. *Independence from irrelevant alternatives*

$R^{(a_i,a_j)} = F^{(a_i,a_j)}(p^{(a_i,a_j)})$, for every pair $(a_i, a_j) \in AxA$, where $R^{(a_i,a_j)}, F^{(a_i,a_j)}, p^{(a_i,a_j)}$ are the contraction of the social preference ordering, the social welfare function (i.e. the social decision function), and the p profile, to the pair (a_i, a_j).

Condition 4. *Non-dictatorship*

There is no dictator in the society, i.e.there is no individual that whenever he prefers a_i to a_j for any a_i and a_j society does likewise regardless of the preferences of other individuals.

Theorem 2.4 *General possibility theorem(Arrow): If there are at least two individuals, and three alternatives, which the members of the society are free to order in any way,(condition 1.) then every social welfare function satisfying condition 2 and 3 and yielding a social ordering and satisfying axioms I. and II. must be dictatorial.*

It means that if a given social welfare function satisfies conditions 1-4, then a contradiction arises. It can be seen that the lexicographic decision function satisfies axioms I. and II. and conditions 1.-3. We now consider the formulation in which there are preference orders $\prec_i$ on the set of alternatives for each criteria along with holistic order $<^L$ on A, see Fishburn [31],[29], May [67], Plott [74]. Here we shall refer to an $n+1$ tuple $(\prec_1, \prec_2, ..., \prec_n, <^L)$ of weak orders on A as a situation. Then we will consider the possibility that any one of a number of potential situation might arise.

Theorem 2.5 *Let us suppose that A contains at least three alternatives, (A is otherwise unlimited) and every n-tuple $(\prec_1, \prec_2, ..., \prec_n)$ of weak orders on A appears in at least one situation. Then preferences are lexicographic, iff the following hold for all situations $(\prec_1, \prec_2, ..., \prec_n, <^L)$ and $(\prec_1^{'}, \prec_2^{'}, ..., \prec_n^{'}, <^{'L})$ and all a_j,$a_k \in A$: $(a_j \sim_i a_k$ for all $i) \Longrightarrow a_j \sim a_k$; $(a_j \precsim_i a_k$ for all i, & $a_j \prec_i a_k$ for some $i) \Longrightarrow a_j <^L a_k$, and $(a_j \prec_i a_k$ iff $a_j \prec_i^{'} a_k)$&$(a_k \prec_i a_j$ iff $a_k \prec_i^{'} a_j$ for all $i) \Longrightarrow (a_j <^L a_k$ iff $a_k <^L a_j)$&$(a_k <^L a_j$ iff $a_j <^L a_k)$*

Now we compare the axioms and conditions of the lexicographic decision method, and the Arrow paradox. We shall refer to the lexicographic method and the Arrow Paradox in this comparison by letters L and A, respectively.

1. *Preference completness and transitivity*

L: The preference is a weak order, so it is strongly complete and transitive.

A: The preference is strongly complete, and transitive.

2. *Universal domain*

L: Every n-tuple $(\prec_1, \prec_2, ..., \prec_n)$ of weak orders on A appears in at least one situation

A: The social welfare function(decision function) f is defined for all possible profiles of individual (criteria)

3. *The Pareto concepts*

L:*The strong Pareto concept* ($a_j \sim_i a_k$ for all i)$\Longrightarrow a_j \sim a_k$; ($a_j \precsim_i a_k$ for all i, & $a_j \prec_i a_k$ for some i)$\Longrightarrow a_j <^L a_k$,

A:*The weak Pareto concept* If $a_k, a_l \in A$ and $a_k \prec_i a_l$ for $i = 1, 2, ..., n$ then $a_k <^L a_l$.

4. *Independence from irrelevant alternatives*

L: ($a_j \prec_i a_k$ iff $a_j \prec_i^{'} a_k$)&($a_k \prec_i a_j$ iff $a_k \prec_i^{'} a_j$ for all i) $\Longrightarrow$ ($a_j <^L a_k$ iff $a_k <^L a_j$)&($a_k <^L a_j$ iff $a_j <^L a_k$)

A: $R^{(a_i,a_j)} = F^{(a_i,a_j)}(p^{(a_i,a_j)})$, for every pair $(a_i, a_j) \in AxA$, where $R^{(a_i,a_j)}$, $F^{(a_i,a_j)}$ and $p^{(a_i,a_j)}$ are the contraction of the social preference ordering, the social welfare function (i.e. the social decision function), and the p profile to the pair (a_i, a_j).

It seems, that the conditions used in Arrow's impossibility theorem for a 'social welfare function' are formally similar to the conditions of Theorem 2.5, or are the same as the condition of this theorem. As we mentioned previously we can set a multicriteria decision situation to a voting situation. Then $\prec_i$ is interpreted as the preference order for the ith individual or voter. The Arrowian axioms, and the condition of universal domain and independence from irrelevant alternatives are the same as the conditions of theorem of the lexicographic decision. The difference is that while Arrow's theorem uses strong Pareto concept, the theorem of the lexicographic decision method uses weak Pareto concept, and Arrow adds the condition that no individual shall be a 'dictator'. The main result of Arrow's theorem, is that all conditions other than the nondictatorship condition imply that some individual is a dictator.

By deleting specific references to dictators and replacing the weak Pareto concept by the strong Pareto concept, as in Theorem 4., we derive a hierarchy of 'dictators' $\sigma(1), \sigma(2), ..., \sigma(n)$, which verify the existence of lexicographic preferences.

Chapter 3

Learning lexicographic orders

3.1 Introduction

In the area of multicriteria decision making many mathematical models exist to handle problems. It is a paradox now to choose the right model. It may also be a multicriteria problem, which can be solved by some multicriteria method. Nearly all of the methods use parameters given by the users. It is important to find the right values. The parameters should be user friendly, i.e. it is important to give its semantical meaning. The most user friendly way to find the right parameters is the order of some alternatives. From a large amount of alternatives for the user it is easy to select some alternatives which order is evident. This fact is a good tool for checking whether the model - equipped by these parameters - is well based. If not, the user begins to play with the parameters, and during this procedure the user implicitly learns to manipulate the decision model too. The question that naturally arises: is whether there are algorithms that find the right parameters with some order information? For the utility approach the solution is the UTA method developed by Jacquet-Lagreze [53] [51] [48] [49][50] [52] [54]. It is an excellent solution to approximate the additive model. Nowadays we can formulate the above mentioned problem statement as learning the parameter using order information. Here we will examine the lexicographic decision from the point of view of learning. Here the model parameters are the order of criteria.
The lexicographic decision model is one of the simplest. In the mid-70's Fishburn [31] wrote a state-of-the-art survey on the method. Although it is very simple, it is the most commonly used decision model in everyday life. Even if the decision-makers use another model, they translate it (if it is possible) to lexicographic decision, because for the verbal communication only this approach is good (see [71]). Lexicographical decisions appear in different areas of research . Usually in multicriteria optimization models the criteria are ordered by importance and the optimal solution is defined by the lexicographic order of the feasible solutions. The lexicographical ordering also appears in the area of linear programming, a version of the simplex algorithm where the pivoting element is selected

by a lexicographic ordering has been developed for the solution of the problem.
In the lexicographical decision model we have an n element set of criteria and a linear importance order on this set. We say that an alternative lexicographically precedes an another one, if it has a more preferred value in the most important criteria where they are not the same.
From the learning aspects the examination of lexicographical decision making was not as an easy task as it could be supposed from this simple model. Several interesting results and algorithms were discovered.
We will use the following mathematical model. There are n criteria, and they are linearly ordered by importance. The decision alternatives are ordered by the lexicographical ordering. Our goal is to learn the importance order of the criteria by decision samples. We suppose that the samples are given by exchange value (EV) vectors which are evaluated by the exchange value evaluation (EVE) function. An EV vector is an arbitrary n-dimensional vector which contains -1, 0 or $+1$ in the coordinates. The $+1$ in the i-th coordinate means that we improve the decision alternative concerning the i-th criterion, the -1 means that we deteriorate the decision alternative concerning the i-th criterion,and the 0 means that we do not change it concerning the i-th criterion. The EVE function has value $+1$ on the EV vector if the resulted alternative precedes the original in lexicographical order, in the opposite case it has value -1.
We present the above notation on the following example. Suppose there are three criteria: A is the most important, B is the second most important and C is the less important, each criterion can get positive integers as value, and in each case the larger values are preferred. Then an EV vector $+1, -1, 0$ belongs to the situation when we increase the value of criterion A and decrease the value of criterion B. Since A is the most important criterion thus the resulted alternative will be the worst, therefore the EVE function has value -1 on this vector.
In Section 2, we present an algorithm called Sample Evaluation algorithm which determines the importance order of the criteria by the sample if it is possible. In the remaining part we investigate how long sequence of EV vectors can be necessary to determine the order of importance. First in Section 3 we consider the best case, we suppose that we can generate the EV vectors and the EVE function evaluates them, we call this case Oracle model. Later in Section 4 we consider the worst case, we suppose that the evaluated EV vectors are generated by an adversary who has the goal to present as long sequence as possible. We call this model adversarial model. The following results can be found in the article [17].

3.2 Sample Evaluation algorithm

In this section we present an algorithm which considers a sample containing a sequence L of EV vectors evaluated by the EVE function. The algorithm evaluates the sequence and as a result it determines the importance order of the criteria if it is possible, or it decides whether the sample is insufficient (it could be generated by more importance orders) or inconsistent (it cannot be obtained by the lexicographical ordering).

This algorithm works in n phases. In the i-th phase it determines the i-th criterion in the importance order. If the algorithm is not able to determine the i-th criterion in the i-th phase then it concludes that the sample is insufficient or it concludes that the sample is inconsistent. In the i-th phase the algorithm examines the EV vectors which may contain useful information - the EV vectors which obtain the value by the $i-1$ most important criteria are already eliminated - and uses these vectors to exclude candidates for the position of the i-th criterion in the importance order. If the set of the candidates contains one element at the end of the phase then it is the i-th criterion in the importance order. If it is empty then the sample is inconsistent, if it contains more elements then the sample will be insufficient or inconsistent in some later phase. We give the algorithm in details below:

SamEv Algorithm

Initialization

$S_1 := \{1 \ldots, n\}$

Iteration part

for $i = 1$ to n do
  $S := S_i$,
  for every $p_k \in L$ do
    if $i > 1$ and $p_k(l_{i-1}) \neq 0$ **then**
      delete p_k from L,
    else if $EVE(p_k) = 1$ **then**
      delete each j with $p_k(j) = -1$ from S,
    else if $EVE(p_k) = -1$ **then**
      delete each j with $p_k(j) = 1$ from S,
  endfor
  if $|S| = 0$ **then**
    stop, the sample is inconsistent,
  if $|S| \geq 2$ then
    note that the sample is not sufficient,
    delete arbitrary $|S| - 1$ elements from S.
  Let $S_{i+1} := S_i \setminus S$,
  let l_i be the index which is contained in S,
endfor

Output:

One importance order is $l_1, \ldots, l_n$, and if we noted at some phase that the sample is insufficient then other importance orders are also consistent with the sample.

Example: Suppose that there are four criteria and the following sample is given: $EVE(1,0,1,-1) = 1$, $EVE(-1,1,0,1) = 1$, $EVE(1,0,-1,0) = -1$, $EVE(-1,0,1,0) = 1$, $EVE(1,0,0,-1) = 1$. Then in the first phase it is determined that the second criterion is the most important. In the next phase the second EV vector is deleted (its second component is not 0) and it is determined that the third criterion is the second most important. In the next phase only the last EV-vectors remains and it follows that the first criterion is the third most important, and the fourth criterion is the less important.

The algorithm solves the problem as the following statement shows.

Theorem 3.1 *If algorithm SamEv does not stop with inconsistent sample then it results in an importance order which is consistent with the sample. Furthermore it determines correctly if the sample is inconsistent, and also determines if the sample is insufficient.*

Proof: First observe that if a criterion is eliminated in the i-th phase from set S then this criterion cannot be the most important one among the criteria in set S_i. If a criterion is the most important in the set S_i then for each EV vector which contains 0 in the coordinates of the criteria not contained in the set S_i and contain $+1$ or -1 in the coordinate of the criterion considered the EVE function evaluates the vector with the same value as the criterion considered. We exclude a criterion if we find a vector which does not satisfies the above condition. Therefore if at the i-th phase the algorithm stopes with the result that the sample is inconsistent then we obtain that none of the criteria in set S_i can be the most important in this set, and this yields that the sample is indeed inconsistent.
Now suppose that the algorithm does not stop with inconsistent sample. We show that the lexicographical decision based on the importance order obtained by the algorithm gives the same evaluation as the EVE function. Consider an arbitrary EV vector from the sample. Let the i-th criterion be the most important one with nonzero coordinate in the vector. Suppose that this nonzero coordinate is $+1$, we can handle the opposite case in the same way. In this case the lexicographical decision will give the value $+1$ to the vector. On the other hand during algorithm $SamEv$ in the first i phases this vector was not eliminated from set L, therefore we examine it in the i-th phase. If the EVE function evaluates it with value -1 then the algorithm excludes the i-th criterion from set S, and this is a contradiction. Therefore it also evaluates the vector with value $+1$ and this proves the statement.
Now we prove that if the algorithm determines an insufficient sample, then there are more importance orders which are consistent with the EVE function. Consider the phase where we noted that the sample is insufficient. Denote by i the criterion which was chosen by the algorithm from the set S. Suppose that a modified algorithm chooses an another criterion from this set S and it is continued in the same way as $SamEv$. We state that this modified algorithm does not determine inconsistent sample, and thus it produces an another importance order which is consistent with the EVE function. We prove this by contradiction. Suppose that at the j-th phase the modified algorithm eliminates

every elements from set S. Let x be the criterion from the modified S_j set which is the most important in the importance order given by the original algorithm. This means that in the phase where x was chosen by the original algorithm all of the input vectors were investigated which are investigated in the j-th phase of the modified algorithm. On the other hand in the original algorithm we did not exclude x from the set S of candidates and this yields the contradiction.

3.3 Oracle model

In this section we examine how long sequence of EV vectors can be necessary to determine the importance order in the best case. We use the following model, which we call the Oracle model. We suppose that we can use the EVE function as an oracle, i.e. we can ask to evaluate a sequence of EV vectors generated by us. We want to find a short sequence of EV vectors which determines the importance order of the criteria. We use the following algorithm to generate the sequence. In the i-th phase we determine the i-th most important criterion by performing a binary search on the set S of the possible candidates. During a phase in each step we half the set C of candidates and we determine which half may contain the desired criterion. At the beginning of the algorithm S is the set of the criteria. We use the following notation. For any two sets C_1, C_2 of criteria the vector $EV(C_1, C_2)$ denotes the EV vector which is $+1$ in the coordinates contained in C_1, -1 in the coordinates contained in C_2, and 0 in the other coordinates.

EV sequence generating algorithm

for $i := 1$ **to** n
    $C := S$
    while $|C| > 1$ do
        let C_1 be the set of first $\lfloor |C|/2 \rfloor$ elements of C
        let C_2 be the set of the other elements of C
        let $V = EV(C_1, C_2)$
        if $EVE(V) = +1$ **then** $C := C_1$
        else $C := C_2$
    endwhile
    C contains the i-th criterion in the importance order
    delete the element of C from S
endfor

Example: Suppose that there are four criteria. Then we generate the first EV vector which is $(1, 1, -1, -1)$. Suppose that it is evaluated as 1. Then we generate $(1, -1, 0, 0)$. Suppose that it is evaluated as -1. Then we conclude that the second is the most important criteria, and we generate $(1, 0, -1, -1)$. Suppose that it is evaluated as -1. Then we generate $(0, 0, 1, -1)$. Suppose that it is evaluated as -1. Then we conclude that the forth is the second most important criteria, and we generate $(1, 0, -1, 0)$. Suppose that it is evaluated as 1, then we conclude that the first is the third most important criterion, and the third is the less important.

It can be seen easily that in the i-th phase we indeed obtain the i-th most important criterion, we exclude every other candidate with the generated EV vectors. Consider now the length of the sequence of the EV vectors. Each phase of the algorithm can be characterized with a binary tree. We can represent each vertex of the tree with a subset of S. The root of the tree is represented with the actual S and for every ν point we divide the C_ν subset belonging to the point into two subsets for which the number of elements are $\lceil C_\nu/2 \rceil$ and $\lfloor C_\nu/2 \rfloor$. These are the children of the C_ν point in the graph. When $|C_\nu| = 1$, then it contains the i-th most important criterion. Using this representation it is clear that the number of the used EV vectors is at most $\lceil \log_2 |S| \rceil$. Therefore the total number of the used EV vectors is at most

$$\lceil \log_2 n \rceil + \lceil \log_2 (n-1) \rceil + ... + \lceil \log_2 2 \rceil = n \lceil \log_2 n \rceil - 2^{\lceil \log_2 n \rceil} + 1.$$

Remark: We can reduce this problem to the classical problem of sorting a set. If we consider an EV vector which has -1 in the i-th component and $+1$ in the j-th component, then this vector gives the importance order of the i-th and j-th criteria. Therefore any of the well-known sorting algorithms (heapsort, quicksort) which are based on comparisons can be transformed to an EV vector generating algorithm, where the number of the used EV vectors is the number of the necessary comparisons of the algorithm. On the other hand these algorithms require at least $\Omega(n \log n)$ comparisons in the worst case (see Knuth 1973) so this approach cannot improve asymptotically our above bound.

3.4 Adversarial models

In contrast with the oracle model in this section we investigate the worst case situation. Here we suppose that the list of the EV vectors is generated by an adversary who has the goal to present as long list as possible. Such kind of worst-case analysis based on adversarial sequences are used in different areas of the theoretical computer science, one can find several applications in the area of the competitive analysis of on-line algorithms or in the area of the analysis of queuing strategies (see [4] and [5] for details).

We can see easily that the adversary can use exponential length sequences, as the following sequence shows. The adversary can start the sequence with the the EV vectors started by 0, the number of such possible nonzero EV vectors is $3^{n-1} - 1$. Using the evaluation of these EV vectors we cannot conclude anything about the importance position of the criterion belonging to the first coordinate. In the problems where the adversary is too strong it is an often used idea to decrease the strength of the adversary by different restrictions, one can find several examples for such restricted adversaries in the area of on-line algorithms in the paper Fiat and Woeginger [26]. In this problem there are some natural ways to restrict the adversary, in the remaining part of the section we investigate these restrictions. We present two cases where even the restricted adversaries can generate exponential length EV vector sequences. We close the section with a restriction (strict distance restriction) where we can give $O(n^2)$ bound on the possible length of the EV vector sequences.

3.4.1 Exclusion of the redundant EV vectors

If we consider all of the possible EV vectors then many redundant information is given. The first restriction which is examined is that we forbid the adversary to generate some types of redundant EV vectors. We use the following rules to decrease redundancy

- If the EVE function takes the value $+1$ on an EV vector, then we know that it evaluates the reverse of the vector (we change each $+1$ to -1 and each -1 to $+1$) to -1. Therefore it is a redundant information to consider both a vector and its reverse. So we suppose that the adversary is allowed only to generate vectors which are evaluated by the EVE function to $+1$.
- Since each nonzero EV vector which does not contain -1 is evaluated to $+1$, we may suppose that the adversary generates only such EV vectors which contain -1.
- If we obtain a vector which is evaluated to $+1$, then any vector which is larger than it (at least as large in every component as the vector considered) is evaluated to $+1$, so these vectors contain only redundant information. Therefore we suppose that the adversary is allowed to generate only such EV vectors which are not larger than any of the vectors already generated.

Unfortunately these restrictions are not enough to force the adversary to use polynomial sequence. It can use the following sequence. Consider all of the EV vectors which are started by 0 in the first and -1 in the second component and contain $\lceil \frac{n-2}{2} \rceil$ times $+1$ and $\lfloor \frac{n-2}{2} \rfloor$ times 0 in the remaining components. Suppose they are all evaluated to $+1$. It follows immediately by the definition that this list satisfies the assumptions given above. Furthermore we cannot determine the place of the first criterion in the importance order, therefore we need a longer list to determine the importance order. On the other hand this sequence contains $\binom{n}{\lfloor \frac{n-2}{2} \rfloor}$ vectors which is an exponential length in n.

3.4.2 Weakly distance restricted adversary

In this part we examine the distance restricted adversaries. We use a metric on the set of the EV vectors. We consider the generalized version of the Hamming distance used in the area of error detecting codes (Hamming, [46]). The distance of two EV vectors is the number of the different components. In this part we suppose that in each step the adversary is only allowed to use such EV vector which has distance 1 from the previous vector. Such sequences are called weakly 1-distance restricted sequences.

We show that the adversary can present an exponential length 1-distance restricted EV sequence which is not enough to determine the importance order of the criteria. First observe that we can list all of the EV vectors in a weakly 1-distance restricted sequence. Let S_i be the weakly 1-distance restricted sequence of the i dimensional EV vectors. We can define these sequences recursively. $S_1 = (1, 0, -1)$. We can obtain S_{i+1} in the order $((1, S_i), (0, (S_i)^{-1}), (-1, S_i))$ where $(S_i)^{-1}$ is the opposite order of S_i, and (c, S_i) is the sequence which contains the sequence of EV vectors which are started by c and in the remaining components have the sequence S_i.
Now let us consider the sequence where the vectors are started by 0 and followed by S_{n-1}. We cannot determine the importance order, because of the first criterion. Furthermore, this is a weakly 1-distance restricted adversarial sequence and it has exponential length.

3.4.3 Strongly distance restricted adversary

In this part we consider strongly distance restricted sequences. A sequence is called strongly k-distance restricted if none of the vectors has larger distance from each other than k. It seems that this is a very strong restriction, but as the following statement show there exist strongly distance restricted sequences which are enough to determine the importance order.

Theorem 3.2 *If $n > 3$ then there is not such strongly 1-distance restricted sequence which contains enough information to determine the importance order.*
If $n \geq 6$ then there is not such strongly 2-distance restricted sequence which contains enough information to determine the importance order.
If $n \geq 8$ then there is not such strongly 3-distance restricted sequence which contains enough information to determine the importance order.
For arbitrary n there exist strongly 4-distance restricted sequences which contain enough information to determine the importance order.

Proof: If we consider a strongly 1-distance restricted sequence, then we have a vector and a second one which differs only in one component (denote this component by i) from it. If we choose a third vector which has different value in some other component than the first two vectors, then it also differs from one of them in the i-th component, which is a contradiction. Therefore the strongly 1-distance restricted sequences contain such vectors which are the same in $n-1$ component. If $n - 1 > 2$ then it is impossible to determine the importance order of these criteria, thus the first statement of the theorem follows.
Now suppose that $n \geq 6$ and we have a strongly 2-distance restricted sequence which can be used to determine the importance order. Consider the 4 most important criteria. We can suppose that these criteria belong to the first 4 coordinates since changing the order of the coordinates keeps the distance restricted property. Since we are able to determine the importance order of the other criteria as well, the sequence must contain a vector which is started by $0, 0, 0, 0$. On the other hand we can

distinguish in the importance order the first and second criteria therefore the sequence most contain a vector which is started by $1, -1$ or by $-1, 1$. We can suppose that it is started with $1, -1$ the other case is completely similar. This vector is continued with $0, 0$ because it is at most 2 distance away from the vector started by $0, 0, 0, 0$. Moreover we are able to determine the importance relation of the third and forth criteria, so the sequence contains a vector which is started by $0, 0$ and continued with 1 and -1 in some order. But this sequence has distance 4 from the vector which is started by $1, -1, 0, 0$ and this leads to a contradiction which proves the second statement of the theorem.
Now suppose that $n \geq 8$ and we have a strongly 3-distance restricted sequence which can be used to determine the importance order. We can obtain a contradiction in a similar way as in the previous case. Consider the 6 most important criteria. We can suppose that these criteria belong to the first 6 coordinates since changing the order of the coordinates keeps the distance restricted property. Since we are able to determine the importance order of the other criteria as well, the sequence must contain a vector which is started by $0, 0, 0, 0, 0, 0$. On the other hand we can distinguish in the importance order the first and second criteria therefore the sequence most contain a vector which is started by $1, -1$ or by $-1, 1$. We can suppose that it is started with $1, -1$ the other case is completely similar. The other components of this vector contains at least three 0 because it is at most 3 distance away from the vector started by $0, 0, 0, 0, 0, 0$. Among these three 0 there must be two which belong to neighboring coordinates. The only possibility to distinguish the importance of these coordinates is a vector which contain $0, 0$ in the first two coordinates and $1, -1$ or $-1, 1$ in the coordinates considered. On the other hand this vector is 4 distance away from the vector which is started by $1, -1$ and contains $0, 0$ in the coordinates considered, and this leads to a contradiction which proves the third statement of the theorem.
To see that we can define strongly 4-distance restricted sequences which determine the importance order, recall that any EV vectors which contain 1 in the i-th coordinate and -1 in the j-th coordinate gives the importance relation of the i-th and j-th criteria. On the other hand any set of such EV vectors is strongly 4-distance restricted, so we can use any sorting algorithm which is based on comparisons.

In the rest of this section we investigate strongly 4-distance restricted sequences. Since the number of vectors which are at most 4-distance away from a fixed vector is $O(n^4)$, thus it is straightforward from the definition that the number of the vectors in a strongly 4-distance sequence is at most $O(n^4)$. On the other hand we can use better bound as the following statement shows.

Theorem 3.3 *The number of the EV vectors in a strongly 4-distance restricted sequence is at most $O(n^2)$.*

Proof: Consider a strongly 4-distance sequence. First suppose that there are two vectors a and b which have distance 4 from each other. We can suppose that they differ in the first 4 coordinates. Denote by A the set of vectors which differ in at most 2 coordinates from the last $n-4$ coordinates

of a. Denote by B the set of vectors which differ in at most 2 coordinates from the last $n-4$ coordinates of b. Observe that every vector from the sequence is in A or B otherwise it would have larger distance than 4 from a or b. But the number of the possible elements in A and also in B is $O(n^2)$, and thus the theorem in this case follows.
Suppose that there are not two vectors which have distance 4 from each other. Then the sequence is strongly 3-distance restricted. Then suppose that there are two vectors a and b which have distance 3 from each other. We can suppose that they differ in the first 3 coordinates. Denote by A the set of vectors which differ in at most 2 coordinates from the last $n-3$ coordinates of a. Denote by B the set of vectors which differ in at most 2 coordinates from the last $n-3$ coordinates of b. Observe that every vector from the sequence is in A or B otherwise it would have larger distance than 3 from a or b. But the number of the possible elements in A and also in B is $O(n^2)$, and thus the theorem in this case follows.
Suppose that there are not two vectors which have distance 4 or 3 from each other. Then the sequence is strongly 2-distance restricted. On the other hand the number of EV vectors which are at most distance 2 from a fixed vectors is $O(n^2)$ and this yields the statement of the theorem in this case.
Using the above theorem we receive the following result about the strongly distance restricted adversary.

Corollary 3.1 *The size of the longest strongly 4-distance restricted sequence of EV vectors which is necessary to determine the importance order of the criteria is at most $O(n^2)$.*

Chapter 4

Process network solution of extended CPM problems with alternatives

4.1 Introduction

The critical path method (CPM) is an algorithmic approach of scheduling a set of activities. CPM is widely used in the field of constructions to software development for projects. Modeling techniques date back in the 1950s. The main criteria, in order to use the CPM technique, are the following. First, duration times of the activities have to be known together with the dependencies among the activities. Based on this information the activity network is developed. With the help of the list of activities together with their duration and dependencies on each other as well as on the logical end points, CPM calculates the longest path of the planned activities together with the earliest and latest times that each activity can start or finish without lengthening the project. In this regard, the critical path is the sequence of the activities which add up to the longest overall duration. It sholud be added that there are activity-on-node and activity-on-arc approaches of the CPM from which representations the later is considered for the present work. For further information, consider the problem definition of Chanas and Zielinski, [9]. Friedler et al. [41] [40] introduced a process network methodology for chemical engineering problems. Based on rigorous mathematical foundations the approach relies both on graph theory as well as combinatorial techniques focusing first on the structure generation of the problem considered. Besides the directed bipartite process network an underlying axiom system is used to derive theorems to generate the potentially feasible structures as well as the so called maximal structure which includes all feasible solutions structures. When the algorithmically and mathematically proven structure generation ends, a mathematical programming model is generated and solved with similar mathematical rigor. One of the main advantages of the methodology developed here is that alternative solutions can also be interpreted for the problems considered. Later on this methodology was used in other areas, like workflow modelling [85] [87]; separation network synthesis problems with multiple feed streams and sharp separators (Kovács et al. [59] [60]); generating and analyzing structural alternatives for supply scenarios [2] [55] [57], determining the thermodynamically dominant pathways in a metabolic network [90], and identifying feasible pathways of the reaction catalyzed by a catalyst with multiactive sites [25]. Process networks were successfully also adopted for solving the routing and scheduling of evacuees, facing a life-threatening situation [44]. The following results can be found in the article [88].

4.2 Mapping of CPM to a process network

In order to solve CPM problems with the help of process network methodology, the two terminologies have to be mapped. First, the basic elements are considered as described in Table 4.1. Both the CPM and the process network methodology may have attributes corresponding to their various objects, depending on the application field. For example, in the CPM an activity has a given duration and may have various resources with various costs; while on the other hand in a process network an operating unit may also have a given operation time and various costs.

CPM	**Process network**
Event (node). Activity (arc). Logical connection between the activities (dependencies between the activities). CPM graph: scheduling problem	Material (node type 1): raw material, intermediate and product. Operating unit (node type 2). Material flow (arcs). Process network: network of the operating units producing the products from the raw materials.

Table 4.1: Basic terminology of the CPM and of the process networks

It worth mentioning that in the CPM networks the task is performed by the resources, while in the process networks process is performed by the operating units. Therefore these are mapped to each other.

A given CPM graph may be mapped into a process network as follows. There is a given start of the CPM problem. In the process network it is represented by a raw material Start . The activity between event 1 and event 2 of the CPM graph is represented in the process network as operating unit O_1_2. The input material of the operating unit O_1_2 in this case is the Start. Activity between event 2 and event 3 of the CPM starts when event 2 is accomplished. In the process network it is mapped into the material End_1_2, which is the output material of O_1_2, which means that the operation of O_1_2 is terminated. Generally speaking, in the process network such materials will be inputs to an operating unit, which have to be completed in the CPM graph before the activity

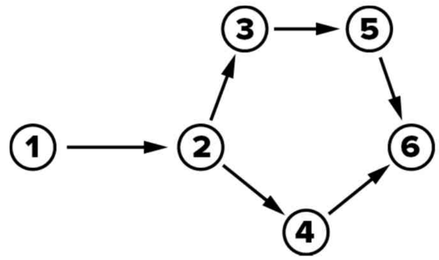


Figure 4.1: CPM graph example

can start. In the process network of the current example operating unit Close will have two input materials, namely End_4_6 and End_5_6, since in the CPM graph both event 4 and event 5 have to be accomplished before the activity between 6 and 7 can start. Each operating unit of the process network has one output, which represents the operational termination of the operating unit. Since the end point of the CPM graph is event 6 with two different preceding activities, in the process network an addition technical operating unit, called Close, has to be inserted. As a result, the CPM graph example can be mapped into the following process network.

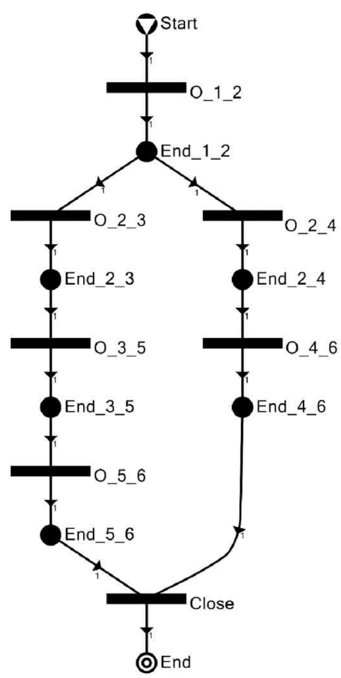


Figure 4.2: Process network of the CPM graph example

The structural mapping is done according to the above given details, the following Table illustrates the logical connections between the CPM and process networks.

CPM	Process network
Activity.	Operating unit.
Event.	Material (raw material, intermediate and product)
Logical connection between the activities (dependencies between the activities).	Material flow (arcs).
CPM graph: scheduling problem.	Process network: network of the operating units producing the products from the raw materials.

Table 4.2: Logical connection between the CPM and process networks

In case the mapping is performed in this way the basic terminology of CPM and process networks can be combined for the sake of simplicity.

4.3 Illustrative example of the mapping

Let us consider the example as illustration published by Chanas and Zielinsky (2001). The CPM graph of the example is given in Figure 4.3.; while Figure 4.4 illustrates the process network representation of the example after the mapping. It should be added that in Figure 4.3 activity between event 4 and event 7 starts only when both activity between event 2 and event 4 and activity between event 3 and event 4 are finished. Similarly, in Figure 4.4 operating unit O_4_7 starts its operation only when both operating unit O_2_4 and operating unit O_3_4 finished their operations, i.e. materials End_2_4 and End_3_4 are available respectively, serving as input materials of O_4_7. The reader should also note that sine the operating units under consideration are different, their end points are also different.

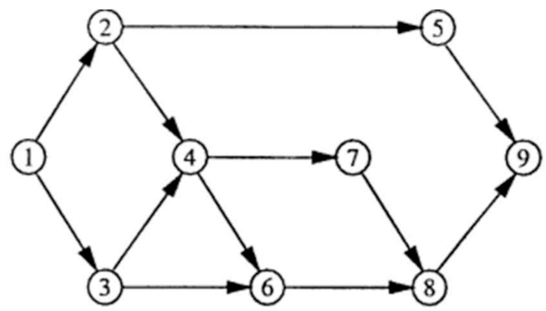


Figure 4.3: A CPM example

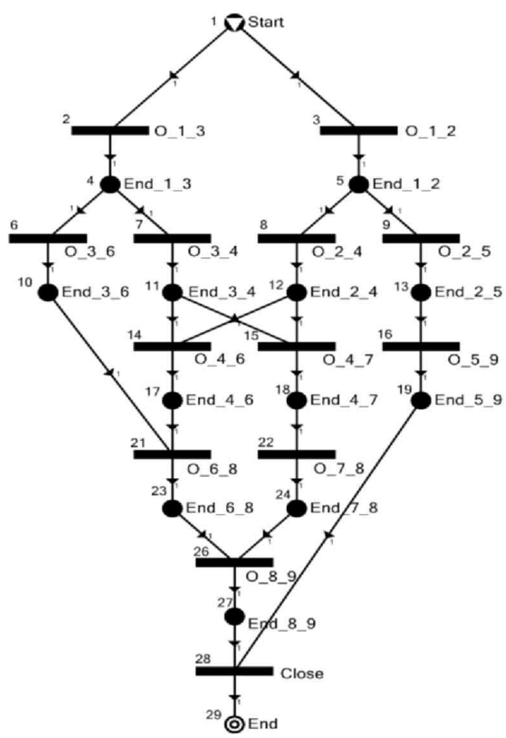


Figure 4.4: Process network of the illustrative example

4.4 Alternatives

Of course real case examples raise the question of possible alternatives. If a given problem can be solved by performing more than one activities or more than one series of activities, then it is called to be the problem of alternatives. This situation is not handled by CPM, moreover, crucial decisions have to be made prior to the depiction of the CPM graph. Nevertheless, these decisions may fundamentally influence the overall duration of the final result of the CPM solution. Obviously, it would be of high importance not to exclude any possibilities at the beginning, but to have these decisions as a result of a solution process also. The transformation presented in this study provides the possibility to have these decisions later on. Alternatives can be added into the process network and thus CPM problems extended with alternatives can be considered within this framework. In the process network representation of the problem these are considered as alternative arcs. This can be easily illustrated on the process network by adding a parallel operating unit for the alternative group, see the following:

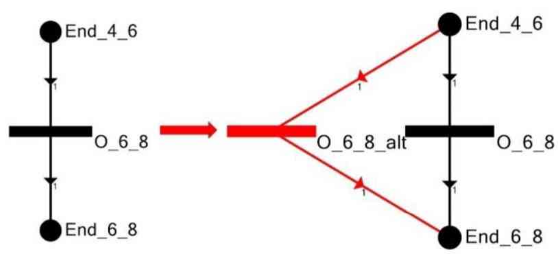


Figure 4.5: Illustration of adding an alternative

Based on the mapping of the basic elements and logical connections of the CPM and process network representations described earlier in this paper, it is obvious that the problem of alternative arcs and alternative paths can be described as alternative operating units of the process networks. As a result, project processes can be illustrated in more details in the process network representation than in CPM graphs. A given activity is performed by one operating unit or another, or one phase of the work is performed by one operating unit and another phase by another operating unit. This can be imagined for example when during a production process a decision maker controls the alternative operating units (arcs), namely which operating unit should operate during the work process. In the process network representation, this means information additional to the structural representation. Until this point structural construction was performed, in other words from the raw materials and set of operating units the production process was generated to produce the desired products with the help of additional intermediate materials. Here, a decision maker may set up priorities and may decide between alternatives. Normally, this information is represented in the cost function of the operating units. All in all, based on the mapping described previously, the CPM graph was transformed into a process network and then alternative solutions were also added into the process network representation, see alternative operating units $O\_6\_8\_alt$ and $O\_8\_9\_alt$ on Figure 4.6 . The redaer should realize that the input and output materials of the added alternative operating units are identical to that of the original operating units.

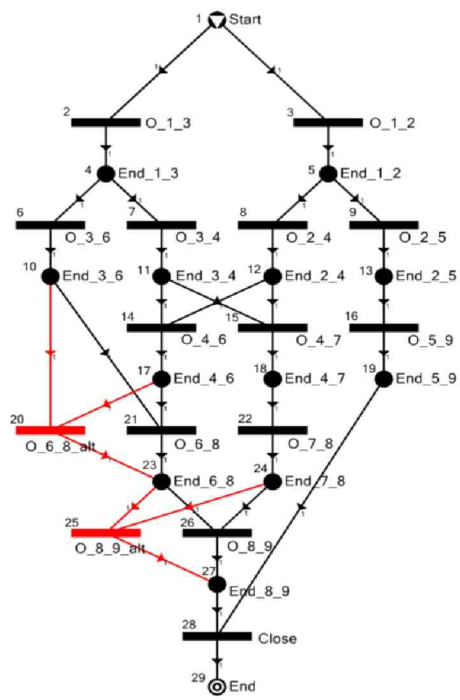


Figure 4.6: Illustrative example with alternatives

The axioms of process networks (see Friedler et al. [41]) determine a solution structure within the process network. According to the terminology of the CPM, the axioms of process networks have to be extended with the followings: each event in any solution structure, which is represented by a material in the process network, has one and only one input arc, except the Start event, which has zero indegree. In other words, it is also important to lock out from the solution point of view the parallel alternatives, therefore it has to be stated that from every operating unit there exists one and only one path to the final product (which is the end of the project in case of the CPM model). In this regards, the solution structures of the process network now correspond to the CPM graph. Therefore, adding the alternatives within the process network, multiple CPM graphs are described for the original problem. As a result, the optimal solution with a given set of constraints of the original problem is generated from the mathematical programming model of the process network with alternatives. The proposed solution method considering the alternatives also is given in Figure

4.7 .

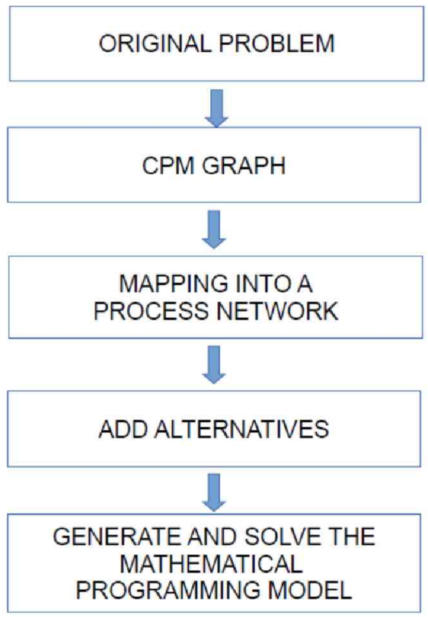


Figure 4.7: Proposed solution method

4.5 Mathematical programming model

Let A, E, D be finite sets, where A denotes the set of Activities, E denotes the set of Events and D denotes the set of edges. Let G be the bipartite process network as follows.
The $G(A, E, D)$ graph has the following properties: $A \cap E = \emptyset$, $D \subsetneq (A \times E) \cup (E \times A)$.
$A = \{i \in N\}$ activities
$E = \{j \in N\}$ events
As an illustration, the above formula means that in Figure 4.6:

$$A = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 20, 21, 22, 25, 26, 28\}$$

$$E = \{1, 4, 5, 10, 11, 12, 13, 17, 18, 19, 23, 24, 27, 29\}$$

$$D = \{(1, 2), (1, 3), (2, 4), (3, 5), ..., \}$$

Let x_i denote the i-th activity in the CPM graph and operating unit in the process network, where

$$x_i = \begin{cases} 1 & \text{if the i-th activity is performed} \\ 0 & \text{if the i-th activity is not performed} \end{cases}$$

t_i=the time from start-up to the i-th event occurs
T_i=the duration the i-th activity
T=planned upper time limit for the total project
C_i=cost of the i-th activity
C=planned upper budget for the total project

$$x_{Close} = 1 \qquad Close \in A \tag{4.1}$$

$$\sum_{\{i:i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{4.2}$$

Line 4.1 refers to the fact that the project has to be finished; line 4.2 refers to the fact that only one alternative can be considered. Let $\{b_1, b_2, ..., b_n\}$ be the solution of the above equations and let $S = \{i : i \in A \quad and \quad b_i = 1\}$ be a subset of A. The set of nodes S and the corresponding edges designates a part of $G(A, E, V)$ process graph, which exactly specify a CPM graph.
As an illustration, all CPM graphs of the illustrative example depicted on Figure 4.6 are as follows for $i \in A$:

CPM graph 1: $x_i = \begin{cases} 0 & \text{if i=20 and i=25} \\ 1 & \text{otherwise} \end{cases}$

CPM graph 2: $x_i = \begin{cases} 0 & \text{if i=21 and i=26} \\ 1 & \text{otherwise} \end{cases}$

CPM graph 3: $x_i = \begin{cases} 0 & \text{if i=20 and i=26} \\ 1 & \text{otherwise} \end{cases}$

CPM graph 4: $x_i = \begin{cases} 0 & \text{if i=21 and i=25} \\ 1 & \text{otherwise} \end{cases}$

4.5.1 Mathematical programming model of the time optimal project plan

The mathematical programming model for the usual CPM problem, extended with alternatives is given below.

$$x_{Close} = 1 \qquad Close \in A \tag{4.3}$$

$$\sum_{\{i:i\in A\,(i,j)\in D\}} x_i = 1 \qquad \forall j \in E \quad and \quad j \neq Start \tag{4.4}$$

$$t_{Start} = 0 \qquad Start \in E \tag{4.5}$$

$$t_k + \sum_{\{i:i\in A\,(i,j)\in D\}} x_i T_i \leq t_j \quad \forall k,j \in E \setminus Start$$

$$where \quad \exists i : (k,i) \quad and \quad (i,j) \in D \tag{4.6}$$

$$t_{End} \longrightarrow min \tag{4.7}$$

The reader should notice that lines 4.3 and 4.4 are equal to lines 4.1 and 4.2 above; while line (4) refers to that only one alternative can be considered, line 4.5 refers to the fact that the project started at time zero, the line 4.6 indicates that the activities have been completed at the earliest time. Line 4.7 refers to the aim, which is to minimize the overall duration of the project. It should also be mentioned that the solution of this mathematical programming model equals to the solution in which the shortest duration time is chosen among the alternatives.

4.5.2 Mathematical programming model of the cost optimal project plan

The mathematical programming model for the resource allocation problem of the given CPM graph, extended with alternatives is given below.

$$x_{Close} = 1 \qquad Close \in A \tag{4.8}$$

$$\sum_{\{i:i\in A\,(i,j)\in D\}} x_i = 1 \qquad \forall j \in E \quad and \quad j \neq Start \tag{4.9}$$

$$\sum_{\{i:i\in E\}} x_i C_i \longrightarrow min \tag{4.10}$$

Note here that lines 4.8 and 4.9 are equal to lines 4.1 and 4.2 above. Line 4.10 refers to the aim, which is to minimize the overall cost of the project.

4.5.3 Mathematical programming model of the time optimal project plan with additional cost constraint

In real case examples, not only the time constraints are important for the project planning but also the different costs of the alternatives are also of importance. Thus the previously described mathematical programming model can be reformulated as follows.

$$x_{Close} = 1 \qquad Close \in A \tag{4.11}$$

$$\sum_{\{i: i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \quad and \quad j \neq Start \tag{4.12}$$

$$t_{Start} = 0 \qquad Start \in E \tag{4.13}$$

$$t_j \geq x_i T_i + t_k \quad \forall j \in E \setminus Start \quad and \quad \forall i : (i,j) \in V \quad and \quad \forall k : (k,i) \in V \tag{4.14}$$

$$\sum_{\{i: i \in A} x_i C_i \leq C \tag{4.15}$$

$$t_{End} \rightarrow min \tag{4.16}$$

The reader should notice that lines 4.11 and 4.12 are eqivalent to lines 4.1 and 4.2 above; while line 4.13 refers to the fact that the project started at time zero, the line 4.14 indicates that the activities have been completed at the earliest time, while line (15) refers to the fact that the total project cost should not exceed the given upper budget limit. Line 4.16 refers to the aim, which is to minimize the overall duration of the project.

4.5.4 Mathematical programming model of the cost optimal project plan with time constraint

Finally, the mathematical programming model of the cost optimal project plan with time constraint can be given as follows.

$$x_{Close} = 1 \qquad Close \in A \tag{4.17}$$

$$\sum_{\{i: i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \quad and \quad j \neq Start \tag{4.18}$$

$$t_{Start} = 0 \qquad Start \in E \tag{4.19}$$

$$t_j \geq x_i T_i + t_k \quad \forall j \in E \setminus Start \quad and \quad \forall i : (i,j) \in V \quad and \quad \forall k : (k,i) \in V \tag{4.20}$$

$$t_{End} \leq T \qquad End \in E \tag{4.21}$$

$$\sum_{\{i:i \in A} x_i C_i \rightarrow min \tag{4.22}$$

Note here that lines 4.17 and 4.18 are eqivalent to lines 4.1 and 4.2 above; while line 4.19 refers to the fact that the project started at time zero, the line 4.20 indicates that the activities have been completed at the earliest time, while line 4.21 refers to the fact that the total project duration should not exceed the given upper time limit. Line 4.22 refers to the aim, which is to minimize the overall cost of the project.

4.6 Illustrative example

The illustrative example taken from Chanas S. and P. Zielinski [9] is revisited below, for details please consider the process network given in Figure 4.6 . Note here that the given fuzzy times are defuzzified using the Mean Of Maxima (MOM) method.

	a	b	c	d	Defuzzified time values
T_2	2.00	2.00	3.00	5.00	2.50
T_3	0.00	1.00	1.50	2.50	1.25
T_6	6.00	6.00	7.00	9.00	6.50
T_7	0.00	0.00	0.00	0.00	0.00
T_8	0.00	0.00	0.00	0.00	0.00
T_9	1.00	2.00	3.00	5.00	2.50
T_{14}	4.00	5.00	5.00	6.00	5.00
T_{15}	8.00	9.00	9.00	10.00	9.00
T_{16}	6.00	8.00	9.00	13.00	8.50
T_{21}	2.00	4.00	4.00	6.00	4.00
T_{22}	1.00	3.00	4.00	4.00	3.50
T_{26}	4.00	6.00	9.00	12.00	7.50

Table 4.3: Duration of the i-th activity

The mathematical programming model of the time optimal project plan with additional cost constraint given above in line 4.11 – 4.16 was formulated. Let us consider the following costs of the activities given as follows: $C_i = 5 \quad i \in \{2, 3, 6, 7, 8, 9, 14, 15, 16, 22, 28\}$ $C_i = 10 \quad i \in \{21, 26\}$ with the duration of the alternative activities: $T20 = 8$; $T25 = 9.5$; and the costs of the alternative activities: $C20 = 5$; $C25 = 5$. Let us consider the planned upper budget for the total project $C = 80$. Thus, the solution of the mathematical programming model results in the overall project duration is 22.5; and in this optimal solution, nodes 21 and 26 are selected, and the alternatives are excluded. Should the planned upper budget for the total project be $C = 70$; then the solution of the mathematical programming model results in the overall project duration 24.5; while in this optimal solution, alternative node 20 is selected instead of the original 21, and the alternative node 25 is excluded.

4.7 Conclusion

The critical path method (CPM) gives the longest path of the planned activities together with its overall duration. In each case only time is considered and only one solution is found at a time. Resources together with their costs do not appear in CPM graphs. In the present paper a novel method to extend the problem range of CPM problems is given. First, it was illustrated how a CPM problem should be transformed into a process network problem. This mapping has no antecedents in the literature since this situation is not handled in the CPM graph, moreover there is no CPM and process network connection yet. This transformation serves as the key to commonly handle resources together with their costs within the given process network. Since real case examples raise the question of alternatives, namely when more than one activity or more than one series of activities are considered for a subtask, this can also be represented in the process network. CPM methods assign the resources to the activities only without any influence to the CPM graph. When another resource is considered for use in the same activity in most cases all the parameters have to be reset and a new graph has to be depicted, and thus the problem has to be solved again. This entails solving a large number of problems to be solved separately, and the parameters considered for each separate problem are independent from the other separate problems' parameters. However, these parameters are dependent on each other in real case problems. Moreover, CPM graph techniques do not handle at all such cases where a given problem can be solved by performing more than one activities, i.e. alternatives. In other words crucial decisions regarding alternatives have to be made before the CPM graph is depicted. Here as a novel solution alternatives can be added and are handled within the given model. Moreover,when alternatives are added to the process network all resources together with their parameters depending on each other are considered and handled within the model, as in the case of real case problems. After describing the extended problem with alternatives,four different mathematical programming models are given in the present work. These mathematical programming models cover a wider range of problems than that of the CPM methodology. These mathematical programming models can be generated and solved algorithmically. The solution of the mathematical programming model is the optimal solution of the original CPM problem, i.e. the exact project definition, solution. Another novelty of the present work is that all solutions of the original CPM problem can be generated and ranked in order according to the cost function.

Chapter 5

Optimization the process network using new concept of fuzzy linear programming

5.1 Introduction

Around 1950, a new network model technique was introduced called the Critical Path Method(CPM). Today this method is widely used for the time scheduling of projects, in different areas of economics, software development, engineering, research projects and product development. The CPM technique calculates time parameters for the events and for the activities. When the CPM technique calculates the crisp activity time parameter values and crisp event time parameters, it is assumed that these parameters can be calculated exactly. In the case of the uncertainty activity time, a statistical technique has been developed for project time scheduling. This technique is called the Project Evaluation and Review Technique (PERT). It uses statistical methods to estimate the expected activity time based on the three time values, where these time values are supplied by the experts. In the PERT model, the expert gives the most likely duration time, the optimistic duration time and the pessimistic duration time of the activities. The PERT model calculates from the duration time parameters an expected time parameter for activity duration. The mathematical basis of this method is the beta distribution. This method was developed in the 1950s at the same times as the CPM method. Later to handle the activity time uncertainty, the interval activity time [9] [10] [78] and the fuzzy activity time were introduced [69]. The interval activity duration time and the fuzzy activity duration time were also introduced to calculate uncertainty activity times [68]. Fuzzy theory [12] is widely applied in solving uncertainty problems in many areas from mathematical problems throughout engeneering problems, to the area of project management. In project management many methods have been developed for the scheduling optimization [77] [63] [61] [83] [62]. A widely accepted method used in the CPM method to calculate the optimal event parameters is the linear optimization model [65]. If we have a CPM graph events $E_1, ..., En$ and the A_{ij} activity between E_i and E_j events have a duration time T_{ij}, then we can get the event occurrance times $t_1, ..., tn$ by solving the following linear optimization model. Let us denote the set of activities by A. And let

$$t_j - t_i \geq T_{ij} \qquad \forall i, j : \qquad A_{ij} \in A$$

$$t_n - t_1 \rightarrow min$$

In [88], there are the solutions for optimization problems of the process network represented by the CPM method extended with alternatives. In this paper we give a generalization of these linear optimization methods for fuzzy duration times T_{ij}, fuzzy event times E_i and fuzzy cost times C_{ij}. We will present a new approach for solving fuzzy linear optimization problems. Friedler [41] [40] introduced a process network methodology that has become important for modelling chemical engineering processes. In the processes, in many cases we cannot determine exactly the time parameters, namely the beginning time, the duration time and the finish time for an activity. For an example of the fuzzy number application in the P-graph represented workflow management see [86]. This applications requires one to handle the uncertainty timing in the processes in the solution of the process scheduling and optimization. We apply the fuzzy time parameters in the project scheduling for the above-mentioned areas of the process network application. We will use a special representation of fuzzy numbers [18]. Here we will present the new approach for fuzzy linear programming on the process network model described in [88]. The following results can be found in the article [20].

5.2 Linear combination of trapezoid membership function

Fuzzy arithmetic [43] is often used to calculate the fuzzy membership function. There are a number of approaches available to solve this problem. In this article we shall use the following fuzzy number representation.

Definition: Let the normalized cut function be [28]:

$$[x] = min(1, max(0, x)) = \begin{cases} 0 & if \quad x < 0 \\ x & if \quad 0 \leq x \leq 1 \\ 1 & if \quad 1 < x \end{cases}$$

Our representation of this fuzzy number is based on the special fuzzy number representation see [88]. In this representation the parameters are those of the linear left and right side functions of the trapezoidal fuzzy membership function for $i \in 1, 2, ..., n$.

$$L_i(x) = \left[m_{i,l}(x - a_{i,l}) + \frac{1}{2} \right] \qquad and \qquad R_i(x) = \left[m_{i,r}(x - a_{i,r}) + \frac{1}{2} \right] \tag{5.1}$$

In this representation of the fuzzy values the $a_{i,l}$ and the $a_{i,r}$ are called the left hand side and the right hand side mean parameters, and the $m_{i,l}$ and $m_{i,r}$ are called the left hand side and the right hand side tangent parameters. We can write these, using the general cut function for the trapezoid fuzzy number:

$$F_{(m_{i,l},a_{i,l},m_{i,r},a_{i,r})}(x) = \left[\left[m_{i,l}(x - a_{i,l}) + \frac{1}{2} \right] + \left[m_{i,r}(x - a_{i,r}) + \frac{1}{2} \right] - 1 \right] \tag{5.2}$$

This arithmetic-based concept leads to the application of the fuzzy logic operators[66], but this lies outside the scope of this study. The ABFD arithmetic-based fuzzy aggregation concept for fuzzy number linear combination is based on the inverse function [18]. It is proved [18] that the linear combination of a trapezoid function is also a trapezoid function. Let the aggregation of these fuzzy numbers be:

$$F_{(m_l,a_l,m_r,a_r)}(x) = \left[\left[m_l(x - a_l) + \frac{1}{2} \right] + \left[m_r(x - a_r) + \frac{1}{2} \right] - 1 \right] \tag{5.3}$$

It can be proved that:

$$\frac{1}{m_l} = \sum_i^n \frac{1}{m_{i,l}} \qquad a_l = \sum_i^n a_{i,l} \qquad \frac{1}{m_r} = \sum_i^n \frac{1}{m_{i,r}} \qquad a_r = \sum_i^n a_{i,r} \tag{5.4}$$

5.3 ABFC defuzzification methods and its properties

In most fuzzy applications the outputs are fuzzy values. Such an output usually requires a conversion from a fuzzy value to a crisp value. The method used here is called the defuzzyfication method. However there are many well known and widely used defuzzification methods available in the literature [18] [15] [89] [28] [76]. There are maximum location based methods available such as the Mean Of Maximum(MOM), the Smallest Of Maximum(SOM) and the Largest Of Maximum(LOM), the widely used centroid method the Center Of Gravity(COG) and the parametrized defuzzification method the Basic Defuzzification Distributions(BADD) method. The MOM, SOM, LOM defuzzification methods operate on the maximum location value so they compute their crisp value for $M = \{x | \mu_A(x) = max(\mu_A)\}$. We note that in the following x is an output variable while $\mu(x)$, $\mu_A(x)$, and , $\mu_i(x)$ denote membership functions.

Defuzzificatiom method	Calculated value
MOM	$\frac{\int_M x dx}{\int_M dx}$
SOM	$min\{x \vert x \in M\}$
LOM	$max\{x \vert x \in M\}$
COG	$\frac{\int_A \mu_A(x) x dx}{\int_A \mu_A(x) dx}$
BADD(α)	$\frac{\int_A \mu_A^\alpha(x) x dx}{\int_A \mu_A^\alpha(x) dx}$

Here, we will use two defuzzification methods taked from the fuzzy control theory [18] [72], namely Arithmetic Based Fuzzy Control area. These two methods are called the ABFC center point (ABFC-CP) method and the ABFC-I intersection method [88]. Next let x_M be the intersection point of the left and right hand side of the fuzzy number. Namely,

$$x_M = \frac{m_l a_l - m_r a_r}{m_l - m_r} \qquad (ABFC - I)$$

Another defuzzification method is called the ABFC center point [61] (ABFC-CP) method [88], which is a COG based method for trapezoid fuzzy values:

$$x_S = \frac{a_l + a_r}{r} + \frac{1}{24}\frac{1}{a_l - a_r}\left(\frac{1}{m_l^2} - \frac{1}{m_r^2}\right) \qquad (ABFC - CP)$$

In the figures below we can see typical plots got by applying these defuzzification methods.

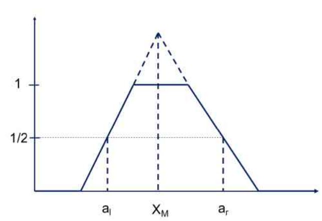


Figure 5.1: Intersection

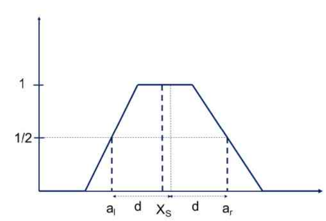


Figure 5.2: Center Point

The advantage of these methods are that we can compute the aggregated fuzzy numbers in those cases where the support of the fuzzy number has an infinite interval,(see Fig. 4,) or where for any fuzzy number of the left hand side or right hand side linear function lies parallel with y axis, so its gradient is infinit (see Figure 5, and 6,).

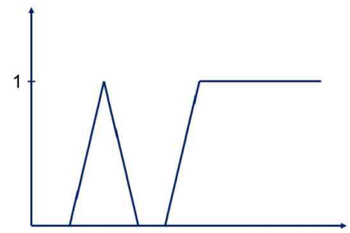


Figure 5.3: Infinite number

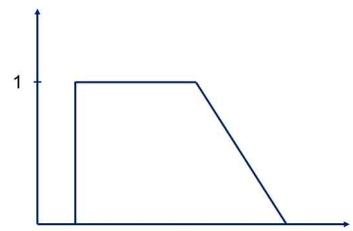


Figure 5.4: Trapezoid

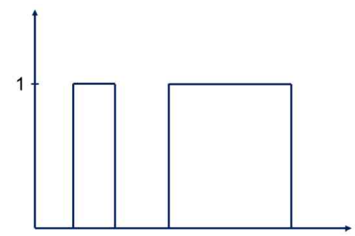
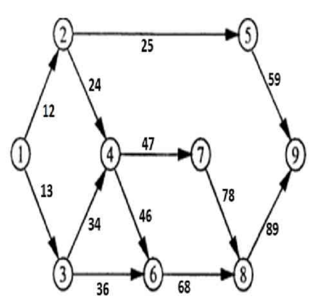


Figure 5.5: Rectangles

The main advantages of these methods can be seen in the fuzzy control systems. With this method we can solve classical problems of the Mamdani method, without classical errors, in problem areas like: range independence [18],effect of fuzzyness [18],efficiency of the calculations [18], and computational speed [18].

5.4 Optimization by using a new type of fuzzy linear programming

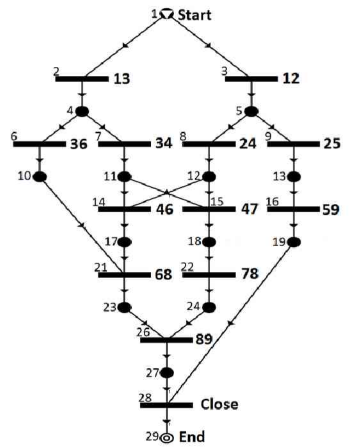
Now, we will define activity times in the new representation. The parameter values of the fuzzy activity times and the parameters for the new representation are listed below. For a better understanding we use the CPM example published by Chanas and Zielinsky (2001). In this chapter **ij** denote the activity between i and j event, and **ij**alt denote its alternative in the process network graph.



	a_l	m_l	a_r	m_r
12	0.50	1.00	2.00	-1.00
13	2.00	∞	4.00	-0.50
24	0.00	∞	0.00	∞
25	1.50	1.00	4.00	-0.50
34	0.00	∞	0.00	∞
36	6.00	∞	8.00	-0.50
46	4.50	1.00	5.50	-1.00
47	8.50	1.00	9.50	-1.00
59	7.00	0.50	11.00	-0.25
68	3.00	0.50	5.00	-0.50
78	2.00	0.50	4.50	-1.00
89	5.00	0.50	10.50	-0.33

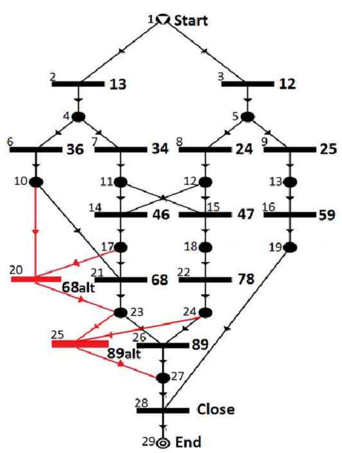
Figure 5.6: CPM graph example

The problem of alternatives in the process network is described in [88]. Here we consider two parameter of activities. These parameters are the cost and the time parameters. From a parameter perspective the alternatives are Pareto optimal. A process network representation of this CPM problem and the extended CPM problem with alternatives are shown in the following figure. Here we apply fuzzy alternative times given in an example described in [88].



	a_l	m_l	a_r	m_r
12	0.50	1.00	2.00	-1.00
13	2.00	∞	4.00	-0.50
24	0.00	∞	0.00	∞
25	1.50	1.00	4.00	-0.50
34	0.00	∞	0.00	∞
36	6.00	∞	8.00	-0.50
46	4.50	1.00	5.50	-1.00
47	8.50	1.00	9.50	-1.00
59	7.00	0.50	11.00	-0.25
68	3.00	0.50	5.00	-0.50
78	2.00	0.50	4.50	-1.00
89	5.00	0.50	10.50	-0.33

Figure 5.7: Process Network and fuzzy activity times



	a_l	m_l	a_r	m_r
12	0.50	1.00	2.00	-1.00
13	2.00	∞	4.00	-0.50
24	0.00	∞	0.00	∞
25	1.50	1.00	4.00	-0.50
34	0.00	∞	0.00	∞
36	6.00	∞	8.00	-0.50
46	4.50	1.00	5.50	-1.00
47	8.50	1.00	9.50	-1.00
59	7.00	0.50	11.00	-0.25
68	3.00	0.50	5.00	-0.50
68_ALT	2.50	0.33	6.50	-0.33
78	2.00	0.50	4.50	-1.00
89	5.00	0.50	10.50	-0.33
89_ALT	5.50	0.33	10.50	-0.33

Figure 5.8: Alternative extended Process Network and fuzzy activity times

In the figure we denote the activities - in the process network they are the operation units -in the same way as we did in the CPM network. This number pairs are in the right side of the activities (operation units). Next, let us denote the vertices in the bipartite process network graph by numbers from 1 to 29 thats are on the left hand side of the activities (operation units) and near the events (material vertices). In the mathematical model we use the above-mentioned new row-wise numbering for the vertices in the process network from 1 to 29. We construct mathematical models for the process network solution of extended CPM problems with alternatives and we introduce a new way of solving the fuzzy linear programming problem. In [88] mathematical optimization models are given for process network represented CPM problems extended with alternatives. We will generalize these models for fuzzy time and cost parameters. In the Chapter 4. we described the mathematical model of the process network. Here the t_i and the T_i time parameters and the C_i cost parameters take fuzzy values. Let the cost fuzzy values be the following.

	a_l	m_l	a_r	m_r
C_2	3,50	1,00	6,50	-1,00
C_3	3,00	0,50	7,50	-1,00
C_6	1,50	1,00	8,00	-0,50
C_7	3,00	0,50	7,50	-0,33
C_8	1,50	1,00	8,00	-0,50
C_9	2,00	0,50	8,00	-0,50
C_{14}	2,00	0,50	8,00	-0,50
C_{15}	2,50	0,33	8,00	-0,25
C_{16}	3,00	0,50	7,00	-0,50
C_{20}	3,50	1,00	7,50	-1,00
C_{21}	3,00	0,50	9,50	-1,00
C_{22}	2,50	0,33	7,50	-1,00
C_{25}	2,50	0,33	7,00	-0,50
C_{26}	3,00	0,50	7,00	-0,50

Table 5.1: Fuzzy cost parameter values

For the fuzzy activity time T_i, let the mean and tangent parameters be denoted by:

$$a_{A,i,l}, \quad m_{A,i,l}, \quad a_{A,i,r}, \quad m_{A,i,r}$$

For the fuzzy event time t_i, the mean and the tangent parameters be denoted by:

$$a_{E,i,l}, \quad m_{E,i,l}, \quad a_{E,i,r}, \quad m_{E,i,r}$$

Similarly let us denote the fuzzy cost mean and tangent time parameters by:

$$a_{C,i,l}, \quad m_{C,i,l}, \quad a_{C,i,r}, \quad m_{C,i,r}$$

In the mathematical model let:

$$M_{A,i,r} = \frac{1}{m_{A,i,r}} \quad M_{E,i,r} = \frac{1}{m_{E,i,r}}$$

5.5 New two step concept for fuzzy linear programming

We summarize in the following, the logical concept of the new fuzzy optimization model.
Step 1.
For the first step we present the left hand side and the right hand side mean parameters of fuzzy time values. We give four mathematical optimization models for time optimization, cost optimization, time optimization with cost constraint, and cost optimization with time constraint.

Time optimization model In the time optimization model we minimize the sum of the average of mean values of the fuzzy time parameters, like so:

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \longrightarrow min$$

In the case where the objective function is

$$\frac{a_{E,29,l} + a_{E,29,r}}{2}$$

the model results in fuzzy values with more uncertainty.

Cost optimization model In the cost optimization model let us minimize the average of the mean values of the fuzzy cost parameters like so:

$$\sum_{\forall s \in E} \frac{a_{C,s,l} + a_{C,s,r}}{2} \longrightarrow min$$

The cost optimization model gives the left hand side and the right hand side mean parameters of fuzzy time values as the output of the first step, but there are not time optimal values.

Time optimization model with cost constraint In the time optimization model with cost constraint we will give an additional cost constraint condition for time optimal model.

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \leq C \tag{5.5}$$

Cost optimization model with time constraint In the cost optimization model with time constraint we give an additional time constraint condition for cost optimal model. This time constraint is associated with the objective function of the time optimality model.

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T \tag{5.6}$$

In the case where the time constraint applies to the value

$$\frac{a_{E,29,l} + a_{E,29,r}}{2}$$

the model results in more uncertainty fuzzy values. For this step the mathematical model generates the optimal structure for the CPM problem in the process network. The activities which are in generated in an optimal structure using Step 1. are the input activities of the model applied using Step 2.

Step 2.

For the second step we solve the time optimal mathematical model to get the average of the left hand side and the right hand side tangent parameter values for the activities of the optimal structure, which was generated using Step 1.

$$\sum_{\forall s \in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \longrightarrow min$$

Now we will optimize the uncertainty of the fuzzy output time parameter values.

The process network which represents the alternative extended CPM problem is a solution structure of the process network that represents the CPM problem.

The events in the CPM problem is represented in the process network graph by the material type vertices. The Start event has zero in-degree in the process network graph.

The mathematical model generates the optimal structure in the process network, which provides the optimal solution of the original CPM problem.

5.5.1 Time optimization model

Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{5.7}$$

$$\sum_{\{i: i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.8}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.9}$$

$$a_{E,k,l} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.10}$$

$$a_{E,k,r} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.11}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \longrightarrow min \tag{5.12}$$

We note that the condition in 5.7 refers to the fact that the project has to terminate and the condition in 5.8 means that only one alternative may be chosen in the optimal structure, while the condition in 5.9 means that the project starts at the zero point of the project time. The conditions for the activities and events on the left hand side means and for the right hand side means in 5.10 and 5.11 refers to the fact that that the activities terminated at the earliest time, applying the aggregation formula 5.4 for the mean parameters, the condition in 5.10 and 5.11 refers to the minimization of the project time means. In Step 1. the mathematical model choses the possible alternatives of the optimal structure. For the output activity set of the optimal structure is denoted by A_T and the set of edges in the optimal structure is denoted by D_T. The set of events E is not changed in Step 1.

$$A_T = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 21, 22, 26, 28\}$$

Now we define a mathematical model for the tangent parameter values for the optimal structure of the process network which contains the alternatives of the A_T set.
Step 2.

$$x_{Close} = 1 \qquad Close \in A_T \tag{5.13}$$

$$\sum_{\{i:i \in A_T\ (i,j) \in D_T\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.14}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.15}$$

$$M_{E,k,l} + \sum_{\{i:i\in A_T\ (i,j)\in D_T\}} x_i M_{A_T,i,l} \leq M_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D_T \quad and \quad (i,j) \in D_T \tag{5.16}$$

$$|M_{E,k,r}| + \sum_{\{i:i\in A_T\ (i,j)\in D_T\}} x_i |M_{A_T,i,r}| \leq |M_{E,j,r}| \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D_T \quad and \quad (i,j) \in D_T \tag{5.17}$$

$$\sum_{\forall s\in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \longrightarrow min \tag{5.18}$$

We note that we get the conditions in (5.16) and (5.17) for the tangent parameters, applying the (5.4) aggregation formula for the tangent parameter values. The condition in the (5.16) and (5.17) and the objective function in (5.18) refers to the fact that the project is terminated with minimal fuzzy event time parameter uncertainty. The mathematical model defines the optimal structure of activities and events. In Table 5.1 we can see the event fuzzy parameter values of the optimal structure.

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	2.00	-1.00
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	2.00	-1.00
t_{13}	2.00	0.50	6.00	-0.33
t_{17}	6.50	0.50	9.50	-0.33
t_{18}	10.50	0.50	13.50	-0.33
t_{19}	9.00	0.25	17.00	-0.14
t_{23}	11.00	0.25	17.00	-0.17
t_{24}	12.50	0.25	18.00	-0.25
t_{27}	17.50	0.17	28.50	-0.11
t_{29}	17.50	0.17	28.50	-0.11

Table 5.2: Time optimal structure solution

We note that formally Step 2. is the same in the four models, but the set of alternatives and the set of edges are different for each model in Step 2.
Here we summarize the time and cost output parameters of the model. We give in the summary

table the mathematical formula and the calculated value of the output parameters.

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	130.75
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.00
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	62.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \lvert M_{E,s,r} \rvert}{2}$	43.00
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\lvert E \rvert}$	0.46
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\lvert E \rvert}$	-0.39
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	66.50

5.5.2 Cost optimization model

Next we define the cost optimization model. Based on our concept the difference between the cost optimalization model and the time optimization model is the objective function. In the cost optimalization model we define a new objective function for the cost parameters.

Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{5.19}$$

$$\sum_{\{i: i \in A \; (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.20}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.21}$$

$$a_{E,k,l} + \sum_{\{i: i \in A \; (i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k, j \in E \quad j \neq Start$$
$$where \qquad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.22}$$

$$a_{E,k,r} + \sum_{\{i: i \in A \; (i,j) \in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k, j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.23}$$

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \longrightarrow min \tag{5.24}$$

In Step 1. the mathematical model choses the alternative set of the optimal structure. For Step 2. the output alternative set of the optimal structure let A_C us denote and let D_C denote the set of edges of the optimal structure. The E set of events is not changed in Step 1.
The output activities of this method are:

$$A_C = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 20, 22, 25, 28\}$$

Next we define the mathematical model for the tangent parameters used for the structure of the process network which contains the alternatives is chosen by the mean model in Step.1.
Step 2.

$$x_{Close} = 1 \qquad Close \in A_C \tag{5.25}$$

$$\sum_{\{i:i\in A_C\ (i,j)\in D_C\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.26}$$

$$M_{E,1,l} = M_{E,1,r} = 0 \tag{5.27}$$

$$M_{E,k,l} + \sum_{\{i:i\in A_C\ (i,j)\in D_C\}} x_i M_{A_C,i,l} \leq M_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_C \quad and \quad (i,j) \in D_C \tag{5.28}$$

$$|M_{E,k,r}| + \sum_{\{i:i\in A_C\ (i,j)\in D_C\}} x_i |M_{A_C,i,r}| \leq |M_{E,j,r}| \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_C \quad and \quad (i,j) \in D_C \tag{5.29}$$

$$\sum_{\forall s\in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \longrightarrow min \tag{5.30}$$

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters.

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	4.50	-1.00
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	4.50	-1.00
t_{13}	2.00	0.50	8.50	-0.33
t_{17}	8.00	0.50	12.00	-0.33
t_{18}	10.50	0.50	14.00	-0.33
t_{19}	9.00	0.25	19.50	-0.14
t_{23}	10.50	0.20	18.50	-0.14
t_{24}	12.50	0.25	18.50	-0.25
t_{27}	18.00	0.13	29.00	-0.10
t_{29}	18.00	0.13	29.00	-0.10

Table 5.3: Cost optimal structure solution

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s\in E} \frac{a_{E,s,l}+a_{E,s,r}}{2}$	139.750
The last event time mean average	$\frac{a_{E,End,l}+a_{E,End,r}}{2}$	23.50
Sum of the difference of means	$\sum_{\forall s\in E} (a_{E,s,l}-a_{E,s,r})$	76.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s\in E} \frac{M_{E,s,l}+\lvert M_{E,s,r}\rvert}{2}$	47.00
The average of the left hand side tangent	$\sum_{\forall s\in E} \frac{M_{E,s,l}}{\lvert E\rvert}$	0.45
The average of the right hand side tangent	$\sum_{\forall s\in E} \frac{M_{E,s,r}}{\lvert E\rvert}$	-0.38
The sum of the cost mean averages	$\sum_{\forall i\in E} x_i \frac{a_{C_i,l}+a_{C_i,r}}{2}$	65.50

5.5.3 Time optimization model with cost constraint

In this model we give a time optimalization with a cost constraint.In the project cost corresponds to the resources. Cost variability appears in the project as resource variability. In the alternative extended process network representation of the CPM problem cost variability is considered as the chosen of alternatives.

In this model we use the total cost limit C parameter for the project. Next, to define this mathematical model we definee an additional cost constraint condition in the time optimization model for Step 1.

Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{5.31}$$

$$\sum_{\{i:i\in A\ (i,j)\in D\}} x_i = 1 \qquad \forall j\in E \qquad j\neq Start \tag{5.32}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.33}$$

$$a_{E,k,l} + \sum_{\{i:i\in A\ (i,j)\in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k,j\in E \quad j\neq Start$$

$$where \qquad \exists i:(k,i)\in D \quad and \quad (i,j)\in D \tag{5.34}$$

$$a_{E,k,r} + \sum_{\{i:i\in A\ (i,j)\in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k,j\in E \quad j\neq Start$$

$$where \qquad \exists i:(k,i)\in D \quad and \quad (i,j)\in D \tag{5.35}$$

$$\forall i\in A: x_{i,l} = x_{i,r} \tag{5.36}$$

$$\sum_{\forall i\in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \leq C \tag{5.37}$$

$$\sum_{\forall s\in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \longrightarrow min \tag{5.38}$$

In Step 1. the mathematical model chose the alternatives of the optimal structure. For the Step 2. the output activity set of the optimal structure let denote $A_{T,CC}$ and let $D_{T,CC}$ denote the set of edges of the optimal structure. The E set of events is not changed in the Step 1.
The output activities of this method are:

$$A_{T,CC} = \{2,3,6,7,8,9,14,15,16,20,22,26,28\}$$

Step 2.

$$x_{Close} = 1 \qquad Close\in A_{T,CC} \tag{5.39}$$

$$\sum_{\{i:i\in A_{T,CC}\ (i,j)\in D_{T,CC}\}} x_i = 1 \qquad \forall j\in E \qquad j\neq Start \tag{5.40}$$

$$M_{E,1,l} = M_{E,1,r} = 0 \tag{5.41}$$

$$M_{E,k,l} + \sum_{\{i:i\in A_{T,CC}\ (i,j)\in D_{T,CC}\}} x_i M_{A_{T,CC},i,l} \leq M_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_{T,CC} \quad and \quad (i,j) \in D_{T,CC} \tag{5.42}$$

$$|M_{E,k,r}| + \sum_{\{i:i\in A_{T,CC}\ (i,j)\in D_{T,CC}\}} x_i |M_{A_{T,CC},i,r}| \leq |M_{E,j,r}| \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_{T,CC} \quad and \quad (i,j) \in D_{T,CC} \tag{5.43}$$

$$\sum_{\forall s\in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \longrightarrow min \tag{5.44}$$

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	2.00	-1.00
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	2.00	-1.00
t_{13}	2.00	0.50	6.00	-0.33
t_{17}	6.50	0.50	9.50	-0.33
t_{18}	10.50	0.50	13.50	-0.33
t_{19}	9.00	0.25	17.00	-0.14
t_{23}	10.50	0.20	18.50	-0.14
t_{24}	12.50	0.25	18.00	-0.25
t_{27}	17.50	0.14	29.00	-0.10
t_{29}	17.50	0.14	29.00	-0.10

Table 5.4: Time optimal structure with cost constraint solution

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters.

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	131.750
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.25
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	65.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \lvert M_{E,s,r}\rvert}{2}$	46.00
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\lvert E\rvert}$	0.45
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\lvert E\rvert}$	-0.38
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	65.75

The value of the cost constraint is 66.00.

5.5.4 Cost optimization model with time constraint

Next, to define this mathematical model we define an additional time constraint condition for the cost optimality model in Step 1.

Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{5.45}$$

$$\sum_{\{i:i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.46}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.47}$$

$$a_{E,k,l} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \qquad and \qquad (i,j) \in D \tag{5.48}$$

$$a_{E,k,r} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \qquad and \qquad (i,j) \in D \tag{5.49}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T \tag{5.50}$$

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \longrightarrow min \tag{5.51}$$

In Step 1. the mathematical model choses the possible alternatives of the optimal structure. In Step 2. the output activity set of the optimal structure and let $D_{C,TC}$ denote the set of edges of the optimal structure . The E set of events is not changed in Step 1. The output activities of the optimal structure are:

$$A_{C,TC} = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 21, 22, 25, 28\}$$

Step 2.

$$x_{Close} = 1 \qquad Close \in A_{C,TC} \tag{5.52}$$

$$\sum_{\{i: i \in A_{C,TC}\ (i,j) \in D_{C,TC}\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.53}$$

$$M_{E,1,l} = M_{E,1,r} = 0 \tag{5.54}$$

$$M_{E,k,l} + \sum_{\{i: i \in A_{C,TC}\ (i,j) \in D_{C,TC}\}} x_i M_{A_{C,TC},i,l} \leq M_{E,j,l} \quad \forall k, j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_{C,TC} \qquad and \qquad (i,j) \in D_{C,TC} \tag{5.55}$$

$$|M_{E,k,r}| + \sum_{\{i: i \in A_{C,TC}\ (i,j) \in D_{C,TC}\}} x_i |M_{A_{C,TC},i,r}| \leq |M_{E,j,r}| \quad \forall k, j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_{C,TC} \qquad and \qquad (i,j) \in D_{C,TC} \tag{5.56}$$

$$\sum_{\forall s \in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \longrightarrow min \tag{5.57}$$

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters.

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	2.00	-1.00
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	2.00	-1.00
t_{13}	2.00	0.50	6.00	-0.33
t_{17}	6.50	0.50	9.50	-0.33
t_{18}	10.50	0.50	13.50	-0.33
t_{19}	9.00	0.25	17.00	-0.14
t_{23}	11.00	0.25	17.00	-0.17
t_{24}	12.50	0.25	18.00	-0.25
t_{27}	18.00	0.14	28.50	-0.11
t_{29}	18.00	0.14	28.50	-0.11

Table 5.5: Cost optimal structure with time constraint solution

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	131.25
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.25
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	61.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \lvert M_{E,s,r} \rvert}{2}$	44.00
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\lvert E \rvert}$	0.45
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\lvert E \rvert}$	-0.39
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	66.25

The value of the time constraint is 131.25.
With the help of the following two model we would like to show the strengthness of the defined two step model, and the necessity of the conditions of this model to find the better approximation of the sharpest solution.

5.6 One step concept for fuzzy linear programming

Next, we summarize the logical concept of this variant of the new fuzzy optimization model. In this model we define the conditons in one step for both the mean and the tangent parameters. In the objective function, we optimalize for the mean parameter values.
We define the mathematical model for optimization on the left hand side and the right hand side mean parameters of fuzzy time values. We provide four mathematical optimization models for time

optimization, cost optimization, time optimization with cost constraint, and cost optimization with time constraint.

Time optimization model In the time optimization model we minimize the sum of the average of mean values of the fuzzy time parameters, like so:

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \longrightarrow min$$

Cost optimization model In the cost optimization model let us minimize the average of the mean values of the fuzzy cost parameters like so:

$$\sum_{\forall s \in E} \frac{a_{C,s,l} + a_{C,s,r}}{2} \longrightarrow min$$

The cost optimization model gives the left hand side and the right hand side mean parameters of fuzzy time values as the output of the first step, but there are not time optimal values.

Time optimization model with cost constraint In the time optimization model with cost constraint we will give an additional cost constraint condition for time optimal model.

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \leq C \tag{5.58}$$

Cost optimization model with time constraint In the cost optimization model with time constraint we give an additional time constraint condition for cost optimal model. This time constraint is associated with the objective function of the time optimality model.

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T \tag{5.59}$$

5.6.1 Time optimization model

$$x_{Close} = 1 \qquad Close \in A \tag{5.60}$$

$$\sum_{\{i: i \in A \ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.61}$$

$$a_{E,1,l} = a_{E,1,r} = M_{E,1,l} = M_{E,1,r} = 0 \tag{5.62}$$

$$a_{E,k,l} + \sum_{\{i:i \in A\,(i,j) \in D\}} x_i a_{A,i,l} \le a_{E,j,l} \quad \forall k,j \in E \quad j \ne Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.63}$$

$$a_{E,k,r} + \sum_{\{i:i \in A\,(i,j) \in D\}} x_i a_{A,i,r} \le a_{E,j,r} \quad \forall k,j \in E \quad j \ne Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.64}$$

$$M_{E,k,l} + \sum_{\{i:i \in A\,(i,j) \in D\}} x_i M_{A,i,l} \le M_{E,j,l} \quad \forall k,j \in E \quad j \ne Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.65}$$

$$|M_{E,k,r}| + \sum_{\{i:i \in A\,(i,j) \in D\}} x_i |M_{A,i,r}| \le |M_{E,j,r}| \quad \forall k,j \in E \quad j \ne Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.66}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \longrightarrow min \tag{5.67}$$

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters.
The output activities are: $A_T = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 21, 22, 26, 28\}$

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.20
t_5	0.50	1.00	2.00	-0.20
t_{10}	8.00	∞	12.00	-0.14
t_{11}	2.00	∞	4.00	-0.20
t_{12}	0.50	1.00	2.00	-0.20
t_{13}	2.00	0.25	6.00	-0.12
t_{17}	6.50	0.50	9.50	-0.14
t_{18}	10.50	0.50	13.50	-0.17
t_{19}	9.00	0.17	17.00	-0.08
t_{23}	11.00	0.25	17.00	-0.11
t_{24}	12.50	0.25	18.00	-0.11
t_{27}	17.50	0.17	28.50	-0.08
t_{29}	17.50	0.17	28.50	-0.08

Table 5.6: Time optimal structure solution

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	130.75
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.00
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	62.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \|M_{E,s,r}\|}{2}$	69.25
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\|E\|}$	0.43
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\|E\|}$	-0.14
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	66.50

5.6.2 Cost optimization model

$$x_{Close} = 1 \qquad Close \in A \tag{5.68}$$

$$\sum_{\{i:i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.69}$$

$$a_{E,1,l} = a_{E,1,r} = M_{E,1,l} = M_{E,1,r} = 0 \tag{5.70}$$

$$a_{E,k,l} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.71}$$

$$a_{E,k,r} + \sum_{\{i:i\in A\,(i,j)\in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.72}$$

$$M_{E,k,l} + \sum_{\{i:i\in A\,(i,j)\in D\}} x_i M_{A,i,l} \leq M_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.73}$$

$$|M_{E,k,r}| + \sum_{\{i:i\in A\,(i,j)\in D\}} x_i |M_{A,i,r}| \leq |M_{E,j,r}| \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.74}$$

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \longrightarrow min \tag{5.75}$$

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	2.00	-0.50
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	2.00	-0.50
t_{13}	2.00	0.29	18.00	-0.17
t_{17}	8.00	0.50	9.50	-0.25
t_{18}	10.50	0.50	13.50	-0.33
t_{19}	9.00	0.18	29.00	-0.10
t_{23}	10.50	0.20	18.50	-0.14
t_{24}	12.50	0.20	18.00	-0.14
t_{27}	18.00	0.13	29.00	-0.10
t_{29}	18.00	0.13	29.00	-0.10

Table 5.7: Cost optimal structure solution

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters. The output activities are: $A_C = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 20, 22, 25, 28\}$

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	145.00
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.50
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	87.00
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \lvert M_{E,s,r} \rvert}{2}$	55.00
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\lvert E \rvert}$	0.41
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\lvert E \rvert}$	-0.28
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	65.50

5.6.3 Time optimization model with cost constraint

$$x_{Close} = 1 \qquad Close \in A \tag{5.76}$$

$$\sum_{\{i:i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.77}$$

$$a_{E,1,l} = a_{E,1,r} = M_{E,1,l} = M_{E,1,r} = 0 \tag{5.78}$$

$$a_{E,k,l} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.79}$$

$$a_{E,k,r} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.80}$$

$$M_{E,k,l} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i M_{A,i,l} \leq M_{E,j,l} \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.81}$$

$$|M_{E,k,r}| + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i |M_{A,i,r}| \leq |M_{E,j,r}| \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k, i) \in D \quad and \quad (i, j) \in D \tag{5.82}$$

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \leq C \tag{5.83}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \longrightarrow min \tag{5.84}$$

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.25
t_5	0.50	1.00	2.00	-0.25
t_{10}	8.00	∞	12.00	-0.17
t_{11}	2.00	∞	4.00	-0.25
t_{12}	0.50	1.00	2.00	-0.25
t_{13}	2.00	0.20	6.00	-0.13
t_{17}	6.50	0.50	9.50	-0.17
t_{18}	10.50	0.50	13.50	-0.20
t_{19}	9.00	0.14	17.00	-0.08
t_{23}	10.50	0.20	18.50	-0.11
t_{24}	12.50	0.20	18.00	-0.11
t_{27}	17.50	0.14	29.00	-0.08
t_{29}	17.50	0.14	29.00	-0.08

Table 5.8: Time optimal structure with cost constraint solution

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters. The output activities of are: $A_{T,CC} = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 20, 22, 26, 28\}$

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	131.75
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.25
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	65.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \lvert M_{E,s,r} \rvert}{2}$	68.25
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\lvert E \rvert}$	0.40
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\lvert E \rvert}$	-0.17
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	65.75

The value of the cost constraint is 66.00.

5.6.4 Cost optimization model with time constraint

$$x_{Close} = 1 \qquad Close \in A \tag{5.85}$$

$$\sum_{\{i:i \in A\,(i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.86}$$

$$a_{E,1,l} = a_{E,1,r} = M_{E,1,l} = M_{E,1,r} = 0 \tag{5.87}$$

$$a_{E,k,l} + \sum_{\{i:i \in A\,(i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.88}$$

$$a_{E,k,r} + \sum_{\{i:i \in A\,(i,j) \in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.89}$$

$$M_{E,k,l} + \sum_{\{i:i \in A\,(i,j) \in D\}} x_i M_{A,i,l} \leq M_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.90}$$

$$|M_{E,k,r}| + \sum_{\{i:i \in A\,(i,j) \in D\}} x_i |M_{A,i,r}| \leq |M_{E,j,r}| \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.91}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T \tag{5.92}$$

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \longrightarrow min \tag{5.93}$$

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters. The output activities are:$A_{C,TC} = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 21, 22, 25, 28\}$

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	2.00	-0.33
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	2.00	-0.33
t_{13}	2.00	0.20	6.00	-0.20
t_{17}	6.50	0.50	9.50	-0.25
t_{18}	10.50	0.50	13.50	-0.25
t_{19}	9.00	0.14	17.00	-0.11
t_{23}	11.00	0.25	17.00	-0.17
t_{24}	12.50	0.25	18.00	-0.17
t_{27}	18.00	0.14	28.50	-0.11
t_{29}	18.00	0.14	28.50	-0.11

Table 5.9: Cost optimal structure with time constraint solution

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s\in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	131.25
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.25
Sum of the difference of means	$\sum_{\forall s\in E} (a_{E,s,l} - a_{E,s,r})$	61.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s\in E} \frac{M_{E,s,l} + \lvert M_{E,s,r}\rvert}{2}$	53.00
The average of the left hand side tangent	$\sum_{\forall s\in E} \frac{M_{E,s,l}}{\lvert E\rvert}$	0.41
The average of the right hand side tangent	$\sum_{\forall s\in E} \frac{M_{E,s,r}}{\lvert E\rvert}$	-0.25
The sum of the cost mean averages	$\sum_{\forall i\in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	66.25

The value of the time constraint is 131.25.

5.7 Two step concept for fuzzy linear programming with different objective function

Next, we summarize in the following, the logical concept of the new fuzzy optimization model. In this model, we follow the standard logical concept of the project management linear programming model. The linear programming model for CPM method optimize for the difference of the last and the first even time parameters. We will now suppose that the time parameters of the first event - Start node in the process network - are zero so the optimization refer to the last event - End node in the process network - parameters. In this model we will follows the new two step fuzzy optimization

method logic.
For the first step we present the left hand side and the right hand side mean parameters of fuzzy time values. We give four mathematical optimization models for time optimization, cost optimization, time optimization with cost constraint, and cost optimization with time constraint. The objective function we define for the last event mean average. The last event in this structure is noted by E_{29}, in the process network thie last event is generally the End event. This model gives greather values for the sum of the difference of means as the other models except the case of cost optimization. This results more uncertainty fuzzy values. To achieve the more exact fuzzy values in this model we give additional time constarints for the sum of the averages of the mean values.

Time optimization model In the time optimization model we minimize the sum of the average of mean values of the fuzzy time parameters, like so:

$$\frac{a_{E,End,l} + a_{E,End,r}}{2} \longrightarrow min$$

Cost optimization model In the cost optimization model let us minimize the average of the mean values of the fuzzy cost parameters like so:

$$\frac{a_{C,s,l} + a_{C,s,r}}{2} \longrightarrow min$$

The cost optimization model gives the left hand side and the right hand side mean parameters of fuzzy time values as the output of the first step, but there are not time optimal values.

Time optimization model with cost constraint In the time optimization model with cost constraint we will give an additional cost constraint condition for time optimal model.

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \leq C \tag{5.94}$$

Cost optimization model with time constraint In the cost optimization model with time constraint we give an additional time constraint condition for cost optimal model. This time constraint is associated with the objective function of the time optimality model.

$$\frac{a_{E,End,l} + a_{E,End,r}}{2} \leq T \tag{5.95}$$

For this step the mathematical model generates the optimal structure for the CPM problem in the process network. The activities which are in generated in an optimal structure using Step 1. are the input activities of the model applied using Step 2.
Step 2.

For the second step we solve the time optimal mathematical model to get the average of the left hand side and the right hand side tangent parameter values for the activities of the optimal structure, which was generated using Step 1. The second step is associated with the objective function

$$\frac{M_{E,End,l} + |M_{E,End,r}|}{2} \longrightarrow min$$

For this model we give an additional time constraint for the sum of the mean averages for the first step:

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T_a \tag{5.96}$$

and for the sum of the tangent parameters for the second step:

$$\sum_{\forall s \in E} \frac{M_{E,End,l} + |M_{E,End,r}|}{2} \leq T_m \tag{5.97}$$

5.7.1 Time optimization model

Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{5.98}$$

$$\sum_{\{i:i\in A\ (i,j)\in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.99}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.100}$$

$$a_{E,k,l} + \sum_{\{i:i\in A\ (i,j)\in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.101}$$

$$a_{E,k,r} + \sum_{\{i:i\in A\ (i,j)\in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.102}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T_a \tag{5.103}$$

$$\frac{a_{E,End,l} + a_{E,End,r}}{2} \longrightarrow min \tag{5.104}$$

In Step 1. the mathematical model choses the possible alternatives of the optimal structure. For the output alternative set of the optimal structure is denoted by A_T and the set of edges in the optimal structure is denoted by D_T. The set of events E is not changed in Step 1.

$$A_T = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 21, 22, 26, 28\}$$

Now we define a mathematical model for the tangent parameter values for the optimal structure of the process network which contains the alternatives of the A_T set.

Step 2.

$$x_{Close} = 1 \qquad Close \in A_T \tag{5.105}$$

$$\sum_{\{i:i\in A_T\ (i,j)\in D_T\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.106}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.107}$$

$$M_{E,k,l} + \sum_{\{i:i\in A_T\ (i,j)\in D_T\}} x_i M_{A_T,i,l} \leq M_{E,j,l} \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k, i) \in D_T \quad and \quad (i, j) \in D_T \tag{5.108}$$

$$|M_{E,k,r}| + \sum_{\{i:i\in A_T\ (i,j)\in D_T\}} x_i |M_{A_T,i,r}| \leq |M_{E,j,r}| \quad \forall k, j \in E \quad j \neq Start$$

$$where \quad \exists i : (k, i) \in D_T \quad and \quad (i, j) \in D_T \tag{5.109}$$

$$\sum_{\forall s\in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \leq T_m \tag{5.110}$$

$$\frac{M_{E,End,l} + |M_{E,End,r}|}{2} \longrightarrow min \tag{5.111}$$

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters.

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	4.00	-1.00
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	4.00	-1.00
t_{13}	2.00	0.50	8.00	-0.33
t_{17}	8.00	0.50	13.00	-0.25
t_{18}	10.50	0.50	13.50	-0.33
t_{19}	9.00	0.25	19.00	-0.14
t_{23}	11.00	0.25	18.00	-0.17
t_{24}	12.50	0.25	18.00	-0.17
t_{27}	17.50	0.17	28.50	-0.11
t_{29}	17.50	0.17	28.50	-0.11

Table 5.10: Time optimal structure solution

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	137.75
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.00
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	73.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \lvert M_{E,s,r} \rvert}{2}$	44.50
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\lvert E \rvert}$	0.46
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\lvert E \rvert}$	-0.37
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	66.50

5.7.2 Cost optimization model

Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{5.112}$$

$$\sum_{\{i: i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.113}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.114}$$

$$a_{E,k,l} + \sum_{\{i: i \in A\ (i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.115}$$

$$a_{E,k,r} + \sum_{\{i: i \in A\ (i,j) \in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.116}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T_a \tag{5.117}$$

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \longrightarrow min \tag{5.118}$$

In Step 1. the mathematical model choses the alternative set of the optimal structure. For Step 2. let A_C denote the output alternative set of the optimal structure and let D_C denote the set of edges of the optimal structure. The E set of events is not changed in Step 1.
The output alternatives of this method are:

$$A_C = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 20, 22, 25, 28\}$$

Next we define the mathematical model for the tangent parameters used for the structure of the process network which contains the alternatives is chosen by the mean model in Step.1.
Step 2.

$$x_{Close} = 1 \qquad Close \in A_C \tag{5.119}$$

$$\sum_{\{i: i \in A_C\ (i,j) \in D_C\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.120}$$

$$M_{E,1,l} = M_{E,1,r} = 0 \tag{5.121}$$

$$M_{E,k,l} + \sum_{\{i: i \in A_C\ (i,j) \in D_C\}} x_i M_{A_C,i,l} \leq M_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_C \quad and \quad (i,j) \in D_C \tag{5.122}$$

$$|M_{E,k,r}| + \sum_{\{i:i \in A_C\ (i,j) \in D_C\}} x_i |M_{A_C,i,r}| \leq |M_{E,j,r}| \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_C \quad and \quad (i,j) \in D_C \tag{5.123}$$

$$\sum_{\forall s \in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \leq T_m \tag{5.124}$$

$$\frac{M_{E,End,l} + |M_{E,End,r}|}{2} \longrightarrow min \tag{5.125}$$

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	4.50	-1.00
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	4.50	-1.00
t_{13}	2.00	0.50	8.50	-0.33
t_{17}	8.00	0.50	12.00	-0.33
t_{18}	10.50	0.50	14.00	-0.33
t_{19}	9.00	0.25	19.50	-0.14
t_{23}	10.50	0.20	18.50	-0.14
t_{24}	12.50	0.25	18.50	-0.14
t_{27}	18.00	0.13	29.00	-0.10
t_{29}	18.00	0.13	29.00	-0.10

Table 5.11: Cost optimal structure solution

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters.

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	139.75
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.50
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	76.50
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \lvert M_{E,s,r} \rvert}{2}$	48.50
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\lvert E \rvert}$	0.45
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\lvert E \rvert}$	-0.38
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	65.50

5.7.3 Time optimization model with cost constraint

Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{5.126}$$

$$\sum_{\{i: i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.127}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.128}$$

$$a_{E,k,l} + \sum_{\{i: i \in A\ (i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k, j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.129}$$

$$a_{E,k,r} + \sum_{\{i: i \in A\ (i,j) \in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k, j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{5.130}$$

$$\forall i \in A : x_{i,l} = x_{i,r} \tag{5.131}$$

$$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \leq C \tag{5.132}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T_a \tag{5.133}$$

$$\frac{a_{E,End,l} + a_{E,End,r}}{2} \longrightarrow min \tag{5.134}$$

In Step 1. the mathematical model chose the alternatives of the optimal structure. For the Step 2. the output alternative set of the optimal structure let denote $A_{T,CC}$ and let $D_{T,CC}$ denote the set of edges of the optimal structure. The E set of events is not changed in the Step 1.
The output alternatives of this method are:

$$A_{T,CC} = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 20, 22, 26, 28\}$$

Step 2.

$$x_{Close} = 1 \qquad Close \in A_{T,CC} \tag{5.135}$$

$$\sum_{\{i:i \in A_{T,CC}\ (i,j) \in D_{T,CC}\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.136}$$

$$M_{E,1,l} = M_{E,1,r} = 0 \tag{5.137}$$

$$M_{E,k,l} + \sum_{\{i:i \in A_{T,CC}\ (i,j) \in D_{T,CC}\}} x_i M_{A_{T,CC},i,l} \leq M_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_{T,CC} \quad and \quad (i,j) \in D_{T,CC} \tag{5.138}$$

$$|M_{E,k,r}| + \sum_{\{i:i \in A_{T,CC}\ (i,j) \in D_{T,CC}\}} x_i |M_{A_{T,CC},i,r}| \leq |M_{E,j,r}| \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_{T,CC} \quad and \quad (i,j) \in D_{T,CC} \tag{5.139}$$

$$\sum_{\forall s \in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \leq T_m \tag{5.140}$$

$$\frac{M_{E,End,l} + |M_{E,End,r}|}{2} \longrightarrow min \tag{5.141}$$

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters.

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.00	∞	4.00	-0.50
t_5	0.50	1.00	4.00	-1.00
t_{10}	8.00	∞	12.00	-0.25
t_{11}	2.00	∞	4.00	-0.50
t_{12}	0.50	1.00	4.00	-1.00
t_{13}	2.00	0.50	8.00	-0.33
t_{17}	8.00	0.50	12.00	-0.33
t_{18}	10.50	0.50	13.50	-0.33
t_{19}	9.00	0.25	19.00	-0.14
t_{23}	10.50	0.20	18.50	-0.14
t_{24}	12.50	0.25	18.50	-0.14
t_{27}	17.50	0.14	29.00	-0.10
t_{29}	17.50	0.14	29.00	-0.10

Table 5.12: Time optimal structure with cost constraint solution

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	138.00
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.25
Sum of the difference of means	$\sum_{\forall s \in E} (a_{E,s,l} - a_{E,s,r})$	75.00
Sum of the reciprocal of the tangent averages	$\sum_{\forall s \in E} \frac{M_{E,s,l} + \lvert M_{E,s,r} \rvert}{2}$	47.50
The average of the left hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,l}}{\lvert E \rvert}$	0.45
The average of the right hand side tangent	$\sum_{\forall s \in E} \frac{M_{E,s,r}}{\lvert E \rvert}$	-0.38
The sum of the cost mean averages	$\sum_{\forall i \in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	65.75

The value of the cost constraint is 66.00.

5.7.4 Cost optimization model with time constraint

Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{5.142}$$

$$\sum_{\{i:i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.143}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{5.144}$$

$$a_{E,k,l} + \sum_{\{i:i\in A\ (i,j)\in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \qquad and \qquad (i,j) \in D \tag{5.145}$$

$$a_{E,k,r} + \sum_{\{i:i\in A\ (i,j)\in D\}} x_i a_{A,i,r} \leq a_{E,j,r} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D \qquad and \qquad (i,j) \in D \tag{5.146}$$

$$\sum_{\forall s\in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \leq T_a \tag{5.147}$$

$$\sum_{\forall i\in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2} \longrightarrow min \tag{5.148}$$

In Step 1. the mathematical model choses the possible alternatives of the optimal structure. In Step 2. the output alternative set of the optimal structure and let $D_{C,TC}$ denote the set of edges of the optimal structure . The E set of events is not changed in Step 1. The output alternatives of the optimal structure are:

$$A_{C,TC} = \{2, 3, 6, 7, 8, 9, 14, 15, 16, 21, 22, 25, 28\}$$

Step 2.

$$x_{Close} = 1 \qquad Close \in A_{C,TC} \tag{5.149}$$

$$\sum_{\{i:i\in A_{C,TC}\ (i,j)\in D_{C,TC}\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{5.150}$$

$$M_{E,1,l} = M_{E,1,r} = 0 \tag{5.151}$$

$$M_{E,k,l} + \sum_{\{i:i\in A_{C,TC}\ (i,j)\in D_{C,TC}\}} x_i M_{A_{C,TC},i,l} \leq M_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_{C,TC} \qquad and \qquad (i,j) \in D_{C,TC} \tag{5.152}$$

$$|M_{E,k,r}| + \sum_{\{i:i \in A_{C,TC}\ (i,j) \in D_{C,TC}\}} x_i |M_{A_{C,TC},i,r}| \leq |M_{E,j,r}| \quad \forall k,j \in E \quad j \neq Start$$

$$where \qquad \exists i : (k,i) \in D_{C,TC} \qquad and \qquad (i,j) \in D_{C,TC} \tag{5.153}$$

$$\sum_{\forall s \in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \leq T_m \tag{5.154}$$

$$\frac{M_{E,End,l} + |M_{E,End,r}|}{2} \longrightarrow min \tag{5.155}$$

	a_l	m_l	a_r	m_r
t_1	0.00	∞	0.00	∞
t_4	2.10	∞	4.00	-0.50
t_5	0.50	1.00	4.00	-1.00
t_{10}	8.10	∞	12.00	-0.25
t_{11}	2.10	∞	4.00	-0.50
t_{12}	0.50	1.00	4.00	-1.00
t_{13}	2.00	0.50	8.00	-0.33
t_{17}	8.10	0.50	13.00	-0.25
t_{18}	10.60	0.50	13.50	-0.33
t_{19}	9.00	0.25	19.00	-0.14
t_{23}	11.10	0.25	18.00	-0.17
t_{24}	12.60	0.25	18.00	-0.17
t_{27}	18.10	0.14	28.50	-0.11
t_{29}	18.10	0.14	28.50	-0.11

Table 5.13: Cost optimal structure with time constraint solution

Here we summarize the time and cost output parameters of the model. We give in the summary table the mathematical formula and the calculated value of the output parameters.

Parameter	Mathematical formula	Output value
The sum of the mean averages	$\sum_{\forall s\in E} \frac{a_{E,s,l} + a_{E,s,r}}{2}$	138.70
The last event time mean average	$\frac{a_{E,End,l} + a_{E,End,r}}{2}$	23.30
Sum of the difference of means	$\sum_{\forall s\in E} (a_{E,s,l} - a_{E,s,r})$	71.60
Sum of the reciprocal of the tangent averages	$\sum_{\forall s\in E} \frac{M_{E,s,l} + \lvert M_{E,s,r}\rvert}{2}$	45.50
The average of the left hand side tangent	$\sum_{\forall s\in E} \frac{M_{E,s,l}}{\lvert E\rvert}$	0.45
The average of the right hand side tangent	$\sum_{\forall s\in E} \frac{M_{E,s,r}}{\lvert E\rvert}$	-0.37
The sum of the cost mean averages	$\sum_{\forall i\in E} x_i \frac{a_{C_i,l} + a_{C_i,r}}{2}$	66.25

The value of the time constraint is 23.30.

5.8 Concluding remarks

In this chapter, we presented a new concept of fuzzy linear programming to solve four optimization problems for process network optimization, i.e. we defined four mathematical models for the time and cost optimization problem, the time optimization, the cost optimization, the time optimization with cost constraint, and the cost optimization with time constraint. The main idea behind our new concept is to search for not only the optimum but also the sharpest solution. To achieve this goal, we had to minimize the objective function which contains the right hand side and the left hand side gradients of the membership function. In this way the mathematical model approximate the sharpest solution. The CPM method extension is given in [88], and with the help of a transformation it is turned into a process network. So the CPM problem is turned into a process network problem. In the process network the CPM problem is extended with different alternatives. Here an alternative means another resource in the CPM problem. In the CPM graph, only the time is handled i.e. the activity time and the event time parameters. The cost in the CPM method is the cost of the resources. In the process network we have to handle the time and the cost parameters together. We assign fuzzy membership function to the activities, events, and the costs. In the process network, the mathematical model gives one optimal solution of the CPM problem. In our new concept of fuzzy linear programming, we solved four optimization problems and we used a special representation of fuzzy membership function, and its aggregation. The input values of the methods are the fuzzy activities and fuzzy costs represented by mean and tangent values. In the new concept we handled a two step optimization of the tasks. First, we optimize the mean parameters. Second, we optimize the tangent parameters i.e. the sharpness of the fuzzy membership function. The output of the new fuzzy linear programming method is the left hand side mean and tangent and the right hand side mean and tangent values of the event fuzzy membership function. The objective function of the time optimization model is the average of the mean parameters of the events - for the first step - and the average of the tangent parameters of the events for the second step. The linear programming algoritm determines in all four cases the optimal structure of the CPM problem. Finally we defined two other new concept for the fuzzy linear programming. First we defined the one step method. For this method we define every constraint conditions in one step. In this method we optimize the average of the sum of the mean average in the objective function. We give in the model a summary table of the output parameter values of the model. The sum of the difference of the means parameter values shows that this model gives same parameter values as the two step method except the cost optimization. In the cost optimization this aggregate model gives greater interval for the difference of the means of the membership function. The tangent parameters of the solution time membership functions of this method are lower as in the two step model. So this model gives more unprecise fuzzy membership functions as the two step model. This shows the importancy of the step for optimize the tangent parameters in the two step methods. In the project management the standard linear optimization model optimize the different of the last and the first time event parameters. Generally it is supposed that the first event time parameter values are zero. So the standard model optimize the last parameter values. In the second new two step optimization concept we optimize for the last parameters of the fuzzy membership functions. In the summary tables of the output parameters we can see that in this concept the tangent parameters of the time fuzzy membership functions are the same as those in the first two step model, but the different of the mean indicators are more greater as in the first two step model or in the one step model(except in the cost optimization). This results that the solution time fuzzy membership functions are the same precize in the point of view of the left hand side and the right hand side uncertainty, but more unprecise in the point of view that this gives greater mean difference intervals. We show that the strengthness of the conditions of the first two step model are necessary to find the sharpest solution. Hence we show, that between these model solutions the first two step model gives more precise results for the event time membership functions so the first model is a well-defined model and its result provide the better approximation of the sharpest solution.

Chapter 6

Summary

In the Neumann-Morgenstern utility theory the preference is transitive. The non-transitive preferences requires special structures. Fishburn introduce a "non-transitive measurable" utility theory. In this theory Fishburn reject the transitivity of Neumann-Morgenstern utility theory. The preference is represented by a positive part of a $\phi(x, y)$ skew symmetric bilinear(SSB) functional. Fishburn supposed $\phi(x, y) = h(x - y)$, $x \geq y$, where h, is a univariate real-valued function which generates the preference. The properties of the preference structure can be characterized by the properties of h. It is showed that with a particular choice of h, we obtain cyclical preference. In the Chapter 1 we present a general characterization of the non-transitivity of preferences in the Fishburn SSB utility theory. We consider the case in which the X set of the decision outcomes is discrete and the probability measures on X are two-valued. The k-cyclicity of a preference is defined for any positive integer k, and it is shown that a k-cyclic preference exists for every k. The presented univariate generating function is given in concave, convex and ε-linear form.

In the Chapter 2 we present a weighted numerical representation of the lexicographic decision function. We defined a universal preference function and a unary operator. We present the lexicographic decision function with the composition of the universal preference function and a unary operator. This construction is applicable into a general framework, which incorporates the different outranking approaches,the lexicographic decision process and the utility-based decision making models. All of these various methods could be constructed with the composition of the universal preference function and a unary operator. We construct our method, that it can apply for the processing the on-line data. Lastly we examine the connection between the lexicographic decision method and the Arrow paradox.

In the Chapter 3 our goal is to learn the lexicographic decision. From the learning aspects the examination of lexicographic decision making means to learn importance order of the criteria. We learn the importance order of the criteria by decision samples. We suppose that the samples are given by exchange value (EV) vectors which are evaluated by the exchange value evaluation (EVE) function. An EV vector is an arbitrary n-dimensional vector which contains -1, 0 or +1 in the coordinates. The +1 in the i-th coordinate means that we improve the decision alternative concerning the i-th criterion, the -1 means that we deteriorate the decision alternative concerning the i-th criterion, the 0 means that we do not change it concerning the i-th criterion. The EVE function has value +1 on the EV vector if the resulted alternative precedes the original in the lexicographical order, in the opposite case it has value -1. We use two models, in the best case which learn the importance order of the criteria we are allowed to generate the EV vectors, we show a method which uses at most $n\lceil\log_2 n\rceil - 2^{\lceil\log_2 n\rceil} + 1$ EV vectors. In the worst case analysis we deal with the adversarial models. In the adversarial model where the list of EV vectors is generated by an adversary who has the goal to present as long list as possible. We consider different types of adversary.We use a metric on the set of the EV vectors. The distance of two EV vectors is the number of the different components. We show a restriction the strongly 4-distance restriction which forces the adversary to use $O(n^2)$ length sequences.

We deal with decision problems of process network in the following. The Critical Path Method (CPM) in the projectmanagement is an algorithmic approach of scheduling a set of activities. The CPM method use the duration times of the activities and the logical dependencies among the activities. Based on this information the activity network is developed. In Chapter 4 a novel method is presented to extend the problem range of CPM problems. First, the CPM problem is transformed into a process network problem. We show how the process network is generated. We give four different linear optimization model. Time optimal, cost optimal, time optimal with additional cost constraints and cost optimal with additional time constraints mathematical programming models were described. If a given problem can be solved by performing more than one activities or more than one series of activities, then it is called to be the problem of alternatives. Alternatives can be added into the process network and thus CPM problems extended with alternatives can be considered within this framework. In the process network representation of the problem these are considered as alternative arcs. Moreover, it is shown how alternative cases may arises in the structure. An example demonstrates the efficiency of the optimization methods.

In the processes, in many cases we cannot determine exactly the time or cost parameters. Fuzzy theory is widely applied in modeling problems with uncertainty parameters in many areas from mathematical problems throughout engeneering problems, to the area of project management. This applications requires one to handle the uncertainty timing in the processes in the solution of the process scheduling and optimization. We introduce a new solution method of the fuzzy linear optimization problem to calculate the optimal fuzzy time and cost parameters for the process network optimization in the Chapter 5. The main idea behind our new concept is to search for not only the optimum but also the sharpest solution, i.e. we wish to be as close as possible for to the the classical sharp crisp solution. This is why in the objective function we handled the slope of the membership function. Here we present three different models for the fuzzy optimization. In this representation the parameters are those of the linear left and right side functions of the trapezoidal fuzzy membership function for. We present the form of trapezoid fuzzy number with the normalized cut function.

$$L(x) = \left[m_l(x - a_l) + \frac{1}{2} \right] \qquad and \qquad R(x) = \left[m_r(x - a_r) + \frac{1}{2} \right]$$

In this representation of the fuzzy values the a_l and the a_r are called the left hand side and the right hand side mean parameters, and the m_l and m_r are called the left hand side and the right hand side tangent parameters. We summarize in the following, the logical concept of the new fuzzy optimization model. New two step concept for fuzzy linear programming. For the first step we present the left hand side and the right hand side mean parameters of fuzzy time values. We give four mathematical optimization models for time optimization, cost optimization, time optimization with cost constraint, and cost optimization with time constraint. For the second step we solve the time optimal mathematical model to get the average of the left hand side and the right hand side tangent parameter values for the activities of the optimal structure, which was generated using Step 1. In the one step concept for fuzzy linear programming we define the conditons in one step for both the mean and the tangent parameters. In the objective function, we optimalize for the sum of the average of the mean parameter values of the fuzzy event time parameters. In the two step concept for fuzzy linear programming with different objective function, we follow the standard logical concept of the linear programming model in the CPM method, we optimalize the last event time mean and tangent parameters. In this model we will follows the logic of the new two step fuzzy optimization method. We show that the strengthness of the conditions of the first two-step concept are necessary to find the sharpest solution.

Chapter 7

Magyar nyelvű összefoglaló

A Neumann-Morgenstern által kidolgozott hasznosságelméletben az alternatívák rendezése mindig tranzitív. A nem-tranzitív struktúrák megalapozása a klasszikus hasznosságelmélettől eltérő megközelítést igényel. Fishburn volt aki először konstruált nem-tranzitív hasznosságon alapuló eljárást. Ebben az koncepcióban Fishburn elveti a Neumann-Morgenstern hasznosságelmélet néhány tulajdonságát. Bevezet egy $\phi(x, y)$ ferdén szimmetrikus bilineáris funkcionált(SSB) ami preferenciaként értelmezhető. Az így kapott eredmények nem feltétlenül tranzitívak. Fishburn feltételezte, hogy a ferdén szimmetrikus bilineáris funkcionál a következő alakú $\phi(x, y) = h(x - y)$, $x \geq y$, ahol h, egyváltozós szigorúan monoton folytonos függvény, amely mint már említettük preferenciát generál. Fishburn azt mutatta meg, hogy speciális esetben ez az egyváltozós függvény ciklikus preferenciát eredményez. Az azonban nyitott kérdés maradt hogy hogyan jellemezhetők ezek a nem-tranzitív struktúrák a h függvény megválasztásával A dolgozat első fejezetében megadjuk a ciklikus preferenciák általános jellemzését a Fishburn-féle SSB hasznossági modellben. Azt az esetet vizsgáljuk, amikor a döntési kimenetek X halmaza diszkrét és az X-en megadott valószínűségi mértékek kétértékűek. Definiáljuk a preferencia k-ciklikusságát minden pozitív egész k-ra és megmutatjuk, hogy minden k-ra létezik k-ciklikus preferencia h megfelelő választásaival. Megmutatjuk továbbá hogy a k-ciklikusság előállítható ha h konkáv, konvex, és egy lineáris függvényhez tetszőleges közeli ε távolságra van.

A dolgozat második fejezetében a preferencia alapú döntésekkel foglalkozunk (azaz az úgynevezett outranking modellekkel).Megmutatjuk, hogy ha az alternatívák fölötti rendezés lexikografikus akkor ez a lexikografikus döntés megkapható az outranking eljárásokra kidolgozott általános modell segítségével. Azaz a lexikografikus döntés is outranking eljárás. Ismeretes, hogy az minden outranking módszer egy rögzített preferenciafüggvény megfelelő módosításával állítható elő. Megmutatjuk, hogy a lexikografikus döntés megkapható egy speciális módosító függvény és a így kapott preferenciák megfelelő súlyozásával. Dolgozatunkban megmutatjuk, a súlyok megválasztásának eljárását.

A dolgozat harmadik fejezetében a lexikografikus döntési eljárás taníthatósága volt a célunk. Tanulás szempontjából ez a döntési kritériumok fontossági sorrendjének megtanulását jelenti. A kritériumok fontossági sorrendjét döntési minták alapján kaphatjuk meg. A konstrukció során feltesszük, hogy a minták "excehange value (EV)" vektorok által adottak, amelyeket egy "excehange value evaluation (EVE)" függvény értékel ki. Egy EV vektor egy n-dimenziós és a vektor elemei +1, 0, -1 értéket vehetnek fel. A +1 az i-edik koordinátában azt jelenti hogy az i-edik kritérium szerint javítjuk a döntési alternatívát, az i-edik kritériumot tekintve, -1 azt jelenti, rontjuk a döntési alternatívát, az i-edik kritériumot tekintve, 0 azt jelenti nem változtatjuk az alternatívát. Az EVE függvény egy EV vektoron +1 kiértékelési értéket ad ha az EV vektor általi eredmény alternatíva megelőzi az eredeti alternatívát a lexikografikus döntési sorrendben, ellenkező esetben -1-et. Az optimális eset vizsgálatánál két modellt adunk meg és a megefelelő EV vektorokat generáljuk. A dolgozatban megadunk egy algoritmust amely legfeljebb $n\lceil\log_2 n\rceil - 2^{\lceil\log_2 n\rceil} + 1$ EV vektort használ. Legrosszabb eset elemzésekor foglalkoztunk az "ellenfél alapú" modellekkel. Ezeken a modellekben az EV vektorok listája "ellenfél" által generáltak. Az Ellenfélnek az a célja hogy olyan a lehető leghosszabb listát generálja. Különböző tipusú "ellenfeleket" vizsgálhatunk a használt távolságfogalmak alapján (a távolság az EV vektorok halmazán értelmezett). Két EV vektor távolsága például lehet a különböző komponensek száma. Továbbá definiálunk egy megszorítást, a szigorú 4-távolságot, amely arra kényszeríti az "ellenfelet", hogy legfeljebb $O(n^2)$ hosszú EV vektor listát használjon.

Folyamathálózatok döntési problémáival foglalkozunk a továbbiakban. A dolgozat negyedik fejezetében a kritikus út módszere (CPM) a projektmenedzsmentben, tevékenységek egy halmazának ütemezésére bevezetett eljárás. A CPM módszertan a tevékenységek elvégzéséhez szükséges idő mellett azok logikai összefüggéseit is felhasználja és ezek alapján építi fel a tevékenységek ütemterv hálóját. A dolgozat negyedik fejezete egy új módszert mutat be mely a CPM problémák kiterjesztését teszi lehetővé. Megadjuk első lépésben a CPM probléma folyamathálózatokra történő transzformációját. Megmutatjuk, hogyan generálható a folyamathálózatban a CPM problémát megoldó optimális struktúra. Négy különböző modellt vizsgálunk, időoptimális, költségoptimális, időkorlátos költségoptimális, költségkorlátos időoptimális. Ha egy adott problémát egynél több tevékenységgel vagy egynél több tevékenységsorozattal lehet megoldani, ezek a tevékenységek vagy tevékenységsorozatok mint egymás alternatívái jelennek meg. Modellünkben a megadott folyamathálózat további alternatívákkal bővíthető. A folyamathálózat reprezentációban az alternatívák problémája új élek hozzáadásával jelenik meg. Ebben a fejezetben példán mutatjuk meg, hogy az alternatívákkal bővített CPM probléma miként jelenik meg a folyamathálózatokban. A példa jól szemlélteti a kidolgozott módszer hatékonyságát.

A folyamathálózatokban sok esetben nem tudjuk meghatározni az idő vagy költség paraméterek értékét. A fuzzy elmélet egyik legfontosabb fogalma a halmazhoz tartozási függvény, amivel a fent megfogalmazott bizonytalanságot tudjuk modellezni. A halmazhoz tartozási függvény a klasszikus "karakterisztikus függvény" folytonos függvénnyel való helyettesítése. A bizonytalanság pedig a halmazhoz tartozási függvény és a karakterisztikus függvény "távolsága" ami a fuzziság mértéke.

A fuzzy elmélet jól alkalmazható amikor paraméterek bizonytalanságát modellezzük, ami egyben lehetőséget teremt arra, hogy a folyamathálózatok paramétereinek bizonytalanságát matematikailag egzakt módon kezeljük a folyamatok ütemezésének és optimalizálási feladatainak megoldásaiban. A dolgozat ötödik fejezetében ezért új fuzzy lineáris programozási eljárást vezetünk be a folyamathálózatok optimalizálási feladataihoz, mellyel fuzzy időket és költségeket tudunk számítani. A bizonytalanság csökkentésének eléréséhez a "legélesebb" (a karakterisztikus függvényhez legközelebbi) fuzzy megoldást keressük, azaz a legközelebb szeretnénk kerülni a klasszikus éles megoldáshoz. A dolgozatban kifejtett eljárásban a tagsági függvények élessége megjelenik a célfüggvényben. Modellünkben az egyszerűség kedvéért trapezoid fuzzy tagsági függvényt használunk, melyet bal és jobb oldali támasztó egyenesekkel írunk le. A trapezoid fuzzy tagsági függvények speciális reprezentációját használjuk normalizált vágófüggvénnyel(amit a szögletes zárójellel reprezentálunk).

$$L(x) = \left[m_l(x - a_l) + \frac{1}{2} \right] \qquad and \qquad R(x) = \left[m_r(x - a_r) + \frac{1}{2} \right]$$

A fuzzy tagsági függvények ezen reprezentációjában az a_l és az a_r paramétereket bal és jobb oldali közép paramétereknek nevezzük, az m_l és az m_r paramétereket bal és jobb oldali tangens paramétereknek nevezzük. A fuzzyság mértékét az m_l és az d m_r paraméterek reprezentálják. Az alábbiakban az új fuzzy optimalizációs módszer logikai koncepcióját foglaljuk össze. A három új koncepcióban különböző fuzzy optimalizálási célfüggvényt használunk mind a négy optimalizálási modellre, időoptimalizálás, költségoptimalizálás, időoptimalizálás költségkorláttal és költségoptimalizálás időkorláttal. Elsőként megadunk egy új két lépéses fuzzy lineáris optimalizálási koncepciót. Első lépésben a fuzzy eseményidők baloldali és a jobboldali közép paramétereinek optimalizálását végzi az algoritmus. A második lépésben a fuzziságot minimalizáljuk azon optimális struktúrára melyet az első lépéssel generáltunk. Második koncepciónkban megadunk egy egylépéses fuzzy lineáris optimalizálási módszert, melyben egy lépésben definiáljuk a közép és fuzziság paraméterekre a feladatot és a célfüggvényben a közép paraméterek átlagát mimimalizáljuk. Harmadik koncepcióként megadunk egy kétlépéses fuzzy lineáris optimalizálási módszert új célfüggvénnyel. Ebben a koncepcióban a projektmenedzsment ütemezési problémájában alkalmazott standard optimalizálási logikát követtük, annyiban, hogy az utolsó esemény fuzzy idejére optimalizálunk, az új két lépéses módszer fuzzy optimalizálási logikáját követve. Végezetül példán illusztráljuk a módszerek hatékonyságát és megmutatjuk, hogy az új két lépéses módszer adja a "legélesebb" megoldást. A kapott eredmények bizonytalansági paraméterei mutatják, hogy az új kétlépéses fuzzy optimalizálási koncepcióban megfogalmazott feltételek szigorúsága szükséges a "legélesebb" megoldás megtalálásához.

Bibliography

[1] N. Sz. Bahvalov. *A gépi matematika numerikus módszerei.* Műszaki Könyvkiadó, Budapest, 1977.

[2] M. Barany, B. Bertok, Z. Kovacs, F. Friedler, and L. T. Fan. Optimization software for solving vehicle assignment problems to minimize cost and environmental impact of transportation. *Chemical Engeneering*, 21, 2010.

[3] F. A. Behringer. Lexmaxmin in fuzzy multiobjective decision-making. *Optimization*, 21(1):23–49, 1990.

[4] A. Borodin and R. El-Yaniv. *Online computation and competitive analysis.* cambridge university press, 2005.

[5] A. Borodin, J. Kleinberg, P. Raghavan, M. Sudan, and D. P. Williamson. Adversarial queuing theory. *Journal of the ACM (JACM)*, 48(1):13–38, 2001.

[6] D. Bouyssou. Some remarks on the notion of compensation in MCDM. *European Journal of Operational Research*, 26(1):150–160, 1986.

[7] R. H. Burros. Axiomatic analysis of non-transitivity of preference and of indifference. *Theory and Decision*, 5(2):185–204, 1974.

[8] S. Chanas, D. Dubois, and P. Zielinski. On the sure criticality of tasks in activity networks with imprecise durations. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 32(4):393–407, 2002.

[9] S. Chanas and P. Zielinski. Critical path analysis in the network with fuzzy activity times. *Fuzzy sets and systems*, 122(2):195–204, 2001.

[10] S. Chanas and P. Zielinski. The computational complexity of the criticality problems in a network with interval activity times. *European Journal of Operational Research*, 136(3):541–550, 2002.

[11] Sh. Chen and Ch. Hwang. Fuzzy multiple attribute decision making methods. In *Fuzzy Multiple Attribute Decision Making*, pages 289–486. Springer, 1992.

[12] J. Dombi. Membership function as an evaluation. *Fuzzy sets and systems*, 35(1):1–21, 1990.

[13] J. Dombi. *Fuzzy Logic and Soft Computing Advances in Fuzzy Systems - Application and Theory*, chapter A general framework for the utility/based and outranking methods, pages 202–208. World Scientific Publishing, 1995.

[14] J. Dombi. *Principles of Fuzzy Preference Modelling and Decision Making*, chapter A Common Preference Model for Various Decision Models. Academia Press, Gent, Belgium, 2003.

[15] J. Dombi. Pliant arithmetics and pliant arithmetic operations. *Acta Polytechnica Hungarica*, 6(5):19–49, 2009.

[16] J. Dombi and Zs. Gera. Approximation of the continuous nilpotent operator class. *Acta Polytechnica Hungarica*, 2(1):45–58, 2005.

[17] J. Dombi, Cs. Imreh, and N. Vincze. Learning lexicographic orders. *European Journal of Operational Research*, 183(2):748–756, 2007.

[18] J. Dombi and T. Szepe. Arithmetic-based fuzzy control. *Iranian Journal of Fuzzy Systems*, 2017.

[19] J. Dombi and T. Szépe. Pliant control system: implementation. In *Intelligent Systems and Informatics (SISY), 2010 8th International Symposium on*, pages 225–230. IEEE, 2010.

[20] J. Dombi and N. Vincze. Optimization the process network using new concept of fuzzy linear programming. *under review.*

[21] J. Dombi and N. Vincze. Universal characterization of non-transitive preferences. *Mathematical Social Sciences*, 27(1):91–104, 1994.

[22] J. Dombi and N. Vincze. The lexicographic decision function. *Acta Cybernetica*, 17(1):95–106, 2005.

[23] J. Dombi and N. Vincze. Lexikografikus döntések egy általános döntési modellben. *Szigma*, XXXVI.(3-4.):149–161, 2005.

[24] S. Elizabeth and L. Sujatha. Fuzzy critical path problem for project network. *International Journal of Pure and Applied Mathematics*, 85(2):223–240, 2013.

[25] L. T. Fan, Y. Ch. Lin, S. Shafie, B. Bertok, and F. Friedler. Exhaustive identification of feasible pathways of the reaction catalyzed by a catalyst with multiactive sites via a highly effective graph-theoretic algorithm: application to ethylene hydrogenation. *Industrial & Engineering Chemistry Research*, 51(6):2548–2552, 2012.

[26] A. Fiat and G. Woeginger. *Online algorithms: The State of the Art*, chapter Competitive Odds and Ends, pages 385–394. Springer-Verlag Berlin, 1990.

[27] A. Fiat and G. Woeginger. Online algorithms: The state of the art. volume 1442 of lecture notes in computer science. *Springer-Verlag Inc., pub-SV: adr*, 1:17, 1998.

[28] D. P. Filev and R. R. Yager. A generalized defuzzification method via BAD distributions. *International Journal of Intelligent Systems*, 6(7):687–697, 1991.

[29] P. C. Fishburn. Axioms for lexicographic preferences, 1972.

[30] P. C. Fishburn. *Mathematics of Decision Theory.* Mouton, The Hague, Paris, 1972.

[31] P. C Fishburn. Lexicographic orders, utilities and decision rules. *Management science*, 20(11):1442–1471, 1974.

[32] P. C. Fishburn. Noncompensatory preferences. *Synthese*, 33(1):393–403, 1976.

[33] P. C. Fishburn. Lexicographic additive differences. *Journal of Mathematical Psychology*, 21(3):191–218, 1980.

[34] P. C. Fishburn. *The Foundations of Expected Utility.* Reidel, D. Publishing Co., Dordrecht, 1982.

[35] P. C. Fishburn. Nontransitive measurable utility. *Journal of Mathematical Psychology*, 26(1):31–67, 1982.

[36] P. C. Fishburn. SSB utility theory: An economic perspective. *Mathematical Social Sciences*, 8(1):63–94, 1984.

[37] P. C. Fishburn. SSB utility theory and decision-making under uncertainty. *Mathematical social sciences*, 8(3):253–285, 1984.

[38] P. C. Fishburn. *Non-linear preference and utility theory.* Wheatsheaf Books Ltd., Brighton, 1988.

[39] Ph. Fortemps and M. Pirlot. Conjoint axiomatization of min, discrimin and leximin. *Fuzzy Sets and Systems*, 148(2):211–229, 2004.

[40] F. Friedler, K. Tarjan, Y. W. Huang, and L. T. Fan. Combinatorial algorithms for process synthesis. *Computers & chemical engineering*, 16:S313–S320, 1992.

[41] F. Friedler, K. Tarjan, Y. W. Huang, and L. T. Fan. Graph-theoretic approach to process synthesis: axioms and theorems. *Chemical Engineering Science*, 47(8):1973–1988, 1992.

[42] A. Frisch, B. Hnich, Z. Kiziltan, I. Miguel, and T. Walsh. Global constraints for lexicographic orderings. In *International Conference on Principles and Practice of Constraint Programming*, pages 93–108. Springer, 2002.

[43] R. Fullér and R. Mesiar. Special issue on fuzzy arithmetic. *Fuzzy Sets and System*, 91(2), 1997.

[44] J. C. Garcia-Ojeda, B. Bertok, and F. Friedler. Planning evacuation routes with the p-graph framework. *Chemical Engineering*, 29, 2012.

[45] I. Gazdik. Fuzzy-network planning-FNET. *IEEE Transactions on Reliability*, 32(3):304–313, 1983.

[46] R. W. Hamming. Error detecting and error correcting codes. *Bell Labs Technical Journal*, 29(2):147–160, 1950.

[47] Ch. L. Hwang and M. J. Lin. *Group Decision Making under Multiple Criteria. Methods and Applicatoins.* Springer, 1987.

[48] E. Jacquet-Lagreze. Binary preference indices: A new look on multicriteria aggregation procedures. *European Journal of Operational Research*, 10(1):26–32, 1982.

[49] E. Jacquet-Lagreze. Interactive assessment of preferences using holistic judgments the prefcalc system. In *Readings in multiple criteria decision aid*, pages 335–350. Springer, 1990.

[50] E. Jacquet-Lagreze. An application of the uta discriminant model for the evaluation of R & D projects. In *Advances in multicriteria analysis*, pages 203–211. Springer, 1995.

[51] E. Jacquet-Lagreze, R. Meziani, and R. Slowinski. MOLP with an interactive assessment of a piecewise linear utility function. *European Journal of Operational Research*, 31(3):350–357, 1987.

[52] E. Jacquet-Lagreze and M. F. Shakun. Decision support systems for semi-structured buying decisions. *European Journal of Operational Research*, 16(1):48–58, 1984.

[53] E. Jacquet-Lagreze and J. Siskos. Assessing a set of additive utility functions for multicriteria decision-making, the UTA method. *European journal of operational research*, 10(2):151–164, 1982.

[54] E. Jacquet-Lagreze and Y. Siskos. Preference disaggregation: 20 years of MCDA experience. *European Journal of Operational Research*, 130(2):233–245, 2001.

[55] K. Kalauz, Z. Süle, B. Bertok, F. Friedler, and L. T. Fan. Extending process-network synthesis algorithms with time bounds for supply network design. *Chemical Engineering*, 29, 2012.

[56] D. Kelsey. The structure of social decision functions. *Mathematical Social Sciences*, 8(3):241–252, 1984.

[57] J. Klemess, F. Friedler, I. Bulatov, and P. Varbanov. *Sustainability in the process industry: integration and optimization (Green Manufacturing & Systems Engineering).* McGraw-Hill Professional, 2010.

[58] D. Knuth. Searching and sorting, the art of computer programming, 1973.

[59] Z. Kovács, Zs. Ercsey, F. Friedler, and L. T. Fan. Exact super-structure for the synthesis of separation-networks with multiple feed-streams and sharp separators. *Computers & Chemical Engineering*, 23:1007–1010, 1999.

[60] Z. Kovács, Zs. Ercsey, F. Friedler, and L. T. Fan. Separation-network synthesis: global optimum through rigorous super-structure. *Computers & Chemical Engineering*, 24(8):1881–1900, 2000.

[61] H. Li, J. Zhang, L. Ren, and Zh. Shi. Scheduling optimization of construction engineering based on ant colony optimized hybrid genetic algorithm. *JNW*, 8(6):1411–1416, 2013.

[62] Zh. Li, Y. Liu, and G. Yang. A new probability model for insuring critical path problem with heuristic algorithm. *Neurocomputing*, 148:129–135, 2015.

[63] Zh. Liu, L. Xiao, and J. Tian. An activity-list-based nested partitions algorithm for resource-constrained project scheduling. *International Journal of Production Research*, 54(16):4744–4758, 2016.

[64] R. D. Luce. Lexicographic tradeoff structures. *Theory and Decision*, 9(2):187–193, 1978.

[65] K. U. Madhuri, B. P. Saradhi, and N. R. Shankar. Fuzzy linear programming model for critical path analysis. *Int. J. Contemp. Math. Sciences*, 8(2):93–116, 2013.

[66] T. Marchant. Towards a theory of MCDM: stepping away from social choice theory. *Mathematical Social Sciences*, 45(3):343–363, 2003.

[67] K. O. May. Intransitivity, utility, and the aggregation of preference patterns. *Econometrica: Journal of the Econometric Society*, pages 1–13, 1954.

[68] M. Mazlum and A. F. Güneri. CPM, PERT and project management with fuzzy logic technique and implementation on a business. *Procedia-Social and Behavioral Sciences*, 210:348–357, 2015.

[69] C. S. McCahon and E. S. Lee. Project network analysis with fuzzy activity times. *Computers & Mathematics with applications*, 15(10):829–838, 1988.

[70] K. S. Miller. *An introduction to the calculus of finite differences and difference equations.* Holt New York, 1960.

[71] H. Moshkovich, A. Mechitov, and D. Olson. *Multiple Criteria Decision Analysis*, chapter Verbal Decision Analysis, pages 609–634. Springer-Verlag Berlin, 2005.

[72] H. T Nguyen and M. Sugeno. *Fuzzy systems: modeling and control*, volume 2. Springer Science & Business Media, 2012.

[73] D. L. Olson. *Decision aids for selection problems.* Springer Science & Business Media, 1996.

[74] Ch. R. Plott, J. T. Little, and R. P. Parks. Individual choice when objects have" ordinal" properties. *The Review of Economic Studies*, 42(3):403–413, 1975.

[75] T. Rapcsák. Többszempontú döntési problémák, AHP módszertan. MTA SZTAKI, 2003.

[76] Sh. Roychowdhury and W. Pedrycz. A survey of defuzzification strategies. *International Journal of intelligent systems*, 16(6):679–695, 2001.

[77] N. R. Shankar, P. Ph. B. Rao, S. Siresha, and K. U. Madhuri. Critical path method in a project network using ant colony optimization. *International Journal of Computational Intelligence Research*, 7(1):7–16, 2011.

[78] N. R. Shankar and B. P. Saradhi. Fuzzy critical path method in interval-valued activity networks. *Int. J. Pure Appl. Sci. Technol*, 3(2):72–79, 2011.

[79] N. R. Shankar, V. Sireesha, K. S. Rao, and N. Vani. Fuzzy critical path method based on metric distance ranking of fuzzy numbers. *International Journal of Mathematical Analysis*, 4(20):995–1006, 2010.

[80] N. R. Shankar, V. Sireesha, and P. Ph. B. Rao. An analytical method for finding critical path in a fuzzy project network. *International Journal of contemporary mathematical sciences*, 5(20):953–962, 2010.

[81] T. Solymosi. *Adat, modell elemzés*, chapter Ordinális mérési skálák lexikografikus aggregációja, pages 119–126. Aula, Budapest, 2001.

[82] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis, R. Bartels, W. Gautschi, and C. Witzgall, translators.* Springer-Verlag, New York, Berlin, Heidelberg, 1980.

[83] K. Sunita and B. Snigdha. CPM analysis of Rolai-Rinjlai road construction. *Research Journal of Mathematical and Statistical Sciences*, 1(2):7–15, 2013.

[84] J. Temesi. *A döntéselmélet alapjai.* Aula, Budapest, 2002.

[85] J. Tick. P-graph-based workflow modelling. *Acta Polytechnica Hungarica*, 4(1):75–88, 2007.

[86] J. Tick. Fuzzy extension to p-graph based workflow models. In *Computational Cybernetics, 2009. ICCC 2009. IEEE International Conference on*, pages 109–112. IEEE, 2009.

[87] J. Tick, Cs. Imreh, and Z. Kovács. Business process modeling and the robust PNS problem. *Acta Polytechnica Hungarica*, 10(6):193–204, 2013.

[88] N. Vincze, Zs. Ercsey, T. Kovács, J. Tick, and Z. Kovács. Process network solution of extended CPM problems with alternatives. *Acta Polytechnica Hungarica*, 13(3), 2016.

[89] R. R Yager and D. P. Filev. SLIDE: A simple adaptive defuzzification method. *IEEE Transactions on fuzzy systems*, 1(1):69, 1993.

[90] Ch. Yun, T. Y. Kim, T. Zhang, Y. Kim, S. Y. Lee, S. Park, F. Friedler, and B. Bertok. Determination of the thermodynamically dominant metabolic pathways. *Industrial & Engineering Chemistry Research*, 52(1):222–229, 2012.