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

1 INTRODUCTION

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 [14]introduced the 'non-transitive measurable utility' theory. In this utility theory, the preferences on pairs of lotteries or risky decisions are represented by the positive part of the $\phi(x, y)$ skew-symmetric bilinear (SSB) functional [15][16][17]. Fishburn supposed that $\phi(x, y) = h(x-y),\ x \geq y$, i.e. the value of the SSB functional is given with the h univariate real-valued function. Fishburn gave five lotteries as example, and presented a univariate h function with which we obtain a preference cycle. In the Chapter 1 we give a generalization of the cyclicity given by Fishburn, and to give a class of the h univariate function for the characterization of the generalized cyclical preferences.
In the Chapter 2 we examine the lexicographic decision method [13]. 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 [34] models by Saaty were the most important models developed. This applications generate the requirement for a general representation for this decision making models. Dombi [4][5] gave a framework for the utility based and outranking decision methods, using a general preference function. In the Chapter 2 we 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 the-

ory 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 [22][23][24][25][26][27][28] and the general decision frameworks, uses the different parameters and parameter values. It is necessary 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. In the Chapter 3 we 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. [18][19] 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 process network, [1][12][21][29][30] 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. In the Chapter 5 we give a new concept of fuzzy linear optimization in order to solve the alternative extended CPM problem in a process network.

2 Results of the dissertation

2.1 Universal characterization of non-transitive preferences

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.The purpose of Chapter 1 is to explore and characterize the non-transitivity of preferences in the Fishburn "non-transitive measurable utility" theory. We consider the case in which the decision outcomes are integers and the probability distributions 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. We represent preferences with univariate functions and give one class of k-cyclic preference function. Then, different preference functions were given in a concave, convex and ε-linear form. The following results can be found in the article [9].

Results

We will define the k-cyclicity of any positive integer k.

Definition 1. *Let $X = \{j, j+1, ..., n\}$ 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 $j \leq m, m+rk, m+i \leq n, k < n-j$.

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.

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 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.

We can get the h univariate function in the form $h(m) = v(m)g(m)$, where g(m) is the solution of the functional equation

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

for

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

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

$$v(i) < v(m)v(m{+}i) \quad if \quad i \neq rk v(i) > v(m)v(m{+}i \quad if \quad i = rk \tag{2}$$

where $r \in N, r \le \left[\frac{n}{2k}\right], j \le m, m+i, rk \le n$
The main results may be summarized in the following:

Theorem 1. *For every $k, n \in N$, where $2k \le N$, 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) so that $g(m)$ is positive on $\{1, 2, ..., n\}$ and $v(m)$ is a solution of (2) and $h(m)$ may be convex, concave and ε-linear.*

It follows from the theory that for every positive integer k there exists a k-cyclic preference relation, and for every k we can characterize it by its preference function. We see that for fixed k the preference functions of k-cyclic preference relations may be given in several forms, taken from different classes of functions.

2.2 Lexicographic decision function construction

In multicriteria decision making the idea of the lexicographic decision consists of a hierarchy or ordered set of attributes or criteria [13][35]. 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. 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". Here the lexicographic decision process is presented in a unified way. We construct a lexicographic decision function using a universal preference function and a unary function. The following results can be found in the articles [10][11].

Results

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, the following.

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

The main result of this study is the following theorem:

Theorem 2. *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,* $0 \leq x_{ij} \leq 1$.
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:*

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

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:

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

This is the main idea behind the preference construction of the PROMETHEE, and ELECTRE [33][36] method. 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 construction is applicable into a general framework, which incorporates the different outranking approaches, the lexicographic decision process and the utility-based decision making models.

2.3 Learning lexicographic orders

The lexicographic decision model is one of the simplest. In the mid-70's Fishburn 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 . 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.
The main goal is to learn the order of criteria of lexicographic decision using various reasonable assumptions. We give a sample evaluation and an oracle based algorithm. In the worst case analysis we deal with the adversarial models. We show that if the distances of the samples are less than 4, then it is not learnable, but 4-distance samples are polynomial learnable. The following results can be found in the article [6].

Results

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.

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.

Theorem 3. *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.*

In the following, 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

The total number of the used EV vectors is at most
$\lceil \log_2 n \rceil + \lceil \log_2(n-1) \rceil + ... + \lceil \log_2 2 \rceil = n \lceil \log_2 n \rceil - 2^{\lceil \log_2 n \rceil} + 1.$

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. 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.
In the following, 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. 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.
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 4. *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 or-

der.
For arbitrary n there exist strongly 4-distance restricted sequences which contain enough information to determine the importance order.

We show a restriction, the strongly 4-distance restriction, which forces the adversary to use $O(n^2)$ length sequences.

2.4 Process network solution of extended CPM problems with alternatives

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 order to solve CPM problems with the help of process network methodology, the two terminologies have to be mapped. A given CPM graph may be mapped into a process network. 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. The axioms of process networks 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. Time optimal, cost optimal, time optimal with additional cost constraints and cost optimal with additional time constraints mathematical programming models are described. 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 following results can be found in the article [39].

Results

We give the mathematical programming model for the CPM problem, extended with alternatives.
The $G(A, E, D)$ graph has the following properties: $A \cap E = \emptyset$, $D \subseteq (A \times E) \cup (E \times A)$.

$A = \{i \in N\}$ activities
$E = \{j \in N\}$ events
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
Since the end point of the CPM graph has two different preceeding alternatives, in the process network an additional technical operating unit called Close has to be inserted.

$$x_{Close} = 1 \qquad Close \in A \tag{3}$$

$$\sum_{\{i:i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{4}$$

Line (3) refers to the fact that the project has to be finished; line (4) 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 mathematical programming model of the time optimal project plan for the CPM problem, extended with alternatives is given below.

$$x_{Close} = 1 \qquad Close \in A \tag{5}$$

$$\sum_{\{i:i \in A\,(i,j) \in D\}} x_i = 1 \qquad \forall j \in E \quad and \quad j \neq Start \tag{6}$$

$$t_{Start} = 0 \qquad Start \in E \tag{7}$$

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

$$where \quad \exists i : (k, i) \quad and \quad (i, j) \in D \tag{8}$$

$$t_{End} \longrightarrow min \tag{9}$$

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. Cost optimal, time optimal with additional cost constraints and cost optimal with additional time constraints mathematical programming models are also described.

2.5 Optimization the process network using new concept of fuzzy linear programming

Fuzzy theory [20] 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 [2]. A widely accepted method used in the CPM method to calculate the optimal event parameters is the linear optimization model. For the process network [31][32][37][38][40] optimization in the Chapter 5 we introduced a new solution method of the fuzzy LP problem to calculate the optimal fuzzy time and cost parameters. We searched for the sharpest solution i.e. we wish to be as close as possible for to the the classical sharp solution. This is why in the objective function we handled the slope of the membership function [3]. Here we present three different models for the fuzzy optimization. We apply fuzzy activity times in the process network solution of extended CPM problems with alternatives. We use a special new parametrization for the aggregation of the left hand side and the right hand side linear functions of trapezoid fuzzy numbers [7]. In the 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. The following results can be found in the article [8].

Results

For the fuzzy activity time T_i, let the left hand side (l)and the right hand side (r) 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, let the left hand side (l)and the right hand side (r) 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 left hand side (l)and the right hand side (r) 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:

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

The time optimization model is:
Step 1.

$$x_{Close} = 1 \qquad Close \in A \tag{10}$$

$$\sum_{\{i:i \in A\ (i,j) \in D\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{11}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{12}$$

$$a_{E,k,l} + \sum_{\{i:i \in A\ (i,j) \in D\}} x_i a_{A,i,l} \leq a_{E,j,l} \quad \forall k,j \in E \quad j \neq Start$$
$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{13}$$

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

$$where \quad \exists i : (k,i) \in D \quad and \quad (i,j) \in D \tag{14}$$

$$\sum_{\forall s \in E} \frac{a_{E,s,l} + a_{E,s,r}}{2} \longrightarrow min \tag{15}$$

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.
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.

$$x_{Close} = 1 \qquad Close \in A_T \tag{16}$$

$$\sum_{\{i:i \in A_T \ (i,j) \in D_T\}} x_i = 1 \qquad \forall j \in E \qquad j \neq Start \tag{17}$$

$$a_{E,1,l} = a_{E,1,r} = 0 \tag{18}$$

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

$$\forall k,j \in E \quad j \neq Start$$

$$where \quad \exists i : (k,i) \in D_T \quad and \quad (i,j) \in D_T \tag{19}$$

$$|M_{E,k,r}| + \sum_{\{i: i \in A_T \ (i,j) \in D_T\}} x_i |M_{A_T,i,r}| \leq |M_{E,j,r}|$$
$$\forall k, j \in E \quad j \neq Start$$
$$where \quad \exists i : (k,i) \in D_T \quad and \quad (i,j) \in D_T \tag{20}$$

$$\sum_{\forall s \in E} \frac{M_{E,s,l} + |M_{E,s,r}|}{2} \longrightarrow min \tag{21}$$

Here we present three different models for the fuzzy optimization. We apply fuzzy activity times in the process network solution of extended CPM problems with alternatives. We use a special new parametrization for the aggregation of the left hand side and the right hand side linear functions of trapezoid fuzzy numbers. We searched for the sharpest solution i.e. we wish to be as close as possible for to the the classical sharp solution. This is why in the objective function we handled the slope of the membership function.

References

[1] 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.

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

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

[4] 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.

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

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

[7] J. Dombi and T. Szépe. Arithmetic-based fuzzy control. Iranian Journal of Fuzzy Systems,14(4):51-66, 2017.

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

[9] J. Dombi and N. Vincze. Universal characterization of non-transitive preferences. Mathematical Social Sciences, 27(1):91-104, 1994.

[10] J. Dombi and N. Vincze. The lexicographic decision function. Acta Cybernetica, 17(1):95-106,2005.

[11] J. Dombi and N. Vincze. Lexikografikus döntések egy általános döntési modellben. Szigma, XXXVI.(3-4.):149-161, 2005.

[12] 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.

[13] P. C Fishburn. Lexicographic orders, utilities and decision rules. Management science, 20(11):1442-1471, 1974.

[14] P. C. Fishburn. Nontransitive measurable utility. Journal of Mathematical Psychology, 26(1):31-67, 1982.

[15] P. C. Fishburn. SSB utility theory: An economic perspective. Mathematical Social Sciences, 8(1):63-94, 1984.

[16] P. C. Fishburn. SSB utility theory and decision-making under uncertainty. Mathematical social sciences, 8(3):253-285, 1984.

[17] P. C. Fishburn. Non-linear preference and utility theory. Wheatsheaf Books Ltd., Brighton,1988.

[18] F. Friedler, K. Tarjan, Y. W. Huang, and L. T. Fan. Combinatorial algorithms for process synthesis. Computers & chemical engineering, 16:S313-S320, 1992.

[19] 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.

[20] R. Fullér and R. Mesiar. Special issue on fuzzy arithmetic. Fuzzy Sets and System, 91(2), 1997.

[21] J. C. Garcia-Ojeda, B. Bertok, and F. Friedler. Planning evacuation routes with the p-graph framework. Chemical Engineering, 29, 2012.

[22] E. Jacquet-Lagreze. Binary preference indices: A new look on multicriteria aggregation procedures. European Journal of Operational Research, 10(1):26-32, 1982.

[23] 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.

[24] 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.

[25] 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.

[26] 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.

[27] 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.

[28] E. Jacquet-Lagreze and Y. Siskos. Preference disaggregation: 20 years of MCDA experience. European Journal of Operational Research, 130(2):233-245, 2001.

[29] 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.

[30] 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.

[31] Z. Kovács, Zs. Ercsey, F. Friedler, and L. T. Fan. Exact superstructure for the synthesis of separation-networks with multiple feedstreams and sharp separators. Computers & Chemical Engineering, 23:1007-1010, 1999.

[32] 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.

[33] D. L. Olson. Decision aids for selection problems. Springer Science & Business Media, 1996.

[34] T. Rapcsák. Többszempontú döntési problémák, AHP módszertan. MTA SZTAKI, 2003.

[35] T. Solymosi. Adat, modell elemzés, chapter Ordinális mérési skálák lexikografikus aggregációja, pages 119-126. Aula, Budapest, 2001.

[36] J. Temesi. A döntéselmélet alapjai. Aula, Budapest, 2002.

[37] J. Tick. P-graph-based workflow modelling. Acta Polytechnica Hungarica, 4(1):75-88, 2007.

[38] J. Tick, Cs. Imreh, and Z. Kovács. Business process modeling and the robust PNS problem. Acta Polytechnica Hungarica, 10(6):193-204, 2013.

[39] 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),101-117, 2016.

[40] 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.