MULTIPLE CRITERIA DECISION MAKING
Vol. 13 2018

Tadeusz Trzaskalik*

APPLICATION OF MULTIOBJECTIVE DYNAMIC PROGRAMMING TO THE ALLOCATION AND RELIABILITY PROBLEM

DOI: 10.22367/mcdm.2018.13.09

Abstract

The paper deals with a model of the allocation and reliability problem. This static problem, presented as a multistage decision process, can be solved using multiobjective dynamic programming. The goal of this paper is to formulate the allocation and reliability problem as a multistage decision process, to find the set of all its efficient solutions, to use the weighted sum method for multistage and single-stage criteria, as well as to perform sensitivity analysis.

Keywords: multiobjective dynamic programming, allocation and reliability problem, efficient solutions, scalarization methods.

1 Introduction

Multiple objective dynamic programming (MODP) deals with multistage decision processes, in which multiple objectives are taken into consideration. The term MODP covers models of tasks which allow to solve various problems such as: the multiple criteria knapsack problem (Klamroth, Wiecek, 2000), the problem of space heating under a time-varying price of electricity (Hämäläinen, Mäntysaari, 2002), the supplier selection-order allocation problem (Mafakheri et al., 2011), or the location-routing model for relief logistic planning under uncertainty on demand, travel time, and cost parameters (Bozorgi-Amiri, Khorsi, 2016). Those problems are usually of dynamic character. MODP methods are used to analyse multistage decision processes in which a given (usually finite) period is divided into a fixed number of stages. Dynamic programming is also often used to model appropriately formulated static problems. This is also the

* University of Economics in Katowice, Faculty of Informatics and Communication, Department of Operations Research, Katowice, Poland, e-mail: ttrzaska@ue.katowice.pl, ORCID: 0000-0001-5581-0922.

case for the mathematical economics problem discussed in this paper, namely the problem of allocation and reliability (A&R).

Multicriteria evaluation of a multistage process is performed using a vector criteria function whose multistage components are certain compositions of single-stage evaluations. These components have to be separable and monotone scalar functions (Mine, Fukushima, 1979; Mitten, 1964; Trzaskalik, 1990; Abdelaziz et al., 2018; Chen, Fu, 2005), such as additive or multiplicative compositions.

In multicriteria problems, because of the conflicting nature of the objectives, a dominating solution – that is, a solution whose all multistage components admit "best" values simultaneously – usually does not exist. As vector optimal solutions we take those solutions whose multistage evaluations are not dominated. It is not possible (in the criteria space) to improve the value of any multistage criterion without worsening the value of at least one of the remaining criteria.

The basic method of solving multicriteria problems consists in searching for non-dominated solutions (in the criteria space) and for the corresponding efficient solutions (in the decision space). This is the case also for MODP problems. Often, however, finding all non-dominated solutions is difficult calculation-wise, and the set obtained can be very large. For that reason, finding this set is of little direct help to the decision maker in making the final decision. Therefore, analogously to other multicriteria problems, various scalarization methods can be used, which allow (on the basis of additional preferences of the DM) to find a solution taking into account the DM's preferences as a single-criteria optimization problem. It is generally accepted that the solution obtained using a scalarization method should be an efficient solution.

The scalarization method used in this paper is the method of weighted sum of multistage criteria. It can be proven that to each non-negative vector of coefficients there corresponds an efficient realization (Trzaskalik, 1993). In multicriteria dynamic optimization a new possibility (as compared with static vector optimization) occurs: the DM can express his/her preferences by specifying the preferred relations between stage criteria. In the case of a bicriteria problem it is also possible to perform an effective sensitivity analysis.

As opposed to many other optimization problems, such as linear programming problems, no standard formulation of the dynamic programming model exists. Various problems are mutually related by their solving method, which uses optimality equations, constructed on the basis of the optimality principle (Bellman, 1957) and its vector counterpart (Trzaskalik, 1998). In this paper we will use the standard description of a multistage, multicriteria decision process (Trzaskalik, 1998; 2015).

The allocation problem is one of the static problems which can be solved by means of dynamic programming methods (Bellman, 1957; Nowak and Trzaskalik, 2014). The A&R problem, discussed in the present paper, can be described as follows: Given is a system consisting of T modules and a certain amount of resource R. The profits from the operation of the system and its reliability are related to the amount of the resource allocated to the individual modules. The function expressing the profits resulting from the system's operation is the sum of the profits from the operations of the individual modules, while the function expressing the reliability of the entire system is a multiplicative function. The allocation of the resource for the operation of the individual modules should be planned so as to maximize both the profits resulting from the operation of the entire system and its reliability.

The goal of the present paper is to formulate the A&R problem as a multistage decision process, to find the set of all its efficient solutions, to use the weighted sum method for multistage and single-stage criteria, and to perform sensitivity analysis.

The paper consists of five sections. In Section 2, the A&R problem is presented as a multistage decision process. A discrete problem illustrating this problem is also presented, together with the graph of this process. Section 3 shows a possible application of optimality equations and Bellman's vector optimality principle to finding the complete set of non-dominated solutions in the criteria space and of efficient solutions in the decision space. In Section 4, the problem of applying the weighted sum method is discussed and a sensitivity analysis of the problem is performed. Conclusions end the paper.

2 The A&R problem as a multistage decision process

To present the problem in question as a discrete multistage decision process, one should determine the number of stages, the sets of admissible states and decisions, the transfer function (which describes the transformations of the system in consecutive stages), and the method of evaluating the process.

The A&R problem, presented in the previous section, can be formulated as a multistage decision process as follows. The number of stages is determined by the number of modules, that is, T. The allocation of the resource is performed consecutively for the individual modules: in stage 1 we allocate resources for the operation of module 1, in stage 2 – for module 2, etc., and finally in stage T we allocate resources for the operation of module T. The process state y_t at the beginning of stage t ($t \in 1,...,T$) is the amount of the resource available after the allocation in the previous stages had been performed. The set of all admissible states at the beginning of stage t is denoted by Y_t. Decision x_t at stage t consists in the allocation of the entire remaining resource or its part for the operation of module t. The set of all admissible decisions for stage t, if at the beginning of this stage the process was in state y_t, is denoted by $X_t(y_t)$. The pair consisting of state y_t and the corresponding admissible decision x_t is the stage realization of the process, denoted by $d_t = (y_t,x_t)$.

The transformation of the system from state $y_t \in Y_t$ to state $y_{t+1} \in Y_{t+1}$, when the decision $x_t \in X_t(y_t)$ is made, is described by the transfer function of the form:

$$y_{t+1} = \Omega_t(y_t, x_t) = y_t - x_t \tag{1}$$

The sequence of admissible states and decisions of the process such that:

$$y_1 \in Y_1,\ x_1 \in X_1(y_1),\ y_2 = \Omega_1(y_1, x_1),\ \ldots,\ y_T = \Omega_t(y_{T-1}, x_{T-1}),\ x_T \in X_T(y_T) \tag{2}$$

is an admissible realization of the process, denoted by d. The set of all admissible realizations of the process is denoted by D.

The evaluation of the operation of the individual modules is described by the stage profit functions $F_t^1(y_t,x_t)$ and stage reliability functions $F_t^2(y_t,x_t)$ for $t = 1,...,T$. The evaluation of the operation of the entire system is described by the vector criterion function. The first component of this function describes the profits from the operation of the system; it is an additive function of the form

$$F^1(d) = \Sigma_{t=1}^{T} F^1(d_t), \tag{3}$$

while its second component describes the reliability of the system's operation, is multiplicative, and of the form

$$F^2(d) = \Pi_{t=1}^{T} F^1(d_t) \tag{4}$$

The vector criterion function which describes the operation of the system is of the form

$$F(d) = [F^1(d), F^2(d)]' \tag{5}$$

To illustrate the type of the process discussed we consider a simple system consisting of three modules. Six units of the resource are available. The profits from the individual modules and their reliability depending on the amount of the resource are shown in Table 1.

Table 1: Values of the stage criteria (dummy data)

	Module 1		Module 2		Module 3	
Amount alocated	**Profit**	**Reliability**	**Profit**	**Reliability**	**Profit**	**Reliability**
0	0	0.9	0	0.9	0	0.9
1	1.2	0.97	3	0.94	2.8	0.96
2	2	0.991	4.8	0.964	4.5	0.984
3	2.7	0.9973	5.5	0.9784	6.5	0.9936
4	3.3	0.9992	6.8	0.987	7.8	0.9974
5	3.7	0.9998	7.9	0.9922	9.0	0.999
6	4	0.9999	8.5	0.9953	10	0.9994

We will determine the sets of admissible states of this process at the beginning of the consecutive stages. The initial state is given as 6, that is, $Y_1 = \{6\}$. At the beginning of stage 2 the process can be in state 0 (if the entire remaining resource is allocated to module 1), in state 1 (if module 1 is allocated five units), or else in one of the consecutive states 2, 3, 4, 5 or 6, which are interpreted analogously to states 0 and 1.

At the beginning of stage 3 the process can be in state 0 (if the entire resource had been allocated previously to modules 1 and 2), in state 1 (if modules 1 and 2 had been allocated five units), or else in one of the remaining states 2, 3, 4, 5, 6. Since we plan to allocate the entire resource, the final state is given as 0. We obtain the following sets of admissible states:

$Y_1 = \{6\}$ $Y_2 = (0,1,2,3,4,5,6\}$ $Y_3 = \{0,1,2,3,4,5,6\}$ $y_4 = \{0\}$

Now we will deal with the sets of admissible decisions for the consecutive admissible states. In the first stage, having six units at our disposal, we can either allocate no resource for the realization of module 1 allocate 1, 2, 3, 4, 5, or 6 units. Hence,

$$X_1\{6\} = \{0, 1, 2, 3, 4, 5, 6\}.$$

Analogously, we determine the sets of admissible decisions for the consecutive admissible states of the second stage. We obtain:

$X_2(0) = \{0\}$ $X_2(1) = \{0, 1\}$ $X_2(2) = \{0, 1, 2\}$ $X_2(3) = \{0, 1, 2, 3\}$
$X_2(4) = \{0, 1, 2, 3, 4\}$ $X_2(5) = \{0, 1, 2, 3, 4, 5\}$ $X_2(6) = \{0, 1, 2, 3, 4, 5, 6\}$

Since we have to use up the entire resource, and a certain amount will remain at the beginning of stage 3, we allocate this remaining amount entirely for the realization of module III. Therefore

$X_3(0) = \{0\}$ $X_3(1) = \{1\}$ $X_3(2) = \{2\}$ $X_3(3) = \{3\}$
$X_3(4) = \{4\}$ $X_3(5) = \{5\}$ $X_3(6) = \{6\}$

The following obvious condition has to be satisfied when the sets of admissible decisions for each state y_t are being constructed:

$$y_t \geq x_t$$

The values $F_t^1(y_t, x_t)$ describe the profit from the operation of module t, while $F_t^2(y_t, x_t)$ describes its reliability. Using the values from Table 1, we obtain the following values of $F_t^1(y_t, x_t)$:

$F_1^1(6, 0) = 0$ $F_1^1(6, 1) = 1,2$ $F_1^1(6, 2) = 2$ $F_1^1(6, 3) = 2,7$
$F_1^1(6, 0) = 3,3$ $F_1^1(6, 0) = 3,7$ $F_1^1(6, 0) = 4$
$F_2^1(y_2, 0) = 0$ $F_2^1(y_2, 1) = 3$ $F_2^1(y_2, 2) = 4,8$ $F_2^1(y_2, 3) = 5,5$
$F_2^1(y_2, 4) = 6,8$ $F_2^1(y_2, 5) = 7,9$ $F_2^1(y_2, 6) = 8,5$
$F_3^1(0, 0) = 0$ $F_3^1(1, 1) = 1,8$ $F_3^1(2, 2) = 4,5$ $F_3^1(3, 3) = 6,5$
$F_3^1(4, 4) = 7,8$ $F_3^1(5, 5) = 9$ $F_3^1(6, 6) = 10$

and the values of $F_2^t(y_t, x_t)$:

$F_1^2(6, 0) = 0.9$ $F_1^2(6, 1) = 0.97$ $F_1^1(6, 2) = 0.991$ $F_1^2(6,3) = 0.9973$
$F_1^2(6, 4) = 0.9992$ $F_1^2(6, 5) = 0.9998$ $F_1^2(6, 6) = 0.9999$
$F_2^2(y_2, 0) = 0.9$ $F_2^2(y_2, 1) = 0.94$ $F_2^2(y_2, 2) = 0.964$ $F_2^2(y_2,3)= 0.9784$
$F_2^2(y_2, 4) = 0.987$ $F_2^2(y_2, 5) = 0.9922$ $F_2^2(y_2, 6) = 0.9953$
$F_3^2(0. 0) = 0.9$ $F_3^2(1, 1) = 0.96$ $F_3^2(2, 2) = 0.984$ $F_3^2(3,3) = 0.9936$
$F_3^2(4, 4) = 0.9974$ $F_3^2(5, 5) = 9,999$ $F_3^2(6, 6) = 0.9994.$

By $F^1(y_1, x_1, y_2, x_2, y_3, x_3)$ we denote the profits from the operation of the system, while by F^2 $(y_1, x_1, y_2, x_2, y_3, x_3)$, its reliability. We obtain:

$$F^1(y_1, x_1, y_2, x_2, y_3, x_3) = F_1{}^1(y_1, x_1) + F_2{}^1(y_2, x_2) + F_3{}^1(y_3, x_3)$$
$$F^2(y_1, x_1, y_2, x_2, y_3, x_3) = F_1{}^1(y_1, x_1) \cdot F_2{}^2(y_2, x_2) \cdot F_3{}^2(y_3, x_3)$$

Figure 1 is a graphical representation of the process. The vertices of the graph represent the admissible states of the process, and the edges are the decisions.

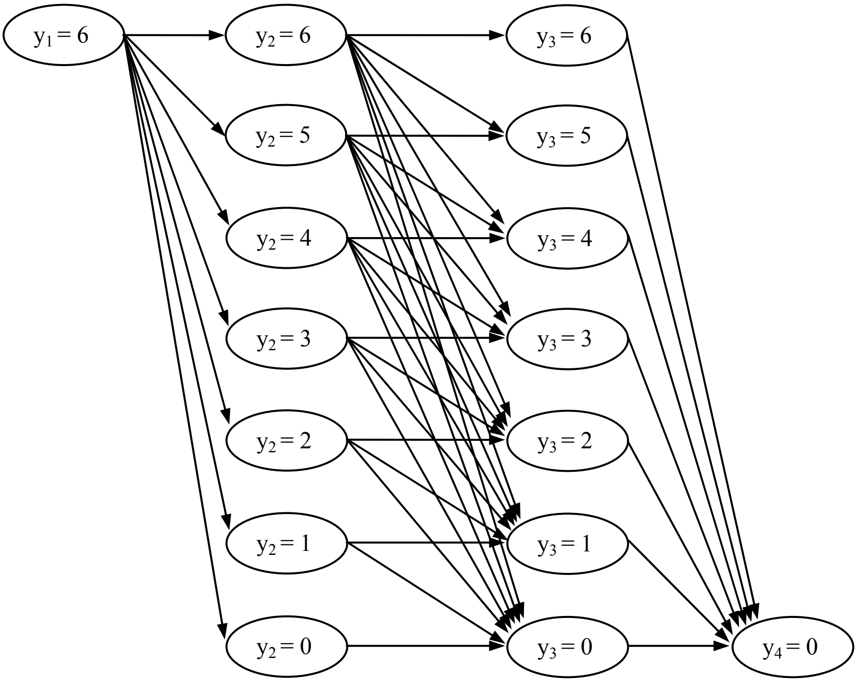

Figure 1: Graph of the process

3 The determination of the set of non-dominated evaluations and of the set of efficient realizations

Realization d' dominates realization d if

$$\forall_{k=1,\ldots,K} F^k(d') \geq F^k(d) \wedge \exists_{l=1,\ldots,K} F^l(d') > F^l(d) \quad (6)$$

which we denote by $F(d') \geq F(d)$

Realization d* is called an efficient realization if no other realization exists whose evaluation dominates the evaluation of d*; that is, if the following condition is satisfied:

$$\sim\exists_{d' \in D} F(d') \geq F(d) \quad (7)$$

The set of all efficient realizations is denoted by D*. The problem of vector maximization for a discrete multistage decision process consists in finding the set D* and the corresponding set F(D*) of non-dominated evaluations. We formulate this problem as follows:

$$\text{‘Max’ } \{F(d) \colon d \in D\} \tag{8}$$

To find the sets F(D*) and D* we use the vector optimality principle, which is a modification of the optimality principle (Klötzer, 1978; Li, Haimes, 1989).

An efficient strategy has the following property: regardless of the initial state and the initial decision, the remaining decisions have to constitute a sequence of decisions efficient with respect to the state resulting from the first decision.

We formulate optimality equations, which in our case are of the form:
for t = T:

$$G_T^*(y_T) = \text{‘max’ } \{F_T(y_T, x_T) \colon x_T \in X_T(y_T)\} \tag{9}$$

for t = T-1,…,1

$$G_t^*(y_t) = \text{‘max’} \{F_t(y_t) \bullet_t G_{t+1}(\Omega_t(y_t, x_t)) : x_t \in X_t(y_t)\} \tag{10}$$

where ‘max’ denotes the set of non-dominated vectors of the given subset, and $\bullet_t$ denotes the stage operator which combines the evaluations in stage t.

The set $G_T^*(y)$ contains non-dominated evaluation vectors for module T. The first component of each vector in this set describes the possible profit from the operation of this module, while the second, its reliability, if y_T units of the resource can be allocated for the operation of this module. The values $G_T^*(y_T)$ are calculated consecutively for all the states $y_T \in Y_T$.

The set $G_t^*(y_t)$ contains non-dominated evaluation vectors for modules from t through T. Their first components express the possible profit from the operation of these modules, while the second, their reliability, if t units of the resource can be allocated for the operation of module t.

Detailed calculations for the numerical example are in Appendix 1. As a result, we obtain four efficient realizations which are shown in Table 2.

Table 2: Efficient realizations and non-dominated evaluation vectors

Efficient realizations	**Non-dominated evaluation vectors**
d^A = (6,0, 6,2, 4,4)	[12.6, 0.8653]
d^B = (6,1, 5,2, 3,3)	[12.5, 0.921]
d^C = (6,2, 4,2, 2,2)	[11.3, 0.94]

4 Weighted sum approach

4.1 Multistage approach

Let $z = [z^1,\dots,z^K]$ be a vector with non-negative, non-zero components, that is, $z \in R_+^K \setminus \{0\}$. For each fixed $z \in R_+^K \setminus \{0\}$, we write the scalar maximization problem in the form:

$$\text{Max } \Sigma_{k=1}^{K} z^k F^k(d) : d \in D\}. \tag{11}$$

Let $D^0(z)$ be the set of all optimal solutions of problem (1). Using the general properties of efficient solutions in multicriteria programming, we can prove the following theorems (Trzaskalik, 1993):

Theorem 1
If for $z^0 \geq 0$ d^0 is an optimal solution of problem (11) and one of the following conditions is satisfied:

$$z^0 > 0 \tag{12}$$

$$\text{card } D^0(z^0) = 1 \tag{13}$$

$$\text{card } F(d^0) = 1 \tag{14}$$

then z^0 is an efficient realization of the given process, that is, $d^0 \in D^*$.

Theorem 2
The following holds:

$$\forall_{z \geq 0}\ D^0(z) \subset D^* \tag{15}$$

These theorems can be used to search for efficient solutions of our A&R problem. First let us note that in this bicriteria problem each criterion is expressed in different units. Hence, to present these criteria jointly as a weighted sum, first we have to normalize the values of the multistage criteria functions. The most convenient way of normalization of multistage criteria is to perform the transformation:

$$\Phi^k(d) = F^k(d)/F^{*k}(d) \tag{16}$$

for $k = 1,\dots, K$, $d \in D$, where

$$F^{*k} = \text{Max } \{F^k(d), d \in D, k = 1,\dots,K\} \tag{17}$$

The results of our numerical experiment are shown in Table 3.

Table 3: Results of the calculations for $z^1 = 0.9$, $z^2 = 0$

1	**2**	**3**	**4**	**5**	**6**	**7**	**11**	**12**	**13**	**8**	**9**	**10**	**14**	**15**	**16**	**17**	**18**
1	6	0	6	0	6	6	0	0	10	0.9	0.9	0.9994	10	0.8095	0.7937	0.8611	0.8004
2	6	0	6	1	5	5	0	3	9	0.9	0.94	0.999	12	0.8452	0.9524	0.8991	0.947
3	6	0	6	2	4	4	0	4.8	7.8	0.9	0.964	0.9974	12.6	0.8653	1	0.9205	0.9921
4	6	0	6	3	3	3	0	5.5	6.5	0.9	0.9784	0.9936	12	0.8749	0.9524	0.9307	0.9502
5	6	0	6	4	2	2	0	6.8	4.5	0.9	0.987	0.984	11.3	0.8741	0.8968	0.9298	0.9001
6	6	0	6	5	1	1	0	7.9	2.8	0.9	0.9922	0.96	10.7	0.8573	0.8492	0.9119	0.8555
7	6	0	6	6	0	0	0	8.5	0	0.9	0.9953	0.9	8.5	0.8062	0.6746	0.8576	0.6929
8	6	1	5	0	5	5	1.2	0	9	0.97	0.9	0.999	10.2	0.8721	0.8095	0.9278	0.8213
9	6	1	5	1	4	4	1.2	3	7.8	0.97	0.94	0.9974	12	0.9094	0.9524	0.9674	0.9539
10	6	1	5	2	3	3	1.2	4.8	6.5	0.97	0.964	0.9936	12.5	0.9291	0.9921	0.9884	0.9917
11	6	1	5	3	2	2	1.2	5.5	4.5	0.97	0.9784	0.984	11.2	0.9339	0.8889	0.9934	0.8993
12	6	1	5	4	1	1	1.2	6.8	2.8	0.97	0.987	0.96	10.8	0.9191	0.8571	0.9777	0.8692
13	6	1	5	5	0	0	1.2	7.9	0	0.97	0.9922	0.9	9.1	0.8662	0.7222	0.9214	0.7421
14	6	2	4	0	4	4	2	0	7.8	0.991	0.9	0.9974	9.8	0.8896	0.7778	0.9463	0.7946
15	6	2	4	1	3	3	2	3	6.5	0.991	0.94	0.9936	11.5	0.9256	0.9127	0.9846	0.9199
16	6	2	4	2	2	2	2	4.8	4.5	0.991	0.964	0.984	11.3	0.94	0.8968	1	0.9071
17	6	2	4	3	1	1	2	5.5	2.8	0.991	0.9784	0.96	10.3	0.9308	0.8175	0.9902	0.8347
18	6	2	4	4	0	0	2	6.8	0	0.991	0.987	0.9	8.8	0.8803	0.6984	0.9365	0.7222
19	6	3	3	0	3	3	2.7	0	6.5	0.9973	0.9	0.9936	9.2	0.8918	0.7302	0.9487	0.752
20	6	3	3	1	2	2	2.7	3	4.5	0.9973	0.94	0.984	10.2	0.9225	0.8095	0.9813	0.8267
21	6	3	3	2	1	1	2.7	4.8	2.8	0.9973	0.964	0.96	10.3	0.9229	0.8175	0.9818	0.8339
22	6	3	3	3	0	0	2.7	5.5	0	0.9973	0.9784	0.9	8.2	0.8782	0.6508	0.9342	0.6791
23	6	4	2	0	2	2	3.3	0	4.5	0.9992	0.9	0.984	7.8	0.8849	0.619	0.9413	0.6513
24	6	4	2	1	1	1	3.3	3	2.8	0.9992	0.94	0.96	9.1	0.9017	0.7222	0.9592	0.7459
25	6	4	2	2	0	0	3.3	4.8	0	0.9992	0.964	0.9	8.1	0.8669	0.6429	0.9222	0.6708
26	6	5	1	0	1	1	3.7	0	2.8	0.9998	0.9	0.96	6.5	0.8638	0.5159	0.9189	0.5562
27	6	5	1	1	0	0	3.7	3	0	0.9998	0.94	0.9	6.7	0.8458	0.5317	0.8998	0.5685
28	6	6	0	0	0	0	4	0	0	0.9999	0.9	0.9	4	0.8099	0.3175	0.8616	0.3719

Description:

Column 1 – realization
Column 2 – state y_1
Column 3 – decision $x_1 \in X_1(y_1)$
Column 4 – state y_2
Column 5 – decision $x_2 \in X_1(y_2)$
Column 6 – state y_3
Column 7 – decision $x_1 \in X_1(y)$
Column 8 – value of $F_1^1(y_1, x_1)$
Column 9 – value of $F_2^1(y_1, x_1)$
Column 10 – value of $F_3^1(y_1, x_1)$
Column 11 –value of $F_2^2(y_1, x_1)$
Column 12 – value of $F_2^2(y_1, x_1)$
Column 13 – value of $F_2^3(y_1, x_1)$
Column 14 – value of $F^1(d)$
Column 15 – value of $F^2(d)$
Column 16 – value of $\Phi^1(d)$
Column 17 – value of $\Phi^2(d)$
Column 18 – $0.9\Phi^1(d) + 0.1\Phi^2(d)$

Thanks to the small size of the problem, we can present all the realizations of the process.

4.2 Sensitivity analysis

In the case of a bicriteria problem, we can write:

$$\text{Max } \{z^1F^1(d) + z^2F^2(d) : d \in D\} \tag{18}$$

Consider an arbitrarily fixed point $\check{z} = [\check{z}^1, \check{z}^2]' \in R_+^2$. Substituting for z the components of vector ž we obtain the problem:

$$\text{Max } \{\check{z}^1F^1(d) + \check{z}^2F^2(d) : d \in D\} \tag{19}$$

which allows to generate the efficient realization corresponding to vector ž. By $Z^+(\check{z})$ we denote the set of the points of the half-line starting at [0, 0] and passing through ž, without the point [0, 0], that is,

$$Z^+(\check{z}) = \{[z^1, z^2] : z^2 = (\check{z}^1/\check{z}^2)\cdot z^1\} \tag{20}$$

Solving problem (18) for a fixed z, we obtain efficient realizations corresponding to z. Since $z^2 = (\check{z}^1/\check{z}^2)\cdot z^1$, problem (19) can be written in the form:

$$\text{Max } \{z^1F^1(d) + (\check{z}^1/\check{z}^2)\cdot z^1F^2(d) : d \in D\} \tag{21}$$

Problem (21) is equivalent to the following problem:

$$\text{Max } \{z^1 (\check{z}^1 F^1(d) + \check{z}^2F^2(d)) : d \in D\} \tag{22}$$

which, in turn, is equivalent to (19). This means that each point of half-line Z(ž) generates the same efficient realizations. Therefore, to determine the set of efficient realizations generated by the points of a given line, it suffices to find this set for one point of the line. The most convenient to use are points satisfying the following relationship:

$$z^1 + z^2 = 1.$$

Hence it suffices to consider the problem:

$$\text{Max } \{z^1F^1(d) + z^2F^2(d) : z^1 \geq 0, z^2 \geq 0, z^1 + z^2 = 1, \ d \in D \tag{23}$$

which can be replaced by the equivalent problem:

$$\text{Max } \{\mu F^1(d) + (1 - \mu) F^2(d) : 0 \leq \mu \leq 1, \ d \in D\} \tag{24}$$

To determine the values of parameter μ for which realization d' is efficient, one has to solve the corresponding systems of inequalities.

In our problem there are three efficient realizations: d^A, d^B and d^c. The relevant systems of inequalities are of the following form:

for realization d^A:

$$\mu F^1(d^A) + (1 - \mu) F^2(d^A) \geq \mu F^1(d^B) + (1 - \mu) F^2(d^B)$$
$$\mu F^1(d^A) + (1 - \mu) F^2(d^A) \geq \mu F^1(d^C) + (1 - \mu) F^2(d^C)$$

for realization d^B:

$$\mu F^1(d^B) + (1 - \mu) F^2(d^B) \geq \mu F^1(d^A) + (1 - \mu) F^2(d^A)$$
$$\mu F^1(d^B) + (1 - \mu) F^2(d^B) \geq \mu F^1(d^C) + (1 - \mu) F^2(d^C)$$

for realization d^C:

$$\mu F^1(d^C) + (1-\mu) F^2(d^C) \geq \mu F^1(d^A) + (1-\mu) F^2(d^A)$$
$$\mu F^1(d^C) + (1-\mu) F^2(d^C) \geq \mu F^1(d^B) + (1-\mu) F^2(d^B)$$

Substituting normalized numerical values, we obtain:

for realization d^A:

$$\mu\, 1 + (1-\mu)\, 0{,}920 \geq \mu\, 0{,}992 + (1-\mu)\, 0{,}988$$
$$\mu\, 1 + (1-\mu)\, 0{,}920 \geq \mu\, 0{,}897 + (1-\mu)\, 1$$

for realization d^B:

$$\mu\, 0{,}992 + (1-\mu)\, 0{,}988 \geq \mu\, 1 + (1-\mu)\, 0{,}920$$
$$\mu\, 0{,}992 + (1-\mu)\, 0{,}988 \geq \mu\, 0{,}897 + (1-\mu)\, 1$$

for realization d^C:

$$\mu\, 0{,}897 + (1-\mu)\, 1 \geq \mu\, 1 + (1-\mu)\, 0{,}920$$
$$\mu\, 0{,}897 + (1-\mu)\, 1 \geq \mu\, 0{,}992 + (1-\mu)\, 0{,}988$$

Solving these systems of inequalities we see that:

d^A is an efficient realization for $\mu \in [0.885, 1]$,

d^B is an efficient realization for $\mu \in [0.112, 0.885]$,

d^C is an efficient realization for $\mu \in [0, 0.112]$,

Moreover, for μ = 0.112 and μ = 0.885 problem (24) has two optimal solutions. Hence every point of the line $Z^+(0.112, 0.888)$ allows to generate both d^B and d^C, while every point of the line $Z^+(0.885, 0.15)$ allows to generate d^A and d^B. Every point of the plane $R^+\backslash 0\}$ allows to generate an efficient realization. It can happen, however, that one of the systems of inequalities will be inconsistent, which means that there exist efficient realizations which cannot be generated using problem (18).

The solution obtained is illustrated in Figure 2.

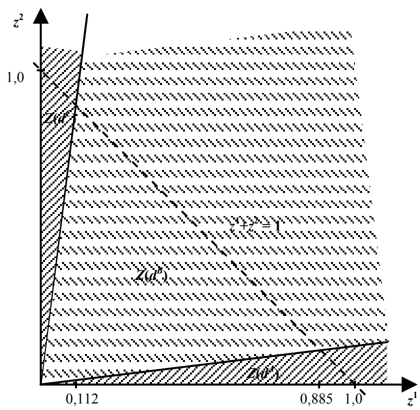

Figure 2: Graphical representation of sets $Z(d^A)$, $Z(d^B)$ and $Z(d^C)$

4.3 Stage weighted sum approach

In this subsection we consider a situation in which the DM intends to express his/her preferences as regards stage values. The preferences will be expressed as utilities, for the DM, of the normalized values of the individual stage criteria. Normalization will be performed with respect to maximal stage values for stage criteria. Hence we define new, normalized values of these criteria as follows:

$$\Phi_t^k(d_t) = F_t^k(d_t)/K \cdot F_t^k(d_t^*) \tag{25}$$

where d_t^* – arg max $\{F_t^k(d_t): d_t \in D_t\}$

We assume that the utility function is in additive form and obtain the problem:

$$\text{Max } \{\Sigma_{k=1}^{K} \Sigma_{t=1}^{T} \alpha_t^k \Phi_t^k(d): d \in D\} \tag{26}$$

in which we assume that α_t^k are non-negative and normalized, that is,

$$\forall_{t=1,\dots,T} \Sigma_{k=1}^{K} \alpha_t^k = 1 \tag{27}$$

Normalization is possible for each non-negative $\{\alpha_t^k\}$; it facilitates the interpretation of the results.

Due to the form of the objective function in problem (26), we can decompose it and solve it using the standard dynamic programming method, using the functional equations

for t = T:

$$g_T(y_T) = \text{Max } \{\Sigma_{k=1}^{K} \alpha_T^k \Phi_T^k(y_T, x_T): x_T \in X_T(y_T)\} \tag{28}$$

for t = T-1,…,1

$$g_t(y_t) = \text{Max } \{\Sigma_{k=1}^{K} \alpha_t^k \Phi_t^k(y_t, x_t) + g_{t+1}(\Omega_t(y_t, x_t): x_t \in X_T(y_t)\} \tag{29}$$

Using these equations we find the optimal realization of the process.

Our discussion will be illustrated by a numerical example. We will use again the numerical data from Table 1, and stage-normalize them using formula (25). The results are shown in Table 4.

Table 4: Normalized values of stage criteria

1	2	3	4	5	6	7	8	9	10	11	12	13
0	0	0	0.9	0.9001	0	0	0.9	0.9042	0	0	0.9	0.90054
1	1,2	0.3	0.97	0.9701	3	0.35	0.94	0.9444	2,8	0.28	0.96	0.960576
2	2	0.5	0.991	0.9911	4,8	0.56	0.964	0.9686	4,5	0.45	0.984	0.984591
3	2,7	0.675	0.9973	0.9974	5,5	0.65	0.9784	0.983	6,5	0.65	0.9936	0.994197
4	3,3	0.825	0.9992	0.9993	6,8	0.8	0.987	0.9917	7,8	0.78	0.9974	0.997999
5	3,7	0.925	0.9998	0.9999	7,9	0.93	0.9922	0.9969	9	0.9	0.999	0.9996
6	4	1	0.9999	1	8,5	1	0.9953	1	10	1	0.9994	1

The consecutive columns are as follows. Column 1 contains the amount of the resource allocated for the operation of each module. Columns 2, 6 and 10 contain profits resulting from the allocation of the given amount of the resource for the operation of modules 1, 2 and 3, respectively, while columns 3, 7, and 11 contain the values normalized using formula (25). Columns 4, 8 and 12 describe the reliability of the modules related to the amount of the resource allocated, while columns 5, 9 and 13 contain the normalized values.

We will consider three example problems:

Problem 1

The DM assumed that the profits from the operation of all the modules are less important than their reliability. This situation can be described by the following example data:

$\alpha_1^1 = 0.25$ $\alpha_1^2 = 0.75$ $\alpha_2^1 = 0.2$, $\alpha_2^2 = 0.8$ $\alpha_3^1 = 0.1$ $\alpha_3^2 = 0.9$

Problem 2

The DM assumed that the profits from the operation of all the modules are equally important as their reliability. This situation can be described by the following example data:

$\alpha_1^1 = 0.5$ $\alpha_1^2 = 0.5$ $\alpha_2^1 = 0.5$, $\alpha_2^2 = 0.5$, $\alpha_3^1 = 0.5$, $\alpha_3^2 = 0.5$

Problem 3

The DM assumed that the profits from the operation of all the modules are more important than their reliability. This situation can be described by the following example data:

$\alpha_1^1 = 0.75$ $\alpha_1^2 = 0.25$ $\alpha_2^1 = 0.8$, $\alpha_2^2 = 0.2$ $\alpha_3^1 = 0.9$ $\alpha_3^2 = 0.1$

Calculations using formulas (25) and (26) result in the solutions shown below.

Realization $d^{(10)}$ is the solution of problem 1. The value of the objective function is 0.5991.

Realization $d^{(16)}$ is the solution of problem 2. The value of the objective function is 0.7432.

Realization $d^{(21)}$ is the solution of problem 3. The value of the objective function is 0.8990.

We will compare these results with the solution of the A&R problem in its initial formulation obtained by searching for the complete set of efficient realizations. It turns out that realizations $d^{(10)}$ and $d^{(16)}$ are efficient realizations of the initial problem, while realization $d^{(21)}$, which is a solution of problem 3, is not an efficient realization. Hence, we perform efficiency testing and generate efficient realizations better that the tested one – if such realizations exist (Trzaskalik, 1990). In the case of $d^{(21)}$, realization $d^{(3)}$ is a better efficient realization.

5 Conclusions

In the paper we have presented a bicriteria A&R problem. Both multistage criteria considered – profit and reliability – are stage-wise separable and monotone, which allows to decompose the problem and to apply optimality equations to find the complete set of efficient realizations. A combination of these two criteria in one objective function, however, is not a separable scalar function, and therefore it is not possible to find optimal solutions using functional equations. In this case it is necessary to apply brute force or else approximation methods, using, for instance, genetic algorithms.

In the case of a weighted sum problem with stage values we can obtain solutions which are not efficient solutions of the initial problem. To check the efficiency of the realization obtained, we use the algorithm for checking efficiency and generating efficient realizations better than the realization tested, if such realizations exist.

In our case the weighted sum of multistage components was not separable. An open question remains: In the case of a separable function and an arbitrary choice of coefficients of stage functions, would we always obtain an efficient solution?

The problem of finding the set of non-dominated solutions in the criteria space and the corresponding set of efficient realizations has been discussed in detail in previous papers (Trzaskalik, 1990, 1998). It would be interesting to further investigate the issue of sensitivity analysis for MODP problems, since it has not been thoroughly researched so far. Another issue worth investigating in detail is that of the properties of the stage weighted sum approach.

The approach used in this paper is based on the application of a linear utility function. Another direction of research should be investigating the possibility of ordering efficient realizations from the most satisfying to the least satisfying based on determination of decision rules by means of rough sets. An example of such an application can be found in the paper by Renaud et al. (2007).

References

Abdelaziz F.B., Colapinto C., La Torre D., Liuzzi D. (2018), *A Stochastic Dynamic Multiobjective Model for Sustainable Decision Making*, Annals of Operations Research, 2018.

Bellman R. (1957), *Dynamic Programming*, Princeton University Press, Princeton.

Bozorgi-Amiri A., Khorsi M. (2016), *A Dynamic Multi-objective Location-routing Model for Relief Logistic Planning under Uncertainty on Demand, Travel Time, and Cost Parameters*, The International Journal of Advanced Manufacturing Technology, 85(5-8), 1633-1648.

Chen S., Fu G. (2005), *Combining Fuzzy Iteration Model with Dynamic Programming to Solve Multiobjective Multistage Decision Making Problems*, Fuzzy Sets and Systems, 152(3), 499-512.

Hämäläinen R.P., Mäntysaari J. (2002), *Dynamic Multi-objective Heating Optimization*, European Journal of Operational Research, 142(1), 1-15.

Klamroth K., Wiecek M.M. (2000), *Dynamic Programming Approaches to the Multiple Criteria Knapsack Problem*, Naval Research Logistics, 47(1), 57-76.

Klötzler R. (1978), *Multiobjective Dynamic Programming*, Mathematics Operations Horsch Statistics Series Optimization, 9(3), 423-426.

Li D., Haimes Y.Y. (1989), *Multiobjective Dynamic Programming: The State of the Art*, Control Theory and Advanced Technology, 5, 4, 471-483.

Mafakheri F., Breton M., Ghoniem A. (2011), *Supplier Selection-order Allocation: A Two-stage Multiple Criteria Dynamic Programming Approach*, International Journal of Production Economics, 132(1), 52-57.

Mine H., Fukushima M. (1979), *Decomposition of Multiple Criteria Mathematical Programming Problems by Dynamic Programming*, International Journal of System Science, 10, 5, 557-566.

Mitten L.G. (1964), *Composition Principles for Synthesis of Optimal Multistage Processes*, Operations Research, 12, 610-619.

Nowak M., Trzaskalik T. (2014), *Interactive Approach Application to Stochastic Multiobjective Allocation Problem – a Two-phase Approach*, Multiple Criteria Decision Making, 9, 84-100

Renaud J., Thibault J., Lanouette R., Kiss L.N, Zaras K., Fonteix C. (2007), *Comparison of Two Multicriteria Decision Aid Methods: Net Flow and Rough Set Methods in a High Yield Pulping Process*, European Journal of Operational Research, 177, 1418-1432.

Trzaskalik T. (2015), *MCDM Applications of Near Optimal Solutions in Dynamic Programming*, Multiple Criteria Decision Making, 10, 166-184.

Trzaskalik T. (1998), *Multiobjective Analysis in Dynamic Environment*, The Karol Adamiecki University of Economics Press, Katowice.

Trzaskalik T. (1993), *Weighted Sum Approach to Multiple Criteria Discrete Dynamic Programming*, Proceedings of the Administrative Sciences Association of Canada, 14, 2.

Trzaskalik T. (1990), *Wielokryterialne dyskretne programowanie dynamiczne. Teoria i zastosowania w praktyce gospodarczej*, Wydawnictwo Akademii Ekonomicznej im. Karola Adamieckiego, Katowice.

Appendix 1

Stage 3

Assume that at the beginning of Stage 3 we have y_3 resource units available, $y_3 \in Y_3$. We find the sets:

$$G_3^*(y_3) = \text{‘max’ } \{F_3(y_3, x_3): x_3 \in X_3(y_3)\}$$

We calculate:

$G_3^*(0) = \text{‘max’ } \{[0, 0.9]\} = \{[0, 0.9]\}$ $\quad x_3^*(0) = \{0\}$

$G_3^*(1) = \text{‘max’ } \{[2.8, 0.96]\} = \{[2.8, 0.96]\}$ $\quad x_3^*(1) = \{1\}$

$G_3^*(2) = \text{‘max’ } \{[4.5, 0.984]\} = \{[4.5, 0.984]\}$ $\quad x_3^*(2) = \{2\}$

$G_3^*(3) = \text{‘max’ } \{[6.5, 0.9936]\} = \{[6.5, 0.9936]\}$ $\quad x_3^*(3) = \{3\}$

$G_3^*(4) = \text{‘max’ } \{[7.8, 0.9974]\} = \{[7.8, 0.9974]\}$ $\quad x_3^*(4) = \{4\}$

$G_3^*(5) = \text{‘max’ } \{[9.0, 0.999]\} = \{[9.0, 0.999]\}$ $\quad x_3^*(5) = \{5\}$

$G_3^*(6) = \text{‘max’ } \{[10, 0.9994]\} = \{[10, 0.9994]\}$ $\quad x_3^*(6) = \{6\}$

Stage 2

Assume that at the beginning of Stage 2 we have y_2 resource units available, $y_2 \in Y_2$. We find the sets:

$$G_2^*(y_2) = \text{‘max’ } F_2(y_2, x_2) + G_3^*(y_2 - x_2): x_2 \in X_3(y_2)$$

We calculate:

$G_2^*(0) = \text{‘max’ } \{[0, 0.9] \bullet_2 [0, 0.9]\} = \text{‘max’ } \{[0, 0.81]\} = [0, 0.81]$

and $x_2^*(0) = \{0\}$

$$G_2^*(1) = \text{‘max’ } \begin{array}{l} \{[0, 0.9 \bullet_2 [2.8, 0.96] \\ \;[3, 0.94] \bullet_2 [0, 0.9]\} \end{array} = \text{‘max’ } \begin{array}{l} \{[2.8, 0.864] \\ \;[3.0, 0.846]\} \end{array} = \begin{array}{l} \{[2.8, 0.864] \\ \;[3, 0.846]\} \end{array}$$

and $x_2^*(1) = \{0, 1\}$

$$G_2^*(2) = \text{‘max’ } \begin{array}{l} \{[0, 0.9] \bullet_2 [4.5, 0.984] \\ \;[3, 0.94] \bullet_2 [2.8, 0.96] \\ \;[4.8, 0.964] \bullet_2 [0, 0.9]\} \end{array} = \text{‘max’ } \begin{array}{l} \{[4.5, 0.8856] \\ \;[5.8, 0.9024] \\ \;[4.8, 0.8676]\} \end{array} = \{[5.8, 0.9024\}$$

and $x_2^*(2) = \{1, 2\}$

$$G_2^*(3) = \text{‘max’ } \begin{array}{l} \{[0, 0.9] \bullet_2 [6.5, 0.9936] \\ \;[3, 0.94] \bullet_2 [4.5, 0.984] \\ \;[4.8, 0.964] \bullet_2 [2.8, 0.96] \\ \;[5.5, 0.9784] \bullet_2 [0. 0.9]\} \end{array} = \text{‘max’ } \begin{array}{l} \{[6.5, 0.8942] \\ \;[7.5, 0.9250] \\ \;[7.6, 0.9254] \\ \;[5.5, 0.8806]\} \end{array} = \{[7.6, 0.9254]\}$$

and $x_2^*(3) = \{2\}$

$$G_2^*(4) = \text{‘max’} \left\{ \begin{array}{l} [0, 0.9] \bullet_2 [7.8, 0.9974] \\ [3, 0.94] \bullet_2 [6.5, 0.9936] \\ [4.8, 0.964] \bullet_2 [4.5, 0.984] \\ [5.5, 0.9784] \bullet_2 [2.8, 0.96] \\ [6.8, 0.987] \bullet_2 [0, 0.9] \end{array} \right\} = \text{‘max’} \left\{ \begin{array}{l} [7.8, 0.8977] \\ [9.5, 0.9340] \\ [9.3, 0.9486] \\ [8.3, 0.9393] \\ [6.8, 0.8883] \end{array} \right\} = \left\{ \begin{array}{l} [9.5, 0.9340] \\ [9.3, 0.9486] \end{array} \right\}$$

and $x_2^*(4) = \{1, 2\}$

$$G_2^*(5) = \text{‘max’} \left\{ \begin{array}{l} [0, 0.9] \bullet_2 [9.0, 0.999] \\ [3, 0.94] \bullet_2 [7.8, 0.9974] \\ [4.8, 0.964] \bullet_2 [6.5, 0.9936] \\ [5.5, 0.9784] \bullet_2 [4.5, 0.984] \\ [6.8, 0.987] \bullet_2 [2.8, 0.96] \\ [7.9, 0.9922] \bullet_2 [0, 0.9] \end{array} \right\} = \text{‘max’} \left\{ \begin{array}{l} [9.0, 0.8991] \\ [10.8, 0.9376] \\ [11.3, 0.9578] \\ [10.0, 0.9627] \\ [9.6, 0.9475] \\ [7.9, 0.8230] \end{array} \right\} = \left\{ \begin{array}{l} [11.3, 0.9578] \\ [10.0, 0.9627] \end{array} \right\}$$

and $x_2^*(5) = (2, 3)$

$$G_2^*(6) = \text{‘max’} \left\{ \begin{array}{l} [0, 0.9] \bullet_2 [10, 0.9994] \\ [3, 0.94] \bullet_2 [9.0, 0.999] \\ [4.8, 0.964] \bullet_2 [7.8, 0.9974] \\ [5.5, 0.9784] \bullet_2 [6.5, 0.9936] \\ [6.8, 0.987] \bullet_2 [4.5, 0.984] \\ [7.9, 0.9922] \bullet_2 [2.8, 0.96] \end{array} \right\} = \text{‘max’} \left\{ \begin{array}{l} [10.0, 0.8995] \\ [12.0, 0.9391] \\ [12.6, 0.9615] \\ [12.0, 0.9721] \\ [11.3, 0.9712] \\ [10.7, 0.8930] \end{array} \right\} = \left\{ \begin{array}{l} [12.6, 0.9615] \\ [12.0, 0.9721] \end{array} \right\}$$

and $x_2^*(6) = (2, 3)$

Stage 1

At the beginning of Stage 1 the process is in state $y_1 = 6$. We find:

$$G_3^*(y_3) = \text{‘max’} \ \{F_3(y_3, x_3) : x_3 \in X_3(y_3)\}$$

that is,

$$G_1^*(6) = \text{‘max’} \left\{ \begin{array}{l} [0, 0.9] \bullet_1 [12.6, 0.9615] \\ [0, 0.9] \bullet_1 [12.0, 0.9721] \\ [1.2, 0.97] \bullet_1 [11.3, 0.9578] \\ [1.2, 0.97] \bullet_1 [10.0, 0.9627] \\ [2, 0.991] \bullet_1 [9.5, 0.9340] \\ [2, 0.991] \bullet_1 [9.3, 0.9486] \\ [2.7, 0.9973] \bullet_1 [7.6, 0.9254] \\ [3.3, 0.9992] \bullet_1 [5.8, 0.9024 \\ [3.7, 0.9998] \bullet_1 [2.8, 0.864] \\ [3.7, 0.9998] \bullet_1 [3, 0.846] \\ [4, 09999] \bullet_1 [0, 0.81] \end{array} \right\} = \text{‘max’} \left\{ \begin{array}{l} [12.6, 0.8654] \\ [12.0, 0.8749] \\ [12.5, 0.9291] \\ [11.2, 0.9338] \\ [11.5, 0.9256] \\ [11.3, 0.9401] \\ [0.3, 0.9229] \\ [9.1, 0.9017] \\ [6.5, 0.8638] \\ [6.7, 0.8458] \\ [4.0, 0.8099] \end{array} \right\} = \left\{ \begin{array}{l} [12.6, 0.8654] \\ [12.5. 0.9291] \\ [11.3, 0.9401] \end{array} \right\}$$

and $x_1^*(6) = (0, 1, 2)$

We obtain the following efficient realizations:

$d^A = (6,0, 6,2, 4,4)$	$F(d^A) = [12.6, 0.8654]$
$d^B = (6,1, 5,2, 3,3)$	$F(d^B) = [12.5, 0.9291]$
$d^C = (6,2, 4,2, 2,2)$	$F(d^C) = [11.3, 0.9401]$