NETWORK PROGRAMMING THEORY APPLICATION TO PROJECT PORTFOLIO FORMATION

Irina Burkova[a], Boris Titarenko[b], Amir Hasnaoui[c] and Roman Titarenko[d*]

*[a]Institute of Control Sciences of Russian Academy of Science,
Leninskiy avenue 14, Moscow, Russia*
[b]Moscow State University of Civil Engineering, Yaroslavskoye Hwy 26, Moscow, Russia
[c]La Rochelle Business School, 102 Rue de Coureilles, La Rochelle, France
[d]Synergy University, Leningradskiy avenue 80, Moscow, Russia

Abstract

The paper deals with an application of the network (dichotomous) programming method for solving multi-extremal problems and discrete optimization problems. The concept of a generalized dual problem is introduced and a theorem on its convexity is proved. Network programming method is used to build a business-forming project portfolio as well as an algorithm for solving completely dependent related projects is developed. Also, this method is used to solve the problem of building a business-supporting project portfolio for which the lower cost estimate is obtained. A computational experiment is carried out to evaluate the suggested algorithm, which showed that for large dimensions of the problem it is more effective than for solving the problem by linear programming methods.

Keywords: project, project portfolio, network programming, generalized dual problem

1. INTRODUCTION

The recent scientific literature suggests many definitions of project portfolio. According to Meredith and Mandel (2010) a portfolio is 'a group or set of projects with varying characteristics'. Turner and Müller (2003) propose the following definition of a portfolio: 'An organization where projects are managed together to coordinate interfaces, prioritize resources between projects, and thereby reduce uncertainty'. Artto et al. (2001) consider a project portfolio as 'a collection of projects that are carried out in the same business unit sharing the same strategic objectives and the same resource pool'.

* *Corresponding author: titar1@mail.ru*

International Project Management Association argues that a project portfolio can be seen as a collection of projects or programs that are not necessarily linked to one goal. Projects and programs are combined into a portfolio in order to ensure optimal use of the organization's resources and achieve the strategic goals of the organization while minimizing portfolio risks (Coesmans et al., 2018).

Project portfolios may include not only individual projects but also groups of interrelated projects combined into programs. For example, an information technology development program may be a part of the portfolio of organizational development projects.

Based on the above definitions we consider in this paper a project portfolio as a set of projects that are grouped in order to increase management efficiency and to achieve the strategic goals of the organization.

Building of the effective project portfolio is an important and complex task, which implies taking into account the compliance of project goals with the strategic goals of the organization, and maximization of the portfolio value for the organization under existing resource constraints.

The main goal of this paper is the application of the network (dichotomous) programming method for solving multi-extremal problems and discrete optimization problems that allows to build a business-forming project portfolio and a business-supporting project portfolio.

This paper has the following structure. In Section 3 the concept of a generalized dual network programming problem is introduced and the theorems are proved, the proofs of which are used to solve the problems of project portfolio formation. In Section 4 the

problem of business-forming project portfolio building is analyzed and the algorithm for its solving is developed. In Section 5 the problem of building a business-supporting project portfolio is examined, an algorithm for its solution based on branch and bound method is suggested as well as a computational experiment is carried out to estimate the computational complexity of the algorithm. Section 6 contains a discussion of the results and perspectives of possible applications.

2. LITERATURE REVIEW

At the present time due to a high competition in fast changing environment efficient management of the project portfolio is an important tool for the success of any company. Project portfolio management (PPM) helps to achieve the strategic goals of an organization by means of management and optimization activities applied to a project portfolio under certain restrictions (Archer & Ghasemzadeh, 1999; Hyväri, 2014; Momćilović et al., 2014). According to (Levine, 2005) PPM solves key problems of project-based organizations: eliminates the gap between project and operating management and becomes a core of all organizational activities. Project portfolio synergistic effect may be described as a simultaneous achievement of the best financial, economic, social and other results (Maizlish & Handler, 2005).

Most organizations seek to get the maximum benefit from the projects implementation included in the corporate portfolio. At the same time, they operate under a variety of constraints, such as time deadline and resource limitations, product project linkages and environmental constraints.

Over the past decades, a variety of approaches have been suggested to optimize the process of portfolio formation and management, covering various aspects of project management. Most approaches are based on solving optimization problems where it's necessary to build a target function describing the value of the portfolio and to determine the system of constraints imposed on its components.

The list of such approaches and examples of their use includes: linear programming method (Titarenko et al., 2018); dynamic programming method (Gutjahr, 2011); branch and bound method (Sharifghazvini, 2018); genetic algorithm (Yu et al., 2012); knapsack method (Tavano et al., 2013); probabilistic method including the use of robust approach (Qingguo, 2015; Marcondes et al., 2017; Yang et al., 2015; Yousefi et al., 2018).

In 2003, the theory of network programming was developed to obtain exact solutions or upper (lower) estimates of multi-extremal (in the particular case - discrete) optimization problems (Burkov et al., 2003) and, subsequently, it was advanced in the following works (Burkov et al., 2005; Burkova, 2009).

The theory of network programming allows to represent a complex function as a superposition of simpler functions. Such a representation is called a network representation and it is convenient to represent it as a graph of the network type, the initial vertices of which correspond to variables, and the rest correspond to the functions included in the superposition. The network programming method for solving optimization problems is applicable when both the objective function and the constraint func-tion have the same network representations (Burkov et al., 2005;

Burkova, 2009).

Taking into account that the theory of network programming has a wide range of applicability in various spheres including project management (see, e.g. – Burkov et al., 2017; Kondratjev et al., 2019; Burkov et al., 2011) it seems reasonable to apply the theory to the problem of project portfolio formation that the paper deals with.

3. GENERALIZED DUAL PROBLEM IN THE METHOD OF NETWORK PROGRAMMING

The method idea is the possibility of representing a function (or set of functions) in the form of a superposition of simpler functions. Such a superposition is convenient to represent in the form of a directed network without contours the inputs of which correspond to variables (zero level), and the rest of the nodes correspond to functions included in the superposition (network representation).

The problem solution at the final node of the network gives an upper (lower) estimate for the initial problem. If the network representation is a tree, then the solution of the problem in the final node of the network gives the optimal solution of the initial problem.

Definition 1. *Functions (or two sets of functions) are called structurally similar (s-similar) if they have the same network representation structures, i.e. the same network with matching sets of variables at each network node.*

If the objective function and the set of functions representing the constraints of the problem are s-similar, in each node of the network representation (starting from the nodes of the first level), it is possible to solve

simpler optimization problems with fewer variables and (or) constraints.

Assume that we solve the maximum problem and the objective function is a non-decreasing function of its variables.

Theorem 1. *The solution of the problem at the final node of the network representation (network input) gives an upper estimate of the initial problem.*

Proof. The set of solutions of the modified network contains all the solutions of the initial problem. These solutions have the following form. If one arc of the obtained solution enters the node corresponding to the variable, then all arcs entering this node also belong to the obtained solution. It follows that the obtained optimal solution of the modified problem gives a lower bound for the optimal solution of the initial problem.

The requirement of a non-decreasing objective function can be weakened if we consider three-level structures (Figure 1):

$$F(x) = \Phi\left[\phi_1(x), \phi_2(x), \ldots, \phi_m(x)\right] \quad (1)$$

In this case, it is enough to require non-decreasing function $\Phi(\varphi)$ by function φ_j, $j = \overline{1, m}$ (Figure 1).
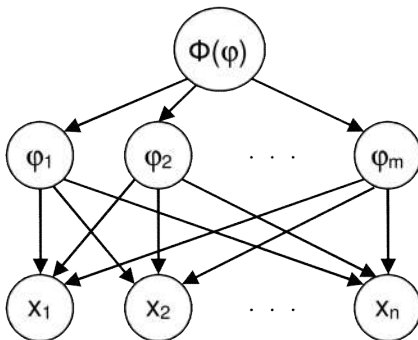

Figure 1. Three-level structure

Thus, the network programming method provides a fairly universal way to obtain upper (lower) estimates of a wide class of discrete optimization problems as well as multi-extremal continuous problems. These estimates can be used in the branch and bound method.

Assume that the structure of the network representation is a tree. In this case, the following theorem takes place.

Theorem 2. *The solution of the problem in the final node of the network representation gives the optimal solution of the initial problem.*

Proof. The proof of Theorem 2 directly follows from the Theorem 1, namely, for the tree not only the set of solutions of the modified network contains all solutions of the initial problem. The opposite is also true – the set of solutions of the initial problem coincides with the set of solutions of the modified network since the set of solutions with input index is greater than when the one is empty. □

Typically, there are many network representations for the optimization problem. Naturally, it raises the issue of a choice of such a network representation for which to get upper (lower) estimates is the minimum (maximum).

Example 1. Let's consider the problem of maximizing the function

$$f(x) = 2x_1 + 3x_2 + 5x_3 \quad (2)$$

under the constraint

$$\varphi(x) = 3x_1^2 x_2 + 2x_2^2 x_3 \le 60, \quad (3)$$

where x_i are 2 or 3. To make $f(x)$ s-similar to $\varphi(x)$, let's write it as

$$f(x) = 2x_1 + x_2 + 2x_2 + 5x_3. \quad (4)$$

In this case, the functions have a single network representation structure (Figure 2), where

$$y_1 = 2x_1 + x_2; \qquad y_2 = 2x_2 + 5x_3;$$
$$z_1 = 3x_1^2 x_2; \qquad z_2 = 2x_2^2 x_3. \tag{5}$$

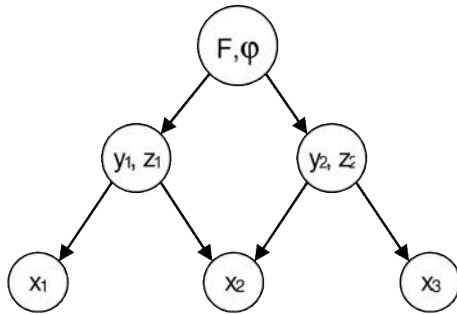*Figure 2. Single network representation structure for functions f (x) and φ(x)*

Solve evaluation problems in node (y_1, z_1) and (y_2, z_2).

At the node (y_1, z_1) we solve the maximization problem $y_1 = 2x_1 + x_2$ under the constraint $z_1 = 3x_1^2 x_2 \leq P_1$, where P_1 is a parameter. The solution of this problem is given in Table 1.

Table 1. The solution of the maximization problem $y_1 = 2x_1 + x_2$ under the constraint $z_1 = 3x_1^2 x_2 \leq P_1$, where P_1 is a parameter.

P_1	24	36	54
F_1	6	7	8

At the node (y_2, z_2) we solve the maximization problem $y_2 = 2x_2 + 5x_3$ under the constraint $z_2 = 3x_2^2 x_3 \leq P_2$, where P_2 is a parameter. The solution of this problem is given in Table 2.

Table 2. The solution of the maximization problem $y_2 = 2x_2 + 5x_3$ under the constraint $z_2 = 3x_2^2 x_3 \leq P_2$, where P_2 is a parameter

P_2	16	24	54
F_2	14	19	21

At the node (f, φ) we solve the maximization problem $F_1(P_1) + F_2(P_2)$ under the constraint $P_1 + P_2 \leq 60$. The solution of this problem is given in Table 3.

Table 3. The solution of the maximization problem $F_1(P_1) + F_2(P_2)$ under the constraint $P_1 + P_2 \leq 60$

$F_2; P_2$ $F_1; P_1$			
24; 19	48; 25	60;26
16; 14	40;20	52; 21
	24;6	36; 7

Optimal solution is defined by the cell (60; 26). Upper estimate is equal to 26. Note that this estimate corresponds to 2 solutions: at the node (y_1, z_1) the solution is $x_2=3, x_1=2$; at the node (y_2, z_2) the solution is $x_2=2, x_3=3$. This pair of solutions does not define a valid solution.

Take another network representation of the function $f(x)$, i.e. take

$$y_1 = 2x_1, \; y_2 = 3x_2 + 5x_3. \tag{6}$$

However, as shown above, solving two evaluation problems we obtain an estimate 25. At the same time we have solutions: at the node (y_1, z_1) the solution is $x_1=2, x_2=2$; at the node (y_2, z_2) the solution is $x_3=3, x_2=2$.

Since the values x_2 in both problems are the same, the pair of these solutions defines the valid one, and hence is the optimal solution of the initial problem.

Definition 2. *The problem of finding a network representation for which the upper*

(lower) estimate is minimal (maximal) is called a generalized dual problem (GDP).

In (Burkova, 2009) there is a constructed GDP for the classical problem of nonlinear optimization

$$
\begin{aligned}
&f(x) \to \max, \\
&\varphi_j(x) \le b_j, \quad j = \overline{1, m}, \\
&x \in X_0,
\end{aligned}
\tag{7}
$$

and simple network representation structure (Figure 3), where x_j is a j problem constraint, $X = \bigcap\limits_{j=0}^{m} x_j$.
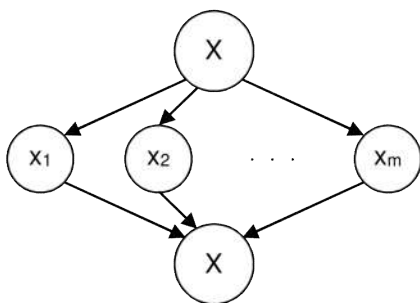

Figure 3. Network representation structure for the problem of nonlinear optimization

Let's introduce generalized (dual) functions $h_j(x)$ so that

$$
\sum_{j=0}^{m} h_j(x) = f(x)
\tag{8}
$$

and consider *(m+1)* evaluation problem. The first *m* evaluation problems are to maximize $h_j(x)$ under the constraint $\varphi_j(x) \le b_j$. The last problem is as follows

$$
h_0(x) = f(x) - \sum_{j=1}^{m} h_j(x) \to \max,
\tag{9}
$$

$$
x \in X.
$$

If we indicate $F_j(h_j)$ as the value of the objective function in the optimal solution of j evaluation problem and $F(h)$ as the value of the objective function in the optimal solution of *(m+1)* evaluation problem, then functional value of

$$
F(h) + \sum_{j=1}^{m} F_j(h_j)
\tag{10}
$$

gives an upper estimate for the initial problem. GDP is the definition of functions $h_j, j = \overline{0, m}$ that satisfy (8) and minimize (10). In particular, if we have $h_j(x) = \lambda; \varphi_j(x), \lambda_j > 0$, then GDP becomes the method of Lagrange multipliers.

In (Burkova, 2009) it was proved that at network representation (Figure 3) GDP is a convex programming problem. Let's give a generalization of this result to arbitrary network representation structures.

Theorem 3. *GDP is a convex programming problem.*

Proof. Let's consider an arbitrary network representation of the constraint system structure. First, let's consider a method of constructing a network representation of the objective function, s-similar network representation of the constraints system. Let the network $G(x,u)$ defines the network representation of the constraints system, Q_l – the set of nodes of the first level (from the nodes of this level begin the solution of evaluation problems), *r* – the number of these nodes. Divide the function $f(x)$ into *r* of parts $h_i(x)$,

$$\sum_{i \in Q_1} h_i(x) = f(x) \qquad (11)$$

In each node i we solve the maximization problem $z_1 = h_i(x)$ under constraint $\varphi_i(x) \leq p_i$ (p_i – parameter). Indicate $z_i^0(p_i, h)$ as a value z_i in the optimal solution of this problem. Indicate u_i as a set of directly following nodes for the node i, r_i as a number of these nodes. Divide the function $z_i^0(p_i, h)$ into r_i of parts $z_{ij}^0(p_{ij}, h)$, $j \in u_i$. Indicate Q_2 as a set of nodes of the second level, v_i as a set of set of immediately preceding nodes for a node $i \in Q_2$. At node i let's solve the maximization problem:

$$z_i = \sum_{j \in v_i} z_{ij}^0(p_j; h) \qquad (12)$$

under the constraint

$$\varphi_i(p^i) \leq p_i, \text{ where } p^i = \{p_j, j \in v_i\}. \qquad (13)$$

Indicate $z_i^0(p_i, h)$ as a value z_i in the optimal solution of this problem. Solving the problem at the first level nodes, let's go to nodes of the second level and etc. until to the final node. Consider the problem (12) – (13) solved at the nodes of the network. Let's take two feasible solutions of the generalized dual problem h_1 and h_2 . It is obvious that their convex linear combination $h = ah^1 + (1-a)h^2$ is also a valid GDP solution.

There is also

$$z_{ji}^0(p_j, h) = \alpha z_{ji}^0(p_j, h^1) + (1-\alpha) z_{ji}^0(p_j, h^2) \qquad (14)$$

therefore

$$
\begin{aligned}
z_i^0(p_i, h) &= \max_{p^i} \sum_{j \in v_i} z_{ji}^0(p_j, h) = \\
&= \max_{p^i} \sum_{j \in v_i} \left[\alpha z_{ji}^0(p_j, h^1) + (1-\alpha) z_{ji}^0(p_j, h^2) \right] \leq \\
&\leq \alpha \max_{p^i} \sum_{j \in v_i} z_{ji}^0(p_j, h^1) + (1-\alpha) \sum_{j \in v_i} z_{ji}^0(p_j, h^2) = \\
&= \alpha z_i^0(p_i, h^1) + (1-\alpha) z_i^0(p_i, h^2).
\end{aligned}
\qquad (15)
$$

Thus, at each node we get a convex functional, which proves the theorem. □

Let's consider examples of network programming theory application to the problems of project portfolio formation.

4. BUILDING OF BUSINESS-FORMING PROJECT PORTFOLIO

4.1. Problem statement

There are m business-forming projects (BF-projects). Each of such projects allows you to implement n_i related business projects. Indicate a_i as an effect of the implementation of í-th business-forming project $i = \overline{1,m}$, a_{ij} – an effect of the implementation of j-th related project, $j = \overline{1, n_i}$ in case if the i-th business-forming project is implemented, δ_{ij} is an effect of the implementation of j-th related project, if i-th BF-project isn't implemented, c_i– implementation costs of i-th BF-project, c_{ij}– implementation costs of j-th project accompanying for the í-th business-forming project $j = \overline{1, n_i}$. Indicate $x_i = 1$, if í-th BF-project is included in the portfolio, $x_i = 0$, otherwise; $x_{ij} = 1$, if j-th project accompanying to i-th BF-project project is included in the portfolio, $x_{ij} = 0$ alternatively.

Define

$$x_i, i = \overline{1,m} \text{ and } x_{ij}, i = \overline{1,m}, j = \overline{1, n_i}, \qquad (16)$$

maximizing

$$A(x) = \sum_{i=1}^{n} x_i \left(\alpha_i + \sum_{j=1}^{n_i} \alpha_{ij} \right) + \sum_{i,j} \delta_{ij} x_{ij} \quad (17)$$

under the constraint

$$\sum_{i=1}^{n} c_i x_i + \sum_{i,j} c_{ij} x_{ij} \le R \quad (18)$$

where R – the value of the investment Fund.

4.2. Solution algorithm for completely dependent related projects

Let's consider the example when the related projects are completely dependent on the business-forming project, i.e. if the BF-project is not implemented, the related BF-projects also can't be implemented. In formal notation, this means that for $\delta_{ij}=0$ all i, j. The peculiarity of this example is the fact that there is a network representation of the problem with the structure of the dichotomous tree type network representation.

To show this, let's consider the general problem of formation a portfolio of interdependent projects. There are n projects. A matrix is given, a_{ij} elements of which determine the effect (we can say, the synergetic effect), if both project i and project j are implemented. The problem is to define $x=\{x_i\}$, $i=\overline{1,n}$ maximizing

$$A(x) = \sum_{i,j} \alpha_{ij} x_i x_j \quad (19)$$

under the constraint

$$\sum c_i x_i \le R \quad (20)$$

It is easy to show that the problem (17) – (18) is a special example of the problem (19) – (20).

Define a graph of project relations. It consists of n nodes (by the number of projects). Two nodes and *j* are connected by an edge *(i,j)*, if $a_{ij}>0$.

Let's give a definition of a special type of graph relations – "Bush". Consider the graph in Figure 4. In fact, this graph reflects the situation with several BF-projects, each of which has several related projects. As is known, a graph consisting of several trees is called a forest. In our example, we are not dealing with trees, but rather with "bushes". It is therefore natural to call a graph consisting of several "bushes" by a "shrub" (Figure 4).
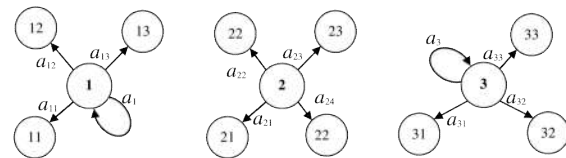

Figure 4. "Bush" type graph

Let's ask when problem (19) – (20) admits a network representation with a dichotomous tree structure. The answer to this question is given by the following theorem.

Theorem 4. *In order that problem (19) – (20) allows a network representation of the dichotomous tree under any $a_{ij}\neq0$, it is necessary and sufficient that the project relations graph is a "shrub".*

Proof. The sufficiency is obvious. Indeed, the "shrub" admits a network representation with a dichotomous tree structure. Such a representation for one bush is shown in Figure 5. □
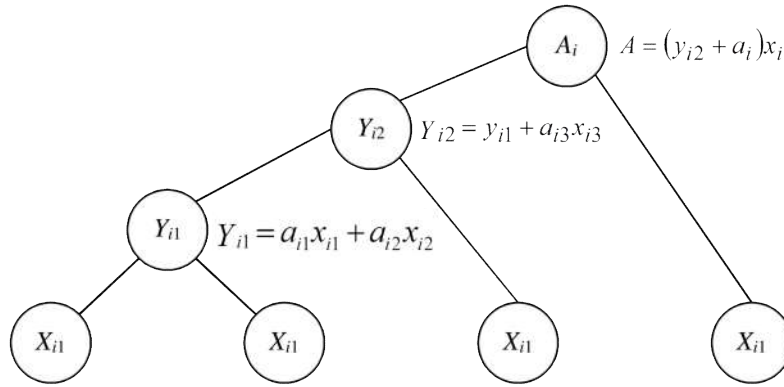
Figure 5. The structure of the "bush" network representation in the form of a dichotomous tree

Necessity. Let the graph of relations is not a "bush", i.e. there is always a chain of length 3 in it. However such a chain can't be realized under arbitrary $a_{ij} \neq 0$ dichotomous tree structure that is easy to verify. □

4.3. Partially dependent related projects

Consider a situation in which one or more related projects have their own businesses independent of the business project. This situation corresponds to the graph of relations (Figure 6).
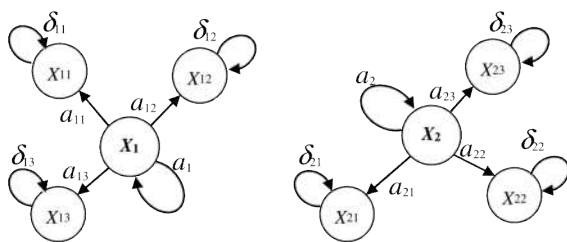

Figure 6. Graph for related projects that have their own businesses

Problem statement. Define $\{x_i\}, \{x_{ij}\}$, $i=\overline{1,m}, j=\overline{1,n_j}$ maximizing

$$A(x) = \sum_{i,j} \alpha_{ij} x_i x_{ij} + \sum_i \alpha_i x_i + \sum_{i,j} \delta_{ij} x_{ij} \quad (21)$$

under the constraint

$$\sum_i c_{ij} x_{ij} + \sum_i c_i x_i \leq R \quad (22)$$

Note that the interaction graph of projects is no longer a "shrub", so there is no network representation with the structure of the dichotomous tree. The exception is the special case when the business-forming project has only one accompanying project.

To solve the problem, let's consider two options $x_i = 0$ or $x_i = 1$ for each "bush".

1 step:

1 variant $x_i = 0.$ Solve problem for each i: define $\{x_{ij}\}, j = \overline{1,n_j}$ maximizing

$$B_i(x) = \sum_j \delta_{ij} x_{ij} \quad (23)$$

under the constraint

$$\sum_j c_{ij} x_{ij} \leq R_i, \quad (24)$$

where R_i – parameter. $0 \leq R_i \leq R$.

As a result, we get the dependence $B_{i1}(R_i)$ of the i-bush maximum effect on the amount of funding R_i. The above problem is solved by the method of dichotomous programming.

2 variant $x_i=1$. Solve problem for each i: define $\{x_{ij}\}$, $j=\overline{1,n_j}$ maximizing

$$B_i(x) = \sum_{i,j} \left(\alpha_{ij} + \delta_{ij}\right)x_{ij} + \alpha_i \qquad (25)$$

under the constraint

$$\sum_j c_{ij}x_{ij} \leq R_i - c_i \qquad (26)$$

As a result, we define the dependence $B_{i2}(R_i)$ of the i-bush maximum effect on the amount of funding R_i.

Find dependence

$$B_i(R_i) = \max\left[B_{i1}(R_i); B_{i2}(R_i)\right]. \qquad (27)$$

2 step:

Solve problem: define R_i, $j=1,m$ maximizing

$$B(R) = \sum_i B_i(R_i) \qquad (28)$$

under the constraint

$$\sum_i R_i \leq R. \qquad (29)$$

The problem, as above, is solved by the method of dichotomous programming. The solution is the reverse method.

5. FORMATION A BUSINESS - SUPPORTING PROJECT PORTFOLIO

5.1. Problem statement

Suppose there are m types of needs required to provide a certain group of businesses. There are also n business-supporting projects (BSP) each of which is able to meet the need for a specific product. Indicate b_j as the implementing cost of the j-BSP, c_{ij}– the cost of providing i-need by j-BSP.

Definition. Portfolio of the Q business-supporting projects will be called complete if for any need there is a BSP $j{\in}Q$ that can respond to this need. Indicate R_i as a set of BSPs that can responds i-need, where

$$Ti = Ri \bigcap Q \qquad (30)$$

a set of BSPs from the full set Q that can respond the i need. It is obvious that consumers (businesses) will choose BSP s, which will provide the need with the minimum cost, that is

$$c_{is} = \min_{j \in T_j} c_{ij}. \qquad (31)$$

The total cost of providing businesses will be

$$C(Q) = \sum_{j \notin Q} b_j + \sum_{i=1}^{m} \min_{j \in T_i(Q)} c_{ij}. \qquad (32)$$

Problem statement. *To determine the full set of Q BSPs in which the total cost (32) is minimal.*

Comment 1. If a large time interval is considered, then to define c_{ij} discounting periods should be taken into account.

Comment 2. It is supposed that the implementation of BSPs is carried out in one period.

5.2. Lower cost estimates obtaining

To obtain the lower cost estimates, apply the network programming method. Indicate P_j as a set of needs that j BSP can provide, m_j is a number of such needs. Divide each j BSP for which $m_j>1$ to m_j elementary BSPs

with implementation cost s_{ij}, $i \epsilon p_j$ so that

$$\sum_{i \in p} s_{ij} = b_j, \quad j = \overline{1, n} \qquad (33)$$

It is obvious that each elementary BSP provides only one need. In this case, the problem is easily solved. Namely, it will provide i- need for BSP k, for which the cost is

$$c_{ik} + s_{ik} = \min_{j \in R_i} c_{ij} + s_{ij} \qquad (34)$$

The total cost is

$$S(Q) = \sum_{i=1}^{m} \min_{j \in R_i} (c_{ij} + s_{ij}) \qquad (35)$$

Theorem 5. *Total cost (35) gives a lower estimate of total cost (32).*

Proof. The proof follows from the fact that any solution to the minimization problem (32) is a valid solution to the problem (33), (35).

Generalized dual problem. Define $\{s_{ij}\}$, satisfying (33) and maximizing (35). $\square$

The problem is easy to reduce to the problem of linear programming. To do this, let's take the variables

$$u_i = \min_{j \in R_i} (c_{ij} + s_{ij}), \ i = \overline{1, n}. \qquad (36)$$

It is obvious that

$$c_{ij} + s_{ij} \geq u_i, \ i = \overline{1, m}, \ j \in R_i. \qquad (37)$$

We have a linear programming problem: maximize

$$U = \sum_i u_i \qquad (38)$$

under the constraints (33) and (37).

Corollary. *If a feasible solution to the initial problem is obtained as a result of the solution of the generalized dual problem, then this solution is optimal.*

Proof. The proof follows from the fact that we obtain an achievable lower estimate, it means that the corresponding solution is optimal. $\square$

5.3. Branch and bound method

The solution results of the generalized dual problem, which give a low estimate for the initial problem, can be used in the branch and bound method.

Note that the exact solution of the generalized dual problem requires the solution of the linear programming problem. This will require computational time that may not be offset by a possible decrease in the number of branches. At the same time, any feasible solution to the generalized dual problem gives a lower estimate for the initial problem. Therefore, we can take any feasible solution of the generalized dual problem to obtain a low estimate of subsets. Computational experiments show that good results are obtained by choosing such s_{ij} values for which values $s_{ij} + c_{ij}$ are equal for all $i \epsilon p_i$.

From $s_{ij} + c_{ij} = u$, $i \epsilon p_i$ define

$$s_{ij} = u_j - c_{ij}, \ i \in p_j \qquad (39)$$

$$\sum_{i \in p_j} s_{ij} = m_j u_j - \sum_{i \in p_j} c_{ij} = b_j \qquad (40)$$

$$u_j = \frac{1}{m_j} \left(b_j + \sum_{i \in p_j} c_{ij} \right) \qquad (41)$$

Comment. If $u_j - c_{ij} < 0$, then assume that $s_{ij} = 0$ and repeat the procedure to define the rest s_{ij}.

5.4. Computational experiment

There was developed a computer program to assess the computational complexity of an algorithm with the help of which problems of varying complexity were solved. The complexity of M problem was estimated by the number of variables of the generalized dual problem, i.e.

$$M = \sum_{j=1}^{n} \left(m_j - 1 \right) \qquad (42)$$

Each problem was solved in two ways. In the first one, a generalized dual problem as a linear programming problem was solved to obtain lower estimate. In the second method, the lower estimates were obtained on the basis of a fixed partition based on expressions (40), (41). 5 problems for each dimension from 3 to 15 were solved.

The results are shown in Table 4.

The table shows that at small dimensions (from 3 to 6) the obtaining estimates by solving the problem of linear programming have less computational complexity. However, for large dimensions, obtain-ing estimates based on a fixed partition is more efficient.

6. CONCLUSION

The paper examines the application of the network programming theory for the project portfolio formation. To this end, the concept of a generalized dual problem as problems of network representation are introduced, for which the upper (lower) estimate is minimal (maximum) and it is proved that the generalized dual problem is a convex programming problem.

This result is used to develop a number of algorithms for the project portfolio formation, including, in particular, business-forming and business-supporting projects. The lower estimates of the portfolio value are obtained and it is shown that the results of solving the generalized dual problem can be used in branch and bound method, often used in solving discrete optimization problems.

Network programming method is used to solve other problems that arise in the project management, e.g., the development of programs for destruction of chemical weapons, providing road safety, and re-structuring of various enterprises (Burkov et al., 2017; Kondratjev et al., 2019; Burkov et al., 2011). The tools of the method can be applied in such areas of project management as project risk management, communications management, quality management, contract management.

Based on the results of the study, it's possible to suggest a universal scheme of the network programming method application for project management problems consisting of the following steps:

1. Building network representations for the target function and constraints with the same structure.
2. Setting initial values of dual variables.
3. Solutions of the assessment tasks.
4. Purposeful change of dual variables values.

Table 4. Computation results (assessing the computational complexity of the algorithm)

M	3	4	5	6	7	8	9	10	11	12	13	14	15
I method, T (m/sec)	0.5	0.7	2.0	5,0	10	21	40	60	100	150	200	300	500
II method, T (m/sec)	0.8	1.0	3.0	4,8	8	15	25	43	60	90	150	250	400

Promising area of the network programming method application can be also the development of effective algorithms for solving various traditional problems of discrete optimization.

These algorithms allow to develop universal program products that solve a wide range of problems of discrete and multi-extreme optimization, creation of intelligent control systems.

The network programming method has certain limitations on the scope of its application. They are associated with the difficulties of formalizing management decisions and taking into account project uncertainty and risk. Therefore to improve the efficiency and reliability of management decisions it is suggested to apply the network programming method in combination with the robust approach for assessing risk parameters.

References

Archer, N., & Ghasemzadeh, F. (1999). An integrated framework for project portfolio selection. Int. J. Project Manage. 17 (4), 207–216.

Artto, K.A., Martinsuo, M., & Aalto, T. (2001). Project Portfolio Management: Strategic Management through Projects, Helsinki, FI: Project Management Association Finland.

Burkov, V., & Burkova, I. (2003). Dichotomous programming method in optimization problems. Moscow, Russia: Central Economic and Mathematical Institute of the Russian Academy of Sciences (in Russian).

Burkov, V., Burkova, I., Popok, M., & Ovchinnikova, T. (2005). Method of network programming. Annals of the Institute of Management Problems under V.A. Trapeznikov of the Russian Academy of Sciences, 3, 25-27 (in Russian).

ПРИМЕНА ТЕОРИЈЕ МРЕЖНОГ ПРОГРАМИРАЊА ЗА ФОРМИРАЊЕ ПОРТФОЛИА ПРОЈЕКТА

Irina Burkova, Boris Titarenko, Amir Hasnaoui, Roman Titarenko

Извод

У раду се говори о примени мрежне (дихотомне) методе програмирања за решавање мулти-екстремних проблема и дискретних проблема оптимизације. Уведен је концепт генерализованог дуалног проблема и доказана теорема о његовој конвексности. Метода мрежног програмирања користи се за изградњу портфолиа пројеката на којима се заснива пословање, као и алгоритам за решавање потпуно зависних повезаних пројеката. Такође, овај метод се користи за решавање проблема изградње пословног портфолиа пројеката за који се добија нижа процена трошкова. Извршен је рачунски експеримент за процену предложеног алгоритма, који је показао да је за велике димензије проблема, овај приступ ефикаснији од решавања проблема методама линеарног програмирања.

Кључне речи: пројекат, портфолио пројеката, мрежно програмирање, уопштени дуални проблем

Burkova, I. (2009). A method of network programming in problems of nonlinear optimization. Automation and Remote Control, 70 (10), 1606–1612.

Burkov, V., Korobets, B., Minaev, V. & Tsepkin, A. (2017). Models, methods and

mechanisms for managing scientific and technical programs. Moscow, Russia: Bauman Moscow State Technical University, 1–202 (in Russian).

Burkov, V., Kondratjev, V. & Tsepkin, A. (2011). Mechanisms for improving road safety. Editorial URSS, Moscow, RU: 1–208 [in Russian].

Coesmans, P., Fuster, M., Schreiner, J,G., Goncalves, M., Huynink, S., Jaques, T., Pugasevskis, V., Sedlmayer, M., Thyssen, D., Tovb, A., Vukomanovic, M., & Young, M. (2018). Individual Competence Baseline for project, program and portfolio management (version 4). Zurich, Switzerland: International Project Management Association.

Gutjahr, W. (2011). Optimal dynamic portfolio selection for projects under a competence development model. OR Spectrum, 33, 173-206.

Hyväri, I. (2014). Project portfolio management in a company strategy implementation, a case study. Procedia – Social and Behavioral Sciences, 119, 229–236.

Kondratjev, V. & Tsepkin, A. (2019). Project management in the implementation of the road safety strategy. Scientific edition of Moscow Automobile and Road Construction State Technical University, 4 (59), 112–119 [in Russian].

Levine, H.A. (2005). Project Portfolio Management. San Francisco, CA, USA: Jossey-Bass. Maizlish, B., & Handler, R. (2005). IT portfolio management step-by-step: unlocking the business value of

technology. Hoboken, NJ, US: John Wiley & Sons.

Marcondes, G., Leme, R., & Silva, M. (2017). Using mean-Gini and stochastic dominance to choose project portfolios with parameter uncertainty. The Engineering Economist, 62 (1), 33–53.

Meredith, J.R., & Mandel, S.J. (2010). Project management: a managerial approach (7th ed.) Hoboken, NJ, US: John Wiley & Sons.

Momćilović, O., Djukic Petromanjanc, L., Doljanica, S., & Rajaković, J. (2014). Organizational context of project portfolio management. Annals of the University of Oradea, 3, 192–196.

Sharifghazvini, M. (2018). Integration of a new MCDM approach based on the DEA, FANP with MONLP for efficiency-risk assessment to optimize project portfolio by branch and bound: a real case study. Economic computation and Economic Cybernetics Studies and Research, 52 (1), 261–278.

Tavano, M., Khalili, K., & Abtahi, A.-R. (2013). A fuzzy multidimensional multiple-choice knapsack model for portfolio selection using an evolutionary algorithm. Annals of Operations Research, 206, 449–483.

Titarenko, B., Hasnaoui, A., Titarenko, R., & Buzuk, L. (2018). Robust data analysis in innovation project portfolio management. SPbWOSCE-2017, MATEC Web of Conferences 170, 01017.

Turner, J. R., & Müller, R. (2003). On the nature of the project as a temporary organization. International Journal of Project Management, 21, 1–7.

Qingguo, T. (2015). Robust estimation for spatial semiparametric varying coefficient partially linear regression. Stat Papers 56, 1137–1161.

Yang, F., Song, S., Huang, W., & Xia, Q. (2015). SMAA-PO: project portfolio optimization problems based on stochastic multicriteria acceptability analysis. Annals of Operations Research, 233, 535 –547.

Yousefi, V., Yakchali, S., Saparauskas, J., & Kiani, S. (2018). The impact made on Project Portfolio optimisation by selection of various risk measures. Inzinerine Ekonomika-Engineering Economics, 29 (2), 168–175.

Yu, L., Wen, S.W., & Lay, K.K. (2012). Genetic algorithm-based multi-criteria project portfolio selection. Annals of Operations Research, 197 (1), 71–86.