UDC 519.857.6
DOI: 10.15587/1729-4061.2023.274144

FEATURES IN SOLVING INDIVIDUAL TASKS TO DEVELOP SERVICE-ORIENTED NETWORKS USING DYNAMIC PROGRAMMING

Olha Kryazhych
Corresponding author
PhD, Senior Researcher
Department of Natural Resources*
Associate Professor
Ternopil Ivan Puluj National Technical University
Ruska str., 56, Ternopil, Ukraine, 46001
E-mail: economconsult@gmail.com
Victoria Itskovych
Postgraduate Student, Deputy Director of Department – Head of Department
Department for Digitalization, Strategic Development of the Field of Digitalization and Information Security
Department of Natural Resources*
Department of Information and Communication Technologies of Executive Body of Kyiv City Council
Kyiv City State Administration
Khreshchatyk str., 36, Kyiv, Ukraine, 01044
Kateryna Iushchenko
Postgraduate Student
Department of Information and Communication Technologies*
Oleksii Kuprin
Postgraduate Student
Department of Information and Communication Technologies*
*Institute of Telecommunications and Global Information Space of National Academy of Sciences of Ukraine
Chokolovsky blvd., 13, Kyiv, Ukraine, 03186

The object of this study is an approach to solving the problems of designing service-oriented networks that warn about emergencies using dynamic programming. The main issue is the complexity of algorithmization of processes that describe the achievement of an optimal solution in multi-stage nonlinear problems. The possibilities of applying the Bellman optimality principle for solving the set tasks for the purpose of their application in the field of engineering and technology are determined. Based on the Bellman functional equation, a model of the optimal number of sensors in the monitoring system for warning of emergencies was built.

A feature of the design is that using the classical Bellman equation, it is proposed to solve problems of various technical directions, provided that the resource determines what exactly makes it possible to optimize work in any way. Important with this approach is the planning of the action as an element of some problem with the augmented state. After that, the proposed structure in formal form extends to other objects.

A problem was proposed and considered, which confirmed the mathematical calculations, as a result of which an optimal plan for replacing the sensors of the system was obtained; and the possibilities of significant cost reduction were identified. In the considered example, an optimal plan for replacing the system sensors was compiled and the possibility of reducing costs by 31.9 % was proved.

The proposed option was used in the development of information technology for modeling a service-oriented network based on energy-efficient long-range protocols; some of the identified features were further developed in the design of a recommendation system for issuing loans and developing an interactive personnel training system

Keywords: sorting solutions, linear objective function, constraints, Bellman optimality principle, synthesis, optimization, discrete quantity, sensor, monitoring, emergency

Received date 21.12.2022
Accepted date 22.02.2023
Published date 28.02.2023

How to Cite: *Kryazhych, O., Itskovych, V., Iushchenko, K., Kuprin, O. (2023). Features in solving individual tasks to develop service-oriented networks using dynamic programming. Eastern-European Journal of Enterprise Technologies, 1 (4 (121)), 34–40. doi: https://doi.org/10.15587/1729-4061.2023.274144*

1. Introduction

Dynamic programming is a special method that is specifically adapted to optimize fast-evolving tasks in which an operation consists of elements with a strong influence on each other. The method of dynamic programming is the most common method for solving problems of optimal control. It is used both for solving problems of a linear objective function and for use for the analysis of nonlinear processes. In addition, dynamic programming is a general principle to solve optimization tasks with limitations. Constraints can be linear and nonlinear, with continuous and discrete variables, but under conditions of the possibility of their decomposition [1].

To solve practical tasks, it is important to use dynamic programming for an objective function given by a table [2].

The Bellman optimality principle [3] underlies dynamic programming and can significantly reduce sorting the solutions in multi-stage nonlinear problems [4]. It is associated with the problem of optimizing a complex system, which consists of many interdependent elements. Elements can include economic units that are part of a single system; nodes of a complex technical system; separate areas of production, construction, emergency zones, or combat operations [5].

Nevertheless, any complex system requires a separate approach when introducing control automation. And in this case, the question arises of how to manage individual elements of the system with maximum performance indicators. In this case, it is not enough to optimize each element separately and prescribe a control algorithm for this. Such a step will certainly lead to an incorrect result [2].

The latter proves the relevance of the research, which requires studying and analysis, especially now, in the process of informatization of society and the active development of digital technologies. After all, the tasks for the solution of which dynamic programming is used are implemented in the control algorithms of complex systems. In particular, this is used in service-oriented networks, smart cities, recommendation systems, modern personnel tools, and other applications. It is dynamic programming that makes it possible to solve such multi-step problems according to the approach of optimal control, that is, determining among all the permissible solutions the optimal one for the task set.

2. Literature review and problem statement

Dynamic programming is widely used by researchers to solve various problems of process automation [6–8]. In particular, in work [6], the features of algorithmization of dynamic programming problems are considered and it is noted that the processes of dynamic programming require optimization of the entire system that they describe. In addition, it is important to analyze the elements of the system and, often, solve additional problems in order to get an adequate solution. In [7], the algebraic style of dynamic programming over the data sequence is presented, which is very important for solving individual applied tasks, in particular, economic, management, algorithmization of various business processes. Paper [7] also details some aspects of the Bellman principle, including some approaches to programming at a convenient level of abstraction.

Among the cited works, study [8] looks somewhat innovative; it presents the tools and philosophy of dynamic programming in relation to the use of network models for building. The work implies not only the influence of stochastic processes and linear algebra, sufficient for practical specialists, but also the analysis and synthesis of features that are interesting for scientists and researchers. In particular, these are hedging and insurance stocks, stability theory for networks, and accelerated modeling methods, network workload models, features of the implementation of cellular communication protocols, data processing in complex networks. The reported equations of dynamic programming allow for a different look at the scope of application and make it possible to radically move from the plane of the economy to the plane of technical aspects of the functioning of complex systems.

Most often, with the help of dynamic programming, optimization problems are solved. That is why the consideration of optimization algorithms [9] makes it possible to implement applications in the fields of automatic control, signal processing and, again, communications and networks. In the cited work [9], it is not only emphasized that optimization algorithms in dynamic programming make it possible to solve the problem efficiently but also makes it possible to somewhat simplify the approach in practice using a programming language. That is, it is possible to implement directly into the management of a service-oriented network or apply individual elements when developing a recommendation system for choosing a solution. In this case, dynamic programming makes it possible to facilitate the process of algorithmization of the problem [10], especially those that use functions with a variable number of arguments and processing of tabular functions [11].

Bellman's principle of optimality raises the question of the optimality of a single element of the system in terms of the optimality of the entire system [12]. When making a decision at a separate stage, it is necessary to choose control at this stage with an eye to the future because it is the overall result that interests you. Any process has an end somewhere, that is, some planning horizon [13]. And based on work [3], the last stage «has no future». It is necessary to optimize it only from the standpoint of this stage. After that, they proceed to the optimization of the $(m–1)$ stage. In this case, we set the state from which the $(m–1)$ step (condition) begins. That is, there is a generalization of the level and the construction of some optimal path, as presented in [14]. Therefore, the function $W_i(S)$ is the conditional optimal gain of some function $W(S)$. Thus, the process of optimization using the method of dynamic programming unfolds from end to beginning, and then from beginning to end [12]. In different tasks, one can know either the initial state or the final state, or both. The Bellman principle has found practical application in the method when any action is planned as an element of some problem with the augmented state [15]. After that, it is enough to extend this structure to the stochastic problems of the system under study and apply the resulting algorithm to the problems of optimal work planning.

But most of the works focus on economic issues, leaving out the technical component. As a rule, technical issues are considered separately, or indirectly, due to the need to investigate a large number of variables. It is from these positions that the main research issue is to find the optimal path in solving a problem with a large number of variables, the transition between which occurs sequentially. Especially so if all variables or some of them are discrete. That is, the result at each step will influence the decisions obtained in other steps. While for economic processes such a task is widely studied [16, 17], for technical directions there are features of solving individual problems of dynamic programming.

3. The aim and objectives of the study

The aim of our study is to identify some features of solving individual problems of developing service-oriented networks that warn about emergencies using dynamic programming. This approach can also be designed to algorithmize processes in recommendation systems, automated systems for selecting and training personnel, etc.

The purpose of the work will be achieved using the following tasks:

– to analyze the possibility of applying the Bellman optimality principle to solve individual problems of developing service-oriented networks using dynamic programming in order to apply them in the field of engineering and technology;

– on the basis of the Bellman functional equation to build a model of the optimal number of sensors of the monitoring system for warning of emergencies for the application of this model in the development of a service-oriented network for warning about emergencies.

4. Materials and research methods

4. 1. The object and hypothesis of research

The object of research in this work is multi-step processes of algorithmization in solving problems of developing service-oriented warning networks in case of emergencies.

The subject of research is the use of dynamic programming tasks to ensure optimal management in the development of service-oriented networks.

The hypothesis of the study assumes that the classical Bellman equation can be used to solve problems of various technical directions, provided that the resource determines what makes it possible in any way to optimize the operation of the system as a whole.

4. 2. The classic problem of resource allocation

The tasks set are solved on the example of the development of a service-oriented network for alerting about the occurrence of an emergency, which receives information from sensors located in different parts of the city. Periodically, sensors need to be replaced due to malfunctions arising from weather factors, or for testing purposes. However, it makes no sense to purchase and store a large number of sensors in warehouses – technologies are changing, obsolete sensors are also being replaced by new ones. Given the above, it is necessary to determine the required number of sensors d_k, which will be needed in each of the n months of the planned period, with the provision of minimum costs for the purchase and maintenance of stocks. At the beginning of the period, there are z_0 sensors in stock. In each of the periods that is planned, no more than A sensors are purchased. But at the same time, no more sensors are stored in stock than B, including those returned for service. The costs associated with the storage and maintenance of sensors X_k, in some month k, is the sum of conditionally fixed costs C, UAH thousand, and variable costs V, UAH thousand, for each unit. Costs that are caused by the storage of one sensor for a month are equal to h UAH thousand.

According to the indicators for the monthly need for sensors (Table 1), and storage costs (Table 2), it is necessary to adjust the available stocks of sensors in stock, under the conditions of a variable number of sensors that are replaced monthly.

Table 1

Indicators of the monthly need for sensors

Indicator	Quantity
Planning period, n, month	6
The minimum required number of sensors per month (permanently), d_k, units	7
The maximum number of sensors purchased monthly, A, units	7
The maximum number of sensors stored in stock, B, units	6
Storage costs of 1 sensor for one month, h, c.u.	5
Number of sensors at the beginning/end of the planned period, z_0/z_n, units	2/0
Fixed costs, C, c.u.	4
Variable costs, V, c.u. per unit	3

Table 2

Storage costs

Volume, x_k, units	0	1	2	3	4	5	6	7
Storage costs, $C(x_k)$, c.u.	0	7	10	13	16	19	22	25

The research is based on the classical problem of resource allocation when there is some initial capital k_0. This approach can be considered formalized. That is, the initial capital can be distributed among several controlled objects $P_1, P_2, ..., P_n$. X_{ij} – the volume of funds that will be invested in the i-th year in some object j. As a result, the following effect will be obtained:

$$W_{ij}=f(X_{ij}). \quad (1)$$

In general, (1) is a nonlinear function, and it is necessary to distribute the initial capital (resource) so that the total effect for all objects for all years is maximum. That is,

$$W_{ij}=\sum f(X_{ij})\rightarrow \max,$$

$$\sum f(X_{ij})\leq k_0. \quad (2)$$

Given that the function W is nonlinear, expression (2) is a classical linear programming problem with a very large number of variables. In addition, in many cases, X_{ji} can have discrete values. And in this case, taking into account [3, 13], the above problem (1) can be solved sequentially by optimizing at each step.

For the first time the principle of optimality of such a task was stated in [3]. That is, optimizing a separate step, it is necessary to think about its consequences, leading to a common result.

The state S of some system involves one or more parameters of the system. For example, some kind of resource. Control U_i in the i-th step is some influence that the system experiences, it changes its state S. Then, taking into account the above, over the i-th step you can get some winnings, which is denoted by $\omega_i(S_i, U_i)$ while the state S goes into S':

$$S\rightarrow S'=\varphi_i(S, U_i). \quad (3)$$

Analyzing expression (3), it is assumed that the functions $\omega_i(S_i, U_i)$ and $\varphi_i(S, U_i)$ are known (Fig. 1).

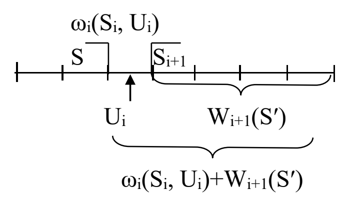

Fig. 1. Graphical analysis of the transition of the state of the system

For representations similar to (3), Bellman introduced the concept of a conditional optimal win $W_i(S)$. This function shows the optimal gain (the best result) obtained in all steps from the i-th to the end, if the i-th step begins with the state S. Then, according to the Bellman optimality principle, analyzing technical systems, if a decision is made at the i-th step, you can choose U_i so that the winnings are the maximum from the i-th step to the end.

4. 3. Graph problems based on the Bellman principle

Some problems of dynamic programming can be solved graphically. Most often [14], it is the search for some minimal path that will make it possible to achieve the goal with maximum success.

This type of task can be illustrated by the example of an unmanned aerial vehicle (UAV) that is gaining altitude and speed.

The UAV is at some altitude h_0 and flies at a speed of v_0. It is necessary to transfer it to height h_1 at a speed of v_1. Condition: $h_1>h_0$, $v_1>v_0$.

To solve the problem, you should divide the flight area from h_0 to h_1 into n parts:

$$\Delta h=\frac{h_1-h_0}{n},$$

$$\Delta v=\frac{v_1-v_0}{n}.$$

The battery charge is known when the system is converted to Δh at v=const and to Δv at h=const. That is, in each state, the UAV has only two possible directions of control (Fig. 2).

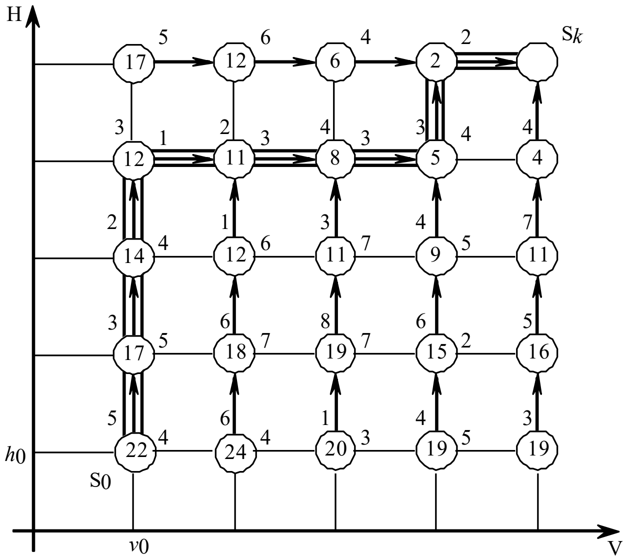

Fig. 2. Illustrating the solution of the graph problem

To represent a solution to the problem, you should start from the end: all nodes (states) are marked with values of conditional (for a given node) optimal battery charge consumption from this node to the end, and the conditional optimal control of the UAV is marked with arrows. These actions in a simplified form demonstrate the considered solution procedure based on the Bellman equation [13].

On the way from the final state to the initial is the result 22. This result is the optimal charge consumption of the UAV battery. Moving along the arrows from the initial state to the final state, you can get unconditional optimal control for transferring the UAV to the desired mode of operation (shown by a double line).

That is, any problem that boils down to finding the minimum path on the graph can be solved by dynamic programming.

4. 4. Bellman functional equation

The operation of any dynamic system can be represented as movement in phase space. This space of states of the system can be expressed by the coordinate vector: $S=(\xi_1, \xi_2, ..., \xi_L)$. And the management of the space of states, as noted in [3], can be step by step. That is, the process of controlling a complex dynamic system will consist of m steps, and direct control takes place at any i-th step. As noted above, the winning function can be represented as $\omega_i(S, U_i)$, S – the state before the i-th step, U_i – control in the i-th step.

It is worth remembering that the $\omega_i(S, U_i)$ value must be known before the start of dynamic programming. If the state before the i-th step was S, and then some kind of control U_i was performed, then the state of the system will already be described as $S'=\varphi_i(S, U_i)$.

However, this function should also be known. If they are not specified, they must be formulated. For example, enter the function of the conditional optimal win $W_i(S)$. This win is a win in all stages from start to finish if the i-th step starts with state S.

The number in m steps is investigated. It is assumed that from the $(i+1)$-th step relative to the system, optimal control is performed. Then the winning value will be $W_{i+1}(S')$. At the i-th step, arbitrary control U_i will be applied, then $\tilde{W}_i(S)$ is a suboptimal win. In order to get the optimal win from the i-th step to the end of the problem, you should change U_i so as to get the Bellman functional equation [13]:

$$W_i(S)=\max_{U_i}\left\{\omega_i(S,U_i)+W_{i+1}(S')\right\};\ S'=\varphi_i(S,U_i);$$

$$\underset{\text{unknown}}{W_i(S)}=\max_{U_i}\left\{\underset{\text{known}}{\omega_i(S,U_i)}+W_{i+1}\left[\underset{\text{unknown}}{\varphi_i(S,U_i)}\right]\right\}. \quad (4)$$

To solve equation (4), actions begin to be performed from the end:

1) $i=m$:

$$W_m(S)=\max_{U_m}\left\{\omega_m(S,U_m)\right\};$$

2) $i=m-1$:

$$W_{m-1}(S)=\max_{U_{m-1}}\left\{\omega_{m-1}(S,U_{m-1})+W_m\left[\varphi_{m-1}(S,U_{m-1})\right]\right\}.$$

Thus, moving from the end to the beginning, the following consecutive expressions can be obtained:

$$W_m(S),\ W_{m-1}(S),\ ...,\ W_1(S),$$

$$U_m(S),\ U_{m-1}(S),\ ...,\ U_1(S).$$

Thus, the transition to the initial state $W_1(S)$, you can substitute $S=S_0$ and $W_1(S_0)=W_{\max}$.

However, this is not the end of the task.

Now it is necessary to obtain unconditional optimal equations by way from beginning to end along a chain:

$$S=S_0\to U_1(S_0)=U_1^*\to\varphi(S_0,U_1^*)=$$
$$=S_1^*\to U_2(S_1^*)=U_2^*\to\varphi(S_1,U_2^*),$$

which ultimately gives the optimal solution:

$$U_1^*,U_2^*,...,U_m^*;W_{\max}.$$

This approach makes it possible to consider applied technical problems, for example, the problem of designing a service-oriented network [18], developing a recommendation system for a bank or credit institution [19], personnel selection systems for many heterogeneous parameters [20], or developing a model for the spread of environmental pollution [21, 22].

The computer experiment and research results were obtained in the MATLAB package.

5. Results of research on solving individual problems of service-oriented network development using dynamic programming

5. 1. Analysis ofthe possibilities of applying the Bellman optimality principle to solve the problem set

Using Table 1 and Table 2, it is possible to build a model for planning the purchase, replacement, and storage of sensors in accordance with the algorithm for modeling dynamic programming problems:

1. The choice of how to divide the management process into steps.

The process of planning the purchase, replacement, and storage of sensors for n months makes it possible to consider it as an n-step process, that is, the number k of the step selected is the number k of the month $\left(k=\overline{1;n}\right)$.

2. Selection of parameters that characterize the state of the system at the beginning of the k-th step and control variables in step k:

– Z – the initial state of the system – the level of stock of finished products at the beginning of the k-th month;

– j_k – the final state of the system – the level of stock of finished products at the end of the pre-th month;

– $X \cdot k(Z)$ – conditionally optimal control at the k-th step – the volume of products produced per month k to ensure the minimum possible costs for purchase, replacement, and storage.

5. 2. Building a model of the optimal number of sensors of the monitoring system for warning of emergencies

For our task, the Bellman optimality principle can be formulated as follows: the optimal plan for the purchase of sensors for any month should have the property of optimality. Limitations should also be taken into account: the cost of purchasing, maintaining, and storing without taking into account how many sensors were purchased last month.

In this case, the objective function can be written as a sum:

$$F=\sum_{k=1}^{n}\left(C\left(x_k\right)+h\cdot j_k\right)\rightarrow \min. \quad (5)$$

The main recurrent ratio according to Bellman makes it possible to find conditionally optimal values of the objective function at some step with its conditionally optimal values known in the previous step.

Let $F_k^*\left(z\right)$ be the minimum (conditionally optimal) costs for the purchase and storage of products for the k-th month, provided that the level of sensor stocks at the beginning of the month is z.

The computational process is performed according to the reverse scheme (from end to beginning), taking into account that the total costs in the following months should be minimal. Thus, the Bellman recurrent ratio is:

$$F_k^*\left(z\right)=\min_{x_k\in U}\left\{C\left(x_k\right)+h\cdot j_k+F_{k+1}^*\left(j_k\right)\right\}, \quad (6)$$

where U is the set of values x_k under the following conditions:

a) $x_k\leq A$ – you can purchase no more than A sensors;

b) $z\left(j_k\right)\leq B$ – the stock of sensors cannot be more than the number B of places for these sensors in stock.

At the end of the period, the stock of sensors should not be more than 0, and in step $k+l$ nothing is bought or stored. Then, for the last step, the Bellman equation for the problem will look like:

$$F_n^*\left(z\right)=\min_{x_n\in U}\left\{C\left(x_n\right)+0+F_{n+1}^*\left(0\right)\right\}$$

or

$$F_n^*\left(z\right)=\min_{x_n\in U}\left\{C\left(x_n\right)\right\}. \quad (7)$$

And the level of stocks at the end of the k-th month will be equal to:

$$j_k=z-d_k+x_k. \quad (8)$$

Conditional process optimization can be carried out sequentially for steps k, k–1, k–2, ..., 2, 1 using an optimization Table 3.

Table 3

Optimization Table for step k

$z \cdot j_k$	0		1		...	B		$F_k^*(z)$	$X_k^*(z)$
0	x_k	$F_k(z)$	x_k	$F_k(z)$	...	x_k	$F_k(z)$	$F_k^*(0)$	$X_k^*(0)$
...	...	...	...	...	...	...	...	...	...
B	x_k	$F_k(z)$	x_k	$F_k(z)$	...	x_k	$F_k(z)$	$F_k^*(B)$	$X_k^*(B)$

The value of x_k is the number of sensors purchased in the k-th month, determined from the formula of states (8):

$$x_k=j_k-z+d_k. \quad (9)$$

When filling in the optimization tables, the fulfillment of the conditions of formula (6) should be taken into account.

In the problem that is analyzed, at least d_k=7 sensors are constantly used monthly; respectively, formula (9) takes the form:

$$x_k=j_k-z+7.$$

When optimizing the last step k=6, the Bellman equation takes the form:

$$F_6^*\left(z\right)=\min_{x_n\in U}\left\{C\left(x_6\right)\right\}=c\left(x_6\right).$$

The peculiarity of this problem is that at the end of the planning period the number of sensors should be zero, that is, the final state at this step will be j_6=0, and Table 3 will only have a zero column.

Then, step by step, the Bellman equation will look like:

a) k=5:

$$F_5^*\left(z\right)=\min_{x_5\in U}\left\{C\left(x_5\right)+h\cdot j_5+F_6^*\left(j_5\right)\right\};$$

b) k=4:

$$F_4^*\left(z\right)=\min_{x_4\in U}\left\{C\left(x_4\right)+h\cdot j_4+F_5^*\left(j_4\right)\right\};$$

c) k=3:

$$F_3^*\left(z\right)=\min_{x_3\in U}\left\{C\left(x_3\right)+h\cdot j_3+F_4^*\left(j_3\right)\right\};$$

d) k=2:

$$F_2^*\left(z\right)=\min_{x_2\in U}\left\{C\left(x_2\right)+h\cdot j_2+F_3^*\left(j_2\right)\right\};$$

e) k=1:

$$F_1^*\left(z\right)=\min_{x_1\in U}\left\{C\left(x_1\right)+h\cdot j_1+F_2^*\left(j_1\right)\right\}.$$

Given that the number of sensors at the beginning of the research period is known and is $z=2$, the optimization Table for the 1st step is also reduced, but to one line. By calculating the primary data, you can get the minimum cost of purchasing and storing sensors. And they amount to UAH 144 thousand.

For unconditional optimization of processes in the MATLAB package based on step-by-step tables, the calculation takes place from the end to the beginning, that is, from steps $k=1$ to $k=6$. The results were obtained that in order to achieve a minimum cost of UAH 144 thousand for the purchase and maintenance of sensors in stock for months, you should:

– in the 1st month, purchase 5 sensors, install 7 sensors;
– in the 2nd month, buy 7, install 7;
– in the following months – the same as in the 2nd month.

Control of calculations can be carried out using formula (5):

$$F=\sum_{k=1}^{n}\left(C\left(x_k\right)+h\cdot j_k\right)=$$
$$=\left(19+5\cdot 0\right)+\left(25+0\right)+\left(25+0\right)+$$
$$+\left(25+0\right)+\left(25+0\right)+\left(25+0\right)=144 \text{ c.u.}$$

Now, with the help of an optimal program for buying and installing sensors, you can adjust the plan for the conditions of a variable number of sensor installations for the system (Table 4).

Table 4

Variable number of sensors by month

n, month	1	2	3	4	5	6
d_k, sensors that need to be replaced	2	5	4	6	7	4

As a result of the calculations, we obtain that in order to achieve an optimal plan for purchasing sensors for the system under the conditions of a variable number of sensors that need to be replaced, according to the developed model, it is necessary:

– in the 1st month, to purchase 0 sensors, replace 2 sensors;
– in the 2nd month, to purchase 5 sensors, replace 5 sensors;
– in the 3rd month, to purchase 4 sensors, replace 4;
– in the 4th month, to buy 6, replace 6;
– in the 5th month, to purchase and replace 7 sensors;
– in the 6th month, to purchase and replace 4 sensors.

In this case, the volume of costs will equal, according to formula (5):

$$F=\sum_{k=1}^{n}\left(C\left(x_k\right)+h\cdot j_k\right)=$$
$$=\left(0+5\cdot 0\right)+\left(19+0\right)+\left(16+0\right)+$$
$$+\left(22+0\right)+\left(25+0\right)+\left(16+0\right)=98 \text{ UAH thousand.}$$

That is, costs are reduced by UAH 46 thousand or by 31.9 %, which confirms the achievement of the goal of reducing the cost of maintaining sensors and makes it possible to offer an optimal maintenance plan for network sensors.

6. Discussion of results of the study of the peculiarities of solving individual problems of service-oriented network development

It is impossible not to agree with [16] that resource allocation is the most common operation. And this can explain the results obtained: it is possible to expand the understanding of the resource by the fact that it is not just a physical or abstract quantity used to produce a useful product [17]. Based on the above task, the resource is time, warehouse area, that is, everything that makes it possible to optimize the work on the provision of services or the production of goods. This was considered on the example of optimizing the number of sensors by months when the number of sensors in stock is reduced to 2–6 pcs., instead of the required 7 pcs.

This expansion of the understanding of the resource, confirmed by example (Table 4), makes it possible to change the view of Bellman's equations and confirm the possibility of its application in the development of technical systems. This may be information technology for managing processes or complex systems. Given that the resource is always limited [16], the task is to distribute the resource between the individual elements of the system so that the total effect is maximum.

And taking into account [5, 15], it is possible to combine economic and technical approaches using the Bellman optimality principle [13], as was given when calculating both the required number of sensors and the cost of storing and maintaining these sensors.

The results of the example make it possible to assert that the initial amount of resources to be distributed is a finite value. However, a simple plan, without optimization, brings only stable costs with unstable sensor replacement. The resource used does not generate income, and costs are stable. This situation can be changed because each step works for the total gain [14]. The state of the system is the number of sensors before the first step. And from this, calculations are carried out simultaneously on the function of costs, and on the optimality of the work on the replacement of sensors.

The limitation of this task is that the presented task still partially remains economic. Because the resource is distributed by objects. For example, if we consider the distribution of resources between n objects, when a winning function is given for each object, then such a task is equivalent to the considered task of optimizing the plan for acquiring sensors in n steps. However, the latter makes a transition to the terms of reference – optimizing the action plan to replace sensors by months.

The above problem can be considered as four subtasks that have their characteristics. In [6], the authors provide the best possible solution to individual problems by dynamic programming, but not always the best solution is optimal for a specific applied problem. For example, conditional process optimization (Table 3) can be carried out not sequentially, given that the sensors are replaced not one after another but depending on the characteristics of operation. Then, accordingly, there will be a replacement of steps and the result will be different. This is indicated in [5] at the level of the hypothesis. In this paper, this was confirmed and allowed us to prove our hypothesis.

It must be said that the above problem was solved by the graph method [6]. Similar to the above, Fig. 2 showed a graph where all sensors are indicated. The graph allowed us to confirm the results obtained, as well as to lay the foundation for the development of research – the identification of exactly those sensors that need priority replacement. Because the latter was a lack of practical expectations of this approach. After all, in practical implementation, it is important to perform work not only optimally but also in a timely manner and ahead of emergencies in the system.

7. Conclusions

1. It is revealed that using the classical Bellman equation, it is possible to solve problems of various technical directions,

recognizing as a resource what makes it possible in any way to optimize the work on the provision of services or the production of goods. It is important to plan the action as an element of some problem with the augmented state and extend this structure to other objects.

2. On the basis of the Bellman functional equation, a model of the optimal number of sensors of the monitoring system for emergency warning was built for the application of this model in the development of a service-oriented network for warning about emergencies. As a result, not only an optimal plan for replacing the system sensors was obtained but also the possibility of significantly reducing the cost by 31.9 % for the purchase and storage of new sensors. In addition, the calculations allow us to offer an optimal maintenance plan for network sensors, if you solve the problem by the graph method.

Conflicts of interest

The authors declare that they have no conflicts of interest in relation to the current study, including financial, personal, authorship, or any other, that could affect the study and the results reported in this paper.

Funding

The study was conducted without financial support.

Data availability

All data are available in the main text of the manuscript.

References

1. Minu, M. (1990). Matematicheskoe programmirovanie. Teoriia i algoritmy. Moscow: Nauka. Gl. red. fiz.mat. lit., 488.
2. Dreyfus, S. (2002). Richard Bellman on the Birth of Dynamic Programming. Operations Research, 50 (1), 48–51. doi: https://doi.org/10.1287/opre.50.1.48.17791
3. Bellman, R. E, Kalaba, R. E. (1959). Dynamic Programming and Feedback Control. RAND Corporation, 1778.
4. Dreyfus, S. (2003). Richard Ernest Bellman. International Transactions in Operational Research, 10 (5), 543–545. doi: https://doi.org/10.1111/1475-3995.00426
5. Kamien, M. I., Schwartz, N. L. (1991). Dynamic Optimization: The Calculus of Variations and Optimal Control in Economics and Management. Amsterdam: Elsevier, 259–263.
6. Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C. (2001). Introduction to Algorithms. MIT Press & McGraw-Hill, 1184.
7. Giegerich, R., Meyer, C., Steffen, P. (2004). A discipline of dynamic programming over sequence data. Science of Computer Programming, 51 (3), 215–263. doi: https://doi.org/10.1016/j.scico.2003.12.005
8. Meyn, S. (2007). Control Techniques for Complex Networks. Cambridge University Press. doi: https://doi.org/10.1017/cbo9780511804410
9. Boyd, S. P., Vandenberghe, L (2004). Convex Optimization. Cambridge University Press, 727. doi: https://doi.org/10.1017/cbo9780511804441
10. Sniedovich, M. (2006), Dijkstra's algorithm revisited: the dynamic programming connexion. Journal of Control and Cybernetics, 35 (3), 599–620.
11. Vasilev, O. M. (2022). Programuvannia movoiu Python. Ternopil: Navchalna kniga – Bogdan, 504.
12. Dreyfus, S. (2002). Richard Bellman on the Birth of Dynamic Programming. Operations Research, 50 (1), 48–51. doi: https://doi.org/10.1287/opre.50.1.48.17791
13. Bellman, R. (2003). Dynamic programming (Dover ed). Mineola: Dover Publications, 384.
14. Jones, M., Peet, M. M. (2021). A generalization of Bellman's equation with application to path planning, obstacle avoidance and invariant set estimation. Automatica, 127, 109510. doi: https://doi.org/10.1016/j.automatica.2021.109510
15. Jones, M., Peet, M. M. (2021). Extensions of the Dynamic Programming Framework: Battery Scheduling, Demand Charges, and Renewable Integration. IEEE Transactions on Automatic Control, 66 (4), 1602–1617. doi: https://doi.org/10.1109/tac.2020.3002235
16. Dixit, A. K. (1990). Optimization in economic theory. Oxford: Oxford University Press, 164.
17. Miao, J. (2014). Economic Dynamics in Discrete Time. MIT Press, 134.
18. Kriazhych, O. O., Itskovych, V. Ye. (2022). Internet rechei v upravlinni skladnymy systemamy. Naukovi pidsumky 2022 roku. Kharkiv: PC TECHNOLOGY CENTER, 19.
19. Kuprin, O. M. (2022). Algorithmization of processes in recommendation systems. Mathematical Machines and Systems, 1, 71–80. doi: https://doi.org/10.34121/1028-9763-2022-1-71-80
20. Yushchenko, K. S. (2022). Approach to automating the recruitment process using 3D resumes. Mathematical Machines and Systems, 2, 29–39. doi: https://doi.org/10.34121/1028-9763-2022-2-29-39
21. Kryazhych, O., Kovalenko, O. (2019). Examining a mathematical apparatus of Z-approximation of functions for the construction of an adaptive algorithm. Eastern-European Journal of Enterprise Technologies, 3 (4 (99)), 6–13. doi: https://doi.org/10.15587/1729-4061.2019.170824
22. Trofymchuk, O., Adamenko, O., Trysniuk, V. (2021). Heoinformatsiini tekhnolohii zakhystu dovkillia pryrodno-zapovidnoho fondu. Instytut telekomunikatsii ta hlobalnoho informatsiinoho prostoru, Ivano-Frankivskyi natsionalnyi tekhnichnyi universytet nafty ta hazu. Ivano-Frankivsk: Suprun V. P., 343.