SAINT-PETERSBURG UNIVERSITY

Artem Sergeevich SHARLAI

SEARCH FOR THE OPTIMAL ROAD ALIGNMENT ON THE TERRAIN IN TERMS OF CONSTRUCTION COST

Scientific specialization
1.2.2. Mathematical modeling, numerical methods and complexes of programs

Dissertation is submitted for the degree of candidate of physical and mathematical sciences

Translation from Russian

Scientific supervisor:
doctor of Physical and Mathematical Sciences
docent Majid E. Abbasov

Saint-Petersburg
2024

Contents

INTRODUCTION

Topicality of the topic and degree of development of the problem in the literature

One of the essential problems in the geometric design of roads, watercourses, pipelines and other transportation networks is the determination of the optimal path [63] in terms of construction cost. Problems of this type naturally arise before various private organizations, government agencies and military structures, is a subject of study for many researchers. Such problems are not only found in civil engineering, but also in other fields such as robotics, space exploration, etc. [47, 89]. Due to the high importance of the problem, many effective methods have been developed to solve the problem. These methods are usually based on graph theory. Here we can, for example, mention the Cost Path Analysis method popular among engineers, which is based on the construction and analysis of a cost lattice. One of the most frequently used is Dykstra's [62] algorithm. To improve the accuracy of the solution when applying this algorithm, we have to increase the density of the lattice. This leads to a sharp increase in computation time and in many cases makes this approach practically inapplicable. To overcome this drawback, various heuristic methods have been proposed, such as the A* [50, 91, 95] algorithm, which is a modification of Dijkstra's algorithm that uses a heuristic function to reduce the number of computations. Another idea is based on constructing random trees in such a way that they expand rapidly to cover the domain under study. Here we can mention RRT [64, 76], RRT* [100], RRT connect [77], T-RRT [71] and others using the same approach [79, 94]. There are many other heuristic procedures for solving the [46, 56, 69, 93, 101] problem. Such methods lead to a satisfactory result, the quality of which usually cannot be guaranteed. In this dissertation study, we propose a method for solving the problem that ensures the optimality of the obtained path, based on variational principles.

Purpose of the study

The purpose of this study is mathematical modeling of the cost of building a road connecting two given points: the starting point, from which the construction materials necessary for laying the path are transported, and the final point. Such a model allows us to rigorously formulate the problem of finding the cost-optimal path. Also the purpose of the work is to analyze the obtained model with the conclusion of the conditions that must be satisfied by the desired trajectory, as well as the construction of methods and algorithms for solving the resulting problem, as well as proving the existence and uniqueness of its solution. Summarizing the above, the global goal of the work is to present mathematical tools for structures and decision makers in issues related to road construction or areas that allow similar mathematical formalization, for more efficient use of resources.

Main tasks

One of the main problems that this dissertation research is aimed at solving is the construction of a mathematical model for the problem of obtaining a cost-optimal path connecting two given points. In order to mathematically formalize the problem, it is necessary to identify the main characteristics on which the path cost depends.

The model is specified by means of an integral cost functional, which defines a mapping between admissible curves and their cost. For this functional, we need to obtain a necessary minimum condition by which the desired optimal path can be determined. We need to propose methods for solving the resulting condition, as well as to study the existence and uniqueness of its solution.

Scientific novelty

In this dissertation work, the problem of finding the optimal construction cost of a path connecting two given points is reduced to a variational calculus problem. The integral cost functional that defines the developed model takes into account the delivery cost of construction materials and the cost of their installation as the main quantities on which the final cost of the entire path depends. The resulting functional contains a summand with a double integral, which is reduced to a simpler form after additional transformation. For the problem obtained in this way, the necessary condition of minimum is derived using the apparatus of calculus of variations, which has the form of an integro-differential equation. Thus, it is shown that the optimal trajectory satisfies the specified integro-differential equation and two boundary conditions. Under some additional conditions, the uniqueness of the solution is proved, and the question of its existence is investigated with the help of Schauder's fixed point principle. Approximate methods for solving the resulting boundary value problem are developed, allowing to obtain the answer in the form of an algebraic or trigonometric polynomial, and a numerical method of solution is constructed, using the ideas of linearization, the shooting method, and the finite difference method.

Research Methods

With the help of mathematical modeling apparatus, an integral cost functional is constructed, the argument in which is a function describing the pathway trajectory. For the formation of the functional, the main values affecting the cost of the path are selected - the cost of delivery of construction materials and the cost of works on their laying. The natural assumption that the cost of laying a unit of road length depends on the distance from the starting point, which is taken as a material base, is used. The apparatus of calculus of variations, methods of higher algebra, algorithms from the field of mathematical programming, numerical methods, theory of differential equations and functional analysis are used to find the optimal function. Optimality conditions that take into account the specificity of the constructed functional are derived.

They are analogous to the classical Euler-Lagrange conditions, but lead not to differential but to integro-differential equations. Numerical methods of finding solutions to systems of nonlinear algebraic equations, decomposition of the desired trajectory by a system of basis functions, as well as the means of the MATLAB mathematical package and the Python programming language are used in the construction of methods for solving the resulting boundary value problem. To prove the existence and uniqueness of the solution, the concepts of uniformly continuous operator, uniform continuity and uniform boundedness and compactness of the set of functions are used.

Theoretical significance and practical relevance

The results obtained in this paper were obtained by the author personally and have theoretical significance for research in the field of civil engineering and other areas in which the problems of constructing an optimal in one sense or another trajectory arise. The approach proposed in the paper makes it possible to lay cost-optimal railroads, highways, pipelines and other transportation infrostructure objects connecting two given points. This makes it possible to solve one of the most important tasks of planning the construction of these objects in the most cost-effective way. On the basis of the constructed model and developed methods it is possible to create a modern software product that allows to obtain a theoretically justified optimal solution of the problem under study.

As it has already been noted, problems from other fields, such as robotics [78, 90] for example, can lead to similar mathematical formulations. Therefore, the results obtained in this paper can be applied not only in the framework of road construction, but also for a wider range of problems.

Boundaries of the study

The study is conducted under the assumption that the height difference on the terrain is insignificant and can be neglected. At the same time, it should be noted that within the framework of the proposed model the terrain can be taken into account by using the construction cost function, which depends on the terrain.

Subject of the study

The subject of the paper is the problem of obtaining the optimal construction cost of a trajectory connecting two given points.

Object of study

The object of the study is the integral functional of track construction cost. The paper deals with the problems of construction of this functional, as well as methods of its minimization.

Provisions for defense

Let us formulate the main results obtained in the paper:

- A method of mathematical modeling of building a cost-optimal road connecting two given points is developed. A mathematical formalization is proposed, within the framework of which a model defined by an integral cost functional is constructed.

- A necessary condition for the minimum of the constructed functional, which takes into account its specificity, is formulated and proved. This condition has the form of an integro-differential equation.

- The existence and uniqueness theorems of the obtained integro-differential equation are formulated and proved.

- Approximate and numerical methods for solving the obtained equation based on the approaches of functional analysis, as well as the apparatus of computational mathematics are developed. The software implementation of the algorithms in MATLAB mathematical package and Python programming language is proposed.

Approbation of the results

The main results of the dissertation work were published in highly rated scientific journals

- Bulletin of St. Petersburg University. Applied mathematics. Informatics. Control processes,
- Mathematical Modeling (M.V. Keldysh Institute of Applied Mathematics of the Russian Academy of Sciences),

and also reported at international conferences

- International Conference «XIV International Conference "Optimization and Applications" (OPTIMA-2023)», Petrovac, Montenegro, September 18-22, 2023.
- 5th International Conference on Problems of Cybernetics and Informatics (PCI 2023), Baku, Azerbaijan, 28-30 August 2023.
- The 8th International Conference on Control and Optimization with Industrial Applications (COIA-2022), Baku, Azerbaijan 24-26 August 2022

and seminars

- Workshop on the intersections of computation and optimisations, Canberra, Australia, November 24, 2021.
- Seminar of the Department of 13 "General Scientific Disciplines" of the Military Academy of Logistics named after Army General A. V. Khrulev, St. Petersburg, Russia, November 25, 2021. V. Khrulev Military Academy of Logistics, St. Petersburg, Russia, November 25, 2021.

In addition, this research was supported by the experts of the Russian Science Foundation, who supported the project 23-21-00027 "Search for optimal trajectory using artificial intelligence algorithms".

Publications

The results have been published in three articles in Russian and international peer-reviewed scientific journals (see [1–3]) and in several abstracts of international scientific conferences (see [38–40]), the list of which is presented above.

Main scientific results

- The method of mathematical modeling of construction of a cost-optimal road connecting two given points, see item 1 of the paper [3], work item 2 [2], work item 2 [1], work [38] from the list of publications of the author of the dissertation (proposed personally by the author of the dissertation)
- For the mathematical formalization within which the model defined by the integral cost functional is constructed, see paragraph 1 of the [3] paper, item 2 of the work [2], item 2 of the work [1] from the list of publications of the author of the thesis (proposed personally by the author of the thesis)
- The necessary condition for the minimum of the constructed functional, which takes into account its specificity. This condition takes the form of an integro-differential equation, see item 2 of the paper [3] from the list of publications of the author of the thesis (personal contribution is at least 80%).
- Existence and uniqueness theorems for the obtained integro-differential equation, see item 3 of the paper [1] from the list of publications of the author of the thesis (personal contribution is at least 80%).
- Approximate and numerical methods for solving the obtained equation, based on the approaches of functional analysis, as well as the apparatus of computational mathematics, see item 3 of the work [3], items 3 and 4 of the work [2], items 2, 3, 4 of the work [1], the work [39] from the list of publications of the author of the thesis (personal contribution is at least 80%).

- Program implementation of the constructed algorithms in MATLAB mathematical package and Python programming language, see item 3 of [3], item 4 of [2], item 4 of [1] from the list of publications of the dissertation author, as well as Appendix A.1 in the dissertation itself (personal contribution is at least 80%).

Provisions for defense

Let us formulate the main results obtained in the paper:

- A method of mathematical modeling of building a cost-optimal road connecting two given points is developed. A mathematical formalization is proposed, within the framework of which a model defined by an integral cost functional is constructed.

- A necessary condition for the minimum of the constructed functional, which takes into account its specificity, is formulated and proved. This condition has the form of an integro-differential equation.

- The existence and uniqueness theorems of the obtained integro-differential equation are formulated and proved.

- Approximate and numerical methods for solving the obtained equation based on the approaches of functional analysis, as well as the apparatus of computational mathematics are developed. The software implementation of the algorithms in MATLAB mathematical package and Python programming language is proposed.

CHAPTER 1

Supporting information

Let us first briefly summarize the supporting information necessary for further exposition.

1.1 Some background on functional analysis

We'll be working in the next normalized prostranzas:

- The space of continuous functions $\mathbb{C}[0,l]$ with norm

$$||x|| = \max_{t\in[0,l]} |x(t)|.$$

- The space $\mathbb{C}^k[0,l]$ k- times continuously differentiable functions with norm

$$||x||_{\mathbb{C}^k[0,l]} = \sum_{i=0}^{k} \max_{t\in[0,l]} |x^i(t)|.$$

- The space $\widetilde{\mathscr{L}_p}[0,l]$ of functions continuous on $[0,l]$ with norm

$$||x||_p = \left(\int_0^l |x(t)|^p \, dt\right)^{\frac{1}{p}}, \quad p \in [1,\infty).$$

Definition 1.1.1 *Let A, B be two sets of the normalized space $\mathbb{X}$. A is called dense in B if $B \subset \bar{A}$, where $\bar{A}$ –the closure of set A. A is called everywhere dense if $\mathbb{E} = \bar{A}$.*

Let $\mathbb{E}$ – Euclidean space

Definition 1.1.2 *A system of elements* $\{x_i\} \subset \mathbb{E}$ *is called complete if and only if the set of all possible linear combinations of its elements is everywhere dense in* $\mathbb{E}$*.*

Definition 1.1.3 *The complete orthogonal system* $\{x_i\}$ *of the Euclidean space* $\mathbb{E}$ *is called an orthogonal basis.*

The space $\widetilde{\mathscr{L}_2}[0,l]$ of functions continuous on $[0,l]$ is Euclidean. of functions is Euclidean. In it we can introduce the scalar product as follows:

$$\langle x, y\rangle = \int_0^l x(t)y(t)\,dt.$$

The most important orthogonal basis in this space is the trigonometric system consisting of functions

$$\frac{1}{2},\ \cos\frac{2\pi k}{l}t,\ \sin\frac{2\pi k}{l}t,\ k = 1, 2, \ldots.$$

Definition 1.1.4 *A function* $x(t)$ *defined on* $[0,l]$ *is called finite if there exists* $[a;b]\ :\ 0 < a;\ b < l$ *outside of which* $x(t) \equiv 0$ *(the function is finite on* $(-\infty;+\infty)$ *if it is zero outside some segment).*

Theorem 1.1.1 *The set of finite, infinitely differentiable on* $[0,l]$ *of functions is dense in* $\widetilde{\mathscr{L}_p}[0,l]$*.*

Inquiry 1.1.1.1 *The set of finite, continuously differentiable on* $[0,l]$ *of functions is dense in* $\widetilde{\mathscr{L}_p}[0,l]$*.*

A detailed statement and proofs of the above results can be found in [18, 21, 25, 31].

1.2 Some information from higher algebra

Definition 1.2.1 *Let's* $x_1, \ldots, x_{n+1} \in \mathbb{R}$. *Matrix*

$$V(x_1, \ldots, x_{n+1}) = \begin{pmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^n \\ 1 & x_2 & x_2^2 & \ldots & x_2^n \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & x_{n+1} & x_{n+1}^2 & \ldots & x_{n+1}^n \end{pmatrix}$$

is called the Vandermonde matrix.

Vandermonde's definition

$$\det V(x_1, \ldots, x_{n+1}) = \prod_{1 \le j < i \le n} (x_i - x_j).$$

For the Vandermonde determinant to be zero, it is necessary and sufficient that there exists at least one pair (x_i, x_j) such that $x_i = x_j$ at $i \neq j$.

A detailed statement and proofs of the above results can be found in [11, 14, 22, 23].

1.3 Some information from calculus of variations

Let $F(x, y, y')$ be given a function $F(x, y, y')$, continuous together with its partial derivatives on all three arguments x, y, y' inclusive. Let also be given two points $A(x_1, y_1)$ and $B(x_2, y_2)$ in the plane Oxy. Any curve expressed by the equations

$$y = y(x),$$

where $y \in \mathbb{C}^1[x_1, x_2]$, passing through the points A and B ($y(x_1) = y_1$, $y(x_2) = y_2$) will be called admissible. Let us formulate the simplest problem of variational calculus. Among all admissible curves, we need to determine the one along which the integral

$$J = \int_{x_1}^{x_2} F(x, y, y')dx$$

takes the largest value.

The method of variations is used to solve this problem. Let us briefly describe it. Let $\eta(x)$– a continuously differentiable finite function defined on the segment $[x_1, x_2]$. The variation of the functional J in y is called the quantity

$$\delta J = \frac{d}{dt} J(y + t\eta)\Big|_{t=0}.$$

Theorem 1.3.1 *For an admissible function $y = y(x)$ to be a minimum of a functional J, it is necessary that the variation*

$$\delta J = 0$$

for any finite continuously differentiable function on the segment $[x_1, x_2]$.

A detailed presentation and proofs of the above results can be found, for example, in [9, 15, 24, 34–36, 84, 88].

CHAPTER 2

Problem statement and necessary conditions of minimization

The main subject of this study is the problem of obtaining the optimal cost-optimized cost-optimal track path construction. Such problems arise in solving a wide range of practical tasks, such as, for example, road construction, robotics, laying pipelines and other transport networks, and therefore arise before various private organizations, government agencies and military organizations. There are a large number of methods used by researchers to solve the problem, most of which are heuristic in nature. For example, one of the most popular engineering approaches to solving this problem is the "textitCost Path Analysis" method, which is based on the construction and analysis of a cost lattice (see [?, 52, 98]). This paper proposes a different path based on the ideas, apparatus and approaches of mathematical modeling. We propose a mathematical formalization of the original problem, which leads to the problem of minimizing the integral cost functional, the argument of which is a function describing the path trajectory. The obtained functional is rewritten in a simpler form after some additional transformations. Thus, the problem is reduced to a variational calculus problem, for which we can derive a necessary optimality condition that takes into account the specificity of the given functional. of this functional. It should be noted that this condition is not differential, as the classical Euler-Lagrange conditions are. Euler-Lagrange conditions, but of an integro-differential equation, which requires the construction of methods for its solution, as well as the elucidation of conditions ensuring the existence and uniqueness of the solution. All these questions constitute the essence of the present work.

Let us begin with the problem statement, as well as with the formulation and discussion of the basic assumptions under which the model is constructed and the integral cost functional is derived.

2.1 Problem statement and basic assumptions

Let the coordinates of the start and end points O and A be given, which need to be connected by a road, spending a minimum amount of money on construction. It is natural to assume that the total construction cost consists of two components:

- delivery costs of building materials;
- paving costs.

To calculate these components, we need additional assumptions. Let us formulate them.

- Delivery of construction materials is always carried out from the from the starting point and on an already constructed road section.

We consider that construction materials are brought from point O, which serves as a material base. In this case, their transportation to the current location of the construction site is carried out exclusively along the already completed road section, i.e. takes place in the same conditions throughout the construction process. It should also be noted that the price of delivery depends on the distance from the base and the volume of transported material.

- The road construction technology is the same at any point along the trajectory.

Since the road paving technology is the same at any point along the trajectory, the amount of materials required to construct a unit length of road is constant. Therefore, it is possible to introduce a constant α equal to the delivery cost per unit of path length (from the base) of the amount of construction materials required to pave a unit of road length.

- The elevation difference in the area where the road is being constructed is insignificant.

This assumption makes it possible to neglect the height difference in the considered area and to carry out further constructions in a two-dimensional coordinate system.

- Construction conditions vary from point to point.

We assume that each point has its own construction conditions due to relief, landscape and other factors. Therefore, we can introduce a function β of construction cost per unit of path length.

Let us introduce a Cartesian coordinate system with origin at point O. Without loss of generality, we may assume that the endpoint A has coordinates $(l, 0)$. Let $y\colon \mathbb{R} \to \mathbb{R}$ is an arbitrary twice continuously differentiable function satisfying the boundary conditions

$$y(0) = 0, \quad y(l) = 0.$$

Any such curve will be called an admissible curve.

Under the formulated assumptions, the functional of road construction cost defined by the function $y(x)$ has the following form

$$J(y) = \int\limits_0^l \alpha\sqrt{1 + y'^2(x)} \int\limits_0^x \sqrt{1 + y'^2(\xi)}\, d\xi\, dx + \\ + \int\limits_0^l \beta(x, y)\sqrt{1 + y'^2(x)}\, dx, \tag{2.1}$$

where α–a constant defining the cost of delivery, and $\beta\colon \mathbb{R}^2 \to \mathbb{R}$ a given non-negative function with continuous partial derivatives up to and including second order defining the cost of paving work.

Let us further assume that there exists twice continuously differentiable admissible curve $y_*(x)$, which gives a minimum to the functional (2.1). It and determines the optimal trajectory of the road of the road.

Thus, we obtain a variational calculus problem with with fixed ends.

2.2 Derivation of necessary minimum conditions for the cost functional

The results stated in this paragraph are obtained by the author in [3]. Let us first formulate and prove an auxiliary result.

Lemma 2.2.1 *For an arbitrary function* $f(x) \in \mathbb{C}[0,l]$ *equality is true*

$$\int_0^l f(x) \int_0^x f(\xi)\, d\xi\, dx = \frac{1}{2}\left(\int_0^l f(x)\, dx\right)^2. \tag{2.2}$$

Proof The left part of the equality (2.2) is a double integral

$$\int_0^l \int_0^x f(\xi) f(x)\, d\xi\, dx = \iint_{G_1} f(\xi) f(x)\, d\xi\, dx,$$

where the region G_1 is depicted by the vertical shading in Fig. 2.1. By changing the order of integration, we obtain

$$\iint_{G_1} f(\xi) f(x)\, d\xi\, dx = \int_0^l \int_\xi^l f(\xi) f(x)\, dx\, d\xi.$$

Taking advantage of the fact that the variables x and ξ are symmetrically into the subintegral expression in the right-hand side, we swap them places

$$\int_0^l \int_\xi^l f(\xi) f(x)\, dx\, d\xi = \int_0^l \int_x^l f(x) f(\xi)\, d\xi\, dx = \iint_{G_2} f(\xi) f(x)\, d\xi\, dx.$$

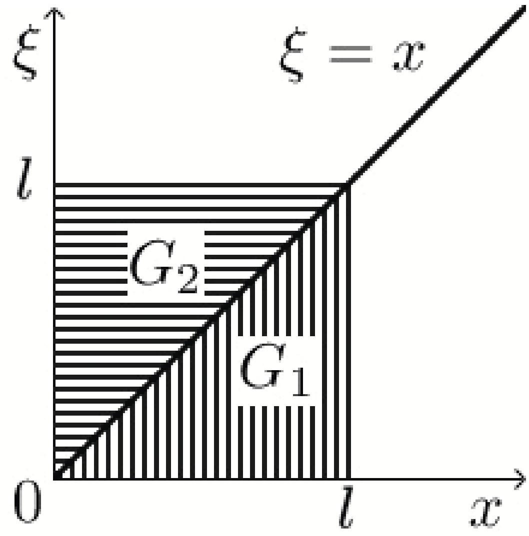

Figure 2.1: Illustration of the areas over which the integration is performed.

Thus, we obtain that the integral over the region G_2, depicted in Fig. 2.1 by horizontal shading, is equal to the integral over the region G_1

$$\iint_{G_1} f(\xi)f(x)\,d\xi\,dx = \iint_{G_2} f(\xi)f(x)\,d\xi\,dx.$$

Therefore, we can write

$$\begin{aligned}\iint_{G_1} f(\xi)f(x)\,d\xi\,dx &= \frac{1}{2}\iint_{G_1\bigcup G_2} f(\xi)f(x)\,d\xi\,dx = \\ &= \frac{1}{2}\int_0^l\int_0^l f(\xi)f(x)\,d\xi\,dx = \\ &= \frac{1}{2}\int_0^l f(x)\,dx\int_0^l f(\xi)\,d\xi = \frac{1}{2}\left(\int_0^l f(x)\,dx\right)^2,\end{aligned}$$

which completes the proof.

Using the lemma 2.2.1, we can rewrite the the functional (2.1) as

$$J(y) = \frac{\alpha}{2}\left(\int_0^l \sqrt{1+y'^2(x)}dx\right)^2 + \int_0^l \beta(x,y)\sqrt{1+y'^2(x)}\,dx. \tag{2.3}$$

The following theorem, which was derived in [3], gives a necessary condition for the minimum of this functional.

Theorem 2.2.1 *In order for the minimum of the cost functional J on the admissible curve $y_*(x) \in \mathbb{C}^2[0,l]$ reaches the minimum of the cost functional J it is necessary that*

$$\frac{y_*''(x)}{1+y_*'^2(x)}\left(\alpha\int\limits_0^l \sqrt{1+y_*'^2(x)}dx + \beta(x, y_*(x))\right) + y_*'(x)\frac{\partial\beta(x,y_*(x))}{\partial x} - \frac{\partial\beta(x,y_*(x))}{\partial y} = 0. \tag{2.4}$$

Proof For convenience, we keep the notation

$$F(y') = \sqrt{1+y'^2}.$$

Then

$$J(y) = \frac{\alpha}{2}\left(\int\limits_0^l F(y'(x))dx\right)^2 + \int\limits_0^l \beta(x,y)F(y'(x))\,dx.$$

Let $\delta(x)$ — continuously differentiable finite function on $[0,l]$, and ε — scalar quantity. Let us write out (see para. 1.3) the variation of the functional

$$\delta J(y_*) = \frac{d}{d\varepsilon}J(y_* + \varepsilon\delta)|_{\varepsilon=0}.$$

According to Theorem 1.3.1, the admissible curve y_*, yielding a minimum of of the functional J satisfies equality

$$\delta J(y_*) = 0.$$

Hence we obtain

$$\delta J(y_*) = \frac{d}{d\varepsilon}\left[\frac{\alpha}{2}\left(\int_0^l F(y'_* + \varepsilon\delta')dx\right)^2 + \right.$$

$$\left.+ \int_0^l \beta(x, y_* + \varepsilon\delta)F(y'_* + \varepsilon\delta')\, dx\right]\Bigg|_{\varepsilon=0} =$$

$$= \alpha\int_0^l \frac{\partial F}{\partial y'}\delta'\, dx \int_0^l F(y'_*)\, dx + \int_0^l \frac{\partial\beta}{\partial y}F(y'_*)\delta\, dx + \int_0^l \beta\frac{\partial F}{\partial y'}\delta'\, dx.$$

Using formula of integration by parts, let us consider separately the expressions in the of the summands in the right-hand side of this equality.

$$\int_0^l \frac{\partial F}{\partial y'}\delta'\, dx = \frac{\partial F}{\partial y'}\delta\Big|_0^l - \int_0^l \frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right)\delta\, dx = -\int_0^l \frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right)\delta\, dx,$$

$$\int_0^l \beta\frac{\partial F}{\partial y'}\delta'\, dx = \beta\frac{\partial F}{\partial y'}\delta\Big|_0^l - \int_0^l \frac{d}{dx}\left(\beta\frac{\partial F}{\partial y'}\right)\delta\, dx = -\int_0^l \frac{d}{dx}\left(\beta\frac{\partial F}{\partial y'}\right)\delta\, dx.$$

Then, taking into account the obtained results, we can write the necessary condition of minimization in the form

$$\delta J(y_*) = \int_0^l \Bigg(-\alpha\frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right)\int_0^l F(y'_*)\, dx + \frac{\partial\beta}{\partial y}F(y'_*) -$$

$$- \frac{d}{dx}\left(\beta\frac{\partial F}{\partial y'}\right)\Bigg)\delta\, dx = 0.$$

The function under the integral, which is the quotient of δ, belongs to $\mathbb{C}[0,l]$. Since $\mathbb{C}[0,l] \subset \widetilde{\mathscr{L}_2}[0,l]$, and the set of continuously of finite functions differentiable on $[0,l]$ according to Theorem 1.1.1 is everywhere dense in $\widetilde{\mathscr{L}_2}[0,l]$, from the last equality follows

$$-\alpha\frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right)\int_0^l F(y'_*)\, dx + \frac{\partial\beta}{\partial y}F(y'_*) - \frac{d}{dx}\left(\beta\frac{\partial F}{\partial y'}\right) = 0.$$

Substituting into this equality $F(y') = \sqrt{1+y'^2}$, and also

$$\frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right) = y''_*(1+{y'_*}^2)^{-\frac{3}{2}},$$

we get

$$\frac{y''_*(x)}{1+{y'_*}^2(x)}\left(\alpha\int\limits_0^l \sqrt{1+{y'_*}^2(x)}dx + \beta(x, y_*(x))\right)+$$
$$+ y'_*(x)\frac{\partial\beta(x, y_*(x))}{\partial x} - \frac{\partial\beta(x, y_*(x))}{\partial y} = 0,$$

which completes the proof.

■

Remark 2.2.1 *Note that we can obtain the same condition (2.4) using the classical results of the calculus of variations. To do this, we need to represent the functional (2.3) in a form suitable for the direct application of the Euler-Lagrange conditions.*

CHAPTER 3

Approximate methods for solving the problem

This chapter deals with approximate methods for solving the problem of obtaining the optimal road trajectory in terms of construction cost. Analytical expressions for the approximate solution will be obtained in the form of algebraic or trigonometric polynomials. This approach in some cases may be convenient for processing and further study of the obtained results.

The results presented in this chapter were obtained by the author in [2, 3, 38, 40].

3.1 Method based on polynomial interpolation

According to Theorem 2.2.1, to obtain an admissible curve satisfying the necessary minimum condition, we need to solve the integro-diphysical algebraic equation

$$\frac{y''}{1+y'^2}\left(\alpha\int\limits_0^l\sqrt{1+y'^2}dx+\beta\right)+y'\beta_x-\beta_y=0, \tag{3.1}$$

the numerical solution of which is an independent problem. It can be solved by considering the values of the function in the nodes as variables, using them to construct an interpolation polynomial. This way leads to a nonlinear system of algebraic equations with respect to the values of the function at the nodes. In [5], we develop an algorithm that realizes the above idea for solving the integro-differential equation under given initial conditions. To apply a similar approach to our problem, we can modify the above algorithm for problems with boundary conditions.

So, let us describe the adaptation of the method from the [5] that takes into account the boundary conditions given in our case.

On the segment $[0, l]$ we introduce a uniform grid containing $n+1$ node. Having the values of the second derivatives of the desired function at the nodes of the grid we can construct an interpolation polynomial for $y''(x)$ of degree n. Integrating the obtained polynomial and using the values of the of the function in the first and last nodes of the grid (ends of the segment $[0, l]$), we obtain interpolation polynomials of degree $n+1$ and $n+2$ for the functions $y'(x)$ and $y(x)$, respectively. Applying any quadrature formula to calculate the integral

$$\int_0^l \sqrt{1+y'(x)^2}dx,$$

reduce Eq. (3.1) to the problem of solving a system of of nonlinear equations with respect to the values of the second derivatives at the nodes of the grid.

Let us denote by $p = (p_1, \ldots, p_{n+1})^T$ the vector of coefficients of the of the interpolation polynomial for $y''(x)$, and

$$\overline{y}^{(2)} = \left(y_1^{(2)}, \ldots, y_{n+1}^{(2)}\right)^T$$

vector whose components are equal to the value of the function $y''(x)$ at the grid nodes, i.e.

$$y_i^{(2)} = y''(x_i), \quad x_i = (i-1)\frac{l}{n}, \quad i = 1, \ldots, n+1.$$

Consider the Vandermonde matrix

$$Z = \begin{pmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^n \\ 1 & x_2 & x_2^2 & \ldots & x_2^n \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & x_{n+1} & x_{n+1}^2 & \ldots & x_{n+1}^n \end{pmatrix}.$$

We have

$$\overline{y}^{(2)} = Zp,$$

whence, by virtue of nondegeneracy Z (see 1.2) we obtain

$$p = Z^{-1}\overline{y}^{(2)}.$$

Hence

$$y''(x) = XZ^{-1}\overline{y}^{(2)},$$

где $X = (1, x, \ldots, x^n)$. Integrating the last equality in the range from x_1 to x_i for each of $i = 1, \ldots, n+1$, we obtain

$$\overline{y}^{(1)} = Iy'(x_1) + S\overline{y}^{(2)},$$

here $I = (1, \ldots, 1)^T$ — n-dimensional unit vector, $\overline{y}^{(1)}$ — vector whose components are equal to the the value of the function $y'(x)$ at the nodes of the grid,

$$S = BZ^{-1},$$

and the matrix

$$B = \begin{pmatrix} 0 & 0 & 0 & \ldots & 0 \\ x_2 - x_1 & \dfrac{x_2^2 - x_1^2}{2} & \dfrac{x_2^3 - x_1^3}{3} & \ldots & \dfrac{x_2^{n+1} - x_1^{n+1}}{n+1} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_{n+1} - x_1 & \dfrac{x_{n+1}^2 - x_1^2}{2} & \dfrac{x_{n+1}^3 - x_1^3}{3} & \ldots & \dfrac{x_{n+1}^{n+1} - x_1^{n+1}}{n+1} \end{pmatrix}.$$

Denoting $\overline{y}$ — a vector whose components are equal to the the value of the function $y(x)$ at the nodes of the grid, we similarly arrive at

$$\overline{y} = Iy(x_1) + SIy'(x_1) + S^2\overline{y}^{(2)}.$$

Given that in our problem we know the value of the desired function $y(x)$ at the point x_{n+1} (at the right end of the segment) $y_n = y(x_{n+1})$, and not $y'(x_1)$, from the last equality express the value we need

$$y'(x_1) = \frac{y(x_{n+1}) - y(x_1) - \left[S^2\overline{y}^{(2)}\right]_{n+1}}{[SI]_{n+1}},$$

где $\left[S^2\overline{y}^{(2)}\right]_{n+1}$ и $[SI]_{n+1}$ denote $n+1$-th components of vectors $S^2\overline{y}^{(2)}$ and SI, respectively. Assuming n is even, calculate the integral using Simpson's formula (see, e.g., [6, 7, 16])

$$\int\limits_0^l \sqrt{1+y'^2}dx \approx \int\limits_0^l F(y'(x))dx =$$
$$= \frac{l}{3n}\sum_{i=1}^{n/2}\left(F(y_{2i-1}^{(1)}) + 4F(y_{2i}^{(1)}) + F(y_{2i+1}^{(1)})\right).$$

Thus, we finally arrive at a nonlinear system of $n+1$ equations with respect to n variables $y_1^{(2)}, \ldots, y_{n+1}^{(2)}$

$$\begin{cases} \dfrac{y_j^{(2)}}{1+\left(y_j^{(1)}\right)^2}\left[\alpha\Phi(\overline{y}^{(1)}) + \beta(x_j, y_j)\right] + y_j^{(1)}\beta_x(x_j, y_j) - \beta_y(x_j, y_j) = 0, \\ j = 1, \ldots, n+1, \end{cases}$$

where

$$y'(x_1) = \frac{y(x_{n+1}) - y(x_1) - \left[S^2\overline{y}^{(2)}\right]_{n+1}}{[SI]_{n+1}},$$
$$\overline{y}^{(1)} = Iy'(x_1) + S\overline{y}^{(2)},$$
$$\overline{y} = Iy(x_1) + SIy'(x_1) + S^2\overline{y}^{(2)},$$
$$\Phi(\overline{y}^{(1)}) = \frac{x_a}{3n}\sum_{i=1}^{n/2}\left(F(y_{2i-1}^{(1)}) + 4F(y_{2i}^{(1)}) + F(y_{2i+1}^{(1)})\right).$$

Primer 3.1.1 *Consider the problem in which $\alpha = 0.1$, $l = y_l = 1$, and the function*

$$\beta(x, y) = 1 + \sin 5x \cdot \sin y.$$

For ease of interpretation, we can assume that the function $\beta(x, y)$ defines the equation of the terrain surface, i.e., the cost of paving the the higher the point is located above the Oxy plane.

Let's use the approach described above with the application of MatLab[1] *the*

[1] see appendix A.1.

cost-optimal trajectory connecting points O and A. trajectory connecting points O and A. Assuming $n = 26$ *(where. it is obvious that the desired curve y is approximated by a polynomial of degree of degree 28), we obtain the numerical results given in the Table 3.1.*

x	0	0.0385	0.0769	0.1154	0.1538	0.1923	0.2308	0.2692	0.3077
y	0	0.0110	0.0218	0.0333	0.0453	0.0588	0.0728	0.0880	0.1058
x	0.3462	0.3846	0.4231	0.4615	0.5000	0.5385	0.5769	0.6154	0.6538
y	0.1228	0.1424	0.1635	0.1859	0.2103	0.2363	0.2645	0.2956	0.3282
x	0.6923	0.7308	0.7692	0.8077	0.8462	0.8846	0.9231	0.9615	1.0000
y	0.3668	0.4061	0.4532	0.5087	0.5739	0.6533	0.7500	0.8526	1.0000

Table 3.1: Calculation results.

In Fig. 3.1.1 u 3.2 the resulting trajectory is depicted on surface

$$z = \beta(x, y).$$

The curve is expected to "pass" elevations.

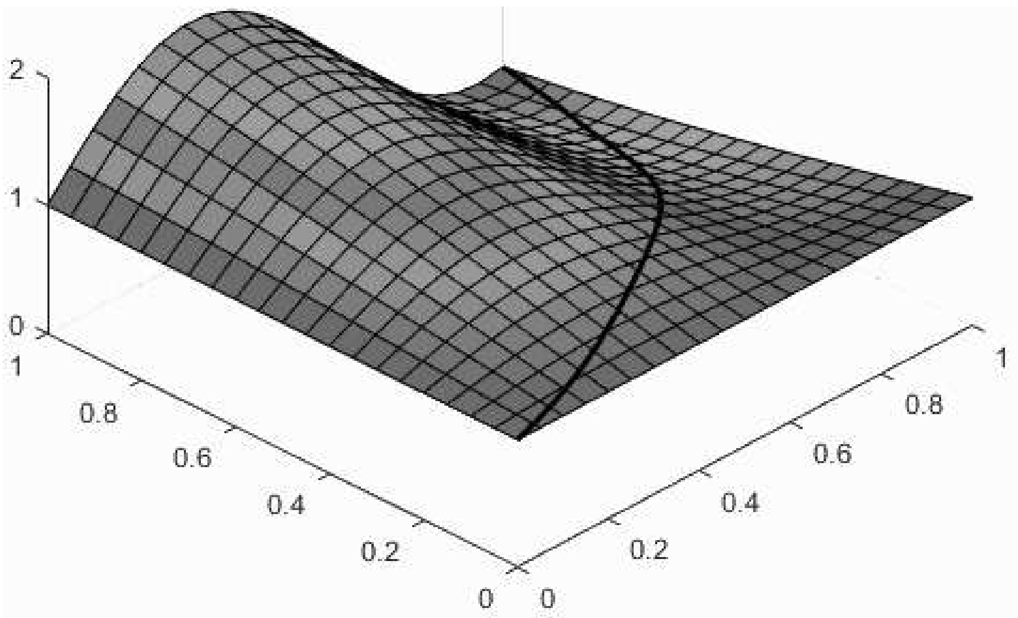

Figure 3.1: The resulting trajectory in Example 3.1.1.

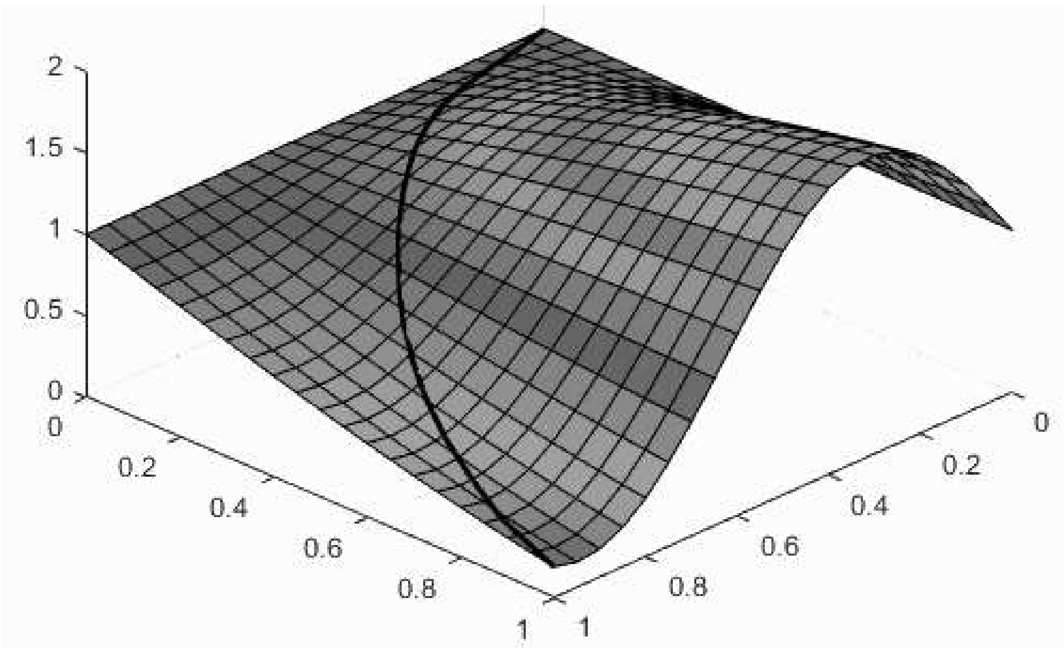

Figure 3.2: View from another point on the obtained trajectory in Example 3.1.1.

Note that the presented method leads to unsatisfactory results with a large number of nodes (numerically unstable with a high degree of the interpolation polynomial). Therefore, there is a need to develop other approaches that allow us to obtain a solution of the equation (4.1) with any desired accuracy.

3.2 Ritz method

To solve our integro-differential equation (3.1) in the previous paragraph, we used a numerical method based on approximating the the desired function and its derivatives by algebraic polynomials. Due to the computational error increases with the degree of the polynomial (see, for example, [6]). this approach becomes practically inapplicable for finding solutions with high degree of accuracy. In this paragraph, in order to to find the minimum of the cost functional, we consider the Ritz method Ritz [8, 13, 26, 27]. It is shown how it can be used to the problem under study can be solved. In addition, a more general formulation of the problem is considered and solved here, in which the cost of material delivery is not a constant value, but depends on the coordinates of a point.

First, we briefly summarize the basic idea of the Ritz method in the general form [19], and then specify it for the problem under study. then we will specify it for the problem under study.

Consider the problem of minimization of the functional

$$I(y) = \int_0^l F(x, y, y')\,dx,$$

where F is a continuous function of its arguments, and $y \in \mathbb{C}^1[0, l]$ under the condition

$$y(0) = y(l) = 0. \tag{3.2}$$

Let's

$$\phi(x, a_1, \dots, a_n), \quad n = 1, 2, \dots\,, \tag{3.3}$$

a sequence of n-parameterized families of functions, each of which is wider than the previous one by adding an additional parameter, and the conditions (3.2) are satisfied for all parameter values. For each family we can set the problem of minimizing the function of n arguments

$$I(a_1, \dots, a_n) = \int_0^l F(x, \phi(x, a_1, \dots, a_n), \phi'(x, a_1, \dots, a_n))\,dx, \tag{3.4}$$

which reduces to solving the system of equations

$$\frac{\partial I}{\partial a_i} = 0, \quad i = 1, \dots, n. \tag{3.5}$$

We denote by $\overline{a}_1, \dots, \overline{a}_n$ the optimal set of parameters, and

$$\overline{y}_n = \phi(x, \overline{a}_1, \dots, \overline{a}_n)$$

the corresponding family function. Due to the expansion of the class of admissible functions as n grows, the sequence $\{I(\overline{y}_n)\}_{n=1}^{\infty}$ is monotonically decreasing:

$$I(\overline{y}_1) \geq I(\overline{y}_2) \geq \dots\,.$$

If the set of functions forming the considered system of families (3.3) is dense in the set of functions from $\mathbb{C}^1[0,l]$ for which the conditions (3.2) are satisfied (see [2, 8]), then we obtain that

$$\lim_{n\to\infty} I(a_n) = I(y^*), \tag{3.6}$$

where y^* is– the function which gives the minimum to the functional I. Indeed, let any continuously differentiable function y defined on $[0,l]$ and going to zero at the ends of this interval can be approximated in the norm of the space $\mathbb{C}^1[0,l]$ by a function belonging to one of the families. Then for any $\delta > 0$ we can choose n and $y_n^* = \phi(x, a_1^*, \ldots, a_n^*)$ such that the inequality

$$\|y^* - y_n^*\|_{\mathbb{C}^1[0,l]} \le \delta.$$

Due to the continuity of F, for any $\varepsilon > 0$ we can find $\delta > 0$ such that for any

$$\|y^* - y_n^*\|_{\mathbb{C}^1[0,l]} \le \delta$$

the inequality

$$I(y_n^*) - I(y^*) \le \varepsilon.$$

is satisfied

Thus, for any $\varepsilon > 0$, the chain of inequalities

$$I(y^*) \le I(\overline{y}_n) \le I(y_n^*) \le I(y^*) + \varepsilon,$$

whence, due to the arbitrariness of the choice of ε, follows (3.6). Relying on the Weierstrass theorem on approximation by a trigonometric polynomial, we can show that the required density property as well as the boundary conditions (3.2) will be satisfied, in particular, by the system

$$\phi(x, a_1, \ldots, a_n) = \sum_{k=1}^{n} a_k \sin\frac{\pi k}{l}x, \quad n = 1, 2, \ldots.$$

Let n be a fixed number. Let us introduce a notation for the vector of coefficients $a = (a_1, \ldots, a_n)^T$, as well as the derivatives of the function ϕ:

$$\phi_x(x,a) = \frac{\partial \phi(x,a)}{\partial x} = \sum_{k=1}^{n} a_k \frac{\pi k}{l} \cos \frac{\pi k}{l} x,$$

$$\phi_{a_k}(x,a) = \frac{\partial \phi(x,a)}{\partial a_k} = \sin \frac{\pi k}{l} x, , \quad k = 1, \ldots, n,$$

$$\phi_{xa_k}(x,a) = \frac{\partial^2 \phi(x,a)}{\partial x \partial a_k} = \frac{\pi k}{l} \cos \frac{\pi k}{l} x, \quad k = 1, \ldots, n.$$

The functional (3.4) for the chosen system of functions will be written as follows:

$$I(a) = \frac{\alpha}{2} \left(\int_0^l \sqrt{1 + \phi_x^2(x,a)}\, dx \right)^2 + \int_0^l \beta(x, \phi(x,a)) \sqrt{1 + \phi_x^2(x,a)}\, dx,$$

and the extremum conditions (3.5) reduce to a system of nonlinear algebraic equations with respect to $a_1, \ldots, a_n$

$$\frac{\partial I(a)}{\partial a_k} = \alpha \left(\int_0^l \sqrt{1 + \phi_x^2(x,a)}\, dx \right) \int_0^l \frac{\phi_{xa_k}(x,a)\phi_x(x,a)}{\sqrt{1 + \phi_x^2(x,a)}}\, dx +$$
$$+ \int_0^l \frac{\partial \beta(x, \phi(x,a))}{\partial y} \phi_{a_k}(x,a) \sqrt{1 + \phi_x^2(x,a)}\, dx +$$
$$+ \int_0^l \beta(x, \phi(x,a)) \frac{\phi_{xa_k}(x,a)\phi_x(x,a)}{\sqrt{1 + \phi_x^2(x,a)}}\, dx = 0, \quad k = 1, \ldots, n.$$

Finding the solution $\overline{a} = (\overline{a}_1, \ldots, \overline{a}_n)^T$ of this of this system, we get an approximate solution $\phi_x(x, \overline{a})$ of the original problem.

The considered approach allows to find an approximate solution for the problem formulated in a more general form, namely, for the situation when the delivery price α is not constant, but varies from point to point due to different construction conditions. For example, a road may require a smaller amount of materials to build a road on hard ground than on swampy terrain. In this case, for the chosen system of functions, the integral cost functional will have the following form

$$I(a) = \int_0^l \alpha(x, \phi(x,a))\sqrt{1+\phi_x^2(x,a)} \int_0^x \sqrt{1+\phi_\xi^2(\xi,a)}\, d\xi\, dx + \\ + \int_0^l \beta(x, \phi(x,a))\sqrt{1+\phi_x^2(x,a)}\, dx,$$

which leads to the following system of nonlinear algebraic equations for search $a_1, \dots, a_n$

$$\frac{\partial I(a)}{\partial a_k} = \int_0^l \frac{\partial \alpha(x,\phi(x,a))}{\partial y}\phi_{a_k}(x,a)\sqrt{1+\phi_x^2(x,a)} \int_0^x \sqrt{1+\phi_\xi^2(\xi,a)}\, d\xi\, dx + \\ + \int_0^l \alpha(x,\phi(x,a))\frac{\phi_{xa_k}(x,a)\phi_x(x,a)}{\sqrt{1+\phi_x^2(x,a)}} \int_0^x \sqrt{1+\phi_\xi^2(\xi,a)}\, d\xi\, dx + \\ + \int_0^l \alpha(x,\phi(x,a))\sqrt{1+\phi_x^2(x,a)} \int_0^x \frac{\phi_{\xi a_k}(\xi,a)\phi_\xi(\xi,a)}{\sqrt{1+\phi_\xi^2(\xi,a)}}\, d\xi\, dx + \\ + \int_0^l \frac{\partial \beta(x,\phi(x,a))}{\partial y}\phi_{a_k}(x,a)\sqrt{1+\phi_x^2(x,a)}\, dx + \\ + \int_0^l \beta(x,\phi(x,a))\frac{\phi_{xa_k}(x,a)\phi_x(x,a)}{\sqrt{1+\phi_x^2(x,a)}}\, dx = 0, \quad k = 1, \dots, n.$$

Note finally that at nonzero boundary conditions we can consider the system of functions

$$\phi(x, a_1, \dots, a_n) = y(0) + \frac{y(l)-y(0)}{l}x + \sum_{k=1}^n a_k \sin\frac{\pi k}{l}x, \quad n = 1, 2, \dots. \tag{3.7}$$

In this case in the above reasoning only the expression for $\phi_x(x,a)$:

$$\phi_x(x,a) = \frac{y(l)-y(0)}{l} + \sum_{k=1}^n a_k \frac{\pi k}{l}\cos\frac{\pi k}{l}x,$$

and the rest will remain unchanged.

Note that the Ritz method is a powerful tool for solving applied problems, which is used by many researchers [37, 41, 42, 59, 61, 67, 74, 81, 82, 86, 96, 99].

Examples of programs implementing the Ritz method for constant and variable α are given in the appendices A.2 и A.3 соответственно.

Primer 3.2.1 *Consider the problem in which* $\alpha = 0.1$, $l = 1$,

$$\beta(x, y) = 1 + \sin 5x \cdot \sin y,$$

and the boundary conditions are

$$y(0) = 0, \;\; y(l) = 1.$$

For ease of understanding the results, we will assume that the function $\beta(x, y)$ *defines the terrain surface, i.e., the cost of the is directly proportional to the height of a point above the Oxy plane. Using the MatLab math package, the following results were obtained for* $n = 5$*:*

$$\overline{a} = (-0.31397, 0.07367, -0.03138, 0.01212, -0.00396)^T.$$

Ha the approximate solution found, which is given as (3.7), we have

$$I(a) = 1.279.$$

Applying the algorithm described in 3.1 and based on approximation of the desired curve by an algebraic polynomial of degree m*, for* $m = 26$ *leads to a solution on which the functional is* 1.325*. In Fig. 3.3 and 3.4 shows the surface* $z = \beta(x, y)$*, where the trajectory constructed using the Ritz method (*$n = 5$*) is highlighted in white, and the solution obtained by the algorithm from 3.1 is shown in black. (*$m = 26$*).*

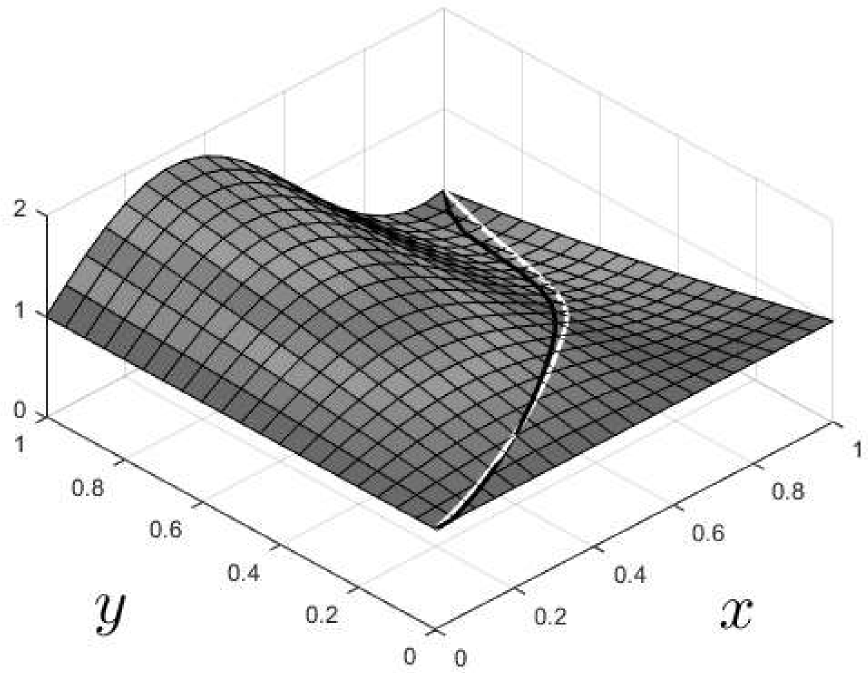

Figure 3.3: View from the starting point on the obtained in the example 3.2.1 траектории.

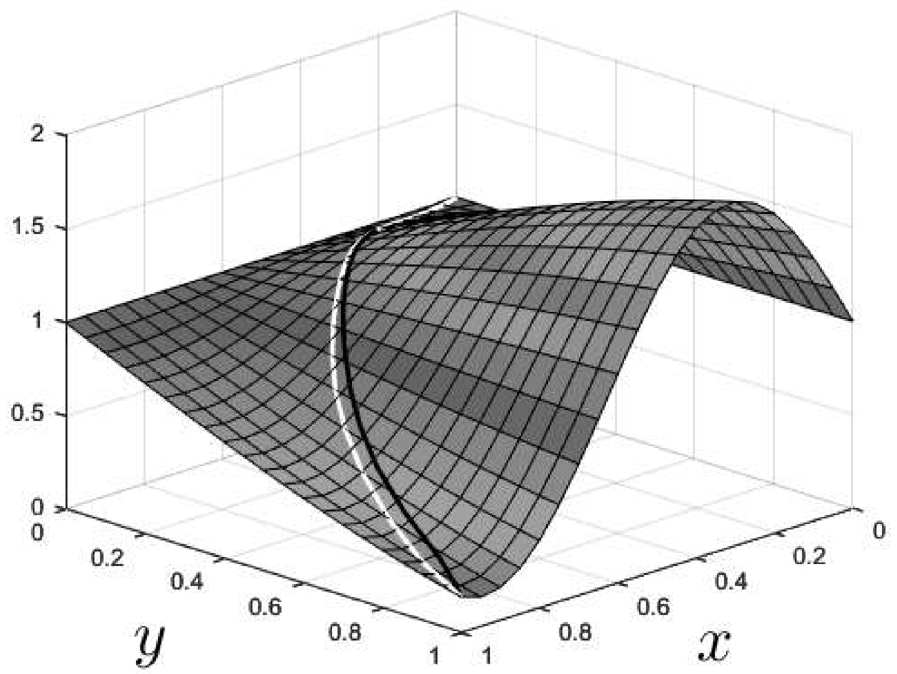

Figure 3.4: View from the endpoint to the ones obtained in the example 3.2.1 траектории.

Primer 3.2.2 *Let's* $l = 1$,

$$\alpha(x, y) = \cos^2 5x \cdot \cos^2 y,$$

$$\beta(x, y) = 1 + \sin 5x \cdot \sin y,$$

boundary conditions have the form

$$y(0) = 0, \;\; y(l) = 1.$$

We are looking for a solution in the form of (3.7) using the Ritz method. Using the mathematical package MatLab, at $n = 5$ *we obtain the following results:*

$$\overline{a} = (-0.32611, 0.11689, -0.0671, 0.03071, -0.01276)^T.$$

On the approximate solution found, we have

$$I(a) = 1.44536.$$

The constructed solution is shown in Fig. 3.5.

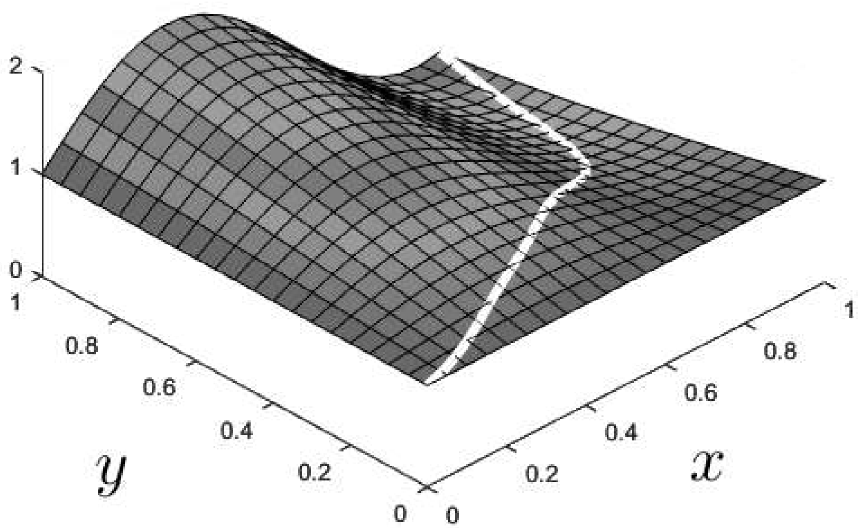

Fig. 3.5: The solution in the example 3.2.2 shown on the surface $z = \beta(x, y)$.

The considered approach to solving the problem allows to obtain the result in the form of a trigonometric polynomial, which with a relatively small number of summands (in the example 3.2.1 there were 5) gives better results, than the solution based on approximation of the desired function by an algebraic polynomial of high degree, proposed by the method of point 3.1 (in the example 3.1.1 this degree was 26). The convergence of the proposed algorithm follows from the convergence of the Ritz method.

3.3 Galerkin method

Let us first describe the basic idea of the Galerkin method (see, e.g., [19]). Consider the operator

$$L(y) = \frac{y''}{1+y'^2}\left(\alpha \int_0^l \sqrt{1+y'^2}dx + \beta(x,y)\right) + y'\frac{\partial\beta(x,y)}{\partial x} - \frac{\partial\beta(x,y)}{\partial y},$$

describing the necessary condition of minimum

$$L(y) = 0. \tag{3.8}$$

We will look for the solution in the form

$$y = \frac{y_l}{l}x + \sum_{k=1}^{\infty} a_k\phi_k(x), \tag{3.9}$$

where $\{\phi_k(x) \mid k = 1, 2, \dots\}$ – is a system of basis functions in the space $C_0^2[0,l]$ of twice continuous, finite functions on the segment $[0,l]$. Obviously, the function (3.9) satisfies the boundary conditions

$$y(0) = y(l) = 0.$$

You can, for example, use the functions

$$\phi_k(x) = \sin\frac{k\pi x}{l}, \quad k = 1, 2, \dots. \tag{3.10}$$

or

$$\phi_k(x) = (l-x)x^k, \quad k = 1, 2, \dots, \tag{3.11}$$

which follows directly from the approximation theorems of Weierstrass.

Let's look at the problem (3.8) in space $\mathscr{L}_2[0,l]$. Obviously, the function y_* satisfies the equation (3.8) if and only if $L(y_*)$ is orthogonal to all functions of the system $\{\phi_k(x) \mid k=1,2,\dots\}$. However, if we work exclusively with the sum of the first n terms of a series (3.9), we can satisfy only n orthogonality conditions, i.e.

$$\int_0^l L(y_*(x))\phi_k(x)dx$$
$$= \int_0^l L\left(\frac{y_l}{l}x + \sum_{k=1}^{n} a_k\phi_k(x)\right)\phi_k(x)dx = 0, \quad k=1,\dots,n.$$

These equations are used to find the unknown coefficients of the basis function decomposition of the solution. More detailed information about the method can be found, for example, in [19].

The Galerkin method and its modifications are used by many researchers to solve the applied [43–45, 48, 53, 57, 66, 68, 70, 72, 75, 80, 83, 85, 87].

Let us consider numerical examples in which we will assume that the terrain surface on which the road is to be built is given by the function β. This assumption allows us to obtain a clear graphical illustration and interpretation of the results. Examples of programs implementing the Galerkin method for the systems of basis functions (3.10) and (3.11) are given in the appendices A.4 and A.5, respectively.

Primer 3.3.1 *Пусть* $\alpha = 0.1$, $l = 1$, $y_l = 1$ *и* $\beta\colon \mathbb{R}^2 \to \mathbb{R}$

$$\beta(x,y) = 1 + \sin 5x \cdot \sin y.$$

First, we use the system of functions (3.10) and find the solution in the form of

$$y(x) = \frac{y_l}{l}x + \sum_{k=1}^{5} a_k \sin\frac{\pi k}{l}x. \tag{3.12}$$

Using the Galerkin method, we obtain (black curve in Fig. 3.6)

$a_1 = -0.31489,\ a_2 = 0.07442,\ a_3 = -0.03199,\ a_4 = 0.01256,\ a_5 = -0.00424.$

Recall that for the same problem, the Ritz method, discussed in the paragraph 3.2, led us (see Example 3.2.1) to a solution whose cost is also equal to $1,279$. *As we can see in Fig. 3.6, these two solutions are almost identical.*

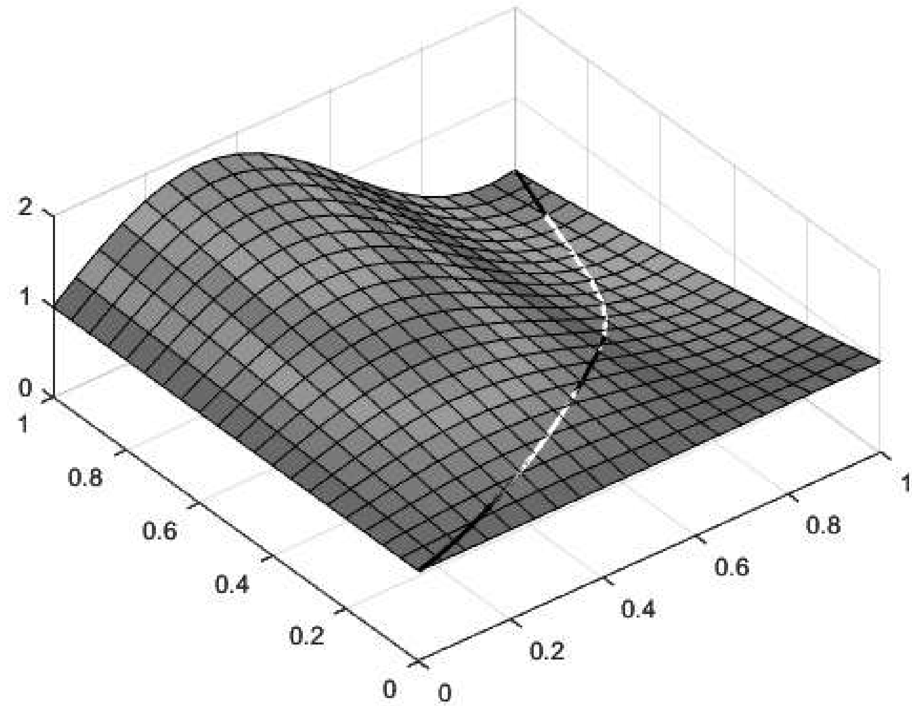

Figure 3.6: Graphs of solutions of the form (3.12) obtained by Galerkin (black curve) and Ritz (white curve) methods in Example 3.3.1.

Now let us use the system (3.11) and find the solution in the form of

$$y(x) = \frac{y_l}{l}x + \sum_{k=1}^{5} a_k x^k (1-x). \tag{3.13}$$

Applying Galerkin's method, we obtain

$a_1 = -0.31489,\ a_2 = 0.07442,\ a_3 = -0.03199,\ a_4 = 0.01256,\ a_5 = -0.00424.$

Fig. 3.7 shows solutions of the form (3.13) obtained by the Galerkin and

Ritz method. The cost of the solution (i.e., the value of the functional J) on the solution obtained

$$a_1 = -0,66947,\ a_2 = -1,03044,\ a_3 = 0,4348,\ a_4 = 0,51721,\ a_5 = -2,37822,$$

amounts to $1,2798$.

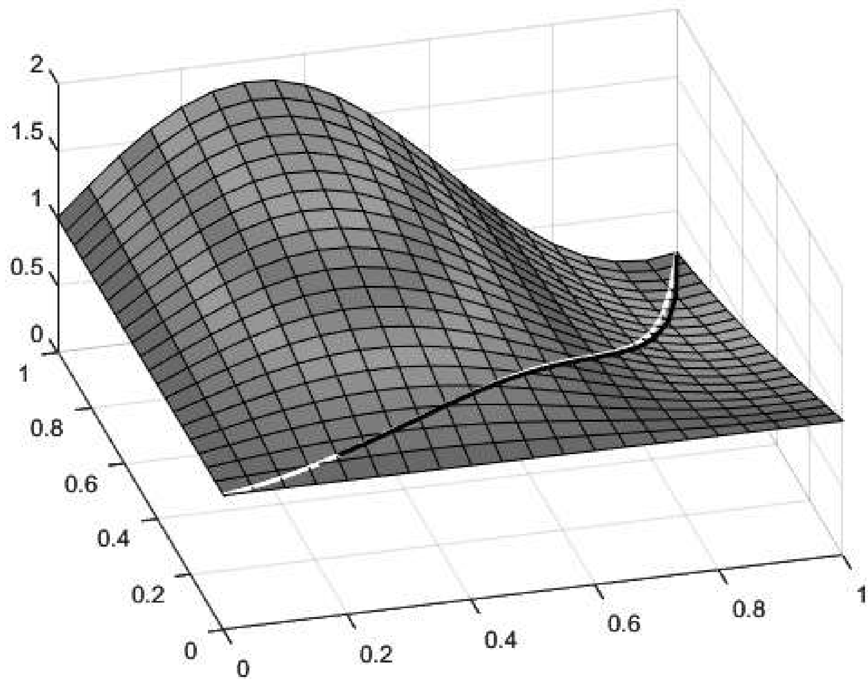

Figure 3.7: Graphs of solutions of the form (3.13), obtained by Galerkin (black curve) and Ritz (white curve) methods in Eq. 3.3.1.

Primer 3.3.2 *Пусть* $\alpha = 0.5$, $l = 1$, $y_l = 1$ *и* $\beta \colon \mathbb{R}^2 \to \mathbb{R}$

$$\beta(x, y) = 5 + 2\cos 2x \cdot \sin y.$$

In Fig. 3.8 an approximate solution of the form (3.13), obtained by Galerkin's method, where

$$a_1 = -0.13852,\ a_2 = 0.02332,\ a_3 = -0.01159,\ a_4 = 0.00629,\ a_5 = -0.03046.$$

The cost of the solution (value of the functional J) is equal to 7.964.

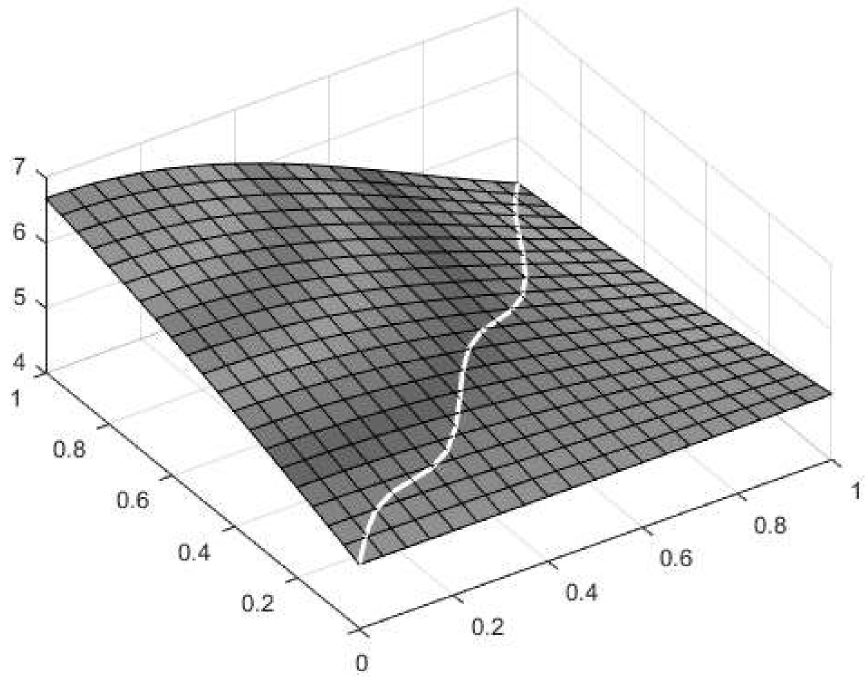

Figure 3.8: Graph of a solution of the form (3.13), obtained by the Galerkin method in Example 3.3.2.

Primer 3.3.3 *Пусть* $\alpha = 0.1$, $l = 1$, $y_l = 1$ *и* $\beta\colon \mathbb{R}^2 \to \mathbb{R}$

$$\beta(x, y) = \sqrt{2.75 - (x - 0.5)^2 - 10(y - 0.5)^2}.$$

Galerkin's method leads us to a solution of the form (3.13), where

$$a_1 = -0,78479,\ a_2 = 0,02804,\ a_3 = 4,52723,\ a_4 = -3,25439,\ a_5 = 0,2608.$$

This solution is shown in Fig. 3.9 and has a value (value of the functional J*)* 1,941.

Primer 3.3.4 *Let's* $\alpha = 0.5$, $l = 1$, $y_l = 1$ *и* $\beta\colon \mathbb{R}^2 \to \mathbb{R}$

$$\beta(x, y) = \sqrt{0.5 + 10(x - 0.5)^2 - (y - 0.5)^2}.$$

Galerkin's method gives us a solution of the form (3.13), where

$$a_1 = -0.43003,\ a_2 = 0.07392,\ a_3 = -7.33056,\ a_4 = 25.32797,\ a_5 = -20.57708.$$

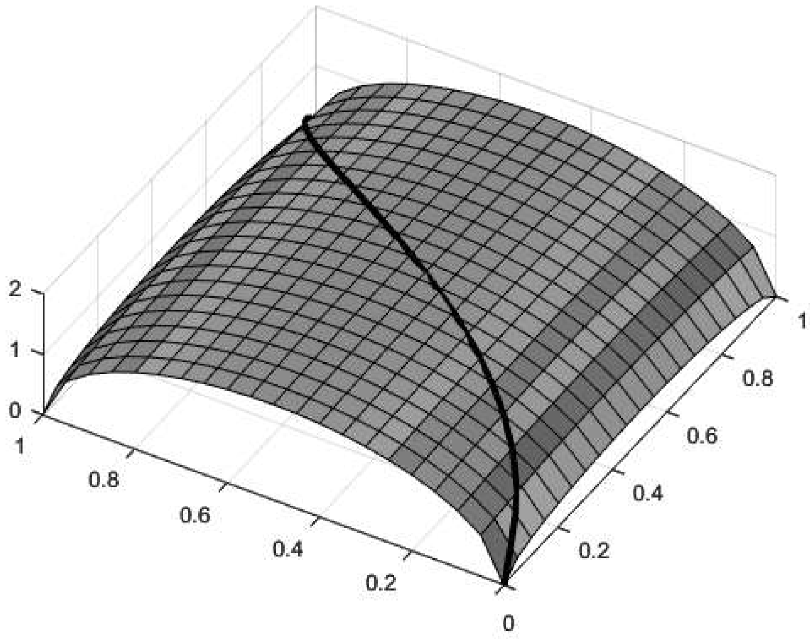

Figure 3.9: Graph of the solution of the form (3.13) obtained by Galerkin's method in Example 3.3.3.

This solution is shown in Fig. 3.10 and has a cost (value of the functional J) 2, 1.

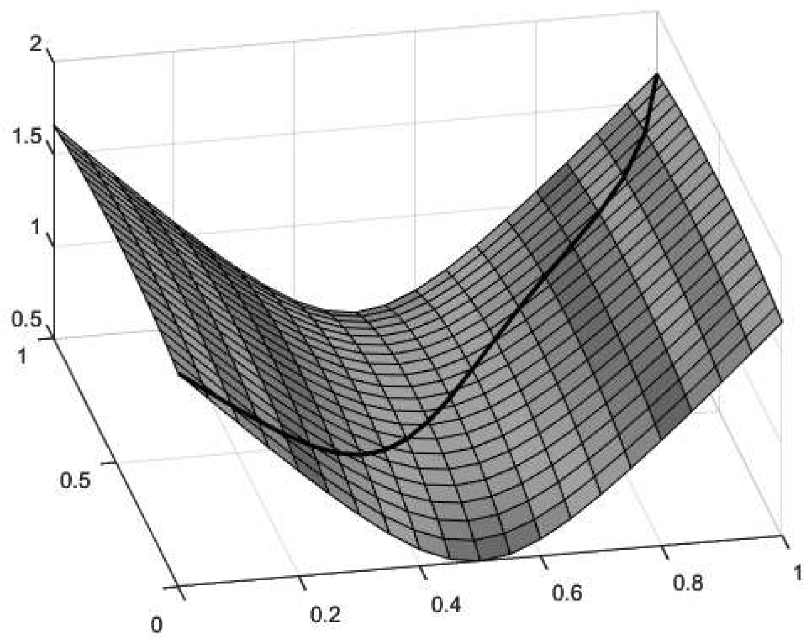

Figure 3.10: Graph of the solution of the form (3.13) obtained by Galerkin's method in Example 3.3.4.

CHAPTER 4

Numerical solution of the problem. Existence and uniqueness of the solution

In this chapter, we describe a numerical method for solving the integro-differential 3.1 based on the well-known and thoroughly studied in the literature shooting method. Under additional assumptions, the existence of the solution is proved using Schauder's fixed point principle. The question of the singularity of the solution is investigated.

The results presented in this chapter were obtained by the author in the paper [1].

4.1 Shooting method for finding the optimal trajectory

Obtaining the optimal trajectory using the approximate methods described in the previous paragraph can be time-consuming, so it is of particular interest to develop numerical methods for solving the integro-differential equation (3.1). The main purpose of this paragraph is to construct an iterative algorithm for finding the solution of the equation (3.1), combining the ideas of linearization and the shooting method.

Thus, it follows from Theorem 2.2.1 that the desired optimal trajectory $y_*(x)$ must satisfy the integro-differential equation

$$y''\left(\alpha\int_0^l \sqrt{1+y'^2}dx+\beta(x,y)\right)+(1+y'^2)\left(y'\beta_x(x,y)-\beta_y(x,y)\right)=0,$$

which can be rewritten as

$$y'' = -\frac{(1+y'^2)\,(y'\beta_x(x,y) - \beta_y(x,y))}{\alpha \int_0^l \sqrt{1+y'^2}dx + \beta(x,y)}, \tag{4.1}$$

where $y(0) = 0$ и $y(l) = 0$.

One of the main problems in the numerical solution of this equation is the calculation of the integral in the denominator of the right-hand side (4.1). On the one hand, when applying the ideas of classical finite-difference schemes, the equation must be used to calculate sequentially the values of the function in the nodes of the grid used, and on the other hand, the values of the function in all nodes must be known in advance to calculate the integral.

In this paragraph, an iterative algorithm based on linearization and the shooting method is proposed. The latter is well studied and described in detail in the literature and is often used in solving applied problems [6, 10, 49, 51, 54, 55, 58, 60, 65, 73, 73, 92]. Linearizing the original problem in the neighborhood of some approximation $y_n(x)$ of the solution serves two purposes: first, we can use $y_n(x)$ to compute the integral in the denominator of the right-hand side (4.1), and, secondly, it is possible to apply the shooting method to the obtained linear boundary value problem to obtain the next approximation. $y_{n+1}(x)$.

Let us denote the right-hand side (4.1) by $f(x, y, y')$ and suppose that an approximate solution is given $y_n(x)$. Then in the neighborhood of $y_n(x)$ the decomposition takes place

$$f(x,y,y') \approx f\Big(x, y_n, y_n'\Big) + \frac{\partial f}{\partial y}\Big(x, y_n, y_n'\Big)(y - y_n) + \frac{\partial f}{\partial y'}\Big(x, y_n, y_n'\Big)(y' - y_n')$$

For convenience, we introduce the notations

$$f_{n0}(x) = f(x, y_n, y_n'),\;\; f_{n1}(x) = \frac{\partial f}{\partial y}(x, y_n, y_n'),\;\; f_{n2}(x) = \frac{\partial f}{\partial y'}(x, y_n, y_n'),$$

where

$$f_{n0}(x) = -\frac{(1+{y_n'}^2)\,(y_n'\beta_x(x,y_n) - \beta_y(x,y_n))}{\alpha \int_0^l \sqrt{1+{y_n'}^2}dx + \beta(x,y_n)},$$

$$
\begin{aligned}
f_{n1}(x) = &- \frac{1}{\left(\alpha \int_0^l \sqrt{1 + {y'_n}^2} dx + \beta(x, y_n)\right)^2} \\
&\Bigg[\left((1 + {y'_n}^2) \left(y'_n \beta_{xy}(x, y_n) - \beta_{yy}(x, y_n) \right) \right) \\
&\times \left(\alpha \int_0^l \sqrt{1 + {y'_n}^2} dx + \beta(x, y_n) \right) - \beta_y(x, y_n)(1 + {y'_n}^2) \\
&\times \left(y'_n \beta_x(x, y_n) - \beta_y(x, y_n) \right) \Bigg],
\end{aligned} \tag{4.2}
$$

$$
\begin{aligned}
f_{n2}(x) = &- \frac{1}{\left(\alpha \int_0^l \sqrt{1 + {y'_n}^2} dx + \beta(x, y_n)\right)^2} \\
&\Bigg[\Big(2y'_n \left(y'_n \beta_x(x, y_n) - \beta_y(x, y_n) \right) \\
&+ (1 + {y'_n}^2) \beta_x(x, y_n) \Big) \left(\alpha \int_0^l \sqrt{1 + {y'_n}^2} dx + \beta(x, y_n) \right) \\
&- \alpha \left((1 + {y'_n}^2) \left(y'_n \beta_x(x, y_n) - \beta_y(x, y_n) \right) \right) \int_0^l \frac{y'_n}{\sqrt{1 + {y'_n}^2}} dx \Bigg].
\end{aligned} \tag{4.3}
$$

Then the new approximation $y_{n+1}(x)$ can be found as the solution of the following boundary value problem

$$
\begin{aligned}
y''_{n+1}(x) = &f_{n0}(x) + f_{n1}(x)(y_{n+1}(x) - y_n(x)) + \\
&+ f_{n2}(x)(y'_{n+1}(x) - y'_n(x)), \\
&y_{n+1}(0) = 0, \; y_{n+1}(l) = 0.
\end{aligned} \tag{4.4}
$$

The grid analog of the problem (4.1) looks like

$$
\begin{aligned}
&\frac{y^{k+1} - 2y^k + y^{k-1}}{h^2} = f(x^k, y^k, y'^k), \quad k = 1, \ldots, N - 1, \\
&y^0 = 0, \; y^N = 0,
\end{aligned}
$$

where

$$
h = l/N, \; x^k = kh,
$$

and y^k is an approximation of $y(x^k)$. Let

$$y_n^k,\ k = 0, \dots, N$$

be the set of values forming the nth approximation to the solution. In the neighborhood of this approximation the decomposition is valid

$$\begin{aligned} f(x^k, y^k, y'^k) \approx & f(x^k, y_n^k, {y'}_n^k) + \\ & + \frac{\partial f}{\partial y}(x^k, y_n^k, {y'}_n^k)(y^k - y_n^k) + \frac{\partial f}{\partial y'}(x^k, y_n^k, {y'}_n^k)(y'^k - {y'}_n^k) = \\ = & f_{n0}(x^k) + f_{n1}(x^k)(y^k - y_n^k) + f_{n2}(x^k)(y'^k - {y'}_n^k) = \\ = & f_{n0}^k + f_{n1}^k(y^k - y_n^k) + f_{n2}^k(y'^k - {y'}_n^k). \end{aligned}$$

Hence, the next approximation can be found from the grid analog of the boundary value problem (4.4)

$$\begin{aligned} & \frac{y_{n+1}^{k+1} - 2y_{n+1}^k + y_{n+1}^{k-1}}{h^2} = f_{n0}^k + f_{n1}^k(y_{n+1}^k - y_n^k) + f_{n2}^k({y'}_{n+1}^k - {y'}_n^k), \\ & k = 1, \dots, N-1, \\ & y_{n+1}^0 = 0,\ y_{n+1}^N = 0. \end{aligned} \tag{4.5}$$

Substituting the symmetric difference derivative in place of ${y'}_{n+1}^k$ and ${y'}_n^k$ in (4.5) we arrive at the discrete problem

$$\begin{aligned} & y_{n+1}^{k+1} = \frac{f_{n0}^k + f_{n1}^k(y_{n+1}^k - y_n^k) - \dfrac{f_{n2}^k}{2h}\left(y_{n+1}^{k-1} + y_n^{k+1} - y_n^{k-1}\right) + 2y_{n+1}^k - y_{n+1}^{k-1}}{1 - \dfrac{hf_{n2}^k}{2}}, \\ & k = 1, \dots, N-1, \\ & y_{n+1}^0 = 0,\ y_{n+1}^N = 0. \end{aligned} \tag{4.6}$$

Note that the integrals in

$$f_{ni}^k,\ i = 1, 2, 3$$

can be calculated numerically, e.g., by the rule of trapezoids.

The boundary value problem (4.6) can be solved by the shooting method. Let y_n^k, $k = 1, \dots, N$ be given. We obtain $\widetilde{y}_{n+1}^k$, $k = 1, \dots, N$ as the solution of the following problem

$$\widetilde{y}_{n+1}^{k+1} = \frac{f_{n0}^k + f_{n1}^k(\widetilde{y}_{n+1}^k - y_n^k) - \dfrac{f_{n2}^k}{2h}\left(\widetilde{y}_{n+1}^{k-1} + y_n^{k+1} - y_n^{k-1}\right) + 2\widetilde{y}_{n+1}^k - \widetilde{y}_{n+1}^{k-1}}{1 - \dfrac{hf_{n2}^k}{2}},$$

$$k = 1, \ldots, N-1, \tag{4.7}$$

$$\widetilde{y}_{n+1}^0 = 0,\ \widetilde{y}_{n+1}^1 = y_n^1,$$

and

$$\widehat{y}_{n+1}^k,\ k = 1, \ldots, N$$

as the solution of the corresponding homogeneous problem

$$\widehat{y}_{n+1}^{k+1} = \frac{f_{n1}^k \widehat{y}_{n+1}^k - \dfrac{f_{n2}^k}{2h}\widehat{y}_{n+1}^{k-1} + 2\widehat{y}_{n+1}^k - \widehat{y}_{n+1}^{k-1}}{1 - \dfrac{hf_{n2}^k}{2}},$$

$$k = 1, \ldots, N-1, \tag{4.8}$$

$$\widehat{y}_{n+1}^0 = 0,\ \widehat{y}_{n+1}^1 = y_n^1.$$

The general solution (4.6) has the form

$$y_{n+1}^k = \widetilde{y}_{n+1}^k + c\widehat{y}_{n+1}^k,$$

where c is a constant. Hence, we can choose the constant such that the boundary condition

$$y_{n+1}^N = 0,$$

that is

$$c = -\frac{\widetilde{y}_{n+1}^N}{\widehat{y}_{n+1}^N}.$$

Finally, as an initial approximation we can choose, for example, a straight line satisfying the boundary conditions

$$y_0^k = 0, \quad k = 1, \ldots, N \tag{4.9}$$

and terminate the calculations as soon as the inequality for some small positive

ε is satisfied

$$|J(y_{n+1}) - J(y_n)| \leq \varepsilon \tag{4.10}$$

4.2 Existence and uniqueness of the solution

We will discuss the questions of existence and uniqueness of the solution of the problem (4.1). We will need auxiliary information (see [18, 21, 25, 31]).

Definition 4.2.1 *Functions from the set* $U \subset \mathbb{C}^1[0,l]$ *are called uniformly bounded if there exists* $p > 0$ *such that*

$$|y(x)| + |y'(x)| \leq p$$

for any $y(x) \in U$ *and any* $x \in [0,l]$.

Definition 4.2.2 *Functions from the set* $U \subset \mathbb{C}^1[0,l]$ *are called equivariantly continuous if for any* $\varepsilon > 0$ *there exists* $\delta > 0$ *such that for any* x_1, x_2 *of* $[0,l]$ *for which* $|x_1 - x_2| < \delta$ *and for all* $y(x) \in U$ *the inequality holds*

$$|y(x_1) - y(x_2)| + |y'(x_1) - y'(x_2)| < \varepsilon.$$

Theorem 4.2.1 (Arzel's theorem) *Let* $U \subset \mathbb{C}^1[0,l]$ *be a set of continuously differentiable functions. For* U *to be compact, it is necessary and sufficient that the functions from the set* U *are uniformly bounded and equally continuous.*

Note that this theorem is a consequence of the classical theorem of Arzel for $\mathbb{C}[0,l]$ and the theorem on honorable differentiation of functional sequences.

Definition 4.2.3 *A continuous operator* A *defined on a set* U *of a linear normalized space* E *with a region of values located in* U *is called quite continuous if every bounded subset of* U *it maps into a compact subset.*

Theorem 4.2.2 (Schauder's fixed point principle) *If a semicontinuous operator* A *maps a bounded closed convex set* S *of Banach space onto its part, then there exists a fixed point of this mapping, i.e., a point* $x \in S$ *such that* $Ax = x$.

Let us now proceed to the proof of existence of the solution of the problem (4.1).

Theorem 4.2.3 *Let's* $M \geq \frac{1}{2l}$, *and exists*

$$c \leq \frac{\alpha}{32l^2M^2},$$

such that the function $\beta(x, y)$ *in the domain of its definition satisfies the condition*

$$\max\{|\beta_x(x, y)|, |\beta_y(x, y)|\} \leq c.$$

Then in the area

$$G = \{y \mid \max_{x\in[0,l]} |y(x)| = \|y\|_{\mathbb{C}[0,l]} \leq 2l^2M, \ \|y'\|_{\mathbb{C}[0,l]} \leq 2lM\}$$

there is a single solution to the problem (4.1).

Proof Note that the original problem (4.1) is equivalent to the operator equation $Ay(x) = y(x)$, where the operator A mapping $\mathbb{C}^1[0, l]$ to $\mathbb{C}^1[0, l]$ is defined as follows:

$$Ay(x) = \int_0^x d\xi \int_0^\xi f(\eta, y(\eta), y'(\eta))d\eta - \frac{x}{l}\int_0^l d\xi \int_0^\xi f(\eta, y(\eta), y'(\eta))d\eta.$$

If the sequence $y_n(x)$ belonging to the set G converges in the sense of the norm of the space $\mathbb{C}^1[0, l]$ to the function $y(x)$ belonging, obviously, to the same set, then by the continuity of the function $f(x, y, y')$ we have $f(x, y_n, y_n') \to f(x, y, y')$ uniformly on $[0, l]$. Hence, due to the possibility of passing to the limit under the sign of the integral at uniform convergence, we obtain $Ay_n \to Ay$, i.e., the operator A is continuous on G.

We prove that if y belongs to G, then Ay is also contained in G. Indeed, given that $\beta \geq 0$, for any y of G and all x of $[0, l]$ we have

$$|f(x, y, y')| = \left|\frac{(1 + y'^2)(y'\beta_x - \beta_y)}{\alpha\int_0^l \sqrt{1 + y'^2}dx + \beta}\right| \leq \frac{(1 + (2lM)^2)(1 + 2lM)c}{\alpha l}.$$

Given that

$$1 \leq 2lM,$$

we obtain

$$|f(x,y,y')| \leq M,$$

whence for any $x \in [0,l]$ the following inequalities are satisfied

$$\begin{aligned} |Ay(x)| \leq \int_0^x d\xi \int_0^\xi |f(\eta, y(\eta), y'(\eta))|d\eta + \\ + \frac{x}{l}\int_0^l d\xi \int_0^\xi |f(\eta, y(\eta), y'(\eta))|d\eta \leq 2l^2M, \end{aligned} \tag{4.11}$$

$$\begin{aligned} \left|\frac{d}{dx}Ay(x)\right| \leq \int_0^x |f(\eta, y(\eta), y'(\eta))|d\eta + \\ + \frac{1}{l}\int_0^l d\xi \int_0^\xi |f(\eta, y(\eta), y'(\eta))|d\eta \leq 2lM. \end{aligned} \tag{4.12}$$

Thus $Ay(x) \in G$, and hence the operator A transforms this set into itself.

Let x_1, x_2 be arbitrary points of the segment $[0,l]$, then the following inequalities are true

$$|Ay(x_1) - Ay(x_2)| \leq 2lM|x_1 - x_2|, \tag{4.13}$$

$$\left|\frac{d}{dx}Ay(x_1) - \frac{d}{dx}Ay(x_2)\right| \leq M|x_1 - x_2|. \tag{4.14}$$

Inequalities (4.11), (4.12), (4.13), (4.14) by virtue of the above theorem of Arzell show that the operator A transforms G into a compact set.

It is also obvious that the set G is bounded, closed and convex.

Thus all conditions of Schauder's theorem are satisfied, so there exists a fixed point of the operator A, which completes the proof.

■

Let us turn to the question of the uniqueness of the solution. It, as a rule, is proved on the basis of the Lipschitzianity of the right part of the equation under study [4, 28–30, 32, 33].

We will still consider the problem on the domain G under the condition that the function $\beta(x,y)$ and all its first and second order partial derivatives are bounded. From (4.2) and (4.3) we see that under the assumptions made, the function $f(x, y, y')$ satisfies the Lipschitz condition on its last two variables for any x in the segment $[0, l]$. That is, there is a constant L such that for any x of the given segment the inequality holds for any x of the given segment

$$|f(x, y_1, y_1') - f(x, y_2, y_2')| \leq L\left(|y_1(x) - y_2(x)| + |y_1'(x) - y_2'(x)|\right). \quad (4.15)$$

Theorem 4.2.4 *Let all conditions of Theorem 2 be satisfied and the partial derivatives of $\beta_{xy}(x, y)$, $\beta_{yyy}(x, y)$ be bounded. Then the solution of the problem (4.1) in the domain G is singular.*

Proof Suppose that in G there exists another solution $z(x)$ to the problem (4.1) besides $y(x)$. Let the point $x_0 \in [0, l]$ be such that

$$y(x) = z(x), \quad y'(x) = z'(x)$$

for any $x \in [0, x_0]$, but

$$y(x) \neq z(x), \quad y'(x) \neq z'(x) \quad (4.16)$$

in any open right neighborhood of the point x_0. Let us take an arbitrary small $\varepsilon > 0$. Since there are points in the segment $[x_0, x_0 + \varepsilon]$ for which (4.16) holds, the function

$$|y(x) - z(x)| + |y'(x) - z'(x)|$$

reaches on it at some point γ its largest value $\theta > 0$. Obviously $\gamma \neq x_0$. The following equations are true

$$y(x) = \int_0^x d\xi \int_0^\xi f(\eta, y(\eta), y'(\eta))d\eta - \frac{x}{l}\int_0^l d\xi \int_0^\xi f(\eta, y(\eta), y'(\eta))d\eta.$$

$$z(x) = \int_0^x d\xi \int_0^\xi f(\eta, z(\eta), z'(\eta))d\eta - \frac{x}{l}\int_0^l d\xi \int_0^\xi f(\eta, z(\eta), z'(\eta))d\eta.$$

Let us subtract one of these equalities from the other and use the Lipschitz condition for estimation (4.15).

$$
\begin{aligned}
|y(\gamma) - z(\gamma)| + |y'(\gamma) - z'(\gamma)| = \theta \leq \\
& \int_{x_0}^{\gamma} d\xi \int_{x_0}^{\xi} |f(\eta, y(\eta), y'(\eta)) - f(\eta, z(\eta), z'(\eta))| d\eta + \\
& + \frac{\gamma}{l} \int_{x_0}^{l} d\xi \int_{x_0}^{\xi} |f(\eta, y(\eta), y'(\eta)) - f(\eta, z(\eta), z'(\eta))| d\eta \leq \\
& \leq 2(\gamma - x_0) \int_{x_0}^{l} L(|y(\eta) - z(\eta)| + |y'(\eta) - z'(\eta)|) d\eta \leq \\
& \leq 2Ll\varepsilon\theta.
\end{aligned}
$$

Thus,

$$\theta \leq 2Ll\varepsilon\theta,$$

whence, by virtue of the positivity of $\theta > 0$, we obtain the inequality

$$1 \leq 2Ll\varepsilon,$$

leading to a contradiction, since ε can be chosen as small as desired.

■

In conclusion, let us illustrate the work of the 4.1 method constructed in paragraph 4.1 with an example. An example program implementing the method proposed in this chapter is given in the appendix A.6.

Primer 4.2.1 *Assume that the construction cost is directly proportional to the elevation of the terrain, and that the terrain surface is defined by the function* $z = \beta(x, y)$. *Let* $l = 1$, $\alpha = 1$ *and the function* $\beta \colon \mathbb{R}^2 \to \mathbb{R}$ *has the form*

$$\beta(x, y) = 1 + \frac{1}{100} \sin 5x \cdot \sin y.$$

Here we can put $M = 0.5$, *and* c *is obviously equal to* 0.05. *Thus, the conditions of Theorems 2 and 3 are satisfied, so in the region of*

$$G = \{y \mid \|y\|_{\mathbb{C}[0,l]} \leq 1,\ \|y'\|_{\mathbb{C}[0,l]} \leq 1\}$$

the solution to the problem exists and is unique.

Applying the proposed method for $N = 1000$*, initial approximation (4.9) and stopping criterion (4.10) at* $\varepsilon = 10^{-6}$ *we come to the solution (see Fig. 4.1) for* $n = 3$ *iterations and 0.17 seconds*[1]*. The value of the cost functional on the obtained solution is* 1.5.

Ritz method, in which the approximate solution is sought in the form of

$$y(x) = \sum_{k=1}^{5} a_k \sin \frac{\pi k}{l} x, \tag{4.17}$$

leads to a solution (see Fig. 4.1) in 0.57 seconds. The following coefficients are obtained

$$a_1 = -0.0002,\ a_2 = -0.0001,\ a_3 = 0.00002,\ a_4 = 0,\ a_5 = 0,$$

and the value of the cost functional on the obtained solution is equal to 1.5.

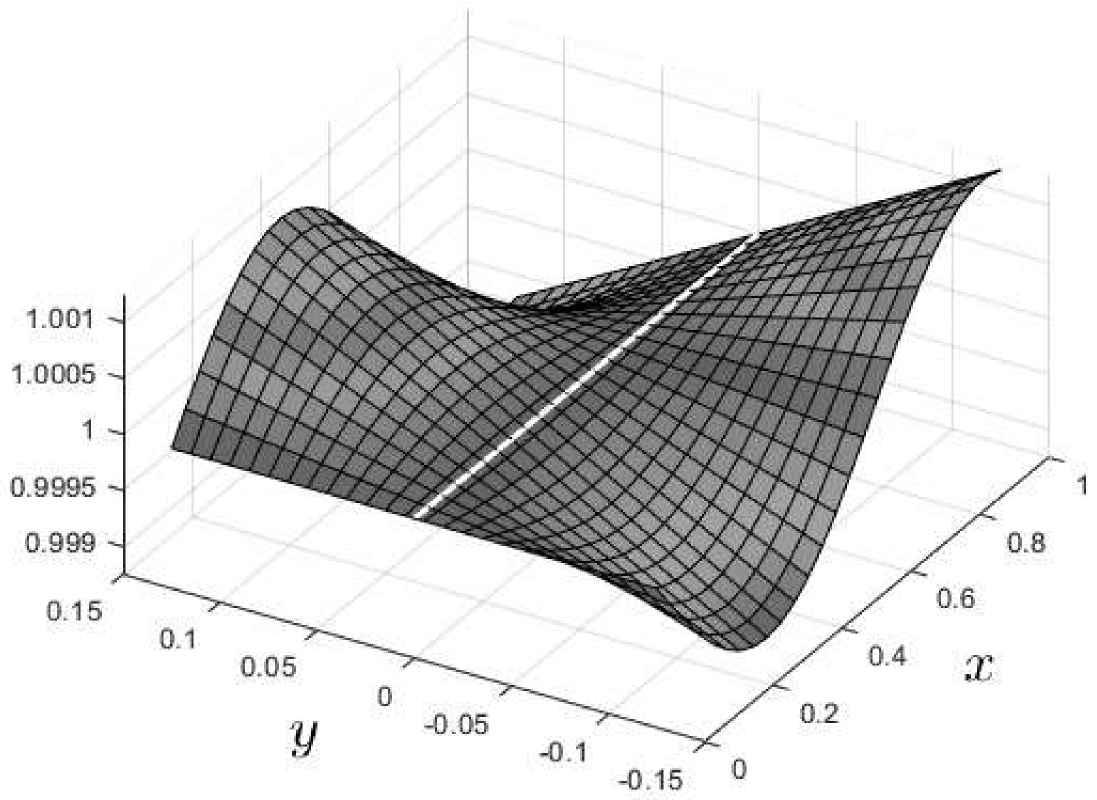

Figure 4.1: Images of visually indistinguishable solutions obtained by the Ritz and gunning methods in Example 4.2.1.

[1]Calculations were performed in the program MatLab R2022b on a PC with Core Duo P8600 processor and 4Gb RAM

In this study, we develop a method for finding the cost-minimal trajectory that uses ideas and approaches of calculus of variations and, therefore, guarantees the optimality of the obtained solution. The cost functional is constructed and the necessary condition of its minimum, which has the form of an integro-differential equation, is derived. Methods of approximate and numerical solution of this equation are proposed. Using the Schuader fixed point principle, the existence theorem of the solution is proved, and the question of its uniqueness is investigated.

Conclusion

We give a brief overview of the results obtained in the paper.

The introduction gives a review of the literature on the topic of the work, discusses the relevance of the study, its theoretical and practical value, and scientific novelty.

The first chapter summarizes the main definitions and auxiliary results from functional analysis, higher algebra, calculus of variations, which are used in the further presentation.

In the second chapter, the formulation of the problem of searching for a cost-optimal trajectory is considered, and the assumptions under which the mathematical model is constructed are introduced. The problem is reduced to the search for the function on which the minimum of the integral cost functional is achieved. To solve this problem of calculus of variations, a necessary condition for the minimum of the constructed functional is derived. This condition has the form of an integro-differential equation.

In the third chapter, approximate methods for solving the problem are proposed. Thus, the integro-differential equation obtained in the first chapter is solved using a method based on polynomial approximation, as well as using the Galerkin method. This chapter also considers a problem formulation in which the delivery cost varies from point to point. It is shown how an approximate solution can be obtained for the problem of minimizing the cost functional obtained for this case using the Ritz method.

The fourth chapter is devoted to the study of existence and uniqueness of the solution of the problem. The existence theorem is proved using Schauder's fixed point principle. In this section, an iterative also method for solving the integro-differential equation based on linearization and the method of shooting.

The appendices also contain program listings that implement the methods of solving the problem proposed in this study.

Further research can be carried out in the direction of model refinement, as well as the study of the problem with constraints. Indeed, an important and of great practical interest is the situation when there are «restricted» areas on the relief, in which there are swamps, ponds, rivers, etc. natural obstacles and obstacles. This problem as well as the one considered in this paper requires derivation of necessary conditions and construction of methods for their solution.

APPENDIX A

Appendices

A.1 Listing of MatLab program implementing the method based on polynomial approximation

```
function z = beta(x,y)
z=1+sin(5*x)*sin(y);
end

function z = betax(x,y)
z=5*cos(5*x)*sin(y);
end

function z = betay(x,y)
z=sin(5*x)*cos(y);
end

function z=diff1(S,y)
m=length(y);
y1=diffy1(S,y);
z=ones(m,1)*y1+S*y;
end

function z=diffy1(S,y)
m=length(y);
A=(S*S)*y;
```

```
B=S*ones(m,1);
z=(1-A(m))/B(m);
end

function z=func_y(S,y)
m=length(y);% m=n+1
y1=diffy1(S,y);
z=S*ones(m,1)*y1+(S*S)*y;
end

function J = J(z,n)
alpha=0.1;
h=1/n;
sum1=0;
for i=1:n
sum1=sum1+sqrt(h^2+(z(i+1)-z(i))^2);
end
sum2=0;
for i=1:n
sum2=sum2+sqrt(h^2+(z(i+1)-z(i))^2)*beta((i-1)*h,z(i));
end
J=0.5*alpha*sum1^2+sum2;
end

function F = root(S,yd2)
m = length(yd2);
h=1/(m-1);
z1=diff1(S,yd2);
y=func_y(S,yd2);
intgrl=simpson(S,yd2);
    for i=1:m
      F(i)=(yd2(i)/(1+(z1(i))^2))*(0.1*intgrl+beta(i*h,y(i)))...
      +(z1(i))*betax(i*h,y(i))-betay(i*h,y(i));
    end
```

```
end

function z = simpson(S,y)
z = 0;
m=length(y);
h=1/(m-1);
z1=diff1(S,y);
    for i = 1:(m-1)/2
     z = z+(1+(z1(2*i-1))^2)^0.5+4*(1+(z1(2*i))^2)^0.5+...
     (1+(z1(2*i+1))^2)^0.5;
    end
z = h*z/3;
end

clear all
clc
n=23; %degree of the approximation polynomial
tic
h=1/n;
x = 0:h:n*h;
Zz=fliplr(vander(x));
for i=1:n+1
    for j=1:n+1
        B(i,j)=(x(i)^j-x(1)^j)/j;
    end
end
B;
S=B/Zz;
[X,Y] = meshgrid([0:0.05:1]);
Z = 1+sin(5*X).*sin(Y);
surf(X,Y,Z);
hold on
w0=zeros(n+1,1);
y0=fsolve(@(y)root(S,y),w0);
```

```
y = (func_y(S,y0))';
toc
x=0:h:n*h;
z=1+sin(5*x).*sin(y);
hPlot = plot3(x,y,z,'k');
set( hPlot, 'LineWidth',2);
J(y,n)
```

A.2 Listing of the MatLab program that implements the Ritz method at constant α.

```
function z = beta(x,y)
z=1+sin(5*x).*sin(y);
end

function y = myfunc(x,a)
l=1;
ya=1;
n=length(a);
y = ya*x/l;
    for k=1:n
    y=y+a(k)*sin(x*k*pi/l);
    end
end

function y = myIntegrand(x,c)
l=1;
ya=1;
n=length(c);

z=ya/l;
for k=1:n
    z=z+c(k)*k*pi/l*cos(x*k*pi/l);
```

```
end
y=sqrt(1+z.^2);
end

function F = myIntegrand2(x,c)
l=1;
ya=1;
n=length(c);
y = ya*x/l;
    for k=1:n
    y=y+c(k)*sin(x*k*pi/l);
    end
F=myIntegrand(x,c).*beta(x,y);
end

function F = Costfunc(c)
alpha=0.1;
l=1;
q1 = integral(@(x) myIntegrand(x,c),0,l);
q2=integral(@(x) myIntegrand2(x,c),0,l);
F=q1*q1*alpha/2+q2;
end

clear all
clc
format long
c0=rand(1,5);
tic
[sol,F1]=fminunc(@(c)Costfunc(c),c0);
toc
x=(0:0.05:1)';
y=myfunc(x,sol);
z=1+sin(5*x).*sin(y);
hPlot = plot3(x,y,z,'w');
```

```
set( hPlot, 'LineWidth',2);
hold on
[X,Y] = meshgrid([0:0.05:1]);
Z = 1+sin(5*X).*sin(Y);
surf(X,Y,Z);
Costfunc(sol)
sol
```

A.3 Listing of MatLab program that implements the Ritz method with variable α.

```
function z = alpha(x,c)
y=phi(x,c);
z=(cos(5*x).*cos(y)).^2;
end

function z = beta(x,c)
y=phi(x,c);
z=1+sin(5*x).*sin(y);
end

function y = phi(x,c)
l=1;
ya=1;
n=length(c);
z=ya*x/l;
for k=1:n
    z=z+c(k)*sin(x*k*pi/l);
end
y=z;
end

function y = myIntegrand(x,c)
```

```
l=1;
ya=1;
n=length(c);
z=ya/l;
for k=1:n
    z=z+c(k)*k*pi/l*cos(x*k*pi/l);
end
y=sqrt(1+z.^2);
end


function q1 = firstsummandint(c)
l=1;
fun = @(x,y) alpha(x,c).*myIntegrand(x,c).*myIntegrand(y,c);
xmin = 0;
xmax = l;
ymin = 0;
ymax = @(x) x;
q1 = integral2(fun,xmin,xmax,ymin,ymax,'Method','tiled');
end

function q2 = secondsummandint(c)
l=1;
F=@(x) myIntegrand(x,c).*beta(x,c);
q2=integral(F,0,l);
end

function F = Costfunc(c)
l=1;
q2=secondsummandint(c);
F=firstsummandint(c)+q2;
end

clear all
```

```
clc
format long
c0=rand(1,5);
[sol,F1]=fminunc(@(c)Costfunc(c),c0);
x=(0:0.05:1)’;
y=phi(x,sol);
z=1+sin(5*x).*sin(y);
hPlot = plot3(x,y,z,’w’);
set( hPlot, ’LineWidth’,4);
hold on
[X,Y] = meshgrid([0:0.05:1]);
Z = 1+sin(5*X).*sin(Y);
surf(X,Y,Z);
Costfunc(sol)
sol
```

A.4 Listing of a Pyton program implementing Galerkin’s method for a system of trigonometric polynomials

```
import sympy
import scipy
from scipy import optimize
import matplotlib.pyplot as plt
import numpy as np
import math
from math import *
from sympy import *
from scipy.optimize import fsolve
def f(x, u1, u2, u3, u4, u5):
    f = [0, 0, 0, 0, 0]
    f1 = u1(x[0], x[1], x[2], x[3], x[4])
    f2 = u2(x[0], x[1], x[2], x[3], x[4])
    f3 = u3(x[0], x[1], x[2], x[3], x[4])
```

```
    f4 = u4(x[0], x[1], x[2], x[3], x[4])
    f5 = u5(x[0], x[1], x[2], x[3], x[4])
    f[0] = f1
    f[1] = f2
    f[2] = f3
    f[3] = f4
    f[4] = f5
    return f

def trapec(f, a, b , n):
   h = (b-a)/n
   k = (f.subs(x, a) + f.subs(x, b))/2
   while(a<b-h):
       a = a+h
       k = k + f.subs(x, a)
   return(k*h)

xa=1
ya=1
a = 0.1
x, a1, a2,a3,a4,a5,y = symbols("x a1 a2 a3 a4 a5 y")
yn = ya*x/xa + a1*(sin(math.pi*x/xa)) + a2*(sin(2*math.pi*x/xa))\
 +a3*(sin(3*math.pi*x/xa))+ a4*(sin(4*math.pi*x/xa))\
 +a5*(sin(5*math.pi*x/xa)) #функция n=5
b = 1 + sin(5*x)*sin(yn) #функция рельефа местности
bstr = 1 + sin(5*x)*sin(y)
bx = 5*sin(yn)*cos(5*x)
by = sin(5*x)*cos(yn)
print(yn)
ynp = yn.diff(x) #первая производная
print(ynp)
yyp =sqrt(1+ sympy.Mul(ynp, ynp)) #умножаем производные
ynp2 = ynp.diff(x) #вторая производная
integr =a * trapec(yyp, 0, xa, 100)#внутренний интеграл
```

```
yob = (sympy.Mul(ynp2, 1/(1 + sympy.Mul(ynp, ynp))))*(integr + b)\
 + ynp*bx - by#общее уравнение
yob1 = yob * (sin(math.pi*x/xa))
yob2 = yob * (sin(2*math.pi*x/xa))
yob3 = yob * (sin(3*math.pi*x/xa))
yob4 = yob * (sin(4*math.pi*x/xa))
yob5 = yob * (sin(5*math.pi*x/xa))

urav1 = trapec(yob1, 0, xa , 5)
urav2 = trapec(yob2, 0, xa , 5)
urav3 = trapec(yob3, 0, xa , 5)
urav4 = trapec(yob4, 0, xa , 5)
urav5 = trapec(yob5, 0, xa , 5)

f1 = lambdify((a1, a2, a3, a4, a5), urav1, 'numpy')
f2 = lambdify((a1, a2, a3, a4, a5), urav2, 'numpy')
f3 = lambdify((a1, a2, a3, a4, a5), urav3, 'numpy')
f4 = lambdify((a1, a2, a3, a4, a5), urav4, 'numpy')
f5 = lambdify((a1, a2, a3, a4, a5), urav5, 'numpy')

#print(f(x0, f1, f2, f3, f4, f5))

sol = fsolve(lambda xx: f(xx, f1, f2, f3, f4, f5),\
 np.asarray([0.0, 0.0, 0.0, 0.0, 0.0]))

print(sol)

yn = ya*x/xa + sol[0]*(sin(math.pi*x/xa))\
 + sol[1]*(sin(2*math.pi*x/xa)) +sol[2]*(sin(3*math.pi*x/xa))\
 + sol[3]*(sin(4*math.pi*x/xa))+sol[4]*(sin(5*math.pi*x/xa))
#получившася функция
integr1 = trapec(sqrt(1+ sympy.Mul(yn.diff(x),\
 yn.diff(x))), 0, xa, 200)
integr2 = trapec((1 + sin(5*math.pi/math.pi*x)*sin(yn))\
```

```
*sqrt(1+ sympy.Mul(yn.diff(x), yn.diff(x))), 0, xa, 200)
res = a/2*integr1*integr1+integr2
print(res)

x0 = np.linspace(0,xa,100)
y0 = np.zeros(100)
z0 = np.zeros(100)
for i in range(len(x0)) :
  y0[i]=yn.subs(x, x0[i])
  b1=bstr.subs(x,x0[i])
  z0[i]=b1.subs(y,y0[i])
fig = plt.figure(figsize =(10, 5))
ax = plt.axes(projection ='3d')
#ax.plot_trisurf(x0, y0, z0)
x00 = np.linspace(0, xa, 100)
y00 = np.linspace(0, ya, 100)
x00, y00 = np.meshgrid(x00, y00)
z00 = 1 + np.sin(5 * x00) * np.sin(y00)
ax.plot_surface(x00, y00, z00, alpha=0.5)
ax.plot(x0, y0, z0, color = "red")
plt.show()
```

A.5 Listing of a Pyton program that implements Galerkin's method for a system of algebraic polynomials

```
import sympy
import scipy
from scipy import optimize
import matplotlib.pyplot as plt
import numpy as np
import math
from math import *
```

```
from sympy import *
from scipy.optimize import fsolve
def f(x, u1, u2, u3, u4, u5):
    f = [0, 0, 0, 0, 0]
    f1 = u1(x[0], x[1], x[2], x[3], x[4])
    f2 = u2(x[0], x[1], x[2], x[3], x[4])
    f3 = u3(x[0], x[1], x[2], x[3], x[4])
    f4 = u4(x[0], x[1], x[2], x[3], x[4])
    f5 = u5(x[0], x[1], x[2], x[3], x[4])
    f[0] = f1
    f[1] = f2
    f[2] = f3
    f[3] = f4
    f[4] = f5
    return f

def trapec(f, a, b , n):
   h = (b-a)/n
   k = (f.subs(x, a) + f.subs(x, b))/2
   while(a<b-h):
       a = a+h
       k = k + f.subs(x, a)
   return(k*h)

xa=1
ya=1
a = 0.1
x, a1, a2,a3,a4,a5,y = symbols("x a1 a2 a3 a4 a5 y")
yn = ya*x/xa + a1*x*(1-x) + a2*x**2*(1-x) +a3*x**3*(1-x)+\
 a4*x**4*(1-x)+a5*x**5*(1-x) #функция n=5
b = 1 + sin(5*x)*sin(yn)
bstr = 1 + sin(5*x)*sin(y)
bx = 5*sin(yn)*cos(5*x)
by = sin(5*x)*cos(yn)
```

```
print(yn)
ynp = yn.diff(x) #первая производная
print(ynp)
yyp =sqrt(1+ sympy.Mul(ynp, ynp)) #умножаем производные
ynp2 = ynp.diff(x) #вторая производная
integr =a * trapec(yyp, 0, xa, 100)#внутренний интеграл
yob = (sympy.Mul(ynp2, 1/(1 + sympy.Mul(ynp, ynp))))*(integr + b)\
 + ynp*bx - by#общее уравнение
yob1 = yob * (x*(1-x))
yob2 = yob * x**2*(1-x)
yob3 = yob * x**3*(1-x)
yob4 = yob * x**4*(1-x)
yob5 = yob * x**5*(1-x)

urav1 = trapec(yob1, 0, xa , 5)
urav2 = trapec(yob2, 0, xa , 5)
urav3 = trapec(yob3, 0, xa , 5)
urav4 = trapec(yob4, 0, xa , 5)
urav5 = trapec(yob5, 0, xa , 5)

f1 = lambdify((a1, a2, a3, a4, a5), urav1, 'numpy')
f2 = lambdify((a1, a2, a3, a4, a5), urav2, 'numpy')
f3 = lambdify((a1, a2, a3, a4, a5), urav3, 'numpy')
f4 = lambdify((a1, a2, a3, a4, a5), urav4, 'numpy')
f5 = lambdify((a1, a2, a3, a4, a5), urav5, 'numpy')


#print(f(x0, f1, f2, f3, f4, f5))

sol = fsolve(lambda xx: f(xx, f1, f2, f3, f4, f5),\
 np.asarray([0.0, 0.0, 0.0, 0.0, 0.0]))

print(sol)
```

```
yn = ya*x/xa + sol[0]*x*(1-x) + sol[1]*x**2*(1-x)\
 +sol[2]*x**3*(1-x)+ sol[3]*x**4*(1-x)+sol[4]*x**5*(1-x)
#получившася функция
integr1 = trapec(sqrt(1+ sympy.Mul(yn.diff(x),\
 yn.diff(x))), 0, xa, 200)
integr2 = trapec((1 + sin(5*math.pi/math.pi*x)\
*sin(math.pi/math.pi*yn))*sqrt(1+ sympy.Mul(yn.diff(x), yn.diff(x))),
res = a/2*integr1*integr1+integr2
print(integr1)
print(integr2)
print(res)

x0 = np.linspace(0,xa,100)
y0 = np.zeros(100)
z0 = np.zeros(100)
for i in range(len(x0)) :
  y0[i]=yn.subs(x, x0[i])
  b1=bstr.subs(x,x0[i])
  z0[i]=b1.subs(y,y0[i])
fig = plt.figure(figsize =(10, 5))
ax = plt.axes(projection ='3d')
#ax.plot_trisurf(x0, y0, z0)
x00 = np.linspace(0, xa, 100)
y00 = np.linspace(0, ya, 100)
x00, y00 = np.meshgrid(x00, y00)
z00 = 1 + np.sin(5 * x00) * np.sin(y00)
ax.plot_surface(x00, y00, z00, alpha=0.5)
ax.plot(x0, y0, z0, color = "red")
plt.show()
```

A.6 Listing of MatLab program that implements the shooting method

```
function z = beta(y)
n=length(y)-1;
h=1/n;
x=0:h:1;
z=1+sin(5*x).*sin(y);
z=z(1,1:end-1);
end

function z = betax(y)
n=length(y)-1;
h=1/n;
x=0:h:1;
z=5*cos(5*x).*sin(y);
z=z(1,1:end-1);
end

function z = betaxy(y)
n=length(y)-1;
h=1/n;
x=0:h:1;
z=5*cos(5*x).*cos(y);
z=z(1,1:end-1);
end

function z = betay(y)
n=length(y)-1;
h=1/n;
x=0:h:1;
z=sin(5*x).*cos(y);
z=z(1,1:end-1);
```

```
end

function z = betayy(y)
n=length(y)-1;
h=1/n;
x=0:h:1;
z=-sin(5*x).*sin(y);
z=z(1,1:end-1);
end

function F = Costfunc(y)
alpha=0.1;
k=length(y);
n=k-1;
h=1/n;
z_dify = sqrt(1+diffy(y).^2);
  z=[z_dify z_dify(end)];
  zz=beta(y).*z_dify;
  zz=[zz zz(end)];
  w=0;
  ww=0;
  for i=1:k
   w=w+z(i);
   ww=ww+zz(i);
  end
  q1=h*(w-(z(1)+z(end))/2);
  q2=h*(ww-(zz(1)+zz(end))/2);
  F=q1*q1*alpha/2+q2;
end

function w = denumi(y,dify)
alpha=0.1;
n=length(y)-1;
h=1/n;
```

```
w=alpha*intdy(dify)+beta(y);
end

function z = diffy(y)
n=length(y)-1;
z=ones(1,n);
h=1/n;
z(1)=(y(2)-y(1))/(h);
  for i=2:n
   z(i)=(y(i+1)-y(i-1))/(2*h);
  end
end

function w = intdy(dify)
n=length(dify);
k=n+1;
h=1/n;
  z=[dify dify(end)];
  z=sqrt(1+z.^2);
  w=0;
  for i=1:k
   w=w+z(i);
  end
  w=h*(w-(z(1)+z(end))/2);
end

function w = intdy1(dify)
n=length(dify);
k=n+1;
h=1/n;
  z=[dify dify(end)];
  z=z./sqrt(1+z.^2);
  w=0;
  for i=1:k
```

```
   w=w+z(i);
  end
  w=h*(w-(z(1)+z(end))/2);
end

function z0 = term0(y,dify)
n=length(y)-1;
z0=(1+dify.^2).*(dify.*betax(y)-betay(y));
end

function z1 = term1(y,dify)
n=length(y)-1;
z1=(1+dify.^2).*(dify.*betaxy(y)-betayy(y)).*...
denumi(y,dify)-betay(y).*(1+dify.^2).*(dify.*betax(y)-betay(y));
end

function z2 = term2(y,dify)
alpha=0.1;
n=length(y)-1;
z2=(2*dify.*(dify.*betax(y)-betay(y))+(1+dify.^2).*betax(y)).*...
denumi(y,dify)-alpha*intdy1(dify).*(1+dify.^2).*...
(dify.*betax(y)-betay(y));
end

clear all
clc
%format long
n=1000;
h=1/n;
%y=[0 1*ones(1,n)/n];
y=0:h:1;
x=0:h:1;
%for k=1:10
eps=1e-6;
```

```
flag=0;
cost_prev=Costfunc(y);
cost_new=Costfunc(y)+1e10*eps;
tic
while abs(cost_prev-cost_new)>eps
z=diffy(y);
d=denumi(y,z);
d2=d.^2;
t0=term0(y,z)./d;
t1=term1(y,z)./d2;
t2=term2(y,z)./d2;
yhom=y;
y_old=y;
for i=2:n
    y(i+1)=(-h^2*(t0(i)+t1(i)*(y(i)-y_old(i))-...
    t2(i)*(y(i-1)/(2*h)+z(i)))+2*y(i)-y(i-1))/(1+0.5*h*t2(i));
    yhom(i+1)=(-h^2*(t1(i)*(yhom(i))-...
    t2(i)*(yhom(i-1)/(2*h)))+...
    2*yhom(i)-yhom(i-1))/(1+0.5*h*t2(i));
end
c=(1-y(end))/yhom(end);
y=y+c*yhom;
cost_new=Costfunc(y);
cost_prev=Costfunc(y_old);
flag=flag+1;
Costfunc(y)
end
toc
sol=y;
[X,Y] = meshgrid([0:0.05:1]);
Z = 1+sin(5*X).*sin(Y);
surf(X,Y,Z);
hold on
```

```
ksi=1+sin(5*x).*sin(sol);
hPlot = plot3(x,sol,ksi,'k');
set( hPlot, 'LineWidth',2);
hold on
sol_gal=galerkin(x);
ksi_gal=1+sin(5*x).*sin(sol_gal);
hPlot = plot3(x,sol_gal,ksi_gal,'g');
set( hPlot, 'LineWidth',2);
hold on
Costfunc(sol)
flag
```

List of designations

$\forall$ – universality quantum,

$\exists$ – existence quantum,

$||.||$ – norm,

$|.|$ – unit,

$\det A$ – matrix determiner A,

$\mathbb{C}[0,l]$ – the space of continuous functions with uniform norm on $[0,l]$,

$\mathbb{C}^k[0,l]$ – the space of k-fold continuously differentiable functions on $[0,l]$,

$\widetilde{\mathscr{L}_p}[0,l]$ – the space of continuous functions with integral norm on $[0,l]$,

$\mathscr{L}_p[0,l]$ – Lebesgue space obtained by augmentation of the space $\widetilde{\mathscr{L}_p}[0,l]$,

$\|\cdot\|_{\mathbb{C}^1[0,l]}$ – space norm $\mathbb{C}^k[0,l]$,

$\|\cdot\|_{\widetilde{\mathscr{L}_p}[0,l]}$ – space norm $\widetilde{\mathscr{L}_p}[0,l]$,

$\delta J(\cdot)=0$ – functional variation J.

Bibliography

1. Abbasov M.E., Sharlay A.S. A variational approach to finding the cost-optimal trajectory// Math. modeling. 2023 T. 35(12) pp. 89–100.

2. Abbasov M.E., Sharlay A.S. A method for searching the cost-optimal road trajectory on the terrain surface// Vestnik of St. Petersburg University. Applied mathematics. Informatics. Control processes. 2023. T.19(2). C. 139–147. https://doi.org/10.21638/11701/spbu10.2023.201

3. Abbasov M.E., Sharlay A.S. Search of the cost-optimal road trajectory on the terrain // Bulletin of St. Petersburg University. University. Applied mathematics. Informatics. Control processes. 2021. T.17(1). pp. 4-12.

4. Arnold V. I. Ordinary differential equations. M.: Nauka, 1972. 240 p.

5. Bandurin N. G., Gureeva N. A. Method and program package for numerical solution of systems of essentially nonlinear ordinary ordinary integro-differential-algebraic equations// Math. modeling. 2012. T. 24(2). pp. 3–16.

6. Bakhvalov N. S., Zhidkov N. P., Kobelkov G. M. Calculus methods. M.: Laboratory of Knowledge, 2020. 636 p.

7. Berezin I. S., Zhidkov N. P. Methods of computation, vol. 1 (2nd ed.). Moscow: Fizmatlit, 1962. 464 p.

8. Berezin I. S., Zhidkov N. P. Methods of computation, vol. 2. Moscow: Fizmatlit, 1959. 620 p.

9. Bliss G.A. Lectures on calculus of variations. MOSCOW: IL, 1950. 349 p.

10. Budak B. A. Shooting method for solving equilibrium programming problems// Journal of Computational Mathematics and Mathematical Physics. 2013. T. 53(12). pp. 1819–1824.

11. Winberg E. B. Course of algebra. 2nd edition, revised and supplemented. Moscow: Factorial Press, 2001. 544 p.

12. Vulich B.Z. Introduction to functional analysis (2nd ed.). Moscow: Nauka, 1967. 415 p.

13. Gavurin M. K. Lectures on methods of computation. Moscow: Nauka, 1971

14. Gantmacher F. R. The theory of matrices (5th ed.). M.: Fizmatlit, 2010. 560 p.

15. Gelfand I.M., Fomin S.V. Variational calculus. Moscow: Fizmatlit, 1961. 227 p.

16. Demidovich B. P., Maron I. A. Foundations of computational mathematics (3rd ed.). Moscow: Nauka, 1966. 664 p.

17. Ioshida K. Functional analysis. M.: Mir, 1967. 624 p.

18. Kantorovich L. V., Akilov G. P. Functional analysis. (3rd ed.). M.: Nauka, 1984. 752 p.

19. Kantorovich L. V. V., Krylov V. I. Approximate methods of higher analysis. L.: Fizmatiz, 1962. 708 p.

20. Collatz L. Functional analysis and computational mathematics. M.: Mir, 1969. 448 p.

21. Kolmogorov A. N., Fomin S. V. Elements of the theory of functions and functional analysis (4th ed.). M.: Nauka, 1976. 544 p.

22. Kostrikin A. I. Introduction to algebra Part 1. Fundamentals of algebra (4th ed.). M.: MCNMO, 2000. 272 p.

23. Kurosh A. G. Course of higher algebra. M.: Nauka, 1965. 431 p.

24. Lusternik L. A., Lavrentiev M. A. Course of variational calculus. M.: GONTI, 1938. 192 p.

25. Lusternik L.A., Sobolev V.I. Elements of functional analysis. Moscow: Nauka, 1965. 520 p.

26. Mikhlin S. G. Variational methods in mathematical physics. M.: Fizmatgiz, 1970. 512 p.

27. Mikhlin S. G. Calculus realization of variational methods. Moscow: Fizmatgiz, 1966. 432 p.

28. Petrovskii I. Г. Lectures on the theory of ordinary differential equations. Moscow: Fizmatlit, 2009. 208 p.

29. Pontryagin L. S. Ordinary differential equations. M.: Nauka, 1961. 312 p.

30. Stepanov V. V. Course of differential equations. M.: GIFML, 1958. 468 p.

31. Trenogin V. A. Functional analysis. M.: Nauka, 1980. 495 p.

32. Tricomi F. Differential equations. Per. s Engl. Izd. 4, stereot. M.: Izdatelstvo zarubezhnoi literatury. 2010. 352 p.

33. Filippov A. F. Introduction to the theory of differential equations. 2nd ed., revised. Moscow: ComKniga, 2007. 240 p.

34. Tslaf L.Y. Variation calculus and integral equations. Moscow: Nauka, 1966. 176 p.

35. Chernousko F.L., Banichuk N.V. Variational problems of mechanics and control (Numerical methods). M.: Nauka, 1973. 236 p.

36. Elsholtz L. E. Differential equations and calculus of variations. M.: Nauka, 1969. 424 p.

37. A. Keibolahi, Y. Kiani et al. Nonlinear rapid heating of shallow arches, Journal of Thermal Stresses, Vol. 41(10-12), pp. 1244–1258, (2018). DOI: 10.1080/01495739.2018.1494522

38. Abbasov M.E., Sharlay A.S. Finding a cost optimal road trajectory with respect to the slope of the terrain// PROCEEDINGS of the 8th International Conference on CONTROL AND OPTIMIZATION WITH INDUSTRIAL APPLICATIONS. Volume I. 2023, pp. 27–29.

39. Abbasov M.E., Sharlay A.S. Shooting Method for Finding Cost Optimal Trajectory// 2023 5th International Conference on Problems of Cybernetics and Informatics (PCI). 2023, pp. 1–3, doi: 10.1109/PCI60110.2023.10325965.

40. Abbasov M.E., Sharlay A.S., Belenok A.I. Galerkin method for finding cost optimal trajectory// XIV International Conferenrence on Optimization Methods and Applications OPTIMIZATION AND APPLICATIONS (OPTIMA-2023). BOOK OF ABSTRACTS. 2023. p. 16

41. Akbas S. D., Ersoy H. et al. Dynamic analysis of a fiber-reinforced composite beam under a moving load by the Ritz method. Mathematics 9(9), 1048 (2021). https://doi.org/10.3390/math9091048

42. Ali A. M., Jasim M. H., Al-Kasob B. D. H. Low velocity impact study of a sandwich beams using Ritz method and finite element modelling// Journal of Engineering, Design and Technology. 2022. https://doi.org/10.1108/JEDT-10-2021-0584

43. Altenbach, H., Ochsner, A. (eds) Ritz–Galerkin Methods. In: Encyclopedia of Continuum Mechanics. Springer, Berlin, Heidelberg, p. 2193, (2020). https://doi.org/10.1007/978-3-662-55771-6_300558

44. Amein N.K., Ramadan M. A. A small time solutions for the KdV equation using Bubnov-Galerkin finite element method, Journal of the Egyptian Mathematical Society, Vol. 19(3), pp.118–125, (2011). DOI: 10.1016/j.joems.2011.10.005

45. Ashok R., Manam S. R. Oblique Wave Scattering Problems Involving Vertical Porous Membranes. J. Marine. Sci. Appl., Vol.21, pp. 51–66 (2022). https://doi.org/10.1007/s11804-022-00255-0

46. C. Yu, J. Lee et. al. Extensions to least-cost path algorithms for roadway planning. International Journal of Geographical Information Science. 17(4), pp. 361–376 (2003)

47. Carsten J., Rankin A. et al. Global Planning on the Mars Exploration Rovers: Software Integration and Surface Testing, Journal of Field Robotics 26(4), pp. 337–357, (2009)

48. Chang W. S., Hassan H. et al. Error Analysis on Galerkin Scheme for the Diffusion Problem. In: Rajendran, P., Mazlan, N., Rahman, A., Suhadis, N., Razak, N., Abidin, M. (eds) Proceedings of International Conference of Aerospace and Mechanical Engineering 2019. Lecture Notes in Mechanical Engineering. Springer, Singapore. (2020) https://doi.org/10.1007/978-981-15-4756-0_4

49. Chapr S. C Applied numerical methods with MATLAB for engineers and scientists, 3rd edn. McGraw-Hill, (2012). p. 622

50. Chen G .R., Guo S. et al. Convex optimization and A-star algorithm combined path planning and obstacle avoidance algorithm. Control and Decision. 35, pp. 2907–2914 (2020)

51. Conte S. D., De Boor C. Elementary numerical analysis: an algorithmic approach, 3rd edn. McGraw-Hill, (1980). p. 416

52. Tomlin D. Propagating radial waves of travel cost in a grid// International Journal of Geographical Information Science. 2010. Vol. 24(9). pp. 1391–1413.

53. Das B. C., Soumen D. et al. Oblique scattering by thin vertical barriers: solution by multi-term Galerkin technique using simple polynomials as basis. J. Mar. Sci. Technol., Vol.23, pp. 915–925, (2018). https://doi.org/10.1007/s00773-017-0520-4

54. Davis M. E. Numerical methods and modeling for chemical engineers. Wiley, (1984). p. 258.

55. Deuflhard P. Recent Advances in Multiple Shooting Techniques,In Computational Techniquesfor Ordinary Differential Equations, I. Gladwell and D. K. Sayers (eds.), Academic, London (1980).

56. Douglas D. H. Least cost path in GIS using an accumulated cost surface and slope lines// Cartographica. 1994. Vol. 31. pp. 37–51.

57. El-Dib Y. O., Elgazery N. S. Galerkin's method to solve a fractional time-delayed jerk oscillator. Arch Appl Mech 93, pp. 3597–3607 (2023). DOI: 10.1007/s00419-023-02455-8

58. Esfandiari R. S. Numerical methods for engineers and scientists using MATLAB, 2nd edn. (Chap. 8.4). CRC Press, Inc., (2017). p. 472.

59. Firoozjaee, M.A., Jafari, H., Lia, A. et al. Numerical approach of Fokker–Planck equation with Caputo–Fabrizio fractional derivative using Ritz approximation. J. Comput. Appl. Math. Vol. 339, pp. 367–373 (2018).

60. Fox L., Mayers D. F. Numerical solution of ordinary differential equations. Chapman & Hall, London, (1987). p. 107.

61. Friz P. K., Gassiat P. et al. Short-dated smile under rough volatility: asymptotics and numerics, Quantitative Finance, Vol. 22(3), pp. 463–480, (2022). DOI: 10.1080/14697688.2021.1999486

62. Gass S.I., Harris C.M. Dijkstra's algorithm . In: Gass, S.I., Harris, C.M. (eds) Encyclopedia of Operations Research and Management Science. Springer, New York, NY (2001)

63. Gurara D., Klyuev V. et al. Trends and challenges in infrastructure investment in low-income developing countries. IMF working papers (2017)

64. He D.-Q., Wang H.-B. et al. Robot path planning using improved rapidly-exploring random tree algorithm, IEEE Industrial Cyber-Physical Systems (ICPS), pp. 181–186, (2018)

65. Hoffman J. D. Numerical methods for engineers and scientists, 2nd edn. Marcel Dekker, Inc., (1992). p. 823.

66. Huang X. Y., Wang J. et al. Nonlinear Dynamics Analysis for a Model-Reduced Rotor System with Nonlinear Galerkin Method. In: Dimitrovova, Z., Biswas, P., Goncalves, R., Silva, T. (eds) Recent Trends in Wave Mechanics and Vibrations. WMVC 2022. Mechanisms and Machine Science, Vol. 125. Springer, Cham. (2023) https://doi.org/10.1007/978-3-031-15758-5_51

67. Huchhanagouda H. P., Pitchaimani J. et al. Buckling and vibration of beams using Ritz method: Effects of axial grading of GPL and axially varying load, Mechanics of Advanced Materials and Structures, 2023. DOI: 10.1080/15376494.2023.2185711

68. Iqbal A., Abbas M. et al. A Galerkin approach to solitary wave propagation for the second-order nonlinear evolution equation based on quartic B-spline functions, International Journal of Computer Mathematics, Vol. 99(11), pp. 2205–2220, (2022). DOI: 10.1080/00207160.2022.2039388

69. J. Bruce, M. Veloso RoboCup 2002: Robot Soccer World Cup VI, In Real-Time Randomized Path Planning for Robot Navigation (Lecture Notes in Computer Science, Berlin: Springer). 2752, pp. 288–295, (2003)

70. Jahan S., Ferdows M. et al. Convective flow of hybrid nano particles in combination of TiO2+CuO/engine oil MoS_2+ZnO/engine oil and Al_2O_3+Cu/engine oil with viscous dissipation over vertically moving surface: Numerical and Galerkin approach, Numerical Heat Transfer, Part A: Applications, (2023). DOI: 10.1080/10407782.2023.2261145

71. Jaillet L., Cortes J., Simeon T. Sampling-Based Path Planning on Costmaps Configuration-space, Ieee Trans. Robot. 26(4), pp. 635–646 (2010)

72. Kalashnikov S., Gurova E. et al. Influence of the Type of Basis Functions in the Bubnov-Galerkin Method in the Deformation Analysis of a Compressed-Curved Rod with Induced Anisotropy. In: Klyuev, S.V., Vatin, N.I., Sabitov, L.S. (eds) Industrial and Civil Construction 2022. ISCICC 2022. Lecture Notes in Civil Engineering, Vol 436. Springer, Cham. (2024). DOI: 10.1007/978-3-031-44432-6_6

73. Keskin A. U. The Shooting Method for the Solution of One-Dimensional BVPs. In: Boundary Value Problems for Engineers. Springer, Cham, pp. 167–258, (2019). https://doi.org/10.1007/978-3-030-21080-9_5

74. Khalili M. M., Keibolahi A. et al. Application of Ritz method to large amplitude rapid surface heating of FGM shallow arches. Arch Appl Mech 92, 1287–1301 (2022). https://doi.org/10.1007/s00419-022-02106-4

75. Kulagin N. E., Lerman L. M. On periodically modulated rolls in the generalized Swift–Hohenberg equation: Galerkin' approximations, Physica D: Nonlinear Phenomena, Vol. 454, 133845, (2023). DOI:10.1016/j.physd.2023.133845

76. LaValle S. M. Rapidly-exploring random trees: a new tool for path planning, The annual research report, 1998

77. LaValle S. M., Kuffner J. J. RRT-connect: An efficient approach to single-query path planning, IEEE International Conference on Robotics and Automation (2000)

78. Li X., Zhao G. et. al. Generating optimal path by level set approach for a mobile robot moving in static/dynamic environments// Applied Mathematical Modelling. 2020. Vol. 85. pp. 210–230.

79. Li Y., Wei W. et al. PQ-RRT*: an improved path planning algorithm for mobile robots. Expert Syst Appl. 152:113425, (2020)

80. Lin H., Atluri N. S. The Meshless Local Petrov–Galerkin (MLPG) method for solving incompressible Navier–Stokes equations. Comput. Model. Eng. Sci., Vol. 2, pp. 117—142, (2001)

81. Lytvyn O. M., Lytvyn O. O., Tomanova I. S. Solving the biharmonic plate bending problem by the Ritz method using explicit formulas for splines of degree 5// Cybern Syst Anal. 2018. Vol. 54, pp. 944–947.

82. Mamehrashi K., Yousefi S. A. A numerical method for solving a nonlinear 2-D optimal control problem with the classical diffusion equa-

tion. International Journal of Control. Vol 90(2), pp. 298–306. (2016). doi:10.1080/00207179.2016.1178807

83. Mandal B. N., De S. Use of Galerkin Technique in Some Water Wave Scattering Problems Involving Plane Vertical Barriers. In: Dutta, H., Peters, J. (eds) Applied Mathematical Analysis: Theory, Methods, and Applications. Studies in Systems, Decision and Control, Vol.177. Springer, Cham., (2020). https://doi.org/10.1007/978-3-319-99918-0_13

84. Mordukhovich B. S. Variational Analysis and Generalized Differentiation II. Springer-Verlag Berlin Heidelberg, 2006. XXII, 610 p.

85. Nahid N., Nelakanti G. Convergence analysis of Galerkin and multi-Galerkin methods for nonlinear-Hammerstein integral equations on the half-line using Laguerre polynomials, International Journal of Computer Mathematics, Vol. 99(4), pp. 808–836, (2022). DOI: 10.1080/00207160.2021.1937612

86. Nemat A., Yousefi S. A. A numerical method for solving fractional optimal control problems using Ritz method// ASME. J. Comput. Nonlinear Dynam. 2016. Vol. 11(5) N 051015.

87. Nguyen A. T., Nguyen C. T. al. A novel weighted local averaging for the Galerkin method with application to elastic buckling of Euler column, International Journal for Computational Methods in Engineering Science and Mechanics, 24:2, pp. 128–142, (2023). DOI: 10.1080/15502287.2022.2080612

88. Rindler F. Calculus of Variations. Springer International Publishing, 2018. XII, 444 p.

89. Saranya C., Unnikrishnan M. et al., Lalithambika VR, Terrain Based D* Algorithm for Path Planning. IFAC-PapersOnLine. 49(1), pp. 178–182 (2016)

90. Sprunk C., Lau B. et. al. An accurate and efficient navigation system for omnidirectional robots in industrial environments// Autonomous Robots. 2016. Vol. 41(2), pp. 473–493.

91. Sudhakara P., Ganapathy V. Trajectory planning of a mobile robot using enhanced A-star algorithm. Indian J. Sci. Technol. 9(41), pp. 1–10 (2016)

92. Teukolsky W H, Vetterling S. A. et al. "Section 18.1. The Shooting Method", Numerical Recipes: The Art of Scientific Computing (3rd ed.), New York: Cambridge University Press, (2007). ISBN 978-0-521-88068- 8

93. D. Tomlin, Propagating radial waves of travel cost in a grid, International Journal of Geographical Information Science. 24(9), pp. 1391–1413 (2010)

94. Wang W., Zuo L. et al. A learning-based multi-RRT approach for robot path planning in narrow passages. J Intell Robot Syst. 90(1), pp. 81–100 (2018)

95. Xiong X., Min H. et al. Application improvement of A* algorithm in intelligent vehicle trajectory planning. Math Biosci Eng MBE 18(1). pp. 1–21, (2021)

96. Xue J., Wang Y. Free vibration analysis of a flat stiffened plate with side crack through the Ritz method// Arch. Appl. Mech. 2019. Vol. 89. pp. 2089–2102.

97. Yates, J., Wang, X. and Chen, N. Assessing the effectiveness of k-shortest path sets in problems of network interdiction. Optim Eng. 15, pp. 721–749 (2014).

98. Yu C., Lee J. et. al. Extensions to least-cost path algorithms for roadway planning// International Journal of Geographical Information Science. 2003. Vol. 17(4). pp. 361–376.

99. Yuan J., Yang Y. et al. Frequency analysis of beams with LASMP using Ritz method. Ferroelectrics, 555(1), pp. 211–223. (2020). doi:10.1080/00150193.2019.1691397

100. Yuan Yi J., Sun Q. R. et al. Path planning of a manipulator based on an improved P_RRT* algorithm. Complex Intell. Syst. 8, pp. 2227–2245 (2022)

101. Zazai M. F., Fugenschuh A. R Computing the trajectories for the development of optimal routes. Optimization and Engineering. 22, pp. 975–999 (2021)