ᲘᲕᲐᲜᲔ ᲯᲐᲕᲐᲮᲘᲨᲕᲘᲚᲘᲡ ᲡᲐᲮᲔᲚᲝᲑᲘᲡ ᲗᲑᲘᲚᲘᲡᲘᲡ ᲡᲐᲮᲔᲚᲛᲬᲘᲤᲝ

ᲣᲜᲘᲕᲔᲠᲡᲘᲢᲔᲢᲘ

ᲡᲐᲛᲐᲒᲘᲡᲢᲠᲝ ᲜᲐᲨᲠᲝᲛᲘ

ავტორი: ზურაბ ვაშაკიძე

ვარიაციულ-სხვაობიანი სქემა კირხოფის ორგანზომილებიანი არაწრფივი დინამიური განტოლებისათვის

სამაგისტრო პროგრამა: გამოყენებითი მათემატიკა

ნაშრომი წარმოდგენილია მეცნიერებათა მაგისტრის აკადემიური ხარისხის მოსაპოვებლად

ხელმძღვანელები:

ჯემალ როგავა, ფიზიკა-მათემატიკის მეცნიერებათა დოქტორი, ასოცირებული პროფესორი

არჩილ პაპუკაშვილი, ფიზიკა-მათემატიკის მეცნიერებათა კანდიდატი

თბილისი, 2017

IVANE JAVAKHISHVILI TBILISI STATE UNIVERSITY

MASTER'S THESIS

Author: Zurab Vashakidze

VARIATIONAL–DIFFERENCE SCHEME FOR KIRCHHOFF TWO–DIMENSIONAL NONLINEAR DYNAMICAL EQUATION

Master's Degree Program: Applied Mathematics

A thesis presented for the academic degree of Master of Science

Supervisors:
Jemal Rogava, Doctor of Physico–Mathematical Sciences, Associate Professor
Archil Papukashvili, Candidate of Physico–Mathematical Sciences

Tbilisi, 2017

სარჩევი

ანოტაცია

ნაშრომში სიმისთვის კირხოფის არაწრფივი კლასიკურ განტოლებასთან ერთად განხილულია მისი ორგანზომილებიანი განზოგადება. ჩვენი მიზანია ამ განტოლებებისათვის დასმული საწყის-სასაზღვრო ამოცანის მიახლოებითი ამოხსნა. ამისთვის ჩვენ ვიყენებთ სამშრიან, სიმეტრიულ ნახევრადდისკრეტულ სქემას დროითი ცვლადის მიხედვით, სადაც გრადიენტის მნიშვნელობა არაწრფივ წევრში აღებულია შუა წერტილში. ეს ნიუანსი მნიშვნელოვანია, რადგან მიახლოებითი ამონახსნის ყოველ დროით ბიჯზე გამოთვლისათვის საკმარისია წრფივი ოპერატორის შებრუნება. სივრცითი ცვლადების მიხედვით გამოყენებულია ვარიაციული მეთოდი. საკორდინატო ფუნქციებად აღებულა სინუსები და ლეჟანდრის პოლინომების სხვაობა. საბაზისო ფუნქციებად ლეჟანდრის პოლინომების სხვაობის აღება მნიშვნელოვანია რიცხვითი რეალიზაციის თვალსაზრისით. ამ შემთხვევაში მიიღება ისეთი სისტემა, რომლის სტრუქტურა არსებითად არ განსხვავდება შესაბამის სხვაობიან განტოლებათა სისტემისაგან, რაც გვაძლევს საშუალებას გამოყენებულ იქნეს სხვაობიანი სისტემის ამოხსნისათვის დამუშავებული მეთოდები.

კირხოფის სივრცით ერთგანზომილებიანი განტოლების შესაბამისი წრფივი ვარიაციული ამოცანისთვის შეფასებულია მიახლოებითი ამონახსნის ცდომილება და დადგენილია კრებადობის რიგი საკოორდინატო ფუნქციების რიცხვის მიხედვით. ზოგადი ოპერატორული განტოლებისთვის, სიმეტრიული ოპერატორით, დამტკიცებულია ვარიაციული სისტემის შესაბამისი მატრიცის დადებითად განსაზღვრულობა, როცა საკოორდინატო ფუნქციები აკმაყოფილებენ გარკვეულ ბუნებრივ პირობებს.

შემოთავაზებული ალგორითმის საფუძველზე შეიქმნა რიცხვითი რეალიზაციის პროგრამა შესაბამისი ინტერფეისით. ჩატარდა რიცხვითი გათვლები სხვადასხვა ტიპის მოდელური ამოცანებისათვის, როგორც ერთგანზომილებიანი ასევე ორგანზომილებიანი შემთხვევისათვის. მიღებულ თეორიულ შედეგებზე და რიცხვით გათვლებზე დაყრდნობით გაკეთდა პრაქტიკული დასკვნები მეთოდის მდგრადობისა და კრებადობის შესახებ.

Annotation

In the present work, we consider the classical nonlinear Kirchhoff string equation and study its two-dimensional generalization. Our goal is to find an approximate solution to the initial-boundary value problem for this equation. To do so, we apply a three-layer symmetrical semi-discrete scheme with respect to time variable, in which the gradient value of a nonlinear term is taken at the middle point. This detail is essential, because the inversion of the linear operator is sufficient for computation of approximate solutions for each time step. The variation method is applied for spatial variables. Sine function and differences of the Legendre polynomials were used as coordinate functions. This choice of Legendre polynomials is also important for numerical realization. This way we obtain a system whose structure does not essentially differ from the corresponding difference equations system, allowing us to use the methods developed for solving difference equations system.

Linear variation problem for one-dimensional Kirchhoff equation (for spatial dimension) is considered, error of approximate solution is estimated, and convergence order considering the number of spatial functions is found. General operator equation is considered for symmetric operators. We prove that the matrix corresponding to its variation system is positively defined, when coordinate functions satisfy some natural conditions that will be specified in this work.

Numerical realization program with corresponding interface was created based on the offered algorithm, and numerical computations were carried out for model problems both for one-dimensional and two-dimensional cases. Based on the obtained theoretical results and numerical computations, the practical conclusions about the stability and convergence of the offered method were drawn.

შესავალი

კირხოფის სიმის განტოლებისთვის საწყის-სასაზღვრო ამოცანის, როგორც ლოკალური ასევე გლობალური ამონახსნის არსებობისა და ერთადერთობის საკითხები პირველად შესწავლილ იქნა Bernstein-ის მიერ 1940 წელს (იხ. [3]). კლასიკური და განზოგადებული კირხოფის განტოლების ამოხსნადობის საკითხები შემდგომში განხილული იყო მრავალი ავტორის მიერ: A. Arosio, S. Panizzi [2], L. Berselli, R. Manfrin [4], P. D'Ancona, S. Spagnolo [9, 10], R. Manfrin [15], L.A. Medeiros [17], M. Matos [16], K. Nishihara [18], S. Panizzi [20]. [2, 9, 10, 15] და [18] შრომებში კირხოფის განზოგადებული განტოლებისათვის საფუძვლიანად არის შესწავლილი კორექტულობისა და გლობალური ამოხსნადობის საკითხები. [20]-ში კირხოფის ტიპის განტოლებისათვის შესწავლილია გლობალური ამონახსნის არსებობა სუსტი რეგულარულობით.

კირხოფის ტიპის ძელის განტოლების აბსტრაქტული ანალოგი განხილულია L.A. Medeiros-ის მიერ [17], სადაც დამტკიცებულია კოშის ამოცანის რეგულარული ამონახსნის არსებობისა და ერთადერთობის თეორემები. იგივე არაწრფივი განტოლება, გაძლიერებული პირველი რიგის დროითი წარმოებულით, განხილულია ნაშრომებში P. Biler-ის და E.H. Brito-ს მიერ (იხ. [5,6]), სადაც მნიშვნელოვანი ყურადღება ექცევა კოშის ამოცანის ამონახსნის ყოფაქცევის შესწავლას. გვინდა აღვნიშნოთ, რომ ძირითადი ოპერატორის კვადრატის მონაწილეობა ამ განტოლების წრფივ ნაწილში არსებითად გვეხმარება საჭირო აპრიორული შეფასების მიღებაში.

კირხოფის კლასიკური და განზოგადებული განტოლებებისათვის საწყის-სასაზღვრო ამოცანის მიახლოებით ამოხსნას ეძღვნება შემდეგი შრომები: A.I. Christie, J. Sanz-Serna [8], I.S. Liu, M.A. Rincon [14], J. Peradze [24], J. Rogava, M. Tsiklauri [27, 28]. კირხოფის განტოლების რიცხვითი ამოხსნის კუთხით აღვნიშნოთ ასევე შრომებში [21], [23] და [22].

ნაშრომში სიმისთვის კირხოფის არაწრფივი კლასიკურ განტოლებასთან ერთად განხილულია მისი ორგანზომილებიანი განზოგადება. ჩვენი მიზანია ამ განტოლებებისათვის დასმული საწყის-სასაზღვრო ამოცანის მიახლოებითი ამოხსნა. მიახლოებით ამონახსნს ვეძებთ ლოკალურად წრფივი, ნახევრადდისკრეტული სამშრიანი სქემის გამოყენებით, სადაც არაწრფივ წევრში შემავალი გრადიენტის მნიშვნელობა აღებულია შუა წერტილში. ეს ნიუანსი მნიშვნელოვანია, რადგან მიახლოებითი ამონახსნის ყოველ დროით ბიჯზე გამოთვლისათვის გვიხდება წრფივი ამოცანის ამოხსნა. განსხვავებით სხვა ავტორების მიერ განხილული სქემისგან, სადაც დროის ყოველ ბიჯზე ხდება არაწრფივი ამოცანის ამოხსნა იტერაციის გამოყენებით. სივრცითი ცვლადების მიხედვით გამოყენებულია ვარიაციული მეთოდი. საკორდინატო ფუნქციებად აღებულა სინუსები და ლეჟანდრის პოლინომების სხვაობა. საბაზისო ფუნქციებად ლეჟანდრის პოლინომების სხვაობის აღება მნიშვნელოვანია რიცხვითი რეალიზაციის თვალსაზრისით. ამ

შემთხვევაში მიიღება ისეთი სისტემა, რომლის სტუქტურა არსებითად არ განსხვავდება შესაბამის სხვაობიან განტოლებათა სისტემისაგან, რაც გვაძლევს საშუალებას გამოყენებულ იქნეს სხვაობიანი სისტემის ამოხსნისათვის დამუშავებული მეთოდები.

წარმოდგენილ ნაშრომში სივრცითი ცვლადის მიხედვით ერთი და ორგანზომილებიანი კირხოფის არაწრფივი დინამიური განტოლებების მიახლოებითი ამოხსნის სქემების აგება, ერთის მხრივ ეფუძნება დროითი ცვლადის მიხედვით წარმოებულების სასრული სხვაობებით აპროქსიმაციას, ხოლო მეორის მხრივ კი სივრცითი ცვლადების მიხედვით ვარიაციული მეთოდის გამოყენებას. ამ ორი მეთოდის კომბინირებით მიღებულ სქემას, ბუნებრივია ვუწოდოთ ვარიაციულ-სხვაობიანი სქემა.

ნაშრომში მნიშვნელოვანი ნაბიჯია გადადგმული რიცხვითი რეალიზაციის კუთხით. საერთოდ უნდა აღინიშნოს, რომ ლიტერატურაში კირხოფის არაწრფივ განტოლების რიცხვით რეალიზაციას ნაკლები ყურადღება აქვს დათმობილი. წარმოდგენილ სამაგისტრო ნაშრომში განხორციელდა რიცხვითი რეალიზაცია, როგორც სივრცით ერთგანზომილებიანი ისე ორგანზომილებიანი ამოცანებისათვის.

თავი 1

ვარიაციულ-სხვაობიანი სქემა კირხოფის სივრცით ერთგანზომილებიანი არაწრფივი დინამიური განტოლებისათვის

1.1 გალიორკინის მეთოდი და მისი გამოყენება კონკრეტული ამოცანისათვის

ქვემოთ მოყვანილი საკითხები გადმოცემულია [32] და [26] ლიტერატურაში.

H ჰილბერტის სივრცეში განვიხილოთ ოპერატორული განტოლება:

$$Au = f, \ \ f \in H \tag{1.1}$$

სადაც f არის ვექტორი H-დან, ხოლო u არის საძიებელი ვექტორი, A არის წრფივი ასახვა H-დან, H-ში $(A : H \to H)$, $D(A)$ განსაზღვრის არით, რომელიც მკვრივია H-ში $\Big($ეს ნიშნავს, რომ $\overline{D(A)} = H\Big)$. A ოპერატორი საზოგადოდ არის შემოუსაზღვრელი. თუ A შემოსაზღვრულია, მაშინ $D(A) = H$.

შევნიშნოთ, რომ როცა A არის დიფერენციალური ოპერატორი (მაგ. $A = -\Delta$), მაშინ ის შემოუსაზღვრელია.

ვთქვათ H სივრცეში არსებობს

$$\varphi_1, \varphi_2, \dots, \varphi_n, \dots \tag{1.2}$$

ვექტორთა სისტემა, რომელიც სრულია. ეს ნიშნავს, რომ H-ში ნებისმიერ u ვექტორს შეგვიძლია მივუახლოვდეთ ნებისმიერი სიზუსტით (1.2) სისტემის სასრული წრფივი კომბინაციისთ. უფრო ზუსტად $\forall \varepsilon > 0$ ნატურალური რიცხვისათვის არსებობს (მოიძებნება) ისეთი $c_1, c_2, \dots, c_N$

რიცხვები, რომ შესრულდება უტოლობა:

$$\left\| u - \sum_{k=1}^{N} c_k \varphi_k \right\| \le \varepsilon$$

ვთქვათ, ვექტორთა (1.2) სისტემა შედის A ოპერატორის განსაზღვრის არეში, მაშინ ბუნებრივია (1.1) განტოლების ამონახსნი ვეძებოთ შემდეგი კომბინაციის სახით:

$$u = \sum_{k=1}^{\infty} c_k \varphi_k \tag{1.3}$$

სადაც c_k უცნობი კოეფიციენტებია. ჩავსვათ ეს კომბინაცია (1.1) განტოლებაში და მოვითხოვოთ, რომ $(Au - f)$ ვექტორი იყოს ორთოგონალური (1.2) სისტემაში შემავალი ყოველი ვექტორის.

გეომეტრიული მოსაზრება გვეუბნება, რომ თუ g ვექტორი ორთოგონალურია (1.2) სისტემის ყოველი ვექტორის (პირობის თანახმად (1.2) სისტემა სრულია), მაშინ ის წარმოადგენს ნულოვან ვექტორს.

მართლაც, ვთქვათ $g_1, g_2, \ldots, g_n, \ldots$ მიმდევრობის წევრები არიან g ვექტორის მიახლოებები, მაშინ პირობის თანახმად გვაქვს $(g_n, g) = 0$ $\forall n$-თვის. თუ გადავალთ ზღვარზე, როცა $g_n \to g$ მივიღებთ:

$$(g, g) = \|g\|^2 = 0 \;\;\Rightarrow\;\; g = 0.$$

ამ ფაქტზე დაყრდნობით ვღებულობთ:

$$\left(A \left(\sum_{k=1}^{\infty} c_k \varphi_k \right) - f, \varphi_i \right) = 0, \;\; i = 1, 2, \ldots$$

ეს ტოლობა გაშლილი სახით ასე ჩაიწერება:

$$\begin{aligned}
&(A\varphi_1, \varphi_1)\, c_1 + (A\varphi_2, \varphi_1)\, c_2 + \ldots + (A\varphi_N, \varphi_1)\, c_N + \ldots = (f, \varphi_1) \\
&(A\varphi_1, \varphi_2)\, c_1 + (A\varphi_2, \varphi_2)\, c_2 + \ldots + (A\varphi_N, \varphi_2)\, c_N + \ldots = (f, \varphi_2) \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
&(A\varphi_1, \varphi_N)\, c_1 + (A\varphi_2, \varphi_N)\, c_2 + \ldots + (A\varphi_N, \varphi_N)\, c_N + \ldots = (f, \varphi_N) \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots
\end{aligned} \tag{1.4}$$

ამრიგად, c_k კოეფიციენტების მიმართ მივიღეთ (1.4) უსასრულო განტოლებათა სისტემა.

ვთქვათ (1.4) სისტემას აქვს ერთადერთი ამონახსნი. თუ ამ ამონახსნს ჩავსვამთ (1.3) კომბინაციაში, მაშინ აგებული u ვექტორი დააკმაყოფილებს (1.1) განტოლებას.

ცხადია, პრაქტიკულად უსასრულო სისტემის ამოხსნა (გარკვეული კერძო შემთხვევის გარდა) შეუძლებელია. ამიტომ, ვიხილავთ მოჭრილ სისტემას, (1.4) უსასრულო სისტემის პირველ N განტოლებას პირველი N უცნობით:

$$(A\varphi_1, \varphi_1)\, c_1 + (A\varphi_2, \varphi_1)\, c_2 + \ldots + (A\varphi_N, \varphi_1)\, c_N = (f, \varphi_1)$$

$$
\begin{aligned}
&(A\varphi_1,\varphi_2)\,c_1 + (A\varphi_2,\varphi_2)\,c_2 + \ldots + (A\varphi_N,\varphi_2)\,c_N = (f,\varphi_2)\\
&(A\varphi_1,\varphi_3)\,c_1 + (A\varphi_2,\varphi_3)\,c_2 + \ldots + (A\varphi_N,\varphi_3)\,c_N = (f,\varphi_3)\\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
&(A\varphi_1,\varphi_N)\,c_1 + (A\varphi_2,\varphi_N)\,c_2 + \ldots + (A\varphi_N,\varphi_N)\,c_N = (f,\varphi_N)
\end{aligned}
\tag{1.5}
$$

რადგან (1.5) სისტემა განსხვავდება (1.4) სისტემისგან, ბუნებრივია მისი ამონახსნი აღვნიშნოთ $\tilde{c}_k$-ით. ამ კოეფიციენტების მიხედვით აგებული ვექტორი აღვნიშნოთ $\tilde{u}_N$-ით.

$$
\tilde{u}_N = \sum_{k=1}^{N} \tilde{c}_k \varphi_k
$$

სავსებით ბუნებრივია $\tilde{u}_N$ ვექტორი გამოვაცხადოთ (1.1) განტოლების მიახლოებით ამონახსნად.

ზევით აღწერილ მეთოდს უწოდებენ გალიორკინის მეთოდს.

ვიხილავთ კონკრეტულ ამოცანას, კერძოდ პუასონის განტოლებისათვის დირიხლეს ერთგვაროვანი სასაზღვრო ამოცანას მართკუთხოვანი არის და მუდმივი მარჯვენა მხარის შემთხვევაში:

$$
-\Delta u = q, \quad q = \text{const} > 0, \quad (x,y) \in \Omega = \,]0,1[\, \times \,]0,1[\tag{1.6}
$$

$$
u|_{\partial\Omega} = 0 \tag{1.7}
$$

(Ω არის ერთეულოვანი კვადრატი, $\partial\Omega$ არის Ω-ს საზღვარი). ჩვენს შემთხვევაში ოპერატორი $A = (-\Delta)$, განსაზღვრის არით:

$$
D\,(A) = \left\{ u\,(x,y) \in C^2\left(\bar{\Omega}\right) : \, u\,(x,y)|_{\partial\Omega} = 0 \right\}.
$$

(1.6), (1.7) ამოცანის ამონახსნი ვეძებოთ შემდეგი ორმაგი მწკრივის სახით:

$$
u\,(x,y) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} c_{m,n} \varphi_{m,n}\,(x,y) \tag{1.8}
$$

სადაც

$$
\varphi_{m,n}\,(x,y) = \frac{2}{\pi\sqrt{m^2+n^2}} \sin\left(m\pi x\right) \sin\left(n\pi y\right)
$$

ცხადია $\varphi_{m,n}\,(x,y)$ ფუნქციები შედის A ოპერატორის განსაზღვრის არეში. $c_{m,n}$ კოეფიციენტები უნდა ვიპოვნოთ შემდეგი პირობიდან:

$$
\left(-\sum_{i=1}^{\infty} \sum_{j=1}^{\infty} c_{i,j} \Delta\varphi_{i,j}\,(x,y) - q, \varphi_{m,n}\,(x,y) \right) = 0, \;\; m,n = 1,2,\ldots \tag{1.9}
$$

სადაც სკალარული ნამრავლი გვესმის $L_2\,(\Omega)$ სივრცის აზრით. $L_2\,(\Omega)$ სივრცეში $u\,(x,y)$ და $v\,(x,y)$ ფუნქციების სკალარული ნამრავლი განიმარტება შემდეგნაირად:

$$
\left(u\,(x,y)\,,\,v\,(x,y)\right) = \iint\limits_{\Omega} u\,(x,y)\,v\,(x,y)\,dxdy
$$

შენიშვნა 1.1.1. სანამ უშუალოდ ამოცანაში შემავალი ფუნქიების სკალარულ ნამრავლებს გამოვთვლით, მანამდე ერთი კონკრეტული სკალარული ნამრავლი გამოვთვალოთ, რომელიც შემდგომში გამოგვადგება.

განვიხილოთ შემდეგი სკალარული ნამრავლი:

$$(\sin(i\pi x)\sin(j\pi y)\,,\,\sin(m\pi x)\sin(n\pi y)) = \int_0^1\int_0^1 \sin(i\pi x)\sin(j\pi y)\sin(m\pi x)\sin(n\pi y)\,dxdy =$$

$$= \int_0^1 \sin(i\pi x)\sin(m\pi x)\,dx \int_0^1 \sin(j\pi y)\sin(n\pi y)\,dy$$

შემთხვევა, როდესაც $i = m$ და $j = n$, ცხადია

$$\int_0^1 \sin^2(m\pi x)\,dx = \frac{1}{2}\,,$$

მივიღეთ,

$$\int_0^1 \sin^2(m\pi x)\,dx \int_0^1 \sin^2(n\pi y)\,dy = \frac{1}{4}\,. \tag{1.10}$$

შემთხვევა, როდესაც $i \neq m$ ან $j \neq n$. გვაქვს,

$$\int_0^1 \sin(i\pi x)\sin(m\pi x)\,dx = 0\,,$$

ამ შემთხვევაში,

$$\int_0^1 \sin(i\pi x)\sin(m\pi x)\,dx \int_0^1 \sin(j\pi y)\sin(n\pi y)\,dy = 0\,, \tag{1.11}$$

საბოლოოდ ჩვენ გვაქვს:

$$(\sin(i\pi x)\sin(j\pi y)\,,\,\sin(m\pi x)\sin(n\pi y)) = \begin{cases} \frac{1}{4}, & \text{თუ } i = m \text{ და } j = n \\ 0, & \text{თუ } i \neq m \text{ ან } j \neq n \end{cases} \tag{1.12}$$

(1.12) ტოლობა შეგვიძლია გადავწეროთ უფრო კომპაქტურად კრონეკერის სიმბოლოს გამოყენებით:

კომენტარი 1.1.1 (კრონეკერის სიმბოლო)**.**

$$\delta_{ks} = \begin{cases} 1, & \text{თუ } k = s, \\ 0, & \text{თუ } k \neq s. \end{cases}$$

$$(\sin(i\pi x)\sin(j\pi y)\,,\,\sin(m\pi x)\sin(n\pi y)) = \frac{1}{4}\delta_{im}\delta_{jn} \tag{1.13}$$

(1.13)-დან გამომდინარეობს, რომ:

$$\left(\varphi_{i,j}\left(x,y\right),\,\varphi_{m,n}\left(x,y\right)\right)=\frac{1}{\pi^{2}\left(m^{2}+n^{2}\right)}\delta_{im}\delta_{jn}\,. \tag{1.14}$$

ცხადია

$$-\Delta\varphi_{i,j}\left(x,y\right)=\pi^{2}\left(i^{2}+j^{2}\right)\varphi_{i,j}\left(x,y\right) \tag{1.15}$$

(1.15)-დან და (1.14)-დან გამომდინარეობს, რომ:

$$\left(-\Delta\varphi_{i,j}\left(x,y\right),\,\varphi_{m,n}\left(x,y\right)\right)=\delta_{im}\delta_{jn} \tag{1.16}$$

ახლა გამოვითვალოთ შემდეგი სკალარული ნამრავლი, მივიღეთ:

$$\left(-\sum_{i=1}^{\infty}\sum_{j=1}^{\infty}c_{i,j}\Delta\varphi_{i,j}\left(x,y\right),\,\varphi_{m,n}\left(x,y\right)\right)=c_{m,n}\,. \tag{1.17}$$

გამოვთვალოთ სკალარული ნამრავლი: $\left(q,\,\varphi_{m,n}\left(x,y\right)\right)$.

$$\left(q,\varphi_{m,n}\left(x,y\right)\right)=\frac{2q}{\pi^{3}mn\sqrt{m^{2}+n^{2}}}\left(1-(-1)^{m}\right)\left(1-(-1)^{n}\right)\,.$$

აქედან გამომდინარეობს, რომ:

$$\left(q,\varphi_{m,n}\left(x,y\right)\right)=\begin{cases}\frac{8q}{\pi^{3}mn\sqrt{m^{2}+n^{2}}}, & \text{როდესაც } m \text{ და } n \text{ კენტია}\\ 0, & \text{როდესაც } m \text{ ან } n \text{ ლუწია}\end{cases} \tag{1.18}$$

(1.17) და (1.18) გათვალისწინებით (1.9)-დან მიიღება:

$$c_{m,n}=\frac{8q}{\pi^{3}mn\sqrt{m^{2}+n^{2}}} \tag{1.19}$$

სადაც m და n ორივე კენტია თუ ჩავსვამთ (1.19)-ს (1.8)-ში, მივიღებთ:

$$u\left(x,y\right)=\frac{16q}{\pi^{4}}\sum_{m=1,3,\ldots}\sum_{n=1,3,\ldots}\frac{1}{mn\left(m^{2}+n^{2}\right)}\sin\left(m\pi x\right)\sin\left(n\pi y\right)$$

1.2 ვარიაციულ-სხვაობიანი სქემა კირხოფის განტოლებისათვის

1.2.1 ლოკალურად წრფივი, სამშრიანი ნახევრადდისკრეტული სქემა და მიახლოებითი ამონახსნის აგება ტრიგონომეტრიული საბაზისო ფუნქციების გამოყენებით

განვიხილოთ განტოლება:

$$\frac{\partial^2 u(x,t)}{\partial t^2} - \left(\alpha + \beta \int\limits_0^1 \left[\frac{\partial u(x,t)}{\partial x}\right]^2 dx\right) \frac{\partial^2 u(x,t)}{\partial x^2} = f(x,t) \ , \quad (x,t) \in]0,1[\times]0,T] \ , \quad (1.20)$$

სადაც $\alpha > 0$ და $\beta > 0$. მოცემული განტოლებისათვის გვაქვს შემდეგი საწყის–სასაზღვრო პირობები:

$$u(x,0) = \psi_0(x) \ , \quad u'_t(x,0) = \psi_1(x) \ . \quad (1.21)$$

$$u(0,t) = 0 \, , \quad u(1,t) = 0 \, . \quad (1.22)$$

ამასთან სრულდება შეთანხმებულობის პირობა:

$$\psi_0(0) = 0 \, , \quad \psi_0(1) = 0 \, .$$

$[0,T]$ სეგმენტი დავყოთ M თანაბარ ნაწილად, τ თანაბარი ბიჯით:

$$0 = t_0 < t_1 < \cdots < t_M = T \, ,$$

სადაც

$$t_k = k\tau, \quad (k = 0, 1, \ldots, M) \, , \quad \tau = \frac{T}{M} \, .$$

(1.20) – (1.22) ამოცანის მიახლოებით ამონახსნს ვეძებთ შემდეგი ნახევრადდისკრეტული სქემის გამოყენებით:

$$\frac{u_{k+1}(x) - 2u_k(x) + u_{k-1}(x)}{\tau^2} - \frac{1}{2} q_k \left(\frac{d^2 u_{k+1}(x)}{dx^2} + \frac{d^2 u_{k-1}(x)}{dx^2}\right) = f_k(x) \ , \quad (1.23)$$

სადაც

$$q_k = \alpha + \beta \int\limits_0^1 \left(\frac{du_k(x)}{dx}\right)^2 dx \, , \quad (k = 1, 2, \ldots, M-1) \ .$$

(1.23)-დან მივიღებთ:

$$2u_{k+1}(x) - \tau^2 q_k \frac{d^2 u_{k+1}(x)}{dx^2} = 2\tau^2 f_k(x) + 4u_k(x) + \tau^2 q_k \frac{d^2 u_{k-1}(x)}{dx^2} - 2u_{k-1}(x) \ ,$$

თუ შემოვიღებთ შემდეგ აღნიშვნას

$$g_k(x) = 2\tau^2 f_k(x) + 4u_k(x) + \tau^2 q_k \frac{d^2 u_{k-1}(x)}{dx^2} - 2u_{k-1}(x) \ ,$$

მაშინ მივიღებთ შემდეგ განტოლებას:

$$\left(2I - \tau^2 q_k \frac{d^2}{dx^2}\right) u_{k+1}(x) = g_k(x) \ . \quad (1.24)$$

საძებნი ფუნქციის მნიშვნელობები ნულოვან და პირველ შრეზე განისაზღვრება საწყისი პირო-

ბებისა და (1.20) განტოლების საშუალებით:

$$u_0(x) = \psi_0(x) \ , \tag{1.25}$$

$$u_1(x) = \psi_0(x) + \tau\psi_1(x) + \frac{1}{2}\tau^2\left(q_0\frac{d^2\psi_0(x)}{dx^2} + f_0(x)\right) . \tag{1.26}$$

სასაზღვრო პირობები:

$$u_k(0) = 0\,, \quad u_k(1) = 0\,. \tag{1.27}$$

(1.24) – (1.27) ამოცანის ამონახსნი ვეძებოთ შემდეგი სახით:

$$u_k(x) = \sum_{m=1}^{N} c_m^k \varphi_m(x)\,, \tag{1.28}$$

სადაც,

$$\varphi_m(x) = \sin(m\pi x)\,, \quad (m = 1, 2, \ldots, N) \ . \tag{1.29}$$

ყოველ $(k+1)$-ე შრეზე c_m^{k+1} $(m = 1, 2, \ldots, N \quad$ და $\quad k = 1, 2, \ldots, M-1)$ კოეფიციენტები უნდა ვიპოვნოთ შემდეგი პირობიდან:

$$\left(\left(2I - \tau^2 q_k\frac{d^2}{dx^2}\right)u_{k+1}(x) - g_k(x)\,, \varphi_m(x)\right) = 0\,, \tag{1.30}$$

(1.28) ჩავსვათ (1.30), მივიღებთ:

$$\left(\sum_{i=1}^{N} c_i^{k+1}\left(2I - \tau^2 q_k\frac{d^2}{dx^2}\right)\varphi_i(x) - g_k(x)\,, \varphi_m(x)\right) = 0\,,$$

საბოლოოდ,

$$\left(\sum_{i=1}^{N} c_i^{k+1}\left(2I - \tau^2 q_k\frac{d^2}{dx^2}\right)\varphi_i(x), \varphi_m(x)\right) = \bigg(g_k(x)\,, \varphi_m(x)\bigg)\,, \tag{1.31}$$

გამოვთვალოთ სკალარული ნამრავლი:

$$(\varphi_i(x)\,, \varphi_m(x)) = (\sin(i\pi x)\,, \sin(m\pi x)) = \int_0^1 \sin(i\pi x)\sin(m\pi x)\,dx\,,$$

(1.10)-ის და (1.11)-ის გათვალისწინებით გვაქვს:

$$(\varphi_i(x)\,, \varphi_m(x)) = \frac{1}{2}\delta_{im}\,. \tag{1.32}$$

ნაწილობითი ინტეგრებითა და (1.27) სასაზღვრო პირობების გათვალისწინებით მივიღებთ,

$$\int_0^1 \left(\frac{du_k(x)}{dx}\right)^2 dx = -\int_0^1 \frac{d^2u_k(x)}{dx^2}u_k(x)\,dx\,. \tag{1.33}$$

(1.29)-ის თანახმად:

$$-\frac{d^2\varphi_i(x)}{dx^2} = i^2\pi^2\varphi_i(x)\ . \tag{1.34}$$

(1.28), (1.33) და (1.34) ტოლობების გათვალისწინებით,

$$q_k = \alpha + \frac{\beta}{2}\pi^2 \sum_{m=1}^{N} \left(mc_m^k\right)^2 . \tag{1.35}$$

გამოვთვალოთ:

$$\left(2I - \tau^2 q_k \frac{d^2}{dx^2}\right)\varphi_i(x) = \left(2 + i^2\pi^2\tau^2 q_k\right)\varphi_i(x)\ . \tag{1.36}$$

(1.32)-ის და (1.36)-ის გათვალისწინებით მივიღებთ

$$\left(\sum_{i=1}^{N} c_i^{k+1}\left(2I - \tau^2 q_k \frac{d^2}{dx^2}\right)\varphi_i(x), \varphi_m(x)\right) = \frac{2 + m^2\pi^2\tau^2 q_k}{2} c_m^{k+1}\,. \tag{1.37}$$

მარჯვენა მხარეში შემავალი ინტეგრალების გამოთვლა კუბური სპლაინ აპროქსიმაციით

ამ ნაწილში კუბური სპლაინის აგება არ იქნება გადმოცემული, ასევე საჭირო დებულებები და თეორემები იქნება მოცემული დამტკიცების გარეშე. აღნიშნულ თემასთან დაკავშირებით უფრო დაწვრილებით შეგიძლიათ იხილოთ შემდეგი ლიტერატურა (იხ. [1] და [31]).

ავაგოთ $y = f(x)$ ფუნქციის მიახლოებითი მრუდი, რომელიც წარმოადგენს კუბური პოლინომებისგან შემდგარ მაინტერპოლებელ ტეხილს.

ჩავთვალოთ, რომ $y = f(x)$ განსაზღვრულია $a \le x \le b$ შუალედში. დავყოთ $[a, b]$ შუალედი n ნაწილად:

$$\Delta: \quad a = x_0 < x_1 < \cdots < x_n = b\,,$$
$$h_j = x_j - x_{j-1}\ (j = 1, 2, \ldots, n)\ .$$

ამასთან ერთად ვიგულისხმოდ, რომ ცნობილია ასევე შესაბამისი ორდინატები:

$$Y: \quad y_0, y_1, y_2, \ldots, y_{n-2}, y_{n-1}, y_n.$$

საძიებელია ფუნქცია $S_\Delta(x)$, რომელიც უწყვეტია $[a, b]$ შუალედში თავის პირველი და მეორე რიგის წარმოებულებთან ერთად და ყოველი $x_{j-1} \le x \le x_j$ $(j = 1, 2, \ldots, n)$ მონაკვეთზე ემთხვევა მესამე რიგის პოლინომებს. ამასთან ვთვლით, რომ სრულდება ინტერპოლების პირობა:

$$S_\Delta(x_j) = y_j\,, \quad (j = 0, 1, 2, \ldots, n)\ .$$

$S_\Delta(x)$-ს უწოდებენ მაინტერპოლებელ სპლაინს Δ ბადის მიმართ. სპლაინს ეწოდება პერიოდული (პერიოდით $b - a$), თუ:

$$S_\Delta^{(p)}(a+) = S_\Delta^{(p)}(b-)\ , \quad (p = 0, 1, 2)\ .$$

შემოვიღოთ შემდეგი სიდიდეები:

$$S''_{\Delta}(x_j) = M_j \ (j = 0, 1, 2, \ldots, n) \ .$$

$M_j \ (j = 0, 1, 2, \ldots, n)$ სიდიდეებს ეწოდებათ "მომენტები". ბუნებრივია ისინი ჩვეულებრივი აზრით არ წარმოადგენენ ძელის მომენტებს.
ყოველ $(x_{j-1} \leq x \leq x_j)$ შუალედზე $S''_{\Delta}(x)$ იქნება წრფივი ფუნქცია. ამიტომ

$$S''_{\Delta}(x_j) = M_{j-1}\frac{x_j - x}{h_j} + M_j\frac{x - x_{j-1}}{h_j} \, . \tag{1.38}$$

ცხადია (1.38) წარმოადგენს ორ კვანძიან ლაგრანჟის საინტერპოლაციო ფორმულას.
თუ (1.38) ტოლობას ვაინტეგრებთ ორჯერ და ინტეგრების შედეგად გაჩენილ ორ უცნობ მუდმივს ვიპოვნით ინტერპოლების პირობიდან x_{j-1} და $x_j \ (j = 1, 2, \ldots, n)$ კვანძებში, მაშინ ჩვენ მივიღებთ:

$$\begin{aligned} S_{\Delta}(x) = M_{j-1}\frac{(x_j - x)^3}{6h_j} + M_j\frac{(x - x_{j-1})^3}{6h_j} + \left(y_{j-1} - \frac{M_{j-1}h_j^2}{6}\right)\frac{x_j - x}{h_j} + \\ + \left(y_j - \frac{M_j h_j^2}{6}\right)\frac{x - x_{i-1}}{h_j}\, , \quad x \in [x_{j-1}, \, x_j] \ . \end{aligned} \tag{1.39}$$

(1.39) ფორმულით მოცემულ მესამე ხარისხის პოლინომს ეწოდება კუბური სპლაინი. ახლა მიღებული კუბური პოლინომი მივიყვანოთ სტანდარტულ ფორმაზე. ჩვენ შეგვიძლია (1.39) ტოლობა გავამარტივოთ, მსგავსი წევრები შევაერთოთ და მივიღოთ მესამე ხარისხის პოლინომი სტანდარტული ფორმით.

შემოგთავაზებთ განსხვავებულ გზას. (1.39) და სტანდარტული ფორმით ჩაწერილი კუბური პოლინომი $\tilde{S}_{\Delta}(x)$ გავაწარმოოთ მესამე რიგამდე ჩათვლით. განვიხილოთ შემდეგი ამოცანა:

$$S_{\Delta}^{(p)}(0) = \tilde{S}_{\Delta}^{(p)}(0) \ , \quad (p = 3, 2, 1, 0) \ .$$

ამრიგად, (1.39) მიიღებს შემდეგ სახეს:

$$S_{\Delta}(x) = a_j x^3 + b_j x^2 + c_j x + d_j, \quad (a_j \neq 0)\, , \quad x \in [x_{j-1}, x_j]\, , \quad (j = 1, 2, \ldots, n)\, , \tag{1.40}$$

სადაც:

$$a_j = \frac{1}{6h_j}\left(M_j - M_{j-1}\right) \, , \tag{1.40a}$$

$$b_j = \frac{1}{2}M_j - 3x_j a_j \, , \tag{1.40b}$$

$$c_j = -h_j^2 a_j - x_j b_j - \frac{1}{2}x_{j-1}M_j + \frac{1}{h_j}\left(y_j - y_{j-1}\right) \, , \tag{1.40c}$$

$$d_j = -x_j\left(x_j^2 + h_j^2\right)a_j - \left(x_j^2 + \frac{1}{3}h_j^2\right)b_j - x_j c_j + \frac{1}{6}h_j^2 M_j + y_j \, , \tag{1.40d}$$

(1.39) და (1.40) ტოლობაში უცნობებს წარმოადგენენ M_0, M_1, $\ldots$, M_n სიდიდეები. ამ სიდი-

დეების საპოვნელად განვიხილოთ შემდეგი ტოლობა $S'_{\Delta}(x_j-) = S'_{\Delta}(x_j+)$, აქედან ვიღებთ შემდეგ წრფივ ალგებრულ განტოლებათა სისტემას:

$$\mu_j M_{j-1} + 2M_j + \lambda_j M_{j+1} = d_j \quad (j = 1, 2, \ldots, n-1) \tag{1.41}$$

სადაც:

$$\lambda_j = \frac{h_{j+1}}{h_j + h_{j+1}}, \ \mu_j = \frac{h_j}{h_j + h_{j+1}} = 1 - \lambda_j, \ d_j = 6\frac{\frac{y_{j+1}-y_j}{h_{j+1}} - \frac{y_j - y_{j-1}}{h_j}}{h_j + h_{j+1}} \quad (j = 1, 2, \ldots, n-1)$$

არაპერიოდული შემთხვევა

(1.41) განტოლებათა სისტემა წარმოადგენს $(n-1)$ განტოლებას $(n+1)$ უცნობით. აღნიშნული განტოლებათა სისტემის ამოსახსნელად საჭიროა M_0 და M_n სიდიდეების ცოდნა.

განვიხილოთ შემდეგი ტოლობები:

$$2M_0 + \lambda_0 M_1 = d_0\,, \quad \mu_n M_{n-1} + 2M_n = d_n\,. \tag{1.42}$$

(1.42) პირობებიდან მიიღება ორი შემთხვევა:

(a) თუ $\lambda_0 = \mu_n = 0,$ მაშინ $d_0 = 2y''_0$ და $d_n = 2y''_n$

(b) თუ $\lambda_0 = \mu_n = 1,$ მაშინ $d_0 = \frac{6}{h_1}\left(\frac{y_1 - y_0}{h_1} - y'_0\right)$ და $d_n = \frac{6}{h_n}\left(y'_n - \frac{y_n - y_{n-1}}{h_n}\right)$

შენიშვნა 1.2.1. შეგვიძლია განვიხილოთ შერეული პირობაც.

ამგვარად, (1.42) და (1.41) განტოლებები, რომლებიც წარმოადგენენ $(n+1)$ განტოლებას, $(n+1)$ უცნობით.

პერიოდული შემთხვევა

პერიოდულ შემთხვევაში გვაქვს შემდეგი პირობები:

$$y_n = y_0,\, M_n = M_0,\, y_{n+1} = y_1,\, M_{n+1} = M_1,\, h_{n+1} = h_1$$

ამ პირობის გათვალისწინებით, ჩვენ მივიღებთ:

$$M_0 = M_n,\, \lambda_n = \frac{h_1}{h_n + h_1},\, \mu_n = \frac{h_n}{h_n + h_1}$$

ამ შემთხვევაში მიიღება n უცნობიანი განტოლებათა სისტემა n განტოლებით.

სასაზღვრო პირობები M_j სიდიდეებისათვის კლასიფიცირდება შემდეგნაირად:

$$
\begin{aligned}
& && \text{როდესაც} \quad x=a \\
&(i) && 2M_0+M_1=\frac{6}{h_1}\left(\frac{y_1-y_0}{h_1}-{y'}_0\right), \\
&(ii) && 2M_0=2{y''}_0\,, \\
&(iii) && 2M_0+\lambda_0 M_1=d_0\,, \\
& && \text{როდესაც} \quad x=b \\
&(i) && M_{n-1}+2M_n=\frac{6}{h_n}\left({y'}_n-\frac{y_n-y_{n-1}}{h_n}\right), \\
&(ii) && 2M_n=2{y''}_n\,, \\
&(iii) && \mu_n M_{n-1}+2M_n=d_n\,.
\end{aligned} \tag{1.43}
$$

თეორემა 1.2.1 (ფუნქციის აპროქსიმაციის შესახებ კუბური სპლაინების გამოყენებით). *ვთქვათ $f(x)\in C^m([a,b])$, (სადაც $m=2,3$) და $\{\Delta_k\}$ არის ბადეების მიმდევრობა $[a,b]$-ზე, შემდეგი პირობით $\lim\limits_{k\to\infty}\|\Delta_k\|=0$. მაშინ $S_{\Delta_k}(x)$ მაინტერპოლებელი სპლაინისათვის, რომელიც აკმაყოფილებს (1.43) (i) ან (ii) სასაზღვრო პირობებს, ან პერიოდულია (თუ $f(x)$ პერიოდულია), ჩვენ გვაქვს შეფასება:*

$$\left[f^{(p)}(x)-S_{\Delta_k}^{(p)}(x)\right]=\mathcal{O}\left(\|\Delta_k\|^{m-p}\right) \quad \left(p=\overline{0,m}\right) \tag{1.44}$$

თანაბრად $x\in[a,b]$-ს მიმართ.
თუ $f^{(m)}(x)$ აკმაყოფილებს ჰოლდერის პირობას $[a,b]$-ზე, α $(0<\alpha\le 1)$ რიგით, მაშინ:

$$\left[f^{(p)}(x)-S_{\Delta_k}^{(p)}(x)\right]=\mathcal{O}\left(\|\Delta_k\|^{m+\alpha-p}\right) \quad \left(p=\overline{0,m}\right) \tag{1.45}$$

თანაბრად $x\in[a,b]$-ს მიმართ.

ახლა დავუბრუნდეთ ჩვენ ამოცანას. დავუშვათ $f_k(x)$, $\psi_0(x)$ და $\psi_1(x)\in C^3([0,1])$ ჩვენი მიზანია ამ ფუნქციებს მივუახლოვდეთ კუბური სპლაინით.

$[0,1]$ სეგმენტი დავყოთ n ტოლ ნაწილად h ბიჯით.

$$0=x_0<x_1<\cdots<x_n=1\,,$$

სადაც

$$x_j=jh, \quad (j=0,1,\ldots,n)\,, \quad h=\frac{1}{n}\,.$$

გამოვთვალოთ $f_k(x)$, $\psi_0(x)$ და $\psi_1(x)$ ფუნქციების მნიშვნელობა x_j $(j=0,1,\ldots,n)$ წერტილებში:

$$f_k(x_j)=y_j^k\,, \quad \psi_0(x_j)=\tilde{y}_j^0\,, \quad \psi_1(x_j)=\tilde{y}_j^1\,.$$

საძიებელი კუბური სპლაინები $f_k(x)$, $\psi_0(x)$ და $\psi_1(x)$ ფუნქციებისათვის აღვნიშნოთ შესაბამისად: $S_j^k(x)$-ით, $\tilde{S}_j^0(x)$-ით და $\tilde{S}_j^1(x)$-ით.

ე. წ. "მომენტებისათვის" შემოვიღოთ შემდეგი აღნიშვნები:

$$\left(S_j^k\left(x\right)\right)'' = M_j^k\,, \quad \left(\tilde{S}_j^0\left(x\right)\right)'' = \tilde{M}_j^0\,, \quad \left(\tilde{S}_j^1\left(x\right)\right)'' = \tilde{M}_j^1\,.$$

თითოეულ შემთხვევაში ყოველ $[x_{j-1}, x_j]$ $(j = 1, 2, \ldots, n)$ მონაკვეთში საძიებელ კუბურ სპლაინს ექნება შემდეგი სახე:

$$S_j^k\left(x\right) = A_j^k x^3 + B_j^k x^2 + C_j^k x + D_j^k\,, \quad \left(A_j^k \neq 0\right), \tag{1.46}$$

სადაც:

$$A_j^k = \frac{1}{6h}\left(M_j^k - M_{j-1}^k\right)\,, \tag{1.46a}$$

$$B_j^k = \frac{1}{2}M_j^k - 3x_j A_j^k\,, \tag{1.46b}$$

$$C_j^k = -h^2 A_j^k - x_j B_j^k - \frac{1}{2}x_{j-1}M_j^k + \frac{1}{h}\left(y_j^k - y_{j-1}^k\right)\,, \tag{1.46c}$$

$$D_j^k = -x_j\left(x_j^2 + h^2\right)A_j^k - \left(x_j^2 + \frac{1}{3}h^2\right)B_j^k - x_j C_j^k + \frac{1}{6}h^2 M_j^k + y_j^k\,, \tag{1.46d}$$

ანალოგიურად ფორმულებით გამოითვლებიან $\tilde{S}_j^0\left(x\right)$ და $\tilde{S}_j^1\left(x\right)$ სპლაინები. ახლა დავწეროთ თითოეული კუბური სპლაინის ასაგებად საჭირო განტოლებათა სისტემა.

$$\mu_j M_{j-1}^k + 2M_j^k + \lambda_j M_{j+1}^k = d_j^k\,, \quad (j = 1, 2, \ldots, n-1)\,, \tag{1.47}$$

$$\mu_j \tilde{M}_{j-1}^0 + 2\tilde{M}_j^0 + \lambda_j \tilde{M}_{j+1}^0 = \tilde{d}_j^0\,, \quad (j = 1, 2, \ldots, n-1)\,, \tag{1.48}$$

$$\mu_j \tilde{M}_{j-1}^1 + 2\tilde{M}_j^1 + \lambda_j \tilde{M}_{j+1}^1 = \tilde{d}_j^1\,, \quad (j = 1, 2, \ldots, n-1)\,. \tag{1.49}$$

შევნიშნოთ, რომ რადგან $[0, 1]$ მონაკვეთი დავყავით n ტოლ ნაწილად (h თანაბარ ბიჯად), ამიტომ სამივე ((1.47), (1.48) და (1.49)) წრფივ ალგებრულ განტოლებათა სისტემისათვის:

$$\lambda_j = \mu_j = \frac{1}{2}\,, \quad (j = 1, 2, \ldots, n)\,.$$

შენიშვნა 1.2.2. ჩვენ ვიხილავთ (1.42) სასაზღვრო პირობის, (a) შემთხვევას.

თითოეული კუბური სპლაინისათვის (1.42) სასაზღვრო პირობა ჩაიწერება შემდეგი სახით:

$$2M_0^k + \lambda_0 M_1^k = d_0^k\,, \quad \mu_n M_{n-1}^k + 2M_n^k = d_n^k\,, \tag{1.50}$$

$$2\tilde{M}_0^0 + \lambda_0 \tilde{M}_1^0 = \tilde{d}_0^0\,, \quad \mu_n \tilde{M}_{n-1}^0 + 2\tilde{M}_n^0 = \tilde{d}_n^0\,, \tag{1.51}$$

$$2\tilde{M}_0^1 + \lambda_0 \tilde{M}_1^1 = \tilde{d}_0^1\,, \quad \mu_n \tilde{M}_{n-1}^1 + 2\tilde{M}_n^1 = \tilde{d}_n^1\,. \tag{1.52}$$

შენიშვნა 1.2.2.-ის თანახმად $\lambda_0 = \mu_n = 0$, მაშინ:

$$d_0^k = 2\left(y_0^k\right)''\,, \quad d_n^k = 2\left(y_n^k\right)'' \tag{1.53}$$

$$\tilde{d}_0^0 = 2\left(\tilde{y}_0^0\right)''\,, \quad \tilde{d}_n^0 = 2\left(\tilde{y}_n^0\right)'' \tag{1.54}$$

$$\tilde{d}_0^1 = 2\left(\tilde{y}_0^1\right)'', \quad \tilde{d}_n^1 = 2\left(\tilde{y}_n^1\right)'' \tag{1.55}$$

(1.47), (1.48) და (1.49) წრფივ ალგებრულ განტოლებათა სისტემების მარჯვენა მხარეებში მდგომი სიდიდეები განისაზღვრება შემდეგნაირად $(j = 1, 2, \ldots, n-1)$:

$$d_j^k = 3\frac{y_{j+1}^k - 2y_j^k + y_{j-1}^k}{h^2} \tag{1.56}$$

$$\tilde{d}_j^0 = 3\frac{\tilde{y}_{j+1}^0 - 2\tilde{y}_j^0 + \tilde{y}_{j-1}^0}{h^2} \tag{1.57}$$

$$\tilde{d}_j^1 = 3\frac{\tilde{y}_{j+1}^1 - 2\tilde{y}_j^1 + \tilde{y}_{j-1}^1}{h^2} \tag{1.58}$$

შენიშვნა 1.2.3. $\psi_0(x)$ და $\psi_1(x)$ ფუნქციებისათვის შესაბამისად $\tilde{S}_j^0(x)$ და $\tilde{S}_j^1(x)$ მაინტერპოლებელი კუბური სპლაინების აგება დაგვჭირდება ერთჯერადად, ხოლო $f_k(x)$ ფუნქციისათვის $S_j^k(x)$ მაინტერპოლებელი კუბური სპლაინის აგება მოგვიწევს ყოველი $k = 0, 1, 2, \ldots, M-1$ შრისათვის.

შემოვიღოთ შემდეგი აღნიშვნები:

$$I_{j,m}^s = \int\limits_{x_{j-1}}^{x_j} x^s \sin(m\pi x)dx\,, \quad (s = 0, 1, 2, 3)\ .$$

$$I_{j,m}^0 = \frac{2}{m\pi}\sin\left(m\left(x_{j-1} + \frac{h}{2}\right)\pi\right)\sin\left(\frac{mh}{2}\pi\right)\,, \tag{1.59}$$

$$\begin{aligned} &I_{j,m}^1 = \\ &= x_{j-1}I_{j,m}^0 + \frac{1}{m\pi}\left(\frac{2}{m\pi}\sin\left(\frac{mh}{2}\pi\right)\cos\left(m\left(x_{j-1} + \frac{h}{2}\right)\pi\right) - h\cos(m\pi x_j)\right)\,, \end{aligned} \tag{1.60}$$

$$\begin{aligned} &I_{j,m}^2 = \left(x_{j-1}^2 - \frac{2}{m^2\pi^2}\right)I_{j,m}^0 + \\ &+ \frac{2}{m^2\pi^2}\left(2x_{j-1}\sin\left(\frac{mh}{2}\pi\right)\cos\left(m\left(x_{j-1} + \frac{h}{2}\right)\pi\right) + h\sin(m\pi x_j)\right) - \\ &- \frac{x_j^2 - x_{j-1}^2}{m\pi}\cos(m\pi x_j)\,, \end{aligned} \tag{1.61}$$

$$\begin{aligned} &I_{j,m}^3 = \\ &= x_{j-1}^3 I_{j,m}^0 + \frac{1}{m\pi}\left(\frac{3}{m\pi}\left(x_j^2 - x_{j-1}^2\right)\sin(m\pi x_j) - \left(x_j^3 - x_{j-1}^3\right)\cos(m\pi x_j)\right) + \\ &+ \frac{6}{m^2\pi^2}\left(x_{j-1}^2\sin\left(\frac{mh}{2}\pi\right)\cos\left(m\left(x_{j-1} + \frac{h}{2}\right)\pi\right) - I_{j,m}^1\right)\,. \end{aligned} \tag{1.62}$$

ახლა გამოვთვალოთ (1.31) განტოლების, მარჯვენა მხარეს მდგომი სკალარული ნამრავლი:

$$\begin{aligned} &(g_k(x), \varphi_m(x)) = 2\tau^2(f_k(x), \varphi_m(x)) + 4(u_k(x), \varphi_m(x)) - \\ &- \left(\left(2I - \tau^2 q_k\frac{d^2}{dx^2}\right)u_{k-1}(x), \varphi_m(x)\right)\,. \end{aligned} \tag{1.63}$$

შემოვიღოთ შემდგეგი აღნიშვნა $I_m^k = (f_k(x), \varphi_m(x))$,

$$\begin{aligned} I_m^k &= \int_0^1 f_k(x) \sin(m\pi x)\, dx = \sum_{j=1}^{n} \int_{x_{j-1}}^{x_j} S_j^k(x) \sin(m\pi x)\, dx = \\ &= \sum_{j=1}^{n} A_j^k I_{j,m}^3 + \sum_{j=1}^{n} B_j^k I_{j,m}^2 + \sum_{j=1}^{n} C_j^k I_{j,m}^1 + \sum_{j=1}^{n} D_j^k I_{j,m}^0 , \end{aligned} \tag{1.64}$$

(1.64) ფორმულაში A_j^k, B_j^k, C_j^k და D_j^k ცნობილი სიდიდეებია, რომლებიც გამოითვლებიან (1.46a), (1.46b), (1.46c) და (1.46d) ტოლობებით. ასევე ცნობილი სიდიდეებია: $I_{j,m}^0$, $I_{j,m}^1$, $I_{j,m}^2$ და $I_{j,m}^3$, ისინი გამოითვლებიან შესაბამისად (1.59), (1.60), (1.61) და (1.62) ფორმულებით.

(1.32) გათვალისწინებით გამოვთვალოთ (1.63) ტოლობის მეორე შესაკრები:

$$(u_k(x), \varphi_m(x)) = \frac{c_m^k}{2} . \tag{1.65}$$

(1.63) ტოლობის ბოლო (მესამე) შესაკრებისათვის (1.36)-ის და (1.37)-ის გათვალისწინებით მივიღებთ:

$$\left(\left(2I - \tau^2 q_k \frac{d^2}{dx^2} \right) u_{k-1}(x), \varphi_m(x) \right) = \frac{2 + m^2\pi^2\tau^2 q_k}{2} c_m^{k-1} , \tag{1.66}$$

თუ (1.64)-ს, (1.65)-ს და (1.66)-ს ჩავსვამთ (1.63)-ში და გავუტოლებთ (1.37)-ს მივიღებთ:

$$\frac{2 + m^2\pi^2\tau^2 q_k}{2} c_m^{k+1} = 2\tau^2 I_m^k - \frac{2 + m^2\pi^2\tau^2 q_k}{2} c_m^{k-1} + 2c_m^k ,$$

საიდანაც:

$$c_m^{k+1} = \frac{4\tau^2}{2 + m^2\pi^2\tau^2 q_k} I_m^k - c_m^{k-1} + \frac{4}{2 + m^2\pi^2\tau^2 q_k} c_m^k , \quad (k = 1, 2, \ldots, M-1) . \tag{1.67}$$

გვაქვს ორი განსაკუთრებული შემთხვევა, ყოველი c_m^{k+1} $(k = 1, 2, \ldots, M-1)$ სიდიდეების საპოვნელად აუცილებელია, რომ ვიპოვნოთ შემდეგი ორი კოეფიციენტი: c_m^0 და c_m^1.
თუ $u_0(x)$ და $u_1(x)$ სკალარულად გავამრავლებთ $\varphi_m(x)$-ზე მივიღებთ:

$$(u_0(x), \varphi_m(x)) = (\psi_0(x), \varphi_m(x)) , \tag{1.68}$$

$$\begin{aligned} (u_1(x), \varphi_m(x)) &= (\psi_0(x), \varphi_m(x)) + \tau (\psi_1(x), \varphi_m(x)) + \\ &+ \frac{1}{2}\tau^2 q_0 \left(\frac{d^2\psi_0(x)}{dx^2}, \varphi_m(x) \right) + \frac{1}{2}\tau^2 (f_0(x), \varphi_m(x)) . \end{aligned} \tag{1.69}$$

თავდაპირველად გამოვთვალოთ შემდეგი ორი სიდიდე: $\tilde{I}_m^0 = (\psi_0(x), \varphi_m(x))$ და $\tilde{I}_m^1 =$

$(\psi_1(x), \varphi_m(x))$,

$$\tilde{I}_m^0 = \int_0^1 \psi_0(x)\sin(m\pi x)\,dx = \sum_{j=1}^{n}\int_{x_{j-1}}^{x_j} \tilde{S}_j^0(x)\sin(m\pi x)\,dx = \\ = \sum_{j=1}^{n}\tilde{A}_j^0 I_{j,m}^3 + \sum_{j=1}^{n}\tilde{B}_j^0 I_{j,m}^2 + \sum_{j=1}^{n}\tilde{C}_j^0 I_{j,m}^1 + \sum_{j=1}^{n}\tilde{D}_j^0 I_{j,m}^0\,, \tag{1.70}$$

$$\tilde{I}_m^1 = \int_0^1 \psi_1(x)\sin(m\pi x)\,dx = \sum_{j=1}^{n}\int_{x_{j-1}}^{x_j} \tilde{S}_j^1(x)\sin(m\pi x)\,dx = \\ = \sum_{j=1}^{n}\tilde{A}_j^1 I_{j,m}^3 + \sum_{j=1}^{n}\tilde{B}_j^1 I_{j,m}^2 + \sum_{j=1}^{n}\tilde{C}_j^1 I_{j,m}^1 + \sum_{j=1}^{n}\tilde{D}_j^1 I_{j,m}^0\,. \tag{1.71}$$

(1.28), (1.32), (1.68)-დან და (1.70)-დან ვღებულობთ, რომ:

$$c_m^0 = 2\tilde{I}_m^0\,, \tag{1.72}$$

მარტივად მიიღება შემდეგი ტოლობა

$$\left(\frac{d^2\psi_0(x)}{dx^2}, \varphi_m(x)\right) = -\frac{m^2\pi^2}{2}c_m^0\,, \tag{1.73}$$

(1.70), (1.71), (1.72), (1.73), (1.64), (1.28), (1.29), (1.32) და (1.34) ფორმულების გათვალისწინებით (1.69)–დან მივიღებთ,

$$c_m^1 = \left(1 - \frac{1}{2}m^2\pi^2\tau^2 q_0\right)c_m^0 + \tau\left(2\tilde{I}_m^1 + \tau I_m^0\right)\,. \tag{1.74}$$

საბოლოოდ ყოველი $(k = 1, 2, \ldots, M-1)$ შრისათვის (1.72), (1.74) და (1.67) კოეფიციენტები ჩავსვათ (1.28) ტოლობაში. ამრიგად მივიღეთ (1.20)–(1.22) მიახლოებითი ამონახსნი დროის ყოველი $(k = 2, 3, \ldots, M)$ შრისათვის.

1.2.2 ლოკალურად წრფივი, სამშრიანი ნახევრადდისკრეტული სქემა და მიახლოებითი ამონახსნის აგება საბაზისო ფუნქციებად ლეჟანდრის პოლინომების სხვაობების გამოყენებით

განვიხილოთ განტოლება:

$$\frac{\partial^2 u(x,t)}{\partial t^2} - \left(\alpha + \beta\int_{-1}^{1}\left[\frac{\partial u(x,t)}{\partial x}\right]^2 dx\right)\frac{\partial^2 u(x,t)}{\partial x^2} = \\ = f(x,t)\ , \quad (x,t) \in\,]-1,1[\, \times\,]0,T]\ , \tag{1.75}$$

სადაც $\alpha > 0$ და $\beta > 0$. მოცემული განტოლებისათვის გვაქვს შემდეგი საწყის–სასაზღვრო პირობები:

$$u(x,0) = \psi_0(x)\ , \quad {u'}_t(x,0) = \psi_1(x)\ . \tag{1.76}$$

$$u(-1,t)=0\,,\quad u(1,t)=0\,. \tag{1.77}$$

ამასთან სრულდება შეთანხმებულობის პირობა:

$$\psi_0(-1)=0\,,\quad \psi_0(1)=0\,.$$

$[0,T]$ სეგმენტი დავყოთ M თანაბარ ნაწილად, τ თანაბარი ბიჯით:

$$0=t_0<t_1<\cdots<t_M=T\,,$$

სადაც

$$t_k=k\tau,\quad (k=0,1,\ldots,M)\,,\quad \tau=\frac{T}{M}\,.$$

(1.75) – (1.77) ამოცანის მიახლოებით ამონახსნს ვეძებთ შემდეგი ნახევრადდისკრეტული სქემის გამოყენებით:

$$\left(2I-\tau^2 q_k\frac{d^2}{dx^2}\right)u_{k+1}(x)=g_k(x)\ , \tag{1.78}$$

სადაც:

$$q_k=\alpha+\beta\int\limits_0^1\left(\frac{du_k(x)}{dx}\right)^2dx\,,\quad (k=1,2,\ldots,M-1)\ ,$$

$$g_k(x)=2\tau^2 f_k(x)+4u_k(x)+\tau^2 q_k\frac{d^2u_{k-1}(x)}{dx^2}-2u_{k-1}(x)\ .$$

საძებნი ფუნქციის მნიშვნელობები ნულოვან და პირველ შრეზე განისაზღვრება საწყისი პირობებისა და (1.75) განტოლების საშუალებით:

$$u_0(x)=\psi_0(x)\ , \tag{1.79}$$

$$u_1(x)=\psi_0(x)+\tau\psi_1(x)+\frac{1}{2}\tau^2\left(q_0\frac{d^2\psi_0(x)}{dx^2}+f_0(x)\right)\ . \tag{1.80}$$

სასაზღვრო პირობები:

$$u_k(-1)=0\,,\quad u_k(1)=0\,. \tag{1.81}$$

(1.78) – (1.81) ამოცანის ამონახსნი ვეძებოთ შემდეგი სახით:

$$u_k(x)=\sum_{m=1}^N c_m^k\varphi_m(x)\,, \tag{1.82}$$

სადაც საკოორდინატო ფუნქციებად ვიღებთ:

$$\varphi_m(x)=A_m\left(P_{m+1}(x)-P_{m-1}(x)\right)\ ,\quad A_m=\frac{1}{\sqrt{2\,(2m+1)}}\,. \tag{1.83}$$

ყოველ $(k+1)$-ე შრეზე c_m^{k+1} $(m=1,2,\ldots,N\quad$ და $\quad k=1,2,\ldots,M-1)$ კოეფიციენტები უნდა

ვიპოვნოთ შემდეგი პირობიდან:

$$\left(\left(2I-\tau^2 q_k \frac{d^2}{dx^2}\right)u_{k+1}(x)-g_k(x)\,,\varphi_m(x)\right)=0\,, \tag{1.84}$$

(1.82) ჩავსვათ (1.84), მივიღებთ:

$$\left(\sum_{i=1}^{N} c_i^{k+1}\left(2I-\tau^2 q_k \frac{d^2}{dx^2}\right)\varphi_i(x)-g_k(x)\,,\varphi_m(x)\right)=0\,,$$

საბოლოოდ,

$$\left(\sum_{i=1}^{N} c_i^{k+1}\left(2I-\tau^2 q_k \frac{d^2}{dx^2}\right)\varphi_i(x),\varphi_m(x)\right)=\Big(g_k(x)\,,\varphi_m(x)\Big)\,, \tag{1.85}$$

მოვიყვანოთ ლეჟანდრის პოლინომების რამოდენიმე თვისება დამტკიცების გარეშე (იხ. [12], [11], და [30]):

$$(n+1)\,P_{n+1}(x)=(2n+1)\,xP_n(x)-nP_{n-1}(x)\;,\quad P_0(x)=1\,,\quad P_1(x)=x\,, \tag{1.86}$$

$$(2n+1)\,P_n(x)={P'}_{n+1}(x)-{P'}_{n-1}(x)\;, \tag{1.87}$$

$$\int_{-1}^{1} P_i(x)\,P_n(x)\,dx=\frac{2}{\sqrt{(2i+1)\,(2n+1)}}\delta_{in}\,, \tag{1.88}$$

$$\|P_n(x)\|=\sqrt{\int_{-1}^{1}(P_n(x))^2dx}=\sqrt{\frac{2}{2n+1}}\,, \tag{1.89}$$

$$P_n(1)=1\,, \tag{1.90}$$

$$P_n(-x)=(-1)^nP_n(x)\;,\quad (P_n(-1)=(-1)^n)\;, \tag{1.91}$$

შემოვიღოთ შემდეგი აღნიშვნა:

$$\tilde{P}_i(x)=\sqrt{\frac{2i+1}{2}}P_i(x)\;. \tag{1.92}$$

შევნიშნოთ, რომ (1.92) პოლინომთა სისტემა ორთონორმირებულია. თუ ვაინტეგრებთ $\tilde{P}_i(x)$ და გავითვალისწინებთ (1.87) და (1.91), მივიღებთ:

$$\int_{-1}^{x}\tilde{P}_i(t)\,dt=\frac{1}{\sqrt{2\,(2i+1)}}\left(P_{i+1}(x)-P_{i-1}(x)\right)\;.$$

შევნიშნოთ, რომ

$${\varphi'}_m(x)=\tilde{P}_m(x)\;. \tag{1.93}$$

ნაწილობითი ინტეგრებით, (1.88)-ისა და (1.93)-ის გათვალისწინებით მივიღებთ:

$$-\int_{-1}^{1} \frac{d^2\varphi_i(x)}{dx^2}\varphi_m(x)\,dx = \delta_{im}\,, \tag{1.94}$$

(1.88) ტოლობა ჩავწეროთ A_i-ის და A_m-ის საშუალებით:

$$\int_{-1}^{1} P_i(x)\,P_m(x)\,dx = 4A_iA_m\delta_{im}\,. \tag{1.88.a}$$

(1.88.a)-ს გათვალისწინებით გვექნება,

$$\begin{gathered}\int_{-1}^{1} \varphi_i(x)\,\varphi_m(x)\,dx = 4A_iA_m\left(A_{i+1}A_{m+1}\delta_{i+1,m+1} - A_{i+1}A_{m-1}\delta_{i+1,m-1} - \right. \\ \left. -A_{i-1}A_{m+1}\delta_{i-1,m+1} + A_{i-1}A_{m-1}\delta_{i-1,m-1}\right)\,.\end{gathered} \tag{1.95}$$

(1.33) და (1.94) ტოლობების გათვალისწინებით გვექნება:

$$q_k = \alpha + \beta\sum_{m=1}^{N}\left(c_m^k\right)^2\,. \tag{1.96}$$

(1.95) გათვალისწინებით გამოვთვალოთ შემდეგი სკალარული ნამრავლი:

$$\begin{aligned}&(u_{k+1}(x)\,,\varphi_m(x)) = \sum_{i=1}^{N} c_i^{k+1}\int_{-1}^{1}\varphi_i(x)\,\varphi_m(x)\,dx = \\ &= 4\left(-A_{m-2}A_{m-1}^2A_mc_{m-2}^{k+1} + A_m^2\left(A_{m-1}^2 + A_{m+1}^2\right)c_m^{k+1} - A_mA_{m+1}^2A_{m+2}c_{m+2}^{k+1}\right)\,,\end{aligned}$$

ვინაიდან ნულისაგან განსხვავებული წევრები მიიღება i ინდექსის შემდეგი მნიშვნელობებისათვის: $i = m-2$, $i = m$ და $i = m+2$. იგივური ოპერატორი გვაძლევს სამ კვანძს:

$$(m-2)\,,\quad (m)\quad \text{და}\quad (m+2)\,.$$

შემოვიღოთ შემდეგი აღნიშვნები:

$$B_m = 4A_{m-1}A_m^2A_{m+1}\,,\qquad B_m = \frac{1}{(2m+1)\sqrt{(2m-1)(2m+3)}}\,, \tag{1.97}$$

$$C_m = 4A_m^2\left(A_{m-1}^2 + A_{m+1}^2\right)\,,\qquad C_m = \frac{2}{(2m-1)(2m+3)}\,, \tag{1.98}$$

(1.97) და (1.98) აღნიშვნების თანახმად მივიღებთ:

$$(u_{k+1}(x)\,,\varphi_m(x)) = -B_{m-1}c_{m-2}^{k+1} + C_mc_m^{k+1} - B_{m+1}c_{m+2}^{k+1}\,. \tag{1.99}$$

(1.94) ტოლობის გათვალისწინებით გამოვთვალოთ სკალარული ნამრავლი:

$$-\left(\frac{d^2u_{k+1}(x)}{dx^2},\varphi_m(x)\right)=c_m^{k+1}\,, \tag{1.100}$$

ახლა გამოვთვალოთ (1.85) განტოლების მარცხენა მხარეს მდგომი სკალარული ნამრავლი (1.99) და (1.100) გათვალისწინებით, მივიღებთ:

$$\begin{gathered}\left(\sum_{i=1}^{N}c_i^{k+1}\left(2I-\tau^2q_k\frac{d^2}{dx^2}\right)\varphi_i(x),\varphi_m(x)\right)=\\=-2B_{m-1}c_{m-2}^{k+1}+\left(2C_m+\tau^2q_k\right)c_m^{k+1}-2B_{m+1}c_{m+2}^{k+1}\,.\end{gathered} \tag{1.101}$$

ახლა გადავიდეთ (1.85) განტოლების მარჯვენა მხარეში შემავალი სკალარული ნამრავლების გამოთვლაზე.

როგორც წინა პარაგრაფში (იხ. 1.2.1) აქაც $f_k(x)$, $\psi_0(x)$ და $\psi_1(x)$ ფუნქციების აპროქსიმაციას ვახდენთ კუბური სპლაინებით.

შენიშვნა 1.2.4. წინა პარაგრაფში (იხ. 1.2.1) $f_k(x)$, $\psi_0(x)$ და $\psi_1(x)$ ფუნქციებისათვის აგებული კუბური სპლაინების აღნიშვნები ამ პარაგრაფშიც იგივე დარჩება.

(1.86) და (1.87) ტოლობები ჩავწეროთ A_n-ის საშუალებით:

$$2(n+1)A_n^2P_{n+1}(x)=xP_n(x)-2nA_n^2P_{n-1}(x)\ , \tag{1.86.a}$$

$$2A_n^2\left(P'_{n+1}(x)-P'_{n-1}(x)\right)=P_n(x)\ , \tag{1.87.a}$$

შემოვიღოთ შემდეგი აღნიშვნები:

$$I_{j,n}^s=\int\limits_{x_{j-1}}^{x_j}x^sP_n(x)dx\,,\quad(s=0,1,2,3)\ .$$

(1.87.a) თანახმად, მივიღებთ:

$$I_{j,n}^0=2A_n^2\left[\left(P_{n+1}(x_j)-P_{n-1}(x_j)\right)-\left(P_{n+1}(x_{j-1})-P_{n-1}(x_{j-1})\right)\right]\ , \tag{1.102}$$

დანარჩენი სამი ინტეგრალის გამოსათვლელად გამოვიყენოთ (1.86.a) ფორმულა:

$$I_{j,n}^r=2A_n^2\left[(n+1)I_{j,n+1}^{r-1}+nI_{j,n-1}^{r-1}\right]\ ,\quad(r=1,2,3)\ . \tag{1.103}$$

შევნიშნოთ, რომ (1.102) ტოლობაში, როდესაც $n=0$ – ს $P_{n-1}(x_j)$ და $P_{n-1}(x_{j-1})$ წევრების ინდექსი უარყოფითი ხდება. ვიგულისხმოდ, რომ (1.102) და (1.103) ტოლობაში უარყოფით ინდექსიანი წევრები იგივურად ნულის ტოლია. ადვილი დასანახია, რომ $n=0$ და $n=1$ შემთხვევაში $I_{j,0}^s$ და $I_{j,1}^s$ $(s=0,1,2,3)$ ინტეგრალები ცხადი სახით იწერება, ამასთან $I_{j,1}^s$ გამოისახება

$I^s_{j,0}$-ის საშუალებით მარტივად, მართლაც:

$$I^s_{j,0} = \int\limits_{x_{j-1}}^{x_j} x^s P_0\left(x\right) dx = \frac{x_j^{s+1} - x_{j-1}^{s+1}}{s+1}\,, \tag{1.104}$$

$$I^s_{j,1} = \int\limits_{x_{j-1}}^{x_j} x^s P_1\left(x\right) dx = \int\limits_{x_{j-1}}^{x_j} x^{s+1} P_0\left(x\right) dx = I^{s+1}_{j,0}\,, \tag{1.105}$$

ახლა გამოვთვალოთ (1.85) განტოლების, მარჯვენა მხარეს შემავალი სკალარული ნამრავლი:

$$\begin{gathered}\left(g_k\left(x\right), \varphi_m\left(x\right)\right) = 2\tau^2\left(f_k\left(x\right), \varphi_m\left(x\right)\right) + 4\left(u_k\left(x\right), \varphi_m\left(x\right)\right) - \\ - \left(\left(2I - \tau^2 q_k \frac{d^2}{dx^2}\right) u_{k-1}\left(x\right), \varphi_m\left(x\right)\right)\,.\end{gathered} \tag{1.106}$$

შემოვიღოთ შემდეგი აღნიშვნები:

$$I^k_m = \left(f_k\left(x\right), \varphi_m\left(x\right)\right) \quad \text{და} \quad E^s_{j,m} = I^s_{j,m+1} - I^s_{j,m-1}\,, \quad \left(s = 0, 1, 2, 3\right)\,,$$

$f_k\left(x\right)$ ფუნქციის მაინტერპოლებელი კუბური სპლაინი არის $S^k_j\left(x\right)$. თუ $f_k\left(x\right)$–ს შევცვლით $S^k_j\left(x\right)$ მივიღებთ:

$$\begin{gathered}I^k_m = \left(f_k\left(x\right), \varphi_m\left(x\right)\right) = \int\limits_{-1}^{1} f_k\left(x\right) \varphi_m\left(x\right) dx = \\ = A_m \sum_{j=1}^{n}\left(A^k_j E^3_{j,m} + B^k_j E^2_{j,m} + C^k_j E^1_{j,m} + D^k_j E^0_{j,m}\right)\,.\end{gathered} \tag{1.107}$$

(1.107) ფორმულაში A^k_j, B^k_j, C^k_j და D^k_j ცნობილი სიდიდეებია, რომლებიც გამოითვლებიან (1.46a), (1.46b), (1.46c) და (1.46d) ტოლობებით. ასევე ცნობილი სიდიდეებია: $E^s_{j,m} = I^s_{j,m+1} - I^s_{j,m-1}\,,\quad (s = 0, 1, 2, 3)$, ისინი გამოითვლებიან შესაბამისად (1.102), (1.103), (1.104) და (1.105) ფორმულებით. თუ (1.106)-ში, (1.107), (1.99) და (1.100), მაშინ მივიღებთ:

$$\begin{aligned}\left(g_k\left(x\right), \varphi_m\left(x\right)\right) &= \\ &= 2B_{m-1}\left(c^{k-1}_{m-2} - 2c^k_{m-2}\right) - 2C_m\left(c^{k-1}_m - 2c^k_m\right) - \\ &- \tau^2\left(q_k c^{k-1}_m - 2I^k_m\right) + 2B_{m+1}\left(c^{k-1}_{m+2} - 2c^k_{m+2}\right)\,.\end{aligned} \tag{1.108}$$

საბოლოოდ ყოველი k-სთვის $(k = 1, 2, \ldots, M-1)$ (1.101)-ის და (1.108)-ის გათვალისწინებით მიიღება (იგულისხმება, რომ უარყოფით ინდექსიანი წევრები ნულის ტოლია):

$$\begin{aligned}&- 2B_{m-1}c^{k+1}_{m-2} + \left(2C_m + \tau^2 q_k\right) c^{k+1}_m - 2B_{m+1}c^{k+1}_{m+2} = \\ &= 2B_{m-1}\left(c^{k-1}_{m-2} - 2c^k_{m-2}\right) - 2C_m\left(c^{k-1}_m - 2c^k_m\right) - \\ &- \tau^2\left(q_k c^{k-1}_m - 2I^k_m\right) + 2B_{m+1}\left(c^{k-1}_{m+2} - 2c^k_{m+2}\right)\end{aligned} \tag{1.109}$$

(1.109) წრფივ ალგებრულ განტოლებათა სისტემის შესაბამისი მატრიცის სტრუქტურიდან გამომდინარე იგი შეგვიძლია გავხლიჩოთ ორ დამოუკიდებელ ქვესისტემად:

- თუ გვაქვს საკოორდინატო ფუნქციების ლუწი რაოდენობა ($N = 2j, \quad j \in \mathbb{N}$), მაშინ უცნობების გადანომრვა უნდა მოვახდინოთ შემდეგნაირად:
$c_{2s-1}^{k+1} = v_s^{k+1} \quad$ და $\quad c_{2s}^{k+1} = w_s^{k+1}, \quad (s = 1, 2, \dots, j)$
- თუ გვაქვს საკოორდინატო ფუნქციების კენტი რაოდენობა ($N = 2j - 1, \quad j \in \mathbb{N}$), მაშინ უცნობების გადანომრვა უნდა მოვახდინოთ შემდეგნაირად:
$c_{2s-1}^{k+1} = v_s^{k+1}, \quad (s = 1, 2, \dots, j) \quad$ და $\quad c_{2s}^{k+1} = w_s^{k+1}, \quad (s = 1, 2, \dots, j-1)$

ასეთი გადანომრვის შედეგად მიღებული ქვესისტემების შესაბამისი მატრიცები სამდიაგონალურია, მათი ამოხსნა შეგვიძლია ე. წ. ფაქტორიზაციის მეთოდით.

ყოველი c_m^{k+1} $(k = 1, 2, \dots, M-1)$ სიდიდეების საპოვნელად აუცილებელია, რომ ვიპოვნოთ შემდეგი ორი კოეფიციენტი c_m^0 და c_m^1. ამისათვის $u_0(x)$ და $u_1(x)$ სკალარულად გავამრავლებთ $\varphi_m(x)$-ზე.

$$\tilde{I}_m^0 = (\psi_0(x), \varphi_m(x)) = A_m \sum_{j=1}^{n} \left(\tilde{A}_j^0 E_{j,m}^3 + \tilde{B}_j^0 E_{j,m}^2 + \tilde{C}_j^0 E_{j,m}^1 + \tilde{D}_j^0 E_{j,m}^0 \right), \tag{1.110}$$

$$\tilde{I}_m^1 = (\psi_1(x), \varphi_m(x)) = A_m \sum_{j=1}^{n} \left(\tilde{A}_j^1 E_{j,m}^3 + \tilde{B}_j^1 E_{j,m}^2 + \tilde{C}_j^1 E_{j,m}^1 + \tilde{D}_j^1 E_{j,m}^0 \right), \tag{1.111}$$

თუ (1.79) და (1.80) ტოლობებში გავითვალისწინებთ (1.99), (1.100), (1.110) და (1.111), მაშინ მივიღებთ:

$$-B_{m-1} c_{m-2}^0 + C_m c_m^0 - B_{m+1} c_{m+2}^0 = \tilde{I}_m^0, \tag{1.112}$$

$$-B_{m-1} c_{m-2}^1 + C_m c_m^1 - B_{m+1} c_{m+2}^1 = \tilde{I}_m^0 + \tau \tilde{I}_m^1 - \frac{1}{2}\tau^2 \left(q_0 c_m^0 - I_m^0 \right), \tag{1.113}$$

იგულისხმება, რომ (1.112)-ში და (1.113)-ში უარყოფით ინდექსიანი წევრები ნულის ტოლია. (1.112) და (1.113) სისტემებიდან ჩვენ ვპოულობთ c_m^0 და c_m^1 სასტარტო კოეფიციენტებს. მათი საშუალებით (1.109) სისტემიდან შეგვიძლია ვიპოვნოთ (რეკურენტულად) c_m^k კოეფიციენტები, ნებისმიერი $k \geq 2$-თვის. თუ c_m^k კოეფიციენტებს ჩავსვამთ (1.82) ტოლობაში, მივიღებთ დასმული ამოცანის მიახლოებით ამონახსნს.

1.2.3 ლეჟანდრის პოლინომების სხვაობების შესაბამისი სისტემის გამოკვლევა

ქვემოთ მოვიყვანთ ზოგიერთ კარგად ცნობილ განმარტებას წრფივი ოპერატორების შესახებ ჰილბერტის სივრცეში (იხილეთ, მაგალითად [32] და [13]).

განსაზღვრება 1.2.1. A წრფივ ოპერატორს H ჰილბერტის სივრცეში ეწოდება დადებითი, თუ ის სიმეტრიულია

$$(Au, v) = (u, Av), \quad \forall u, v \in D(A)$$

და სრულდება პირობა,

$$(Au,u) \geq 0\,, \quad \forall u \in D\,(A)$$

$(Au,u) = 0$ სრულდება, მხოლოდ მაშინ, როდესაც $u = 0$ (u არის ნულოვანი ვექტორი). ამ განმარტების თანახმად, თუ გვაქვს A და B სიმეტრიული ოპერატორი H-ში, მაშინ ვიტყვით, $A \geq B$, თუ $A - B \geq 0$.

განსაზღვრება 1.2.2. A წრფივ ოპერატორს H ჰილბერტის სივრცეში ეწოდება დადებითად განსაზღვრული, თუ ის სიმეტრიულია და სრულდება პირობა:

$$(Au,u) \geq c\|u\|^2\,, \quad c = const > 0\,, \quad \forall u \in D\,(A)\,.$$

შენიშვნა 1.2.5. ნამდვილ ელემენტებიანი A სიმეტრიული მატრიცი დადებითად განსაზღვრულია $\mathbb{R}^n$ ევკლიდეს სივრცეში, თუ მისი ყველა საკუთრივი რიცხვი დადებითია.

თეორემა 1.2.2. *თუ A არის დადებითი ოპერატორი H ჰილბერტის სივრცეში, მაშინ*

$$Au = f\,, \quad f \in H \tag{1.114}$$

განტოლებას არ შეიძლება ჰქონდეს ერთზე მეტი ამონახსნი.

დამტკიცება. ვთქვათ, (1.114) განტოლებას აქვს ორი ამონახსნი u_1 და u_2, $Au_1 = f$ და $Au_2 = f$. აქედან A ოპერატორის წრფივობის გამო გამომდინარეობს, რომ $A\tilde{u} = 0$, სადაც $\tilde{u} = u_1 - u_2$. თუ ამ ტოლობის ორივე მხარეს გავამრავლებთ სკალარულად $\tilde{u}$-ზე, მივიღებთ $(A\tilde{u},\tilde{u}) = 0$, აქედან კი A ოპერატორის დადებითობიდან გამომდინარეობს, რომ $\tilde{u} = 0$, ან რაც იგივეა: $u_1 = u_2$ ■

თეორემა 1.2.3. *ვთქვათ $y = f\,(x)$, $x \in [a,b]$ არის არაუარყოფითი უწყვეტი ფუნქცია და $\int\limits_a^b f\,(x)\,dx = 0$, მაშინ $f\,(x) \equiv 0$.*

დამტკიცება. ვთქვათ $f\,(x) \not\equiv 0$, მაშინ არსებობს წერტილი $c \in [a,b]$, ისეთი, რომ $f\,(c) \neq 0$. რადგან პირობის თანახმად $f\,(x)$ ფუნქცია უწყვეტია, ამიტომ არსებობს c წერტილის ისეთი ε მიდამო $(c-\varepsilon, c+\varepsilon)$, სადაც $f\,(x) > 0$. ცხადია ამ ფუნქციის გათვალისწინებით გვაქვს

$$\int\limits_a^b f\,(x)\,dx \geq \int\limits_{c-\varepsilon}^{c+\varepsilon} f\,(x)\,dx > 0\,,$$

მივიღეთ წინააღმდეგობა.

თუ $c = a$, მაშინ იარსებებს $[a, a+\varepsilon)$ შუალედი, ხოლო $c = b$, მაშინ იარსებებს $(b-\varepsilon, b]$ შუალედი. ■

ვაჩვენოთ ყოველ შრეზე მიღებული (1.24) ელიფსური განტოლების შესაბამისი ოპერატორის დადებითად განსაზღვრულობა (ეს საკითხი კარგად არის გადმოცემული, მაგალითად [26]–ში). განვიხილოთ ოპერატორი:

$$Au = \left(2I + \tau^2 q_k B\right) u\,,$$

განსაზღვრის არით:

$$D\left(A\right)=\left\{u\left(x\right)\in C^{2}\left(\left[-1,1\right]\right)\middle|\, u\left(-1\right)=u\left(1\right)=0\right\}\ ,$$

სადაც, $B=-\frac{d^2}{dx^2}$, I–იგივური ოპერატორია და $q_k>0$.

ვაჩვენოთ $B=-\frac{d^2}{dx^2}$ ოპერატორის დადებითობა. $\forall u\in D\left(A\right)$-თვის და $\forall v\in D\left(A\right)$-თვის, ნაწილობითი ინტეგრების გამოყენებით მივიღებთ:

$$\left(Bu,v\right)=-\int\limits_{-1}^{1}\frac{d^2u}{dx^2}vdx=-\int\limits_{-1}^{1}\frac{d^2v}{dx^2}udx=\left(u,Bv\right)\ .$$

მივიღეთ, რომ $\left(Bu,v\right)=\left(u,Bv\right)$, ე. ი. B ოპერატორი სიმეტრიულია. $\forall u\in D\left(A\right)$–თვის,

$$\left(Bu,u\right)=-\int\limits_{-1}^{1}\frac{d^2u}{dx^2}udx=-\int\limits_{-1}^{1}u''udx=-u'u\Big|_{-1}^{1}+\int\limits_{-1}^{1}\left(u'\right)^2dx=\int\limits_{-1}^{1}\left(\frac{du}{dx}\right)^2dx\geq 0\,,$$

ე. ი. $\left(Bu,u\right)\geq 0$.

თუ $\left(Bu,u\right)=0$, მაშინ **თეორემა 1.2.3**-ის ძალით $u'=0$, ე. ი. $u=const$, მაგრამ რადგან $u\left(-1\right)=u\left(1\right)=0$ ამიტომ $u\equiv 0$. მივიღეთ, რომ B ოპერატორი დადებითია. ახლა ვაჩვენოთ A ოპერატორის დადებითად განსაზღვრულობა. $\forall u\in D\left(A\right)$–თვის ,

$$\left(Au,u\right)=2\left(u,u\right)+\tau^2q_k\left(Bu,u\right)=2\|u\|^2+\tau^2q_k\left(Bu,u\right)\geq 2\|u\|^2\,.$$

ვაჩვენოთ გალიორკინის მეთოდის გამოყენების შედეგად მიღებული წრფივ ალგებრულ განტოლებათა სისტემის შესაბამისი მატრიცის დადებითად განსაზღვრულობა.

ლემა 1.2.1. *H ჰილბერტის სივრცეში განვიხილოთ ზოგადი ოპერატორული განტოლება*

$$Au=f\,,\quad f\in H\,,$$

სადაც A არის სიმეტრიული ოპერატორი, რომელიც აკმაყოფილებს პირობას

$$\left(Au,u\right)\geq\alpha\left(Bu,u\right)+\nu\|u\|^2\,,\quad\forall u\in D\left(A\right)\subset D\left(B\right)\tag{1.115}$$

სადაც B ასევე სიმეტრიული ოპერატორია, ამასთან $D\left(A\right)\subset D\left(B\right)$; α და ν დადებითი მუდმივებია.

შესაბამისი განტოლებათა სისტემის მატრიცი დადებითად განსაზღვრულია, როდესაც საბაზისო ფუნქციები $\{\varphi_k\}_{k=1}^{\infty}$ არის B–ორთოგონალური. ეს ნიშნავს, რომ

$$\left(B\varphi_k,\varphi_i\right)=\begin{cases}1, & k=i;\\ 0, & k\neq i.\end{cases}\tag{1.116}$$

დამტკიცება. (1.5) სისტემის მატრიცი აღვნიშნოთ S_N-ით.

შემოვიღოთ ვექტორი $v_N=(c_1,c_2,\ldots,c_N)^T$ მარტივად შეგვიძლია ვაჩვენოთ, რომ:

$$S_N v_N = \left((Au_N,\varphi_1),(Au_N,\varphi_2),\ldots,(Au_N,\varphi_N)\right)^T,$$

მართლაც გვაქვს:

$$(Au_N,\varphi_i)=\left(\sum_{k=1}^{N} c_k A\varphi_k,\varphi_i\right)=\sum_{k=1}^{N}(A\varphi_k,\varphi_i)\,c_k\,,\quad (i=1,2,\ldots,N)$$

ამ ტოლობის გათვალისწინებით გვაქვს

$$\begin{aligned}(S_N v_N,v_N)&=c_1(Au_N,\varphi_1)+c_2(Au_N,\varphi_2)+\ldots+c_N(Au_N,\varphi_N)=\\&=(Au_N,c_1\varphi_1)+(Au_N,c_2\varphi_2)+\ldots+(Au_N,c_N\varphi_N)=\\&=\left(Au_N,\sum_{k=1}^{N}c_k\varphi_k\right)=(Au_N,u_N)\ ,\end{aligned}$$

მივიღეთ, რომ

$$(S_N v_N,v_N)=(Au_N,u_N) \tag{1.117}$$

(1.117)-იდან და (1.115)-იდან გამომდინარეობს, რომ

$$(S_N v_N,v_N)\geq\alpha(Bu_N,u_N)+\nu\|u_N\|^2 \tag{1.118}$$

u_N-ს აქვს შემდეგი სახე

$$u_N=\sum_{k=1}^{N}c_k\varphi_k \tag{1.119}$$

(1.118)–ში (1.119)–ის ჩასმით და (1.116)–ის გათვალისწინებით გამომდინარეობს, რომ

$$(S_N v_N,v_N)\geq\alpha\|v_N\|^2 \tag{1.120}$$

■

როგორც ვნახეთ საბაზისო ფუნქციებად ლეჟანდრის პოლინომების სხვაობების გამოყენების შედეგად მიღებული წრფივ ალგებრულ განტოლებათა სისტემის შესაბამისი მატრიცი იხლიჩება ორ ერთმანეთისაგან დამოუკიდებელ სამდიაგონალურ მატრიცებად. შევისწავლოთ ფაქტორიზაციის მეთოდის მდგრადობა.

მოცემულია $A=(a_{ij})\ ,\quad \forall i,j\in\{1,2,\ldots,N\}$, მატრიცი, სადაც:

$$a_{ij}=\begin{cases}2C_i+\tau^2 q_k\,, & \text{როდესაც}\quad j=i\,, \qquad i\in\{1,2,\ldots,N\};\\ -2B_{i-1}\,, & \text{როდესაც}\quad j=i-2\,,\quad i\in\{3,4,\ldots,N\};\\ -2B_{i+1}\,, & \text{როდესაც}\quad j=i+2\,,\quad i\in\{1,2,\ldots,N-2\};\\ 0\,, & \text{სხვა შემთხვევაში.}\end{cases}$$

(1.97) და (1.98) ტოლობებს აქვს შემდეგი სახე:

$$B_m = 4A_{m-1}A_m^2A_{m+1}\,, \qquad B_m = \frac{1}{(2m+1)\sqrt{(2m-1)\,(2m+3)}}\,, \tag{1.97}$$

$$C_m = 4A_m^2\left(A_{m-1}^2 + A_{m+1}^2\right)\,, \qquad C_m = \frac{2}{(2m-1)\,(2m+3)}\,, \tag{1.98}$$

შევნიშნოთ, რომ B_m-თვის $(m = 2, 3, \ldots, N-1)$

$$(2m)^2 < (2m-1)\,(2m+3) < (2m+1)^2\,,$$

მართლაც:

- ვაჩვენოთ მარცხენა უტოლობა

$$\begin{aligned}(2m-1)\,(2m+3) &> (2m)^2\\ 4m^2+4m-3 &> 4m^2\\ 4m &> 3\\ m &> \frac{3}{4}\end{aligned}$$

- ვაჩვენოთ მარჯვენა უტოლობა

$$\begin{aligned}(2m-1)\,(2m+3) &< (2m+1)^2\\ 4m^2+4m-3 &< 4m^2+4m+1\\ -3 &< 1\end{aligned}$$

მივიღებთ, რომ B_{m-1}-თვის $(m = 3, 4, \ldots, N)$, მართებულია უტოლობა

$$4(m-1)^2 < (2m-3)\,(2m+1) < (2m-1)^2\,, \tag{1.121}$$

ცხადია აქედან B_{m+1}-თვის $(m = 1, 2, \ldots, N-2)$ მივიღებთ

$$4(m+1)^2 < (2m+1)\,(2m+5) < (2m+3)^2\,, \tag{1.122}$$

(1.121)-ისა და (1.122)-ის გათვალისწინებით შევაფასოთ $B_{m-1} + B_{m+1} - C_m$ $(m = 3, 4, \ldots, N-2)$.

$$\begin{aligned}B_{m-1} + B_{m+1} - C_m &> \frac{16}{(2m-1)^2(2m+3)^2}\,,\\ B_{m-1} + B_{m+1} - C_m &< \frac{m+4}{(2m-1)\,(2m+3)\,(m-1)\,(m+1)}\,,\end{aligned}$$

საბოლოოდ მივიღეთ,

$$\frac{16}{(2m-1)^2(2m+3)^2} < B_{m-1} + B_{m+1} - C_m < \frac{m+4}{(2m-1)\,(2m+3)\,(m-1)\,(m+1)}\,,$$

ახლა შევაფასოთ ოთხი განსაკუთრებული შემთხვევა:

- როდესაც $m = 1$ და $m = 2$, შესაბამისად $B_{m-1}c_{m-2}^{k+1} = 0$ $\Big($რადგან $c_{-1}^{k+1} = c_0^{k+1} = 0\Big)$.
  (1.122) უტოლობის ძალით
  $$\frac{2m+5}{2\,(2m-1)\,(2m+3)\,(m+1)} < C_m - B_{m+1} < \frac{2m+7}{(2m-1)\,(2m+3)^2}\,,$$
  თუ უტოლობაში შევიტანოთ $m = 1$ და $m = 2$, მივიღებთ შემდეგ შეფასებებს:
  $$\frac{7}{20} < C_1 - B_2 < \frac{9}{25} \tag{1.123}$$
  $$\frac{1}{14} < C_2 - B_3 < \frac{11}{147} \tag{1.124}$$

- განვიხილოთ შემთხვევა, როდესაც $m = N-1$ და $m = N$, შესაბამისად $B_{m+1}c_{m+2}^{k+1} = 0$ $\Big($რადგან $c_{N+1}^{k+1} = c_{N+2}^{k+1} = 0\Big)$.
  (1.122) უტოლობის თანახმად
  $$\frac{2m-7}{2\,(2m-1)\,(2m+3)\,(m-1)} < C_m - B_{m-1} < \frac{2m-5}{(2m-1)^2\,(2m+3)}$$
  მიღებულ გამოსახულებაში ჩავსვათ $m = N-1$ და $m = N$,
  $$\frac{2N-9}{2\,(2N-3)\,(2N+1)\,(N-2)} < C_{N-1} - B_{N-2} < \frac{2N-7}{(2N-3)^2\,(2N+1)} \tag{1.125}$$
  $$\frac{2N-7}{2\,(2N-1)\,(2N+3)\,(N-1)} < C_N - B_{N-1} < \frac{2N-5}{(2N-1)^2\,(2N+3)} \tag{1.126}$$
  (1.125) უტოლობიდან გამომდინარეობს, რომ ოთხივე განსაკუთრებულ წერტილში ერთდროულად მკაცრი დიაგონალური ჭარბობისათვის საკოორდინატო ფუნქციების რაოდენობა უნდა იყოს მინიმუმ ხუთის ტოლი, ე. ი. $N = 5, 6, 7, \ldots$.

  ამრიგად,
  $$C_m + \frac{m+4}{(2m-1)\,(2m+3)\,(m-1)\,(m+1)} > B_{m-1} + B_{m+1}\,, \quad (m = 3, 4, \ldots, N-2)\,,$$

ამ შემთხვევაში შეიძლება ვთქვათ, რომ დიაგონალური ჭარბობა გვაქვს $\mathcal{O}\left(\frac{1}{m^3}\right)$ რიგით, რაც ასევე უზრუნველყოფს ფაქტორიზაციის მეთოდის მდგრადობას.

1.2.4 ხოლეცკის დეკომპოზიცია დადებითად განსაზღვრული მატრიცისათვის

განვიხილოთ დადებითად განსაზღვრული $A = (a_{i,j})_{n \times n}$ კვადრატული მატრიცის შემდეგნაირი დეკომპოზიცია
$$A = LDL^T\,, \tag{1.127}$$

სადაც L არის ქვედა სამკუთხა მატრიცა მთავარ დიაგონალზე ერთის ტოლი ელემენტებით, L^T არის L მატრიცის ტრანსპონირებული და D წარმოადგენს დიაგონალურ მატრიცს, ე. ი.

$$L = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \ell_{2,1} & 1 & 0 & \cdots & 0 \\ \ell_{3,1} & \ell_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \ell_{n,1} & \ell_{n,2} & \cdots & \ell_{n,n-1} & 1 \end{pmatrix}, \quad D = \begin{pmatrix} d_1 & 0 & 0 & \cdots & 0 \\ 0 & d_2 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & d_n \end{pmatrix},$$

A მატრიცს აქვს შემდეგი სახე

$$A = \begin{pmatrix} a_{1,1} & a_{2,1} & a_{3,1} & \cdots & a_{n,1} \\ a_{2,1} & a_{2,2} & a_{3,2} & \cdots & a_{n,2} \\ a_{3,1} & a_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & a_{n,n-1} \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n-1} & a_{n,n} \end{pmatrix}, \tag{1.128}$$

თუ გამოვთვალოთ LDL^T მატრიცების ნამრავლს და გავუტოლებთ (1.128), მივიღებთ:

$$a_{i,j} = \sum_{k=1}^{j} d_k \ell_{i,k} \ell_{j,k}, \quad (i, j \in \{1, 2, \dots, n\}, \quad \text{ამასთან} \quad i \geq j), \tag{1.129}$$

შევნიშნოთ, რომ მივიღებთ $\frac{n(n+1)}{2}$ განტოლებას, A მატრიცის სიმეტრიულობის გამო.
(1.129)–დან გამომდინარეობს, რომ

$$d_j = a_{jj} - \sum_{k=1}^{j-1} d_k \ell_{j,k}^2, \tag{1.130}$$

$$\ell_{i,j} = \frac{1}{d_j} \left(a_{i,j} - \sum_{k=1}^{j-1} d_k \ell_{i,k} \ell_{j,k} \right), \quad (d_j \neq 0), \tag{1.131}$$

თუ გვაქვს

$$Ax = b, \quad (\det(A) \neq 0), \tag{1.132}$$

სადაც A დადებითად განსაზღვრული მატრიცაა, ამასთან თუ A დაშლილი გვაქვს (1.127) სახით, მაშინ (1.132) შეგვიძლია გავხლიჩოთ სამ ქვესისტემად, მართლაც:
(1.127)–ი ჩავსვათ (1.132)–ში, მივიღებთ

$$LDL^T x = b,$$

შემოვიღოთ შემდეგი აღნიშვნები:

- $L^T x = y$,
  აქედან,

- $LDy = b\,,$
  თუ შემოვიღებთ კიდევ აღნიშვნას $Dy = z\,,$ საბოლოოდ მივიღებთ
- $Lz = b\,,$

წრფივ ალგებრულ განტოლებათა სისტემას ექნება შემდეგი სახე

$$\begin{cases} Lz = b, \\ Dy = z, \\ L^T x = y, \end{cases} \tag{1.133}$$

ახლა მოვიყვანოთ (1.133) სისტემის ამოხსნის ალგორითმი

- განვიხილოთ განტოლებათა სისტემა

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \ell_{2,1} & 1 & 0 & \cdots & 0 \\ \ell_{3,1} & \ell_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \ell_{n,1} & \ell_{n,2} & \cdots & \ell_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix},$$

$$\begin{aligned} & z_1 = b_1\,, \\ & z_i = b_i - \sum_{k=1}^{i-1} \ell_{i,k} z_k\,, \quad (i = 2, 3, \ldots, n)\,, \end{aligned} \tag{1.134}$$

- მეორე სისტემისათვის

$$\begin{pmatrix} d_1 & 0 & 0 & \cdots & 0 \\ 0 & d_2 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & d_n \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_n \end{pmatrix},$$

$$y_i = \frac{z_i}{d_i}\,, \quad (i = 1, 2, \ldots, n)\,, \tag{1.135}$$

- მესამე სისტემისათვის

$$\begin{pmatrix} 1 & \ell_{2,1} & \ell_{3,1} & \cdots & \ell_{n,1} \\ 0 & 1 & \ell_{3,2} & \cdots & \ell_{n,2} \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ell_{n,n-1} \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix},$$

$$
\begin{aligned}
&x_n = y_n\,,\\
&x_i = y_i - \sum_{k=i+1}^{n} \ell_{k,i} x_k\,, \quad (i = n-1, n-2, \dots, 1)\,,
\end{aligned}
\tag{1.136}
$$

ახლა განვიხილოთ საკოორდინატო ფუნქციებად ლეჟანდრის პოლინომების სხვაობების გამოყენების შედეგად მიღებული წრფივ ალგებრულ განტოლებათა სისტემის შესაბამისი მატრიცი. როგორც ვნახეთ **ლემა 1.2.1**–ის ძალით იგი დადებითად არის განსაზღვრული. ვიპოვნოთ ამ მატრიცის ხოლეცკის დეკომპოზიცია (1.127) სახით.

ვინაიდან $k = 0$ და $k = 1$ შრეზე შესაბამისად (1.112) და (1.113) წრფივ ალგებრულ განტოლებათა სისტემის შესაბამისი მატრიცი ერთიდაიგივეა, იგი აღვნიშნოთ A–თი

$$
A = \begin{pmatrix}
C_1 & 0 & -B_2 & 0 & \cdots & 0\\
0 & C_2 & 0 & -B_3 & \ddots & \vdots\\
-B_2 & 0 & C_3 & 0 & \ddots & 0\\
0 & -B_3 & 0 & \ddots & \ddots & -B_{N-1}\\
\vdots & \ddots & \ddots & \ddots & C_{N-1} & 0\\
0 & \cdots & 0 & -B_{N-1} & 0 & C_N
\end{pmatrix},
\tag{1.137}
$$

ყოველ $(k+1)$–ე $(k = 1, 2, \dots, M-1)$ შრეზე მიღებული (1.109) წრფივ ალგებრულ განტოლებათა სისტემის შესაბამისი მატრიცი აღვნიშნოთ $A^{(k)} = \left(a^{(k)}_{m,n}\right)_{N\times N}$–თი.

$$
A^{(k)} = \begin{pmatrix}
a^{(k)}_{1,1} & 0 & a^{(k)}_{3,1} & 0 & \cdots & 0\\
0 & a^{(k)}_{2,2} & 0 & a^{(k)}_{4,2} & \ddots & \vdots\\
a^{(k)}_{3,1} & 0 & a^{(k)}_{3,3} & 0 & \ddots & 0\\
0 & a^{(k)}_{4,2} & 0 & \ddots & \ddots & a^{(k)}_{N,N-2}\\
\vdots & \ddots & \ddots & \ddots & a^{(k)}_{N-1,N-1} & 0\\
0 & \cdots & 0 & a^{(k)}_{N,N-2} & 0 & a^{(k)}_{N,N}
\end{pmatrix},
$$

იგი წარმოვადგინოთ შემდეგი სახით:

$$
A^{(k)} = 2A + \tau^2 q_k I\,, \quad \text{სადაც} \quad I \quad \text{ერთეულოვანი მატრიცაა}
\tag{1.138}
$$

(1.138)–დან ჩანს, რომ $A^{(k)}$ მატრიცაში, მხოლოდ დიაგონალური ელემენტებია (k)–ზე დამოკიდებული. მატრიცის აღნიშვნიდან გამომდინარე თავის ყველა ელემენტს დავუწეროთ შრის ნომერი. $A^{(k)}$ მატრიცის ელემენტებს აქვს შემდეგი სახე:

$$
a^{(k)}_{m,n} = \begin{cases}
2C_m + \tau^2 q_k\,, & \text{როდესაც} \quad n = m\,, \qquad m \in \{1, 2, \dots, N\};\\
-2B_{m-1}\,, & \text{როდესაც} \quad n = m-2\,, \quad m \in \{3, 4, \dots, N\};\\
-2B_{m+1}\,, & \text{როდესაც} \quad n = m+2\,, \quad m \in \{1, 2, \dots, N-2\};\\
0\,, & \text{სხვა შემთხვევაში.}
\end{cases}
\tag{1.139}
$$

შენიშვნა 1.2.6. როგორც $A^{(k)}$ მატრიცის სტრუქტურიდან ჩანს, მას აქვს $(3N-4)$ არანულო-

ვანი წევრი და შესაბამისად $\left(N^2-3N+4\right)$ ნულოვანი წევრი.

თუ გამოვიყენებთ ხოლეცკის სქემას (1.137) და (1.138) მატრიცებისათვის მივიღებთ:

$$\ell_{m,m-2}^{(k)}=\frac{1}{d_{m-2}^{(k)}}a_{m,m-2}^{(k)}\,,\quad (m=3,4,\ldots,N)\ , \tag{1.140}$$

$$\begin{aligned} d_1^{(k)}&=a_{1,1}^{(k)}\,,\\ d_2^{(k)}&=a_{2,2}^{(k)}\,,\\ d_m^{(k)}&=a_{m,m}^{(k)}-\frac{1}{d_{m-2}^{(k)}}\left(a_{m,m-2}^{(k)}\right)^2,\quad (m=3,4,\ldots,N)\ . \end{aligned} \tag{1.141}$$

თვალსაჩინოებისათვის ჩავწეროთ (1.140) და (1.141) მატრიცული სახით:

$$L^{(k)}=\begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0\\ 0 & 1 & 0 & 0 & \ddots & \vdots\\ \ell_{3,1}^{(k)} & 0 & 1 & 0 & \ddots & 0\\ 0 & \ell_{4,2}^{(k)} & 0 & 1 & \ddots & 0\\ \vdots & \ddots & \ddots & \ddots & \ddots & 0\\ 0 & \cdots & 0 & \ell_{N,N-2}^{(k)} & 0 & 1 \end{pmatrix},\quad D^{(k)}=\begin{pmatrix} d_1^{(k)} & 0 & 0 & 0 & \cdots & 0\\ 0 & d_2^{(k)} & 0 & 0 & \ddots & \vdots\\ 0 & 0 & d_3^{(k)} & 0 & \ddots & 0\\ 0 & 0 & 0 & d_4^{(k)} & \ddots & 0\\ \vdots & \ddots & \ddots & \ddots & \ddots & 0\\ 0 & \cdots & 0 & 0 & 0 & d_N^{(k)} \end{pmatrix}.$$

შენიშვნა 1.2.7. $L^{(k)}$ მატრიცს აქვს $(2N-2)$ ცალი არანულოვანი ელემენტი და მაშასადამე $\left(N^2-2N+2\right)$ ნულოვანი ელემენტი.

ახლა ამოვხსნათ (1.109) წრფივ ალგებრულ განტოლებათა სისტემა (1.132) სისტემისათვის გამოყენებული ალგორითმით. (1.109) განტოლებათა სისტემის მარჯვენა მხარე აღვნიშნოთ $g^{(k)}$–თი, მივიღებთ

$$A^{(k)}c^{(k+1)}=g^{(k)}\,,\quad (k=1,2,\ldots,M-1)\ . \tag{1.142}$$

$$\begin{cases} L^{(k)}z^{(k+1)}=g^{(k)},\\ D^{(k)}y^{(k+1)}=z^{(k+1)},\\ \left(L^{(k)}\right)^T c^{(k+1)}=y^{(k+1)}, \end{cases} \tag{1.142.1}$$

- განვიხილოთ (1.142.1) სისტემის პირველი ქვესისტემა

$$L^{(k)}z^{(k+1)}=g^{(k)}\,,$$

(1.139)–ის, (1.140)–ისა და $L^{(k)}$ მატრიცის სტრუქტურის გათვალისწინებით (1.134) მიიღებს შემდეგ სახეს:

$$\begin{cases} z_m^{(k+1)}=g_m^{(k)}, & m\in\{1,2\};\\ z_m^{(k+1)}=g_m^{(k)}+\frac{2B_{m-1}}{d_{m-2}^{(k)}}z_{m-2}^{(k+1)}, & m\in\{3,4,\ldots,N\}. \end{cases}$$

- ახლა მეორე ქვესისტემისათვის

$$D^{(k)}y^{(k+1)} = z^{(k+1)}\,,$$

(1.135)–ის გათვალისწინებით გვექნება:

$$y_m^{(k+1)} = \frac{z_m^{(k+1)}}{d_m^{(k)}}\,, \quad m \in \{1, 2, \ldots, N\}\,, \tag{1.143}$$

- ბოლო, მესამე ქვესისტემისათვის

$$\left(L^{(k)}\right)^T c^{(k+1)} = y^{(k+1)}\,,$$

(1.136) მიიღებს შემდეგ სახეს:

$$\begin{cases} c_m^{(k+1)} = y_m^{(k+1)}, & m \in \{N, N-1\}; \\ c_m^{(k+1)} = y_m^{(k+1)} + \frac{2B_{m+1}}{d_m^{(k)}} c_{m+2}^{(k+1)}, & m \in \{N-2, N-3, \ldots, 1\}. \end{cases}$$

საბოლოოდ მივიღეთ, რომ ყოველი $(k+1)$–ე $(k = 1, 2, \ldots, M-1)$ შრისათვის $A^{(k)}c^{(k+1)} = g^{(k)}$ წრფივ ალგებრულ განტოლებათა სისტემის ამონახსნს აქვს სახე:

$$\begin{cases} z_m^{(k+1)} = g_m^{(k)}, & m \in \{1, 2\}; \\ z_m^{(k+1)} = g_m^{(k)} + \frac{2B_{m-1}}{d_{m-2}^{(k)}} z_{m-2}^{(k+1)}, & m \in \{3, 4, \ldots, N\}; \\ y_m^{(k+1)} = \frac{z_m^{(k+1)}}{d_m^{(k)}}, & m \in \{1, 2, \ldots, N\}; \\ c_m^{(k+1)} = y_m^{(k+1)}, & m \in \{N, N-1\}; \\ c_m^{(k+1)} = y_m^{(k+1)} + \frac{2B_{m+1}}{d_m^{(k)}} c_{m+2}^{(k+1)}, & m \in \{N-2, N-3, \ldots, 1\}. \end{cases}$$

სადაც (1.141)–ის და (1.139)–ის თანახმად:

$$\begin{cases} d_m^{(k)} = 2C_m + \tau^2 q_k, & m \in \{1, 2\}; \\ d_m^{(k)} = \left(2C_m + \tau^2 q_k\right) - \frac{4B_{m-1}^2}{d_{m-2}^{(k)}}, & m \in \{3, 4, \ldots, N\}. \end{cases}$$

ახლა ამოვხსნათ წრფივ ალგებრულ განტოლებათა სისტემები, $k = 0$ და $k = 1$ შრისათვის. ამ შემთხვევაში ორივე სისტემის შესაბამისი მატრიცი ერთი და იგივეა და აქვს (1.137) სახე. მარჯვენა მხარეები ამ სისტემების აღვნიშნოთ შესაბამისად $b^{(0)}$–ითა და $b^{(1)}$–ით.
გვაქვს:

$$Ac^{(n)} = b^{(n)}\,, \quad n = 0, 1\,, \tag{1.144}$$

(1.144) სისტემის ამონახსნს ექნება შემდეგი სახე $(n=0 \quad \text{და} \quad n=1)$:

$$\begin{cases} z_m^{(n)} = b_m^{(n)}, & m \in \{1,2\}; \\ z_m^{(n)} = b_m^{(n)} + \frac{B_{m-1}}{d_{m-2}} z_{m-2}^{(n)}, & m \in \{3,4,\ldots,N\}; \\ y_m^{(n)} = \frac{z_m^{(n)}}{d_m}, & m \in \{1,2,\ldots,N\}; \\ c_m^{(n)} = y_m^{(n)}, & m \in \{N, N-1\}; \\ c_m^{(n)} = y_m^{(n)} + \frac{B_{m+1}}{d_m} c_{m+2}^{(n)}, & m \in \{N-2, N-3, \ldots, 1\}. \end{cases}$$

სადაც

$$\begin{cases} d_m = C_m, & m \in \{1,2\}; \\ d_m = C_m - \frac{B_{m-1}^2}{d_{m-2}}, & m \in \{3,4,\ldots,N\}. \end{cases}$$

1.2.5 ვარიაციული მეთოდის ცდომილების შეფასება მეორე რიგის, წრფივი ჩვეულებრივი დიფერენციალური განტოლებისათვის

მოცემული საკითხი, კერძოდ როდესაც საკოორდინატო ფუნქციებად აღებულია სინუსები გადმოცემულია [26] ლიტერატურაში.

განვიხილოთ ამოცანა

$$\frac{d}{dx}\left(p(x)\frac{dy}{dx}\right) - q(x)\,y = f(x)\,, \quad x \in [-1,1]\,, \tag{1.145}$$

$$y(-1) = y(1) = 0\,, \tag{1.146}$$

სადაც $p(x) \geq p_0 > 0\,,\ q(x) \geq 0\,.$

ჩვენი მიზანია (1.145) – (1.146) ამოცანის ამოხსნა რიტცის მეთოდით.

ამოხსნის ქვეშ ჩვენ ვგულისხმობთ:

1. ისეთი y_n (n–ური მიახლოება) ფუნქციის მონახვას, რომელიც ნებისმიერი სიზუსტით დააკმაყოფილებს (1.145)–(1.146) ამოცანას. ეს იმას ნიშნავს, რომ

$$|y(x) - y_n(x)| < \varepsilon\,; \quad \text{როცა} \quad n > \mathcal{N}\,;$$

და

$$y(-1) = y(1) = 0\,;$$

2. კრებადობის რიგის დადგენა. საკოორდინატო ფუნქციებად ავიღოთ შემდეგი ფუნქციები:

$$\varphi_0\,, \varphi_1\,, \ldots\,, \varphi_n\,, \ldots$$
$$\varphi_i(x) = \sqrt{\frac{2i+1}{2}} \int_{-1}^{x} P_i(t)\,dt\,,$$

სადაც $P_i(x)$ არის i–ური ხარისხის ლეჟანდრის პოლინომები.

ცხადია $\varphi_i(-1) = \varphi_i(1) = 0$; შემოვიღოთ აღნიშვნა

$$\tilde{P}_i(x) = \sqrt{\frac{2i+1}{2}} P_i(x) \,,$$

მაშინ

$$\varphi_i(x) = \int\limits_{-1}^{x} \tilde{P}_i(t) dt \,,$$

$$\tilde{P}_0(x) \,,\, \tilde{P}_1(x) \,,\, \ldots \,,\, \tilde{P}_n(x) \,,\, \ldots$$

პოლინომთა ეს სისტემა წარმოადგენს ლეჟანდრის ორთონორმირებულ პოლინომთა სისტემას $[-1,1]$ სეგმენტზე

$$\int\limits_{-1}^{1} \tilde{P}_n(x)\, \tilde{P}_m(x) dx = \delta_{nm} = \begin{cases} 1, & \text{როცა} \quad n = m; \\ 0, & \text{როცა} \quad n \neq m. \end{cases}$$

(1.145)–(1.146) ამოცანა ექვივალენტურია შემდეგი ვარიაციული ამოცანისა:

$$I(y) = \int\limits_{-1}^{1} \Big(p{y'}^2 + qy^2 + 2fy \Big) dx \,; \qquad (1.145')$$

$$y(-1) = y(1) = 0 \,; \qquad (1.146')$$

ის ფუნქცია, რომელიც $I(y)$–ს ანიჭებს მინიმალურ მნიშვნელობას $(1.146')$ პირობებში წარმოადგენს (1.145)–(1.146) ამოცანის ამონახსნს.
ადგილი აქვს ასეთ თეორემას

თეორემა 1.2.4. *თუ $[-1,1]$ სეგმენტზე მოცემულ $f(x)$ ფუნქციას გააჩნია მეორე რიგის უწყვეტი წარმოებული, მაშინ ის შეგვიძლია გავშალოთ თანაბრად კრებად ფურიეს მწკრივად ლეჟანდრის პოლინომების მიხედვით, ე. ი. ნებისმიერი $\varepsilon > 0$ – თვის მოიძებნება ისეთი n_0 ნატურალური რიცხვი, რომ*

$$|f(x) - S_n(x)| < \varepsilon \,, \quad \text{როცა} \quad n > n_0 \,;$$

სადაც

$$S_n(x) = \sum_{k=0}^{n} c_k \tilde{P}_k(x) \,;$$

$$c_k = \int\limits_{-1}^{1} f(x)\, \tilde{P}_k(x) dx \,;$$

ვთქვათ $y(x)$ არის (1.145)–(1.146) ამოცანის ამონახსნი და $\bar{y}(x)$ არინ ნებისმიერი სხვა ფუნქცია, რომელიც აკმაყოფილებს (1.146) სასაზღვრო პირობას $\bar{y}(-1) = \bar{y}(1) = 0$. შემოვიღოთ ფუნქცია $\eta(x) = \bar{y}(x) - y(x)$, ცხადია იგი აკმაყოფილებს (1.146) სასაზღვრო პირობებს $\eta(-1) = \eta(1) = 0$.

ჩვენ გვაქვს

$$I\left(\bar{y}\right)-I\left(y\right)=I\left(y+\eta\right)-I\left(y\right)=\int\limits_{-1}^{1}\Big[p\big(y'+\eta'\big)^{2}+q(y+\eta)^{2}+2f\left(y+\eta\right)\Big]dx-$$

$$-\int\limits_{-1}^{1}\Big[p{y'}^{2}+qy^{2}+2fy\Big]dx=$$

$$=2\int\limits_{-1}^{1}\big[py'\eta'+qy\eta+f\eta\big]dx+\int\limits_{-1}^{1}\Big[p{\eta'}^{2}+q\eta^{2}\Big]dx\,;$$

მიღებულ ბოლო ორ შესაკრებში პირველი წევრი არის ვარიაცია და იგი ნულის ტოლია ($\delta I=0$). საბოლოოდ მივიღეთ, რომ

$$I\left(\bar{y}\right)-I\left(y\right)=\int\limits_{-1}^{1}\Big[p\big(\bar{y}'-y'\big)^{2}+q(\bar{y}-y)^{2}\Big]dx\,;$$

ახლა ვუჩვენოთ ფუნქციათა $\{\varphi_i\left(x\right)\}$ სისტემის სისრულე.

ცხადია

$$\left|y'\left(x\right)-S_n^*\left(x\right)\right|<\varepsilon\,,\quad \text{როცა}\quad n>\mathcal{N}\,; \tag{1.147}$$

სადაც

$$S_n^*\left(x\right)=\sum_{k=0}^{n}c_k^*\tilde{P}_k\left(x\right);$$

$$c_k^*=\int\limits_{-1}^{1}y'\left(x\right)\tilde{P}_k\left(x\right)dx\,,\quad c_0^*=0\,;$$

(1.147) უტოლობის საფუძველზე გვექნება

$$\left|y\left(x\right)-\int\limits_{-1}^{x}S_n^*\left(x\right)dx\right|=\left|\int\limits_{-1}^{x}\big[y'\left(x\right)-S_n^*\left(x\right)\big]dx\right|\le\int\limits_{-1}^{x}\big|y'\left(x\right)-S_n^*\left(x\right)\big|dx\le 2\varepsilon\,;$$

მაგრამ

$$\Phi_n\left(x\right)=\int\limits_{-1}^{x}S_n^*\left(x\right)dx=\sum_{k=1}^{n}c_k^*\varphi_k\left(x\right);$$

$$\Phi'_n\left(x\right)=S_n^*\left(x\right)=\sum_{k=1}^{n}c_k^*\varphi'_k\left(x\right);$$

ე. ი.

$$|y(x)-\Phi_n(x)|=\left|y(x)-\sum_{k=1}^{n}c_k^*\varphi_k(x)\right|<2\varepsilon\,;$$
$$\left|y'(x)-\Phi'_n(x)\right|=\left|y'(x)-\sum_{k=1}^{n}c_k^*\varphi'_k(x)\right|<\varepsilon\,;$$

გვაქვს, რომ

$$\int_{-1}^{1}\Big(py'^2+qy^2+fy\Big)dx=0\,; \tag{1.148}$$

უკანასკნელი ტოლობა მიიღება შემდეგნაირად,
განვიხილოთ

$$\delta I=\int_{-1}^{1}\big(py'\eta'+qy\eta+f\eta\big)dx\,,\quad \eta(-1)=\eta(1)=0\,;$$

ამ ტოლობაში თუ $\eta(x)$–ს მაგივრად ჩავსვამთ $y(x)$–ს მივიღებთ ზემო აღნიშნულ ტოლობას. (1.148)–დან გვექნება

$$\int_{-1}^{1}py'^2dx\le\left|\int_{-1}^{1}fydx\right|\le\|f\|\cdot\|y\|\ ;$$

რადგან $p(x)$ არის უწყვეტი $[-1,1]$–ზე (ამასთან პირობის ძალით $p(x)\ge p_0>0$) და $y'^2(x)$ არის არაუარყოფითი ინტეგრებადი ფუნქცია $[-1,1]$–ზე, ამიტომ სამართლიანია შემდეგი უტოლობა

$$\min p\int_{-1}^{1}y'^2dx\le\int_{-1}^{1}py'^2dx\le\max p\int_{-1}^{1}y'^2dx\,;$$

ზედა, ბოლო ორი უტოლობის თანახმად

$$\left\|y'\right\|^2=\int_{-1}^{1}y'^2dx\le\frac{1}{\min p}\|f\|\cdot\|y\|\ ; \tag{1.149}$$

ახლა გამოვიყვანოთ ერთი დამხმარე უტოლობა.
ცხადია

$$y(x)=\int_{-1}^{x}y'(x)dx\,;$$

$$\|y\|^2=\int_{-1}^{1}\left\{\int_{-1}^{x}y'(t)dt\right\}^2dx\le\int_{-1}^{1}\left\{\int_{-1}^{x}1^2dt\cdot\int_{-1}^{x}y'^2(t)dt\right\}dx\le$$
$$\le\left\|y'\right\|^2\int_{-1}^{1}(x+1)dx=2\left\|y'\right\|^2;$$

მივიღეთ, რომ

$$\|y\|^2 \le 2\|y'\|^2 ;$$
$$\|y\| \le \sqrt{2}\,\|y'\| ; \tag{1.150}$$

ასევე ადგილი აქვს შემდეგ უტოლობას

$$|y\,(x)| = \left|\int_{-1}^{x} y'\,(x)dx\right| \le \sqrt{\int_{-1}^{x} y'^2 dx} \cdot \sqrt{\int_{-1}^{x} 1^2 dx} \le \sqrt{2}\,\|y'\| ;$$

ე. ი.

$$|y\,(x)| \le \sqrt{2}\,\|y'\| ;$$

აქედან კი

$$\max|y\,(x)| \le \sqrt{2}\,\|y'\| ; \tag{1.151}$$

(1.149)–დან და (1.150)–დან გამომდინარეობს, რომ

$$\|y'\|^2 \le \frac{\sqrt{2}}{\min p}\,\|f\| \cdot \|y'\| ;$$

$$\|y'\| \le \frac{\sqrt{2}}{\min p}\,\|f\| ; \tag{1.152}$$

ასევე

$$\|y\|^2 \le \frac{2}{\min p}\,\|f\| \cdot \|y\| ;$$

$$\|y\| \le \frac{2}{\min p}\,\|f\| ; \tag{1.153}$$

(1.151) და (1.152) გვაძლევს

$$\max|y\,(x)| \le \frac{2}{\min p}\,\|f\| ; \tag{1.154}$$

(1.145) განტოლებიდან გვაქვს

$$py'' = -p'y' + qy + f ;$$

$$\begin{aligned}
&\|py''\| \le \|p'y'\| + \|qy\| + \|f\| \le \max|p'| \cdot \|y'\| + \max q \cdot \|y\| + \|f\| \le \\
&\le \frac{\sqrt{2}\max|p'|}{\min p}\,\|f\| + \frac{2\max q}{\min p}\,\|f\| + \|f\| = \left[\frac{\sqrt{2}\max|p'|}{\min p} + \frac{2\max q}{\min p} + 1\right]\|f\| = \\
&= \frac{1}{\min p}\left[\sqrt{2}\max|p'| + 2\max q + \min p\right]\|f\| ;
\end{aligned}$$

საბოლოოდ

$$\|y''\| \le \frac{1}{(\min p)^2}\left[\sqrt{2}\max|p'| + 2\max q + \min p\right]\|f\| ;$$

ე. ი.

$$\|y''\| \le c_1\,\|f\| , \quad \text{სადაც} \quad c_1 = \frac{1}{(\min p)^2}\left[\sqrt{2}\max|p'| + 2\max q + \min p\right] ; \tag{1.155}$$

ახლა დავუბრუნდეთ ჩვენს საკოორდინატო ფუნქციებს

$$\varphi_n(x)=\int\limits_{-1}^{x}\sqrt{\frac{2n+1}{2}}P_n(t)dt=\int\limits_{-1}^{x}\tilde{P}_n(t)dt\,;$$

(1.87) თვისების ძალით

$$P_n(x)=\frac{1}{2n+1}\left[P'_{n+1}(x)-P'_{n-1}(x)\right]\;; \tag{1.87'}$$

ცხადია გვექნება

$$\varphi_n(x)=\frac{1}{\sqrt{2(2n+1)}}\int\limits_{-1}^{x}\left[P'_{n+1}(t)-P'_{n-1}(t)\right]dt=A_n\left[P_{n+1}(t)|_{-1}^{x}-P_{n-1}(t)|_{-1}^{x}\right]=$$
$$=A_n\left[P_{n+1}(x)-P_{n+1}(-1)-P_{n-1}(x)+P_{n-1}(-1)\right]=$$
$$=A_n\left[P_{n+1}(x)-P_{n-1}(x)-(-1)^{n+1}+(-1)^{n-1}\right]=$$
$$=A_n\left[P_{n+1}(x)-P_{n-1}(x)+(-1)^{n-1}(-1+1)\right]=A_n\left(P_{n+1}(x)-P_{n-1}(x)\right)\;;$$

ე. ი.

$$\varphi_n(x)=A_n\left(P_{n+1}(x)-P_{n-1}(x)\right)\;,\quad \text{სადაც}\quad A_n=\frac{1}{\sqrt{2(2n+1)}}\;;$$

შევნიშნოთ, რომ

$$\varphi_0(x)=\frac{1}{\sqrt{2}}\int\limits_{-1}^{x}P_0(t)dt=\frac{1}{\sqrt{2}}\int\limits_{-1}^{x}dt=\frac{1}{\sqrt{2}}(x+1)=\frac{1}{\sqrt{2}}\left(P_1(x)+P_0(x)\right)=$$
$$=A_0\left(P_1(x)-P_{-1}(x)\right)\;;$$

სადაც ვგულისხმობთ, რომ $P_{-1}(x)=-P_0(x)$;
საბოლოოდ მივიღეთ, რომ

$$\varphi_n(x)=A_n\left(P_{n+1}(x)-P_{n-1}(x)\right)\;,\quad n=0,1,2,\ldots$$

ამ ტოლობას შეგვიძლია მივცეთ სხვანაირი სახე, კერძოდ

$$\tilde{P}_n(x)=\sqrt{\frac{2n+1}{2}}P_n(x)=\sqrt{\frac{2(2n+1)}{4}}P_n(x)=\frac{\sqrt{2(2n+1)}}{2}P_n(x)=\frac{1}{2A_n}P_n(x)\;;$$

საიდანაც მივიღებთ, რომ

$$P_n(x)=2A_n\tilde{P}_n(x)\;; \tag{1.156}$$

უკანასკნელი ტოლობა გვაძლევს

$$\varphi_n(x)=2A_n\left(A_{n+1}\tilde{P}_{n+1}(x)-A_{n-1}\tilde{P}_{n-1}(x)\right)\;;\quad n=0,1,2,\ldots$$

შემოვიღოთ შემდეგი აღნიშვნები:

$y^{(k)}(x)$ ფუნქციის ლეჟანდრის პოლინომების მიხედვით ფურიეს მწკრივად განაშლის პირველი n წევრის ჯამი აღვნიშნოთ $S_{n(k)}$–თი.
ე. ი.

$$S_{n(k)}(x) = \sum_{j=0}^{n} c_j^{(k)} \tilde{P}_j(x);$$

სადაც

$$c_j^{(k)} = \int_{-1}^{1} y^{(k)}(x) \tilde{P}_j(x) dx;$$

$$y^{(0)} = y, \;\; y^{(1)} = y', \;\; y^{(2)} = y'', \;\; \ldots$$

ცხადია

$$c_n = c_n^{(0)} = \int_{-1}^{1} y^{(0)}(x) \tilde{P}_n(x) dx = \int_{-1}^{1} y(x) \tilde{P}_n(x) dx; \tag{1.157}$$

$(1.87')$ გადავწეროთ შემდეგი სახით

$$P_n(x) = 2A_n^2 \left(P'_{n+1}(x) - P'_{n-1}(x)\right); \tag{1.87''}$$

ახლა $(1.87'')$ გარდავქმნათ (1.156)–ის გამოყენებით, მივიღებთ

$$2A_n \tilde{P}_n(x) = 4A_n^2 \left(A_{n+1} \tilde{P}'_{n+1}(x) - A_{n-1} \tilde{P}'_{n-1}(x)\right);$$

საიდანაც

$$\tilde{P}_n(x) = 2A_n \left(A_{n+1} \tilde{P}'_{n+1}(x) - A_{n-1} \tilde{P}'_{n-1}(x)\right); \tag{1.87'''}$$

$(1.87''')$–ის თანახმად (1.157) მიიღებს შემდეგ სახეს:

$$c_n = \int_{-1}^{1} y(x) \tilde{P}_n(x) dx = 2A_n \left(A_{n+1} \int_{-1}^{1} y(x) \tilde{P}'_{n+1}(x) dx - A_{n-1} \int_{-1}^{1} y(x) \tilde{P}'_{n-1}(x) dx\right) =$$
$$= 2A_n \left(A_{n-1} c_{n-1}^{(1)} - A_{n+1} c_{n+1}^{(1)}\right);$$

მართლაც

$$A_{n+1} \int_{-1}^{1} y(x) \tilde{P}'_{n+1}(x) dx = A_{n+1} \left[y(x) \tilde{P}_{n+1}(x) \Big|_{-1}^{1} - \int_{-1}^{1} y'(x) \tilde{P}_{n+1}(x) dx \right] =$$
$$= A_{n+1} \left[y(1) \frac{1}{2A_{n+1}} - y(-1) \frac{(-1)^{n+1}}{2A_{n+1}} - c_{n+1}^{(1)} \right] = \frac{y(1) - (-1)^{n+1} y(-1)}{2} - A_{n+1} c_{n+1}^{(1)} =$$
$$= \frac{y(1) + (-1)^n y(-1)}{2} - A_{n+1} c_{n+1}^{(1)};$$

სრულიად ანალოგიურად

$$A_{n-1}\int\limits_{-1}^{1} y\left(x\right)\tilde{P}'_{n-1}\left(x\right)dx=\frac{y\left(1\right)-(-1)^{n-1}y\left(-1\right)}{2}-A_{n-1}c^{(1)}_{n-1}=$$
$$=\frac{y\left(1\right)+(-1)^{n}y\left(-1\right)}{2}-A_{n-1}c^{(1)}_{n-1}\,;$$

თუ წინა ტოლობას გამოვაკლებთ უკანასკნელ ტოლობას, მივიღებთ საძიებელს. ამგვარად, ჩვენ ვაჩვენეთ შემდეგი ტოლობის სამართლიანობა

$$c_n=2A_n\left(A_{n-1}c^{(1)}_{n-1}-A_{n+1}c^{(1)}_{n+1}\right)\;;$$

აქედან

$$|c_n|\leq 2A_n\left(A_{n-1}\left|c^{(1)}_{n-1}\right|+A_{n+1}\left|c^{(1)}_{n+1}\right|\right)\;;$$

უკანასკნელი უტოლობა კი გვაძლევს

$$c_n^2\leq 4A_n^2\left(A_{n-1}^2c^{(1)2}_{n-1}+2A_{n-1}A_{n+1}\left|c^{(1)}_{n-1}\cdot c^{(1)}_{n+1}\right|+A_{n+1}^2c^{(1)2}_{n+1}\right)\leq$$
$$\leq 8A_n^2\left(A_{n-1}^2c^{(1)2}_{n-1}+A_{n+1}^2c^{(1)2}_{n+1}\right)\;;$$

ე. ი.

$$c_n^2\leq 8A_n^2\left(A_{n-1}^2c^{(1)2}_{n-1}+A_{n+1}^2c^{(1)2}_{n+1}\right)\;;$$

თუ მოვახდენთ აჯამვას $(n+1)$–დან ∞–მდე, მივიღებთ

$$\sum_{k=n+1}^{\infty}c_k^2\leq\sum_{k=n+1}^{\infty}8A_k^2\left(A_{k-1}^2c^{(1)2}_{k-1}+A_{k+1}^2c^{(1)2}_{k+1}\right)\leq$$
$$\leq 8A_{n+1}^2\left\{A_n^2\sum_{k=n+1}^{\infty}c^{(1)2}_{k-1}+A_{n+2}^2\sum_{k=n+1}^{\infty}c^{(1)2}_{k+1}\right\}\;;$$

შევნიშნოთ, რომ

$$y'\left(x\right)=\sum_{j=0}^{\infty}c^{(1)}_j\tilde{P}_j\left(x\right);$$
$$S_{n(1)}\left(x\right)=\sum_{j=0}^{n}c^{(1)}_j\tilde{P}_j\left(x\right);$$
$$y'\left(x\right)-S_{n(1)}\left(x\right)=\sum_{j=n+1}^{\infty}c^{(1)}_j\tilde{P}_j\left(x\right);$$
$$\left\|y'\left(x\right)-S_{n(1)}\left(x\right)\right\|^2=\int\limits_{-1}^{1}\left[y'\left(x\right)-S_{n(1)}\left(x\right)\right]^2dx=\int\limits_{-1}^{1}\left[\sum_{j=n+1}^{\infty}c^{(1)}_j\tilde{P}_j\left(x\right)\right]^2dx=$$
$$=\sum_{j=n+1}^{\infty}c^{(1)2}_j\,;$$

საბოლოოდ მივიღეთ, რომ

$$\sum_{j=n+1}^{\infty} c_j^{(1)2} = \left\|y'(x) - S_{n(1)}(x)\right\|^2;$$

ბოლო ტოლობა გვაძლევს, რომ

$$8A_{n+1}^2 \left\{ A_n^2 \sum_{k=n+1}^{\infty} c_{k-1}^{(1)2} + A_{n+2}^2 \sum_{k=n+1}^{\infty} c_{k+1}^{(1)2} \right\} =$$
$$= 8A_{n+1}^2 \left\{ A_n^2 \left\|y'(x) - S_{n-1(1)}(x)\right\|^2 + A_{n+2}^2 \left\|y'(x) - S_{n+1(1)}(x)\right\|^2 \right\} \le$$
$$\le 8A_{n+1}^2 \left(A_n^2 + A_{n+2}^2\right) \cdot \left\|y'(x) - S_{n-1(1)}(x)\right\|^2;$$

ჩვენი აღნიშვნების თანახმად

$$B_m = 4A_{m-1}A_m^2A_{m+1}, \qquad B_m = \frac{1}{(2m+1)\sqrt{(2m-1)(2m+3)}}, \tag{1.97}$$

$$C_m = 4A_m^2\left(A_{m-1}^2 + A_{m+1}^2\right), \qquad C_m = \frac{2}{(2m-1)(2m+3)}, \tag{1.98}$$

შევნიშნოთ, რომ

$$\|\varphi_n(x)\|^2 = \int_{-1}^{1} [\varphi_n(x)]^2 dx = 4A_n^2 \int_{-1}^{1} \left(A_{n+1}\tilde{P}_{n+1}(x) - A_{n-1}\tilde{P}_{n-1}(x)\right)^2 dx =$$
$$= 4A_n^2\left(A_{n-1}^2 + A_{n+1}^2\right);$$

ამასთან

$$C_n = 4A_n^2\left(A_{n-1}^2 + A_{n+1}^2\right) = \frac{2}{(2n-1)(2n+3)} = \frac{8}{2(2(n-1)+1)\cdot 2(2(n+1)+1)} =$$
$$= 8A_{n-1}^2A_{n+1}^2;$$

საბოლოოდ გვექნება

$$C_n = \|\varphi_n(x)\|^2 = 4A_n^2\left(A_{n-1}^2 + A_{n+1}^2\right) = 8A_{n-1}^2A_{n+1}^2;$$

ზედა ტოლობის გათვალისწინებით

$$8A_{n+1}^2\left(A_n^2 + A_{n+2}^2\right) = 2C_{n+1} = 16A_n^2A_{n+2}^2 = \frac{4}{(2n+1)(2n+5)};$$

საბოლოოდ მივიღებთ შემდეგ შეფასებას

$$\left\|y(x) - S_{n(0)}(x)\right\|^2 \le 16A_n^2A_{n+2}^2 \left\|y'(x) - S_{n-1(1)}(x)\right\|^2;$$

სრულიად ანალოგიურად მივიღებთ, რომ

$$\left\|y^{(k)}\left(x\right)-S_{n(k)}\left(x\right)\right\|^{2}\leq16A_{n}^{2}A_{n+2}^{2}\left\|y^{(k+1)}\left(x\right)-S_{n-1(k+1)}\left(x\right)\right\|^{2},\qquad k=0,1,2,\ldots\,; \tag{1.158}$$

განვიხილოთ ფუნქციათა შემდეგი სისტემა

$$\varphi_1\,,\,\varphi_2\,,\,\ldots\,,\,\varphi_n\,,\,\ldots$$

და

$$\tilde{P}_1\,,\,\tilde{P}_2\,,\,\ldots\,,\,\tilde{P}_n\,,\,\ldots$$

შემოვიღოთ შემდეგი აღნიშვნები:

$$\Phi_{n(1)}=\sum_{k=1}^{n}c_k^{(1)}\varphi_k\,,\quad \text{სადაც}\quad c_k^{(1)}=\int\limits_{-1}^{1}y'\tilde{P}_k\left(x\right)dx\,;$$

$$\Phi_n=\sum_{k=1}^{n}a_k\varphi_k\,,\quad \text{სადაც}\, a_k\, \text{არის რიტცის მოჭრილი სისტემის ამონახსნი}\,.$$

ცხადია ადგილი ექნება შემდეგ უტოლობას

$$y-\Phi_{n(1)}=\int\limits_{-1}^{x}\left(y'-{\Phi'}_{n(1)}\right)dt=\int\limits_{-1}^{x}\left[y'\left(t\right)-S_{n(1)}\left(t\right)\right]dt\,;$$

ამ უტოლობის შემდეგ ადვილად გამოდის შემდეგი უტოლობა

$$\begin{aligned}&\left\|y-\Phi_{n(1)}\right\|^2=\int\limits_{-1}^{1}\left\{\int\limits_{-1}^{x}\left(y'-{\Phi'}_{n(1)}\right)dt\right\}^2dx\leq\\&\leq\int\limits_{-1}^{1}\left\{\int\limits_{-1}^{x}1^2dt\cdot\int\limits_{-1}^{x}\left(y'-{\Phi'}_{n(1)}\right)^2dt\right\}dx\leq\left\|y'-{\Phi'}_{n(1)}\right\|^2\int\limits_{-1}^{1}\left(x+1\right)dx=\\&=2\left\|y'-{\Phi'}_{n(1)}\right\|^2;\end{aligned}$$

ე. ი.

$$\left\|y-\Phi_{n(1)}\right\|^2\leq2\left\|y'-{\Phi'}_{n(1)}\right\|^2=2\left\|y'-S_{n(1)}\right\|^2;$$

ცხადია ადგილი აქვს უტოლობას

$$I\left(\Phi_n\right)-I\left(y\right)\leq I\left(\Phi_{n(1)}\right)-I\left(y\right)\,;$$

ან რაც იგივეა

$$\int\limits_{-1}^{1}\left[p\left({\Phi'}_n-y'\right)^2+q\left(\Phi_n-y\right)^2\right]dx\leq\int\limits_{-1}^{1}\left[p\left({\Phi'}_{n(1)}-y'\right)^2+q\left(\Phi_{n(1)}-y\right)^2\right]dx\,;$$

უკანასკნელი უტოლობიდან მიიღება

$$\min p\left\|\Phi'_n - y'\right\|^2 \le \max p\left\|\Phi'_{n(1)} - y'\right\|^2 + \max q\left\|\Phi_{n(1)} - y\right\|^2;$$
$$\min p\left\|\Phi'_n - y'\right\|^2 \le \max p\left\|y' - \Phi'_{n(1)}\right\|^2 + 2\max q\left\|y' - \Phi'_{n(1)}\right\|^2;$$

საბოლოოდ

$$\left\|\Phi'_n - y'\right\|^2 \le \frac{1}{\min p}\left[\max p + 2\max q\right]\left\|y' - \Phi'_{n(1)}\right\|^2;$$

აქედან

$$\left\|\Phi'_n - y'\right\| \le \sqrt{\frac{1}{\min p}\left[\max p + 2\max q\right]} \cdot \left\|y' - \Phi'_{n(1)}\right\|;$$

ან რაც იგივეა

$$\left\|\Phi'_n - y'\right\| \le c_2\left\|y' - \Phi'_{n(1)}\right\|;$$
$$\left\|\Phi'_n - y'\right\| \le c_2\left\|y' - S_{n(1)}\right\|;$$
$$\text{სადაც} \quad c_2 = \sqrt{\frac{1}{\min p}\left[\max p + 2\max q\right]};$$

(1.151)–დან გვაქვს

$$\max\left|\Phi_n - y\right| \le \sqrt{2}\left\|\Phi'_n - y'\right\| \le c_2\sqrt{2}\left\|y' - S_{n(1)}\right\|;$$

ე. ი.

$$\max\left|\Phi_n - y\right| \le c_2\sqrt{2}\left\|y' - S_{n(1)}\right\|;$$

უკანასკნელი უტოლობა (1.158)–თან ერთად მოგვცემს

$$\max\left|\Phi_n - y\right| \le 4c_2A_nA_{n+2}\sqrt{2}\left\|y'' - S_{n-1(2)}\right\| \le 4c_2A_nA_{n+2}\sqrt{2}\left\|y''\right\|;$$
$$\max\left|\Phi_n - y\right| \le 4c_2A_nA_{n+2}\sqrt{2}\left\|y''\right\|;$$

ეს კი თავის მხრივ (1.155)–თან ერთად მოგვცემს

$$\max\left|\Phi_n - y\right| \le 4c_1c_2A_nA_{n+2}\sqrt{2}\left\|f\right\|;$$

უკანასკნელი უტოლობა გაშლილი სახით შეგვიძლია ჩავწეროთ შემდეგნაირად. პირველ რიგში გარდავქმნათ შემდეგი ნამრავლი:

$$4A_nA_{n+2}\sqrt{2} = \frac{2\sqrt{2}}{\sqrt{(2n+1)(2n+5)}};$$

ე. ი.

$$\max\left|\Phi_n - y\right| \le \frac{c_0}{\sqrt{(2n+1)(2n+5)}}\left\|f\right\|;$$

სადაც:

$$c_0 = 2c_1c_2\sqrt{2};$$

$$c_1 = \frac{1}{(\min p)^2}\left[\sqrt{2}\max\left|p'\right| + 2\max q + \min p\right]\;;$$

$$c_2 = \sqrt{\frac{1}{\min p}\left[\max p + 2\max q\right]}\,.$$

თავი 2

ვარიაციულ-სხვაობიანი სქემა კირხოფის სივრცით ორგანზომილებიანი არაწრფივი დინამიური განტოლებისათვის

2.1 ვარიაციულ-სხვაობიანი სქემა

2.1.1 მიახლოებითი ამონახსნის აგება საბაზისო ფუნქციებად სინუსების ნამრავლის გამოყენებით

მოცემულია განტოლება:

$$\begin{gathered}\frac{\partial^2 u\left(x,y,t\right)}{\partial t^2}-\left(\alpha+\beta\int\limits_0^1\int\limits_0^1\left[\left(\frac{\partial u\left(x,y,t\right)}{\partial x}\right)^2+\left(\frac{\partial u\left(x,y,t\right)}{\partial y}\right)^2\right]dxdy\right)\Delta u\left(x,y,t\right)=\\ =f\left(x,y,t\right)\,,\quad (x,y,t)\in\left]0,1\right[^2\times\left]0,T\right]\,,\end{gathered}\tag{2.1}$$

სადაც, $\alpha>0$ და $\beta>0$.
საწყისი პირობები:

$$u\left(x,y,0\right)=\psi_0\left(x,y\right)\,,\tag{2.2}$$

$$u'_t\left(x,y,0\right)=\psi_1\left(x,y\right)\,.\tag{2.3}$$

სასაზღვრო პირობები:

$$u\left(0,y,t\right)=u\left(1,y,t\right)=0\,,\quad u\left(x,0,t\right)=u\left(x,1,t\right)=0\,.\tag{2.4}$$

ამასთან სრულდება შეთანხმებულობის პირობა:

$$\psi_0\left(0,y\right)=\psi_0\left(1,y\right)=0\,,\quad \psi_0\left(x,0\right)=\psi_0\left(x,1\right)=0\,.$$

(2.1)–(2.4) ამოცანისათვის დავწეროთ შესაბამისი ვარიაციულ-სხვაობიანი სქემა.

$t \in [0, T]$ დავყოთ M თანაბარ ნაწილად, რომლის ბიჯის სიგრძეა τ.

$$0 = t_0 < t_1 < \cdots < t_M = T\,,$$

სადაც,

$$t_k = k\tau, \quad (k = 0, 1, \ldots, M)\,, \quad \tau = \frac{T}{M}\,,$$

$$\begin{aligned}&\frac{u_{k+1}(x,y) - 2u_k(x,y) + u_{k-1}(x,y)}{\tau^2} - \frac{1}{2} q_k \left(\Delta u_{k+1}(x,y) + \Delta u_{k-1}(x,y)\right) = \\ &= f_k(x,y)\,,\end{aligned} \tag{2.5}$$

სადაც

$$q_k = \alpha + \beta \int\limits_0^1 \int\limits_0^1 \left[\left(\frac{\partial u_k(x,y)}{\partial x}\right)^2 + \left(\frac{\partial u_k(x,y)}{\partial y}\right)^2\right] dxdy\,, \quad (k = 1, 2, \ldots, M-1)\,,$$

(2.5)-დან მივიღებთ:

$$2u_{k+1}(x,y) - \tau^2 q_k \Delta u_{k+1}(x,y) = 2\tau^2 f_k(x,y) + 4u_k(x,y) + \tau^2 q_k \Delta u_{k-1}(x,y) - 2u_{k-1}(x,y)\,.$$

თუ შემოვიღებთ შემდეგ აღნიშვნას

$$g_k(x,y) = 2\tau^2 f_k(x,y) + 4u_k(x,y) - \left(2I - \tau^2 q_k \Delta\right) u_{k-1}(x,y)\,, \tag{2.6}$$

მაშინ მივიღებთ შემდეგ განტოლებას:

$$\left(2I - \tau^2 q_k \Delta\right) u_{k+1}(x,y) = g_k(x,y)\,. \tag{2.7}$$

საძებნი ფუნქციის მნიშვნელობები ნულოვან და პირველ შრეზე განისაზღვრება საწყისი პირობებისა და (2.1) განტოლების საშუალებით:

$$u_0(x,y) = \psi_0(x,y)\,, \tag{2.8}$$

$$u_1(x,y) = \psi_0(x,y) + \tau\psi_1(x,y) + \frac{1}{2}\tau^2 \left(q_0 \Delta\psi_0(x,y) + f_0(x,y)\right)\,. \tag{2.9}$$

სასაზღვრო პირობები:

$$u_k(0,y) = u_k(1,y) = 0\,, \quad u_k(x,0) = u_k(x,1) = 0\,. \tag{2.10}$$

(2.7) – (2.10) ამოცანის ამონახსნი ვეძებოთ შემდეგი სახით:

$$u_k(x,y) = \sum_{m=1}^{N}\sum_{n=1}^{N} c_{m,n}^{k}\varphi_m(x)\,\varphi_n(y)\,. \tag{2.11}$$

საბაზისო ფუნქციებად ვიღებთ

$$\varphi_m(x) = \sin(m\pi x) \quad \text{და} \quad \varphi_n(y) = \sin(n\pi y)\ ,$$

ყოველ $(k+1)$-ე შრეზე $c_{m,n}^{k+1}$ $(k = 1, 2, \ldots, M-1)$ კოეფიციენტები უნდა ვიპოვნოთ შემდეგი პირობიდან:

$$\left(\left(2I - \tau^2 q_k\Delta\right)u_{k+1}(x,y) - g_k(x,y)\,, \varphi_m(x)\,\varphi_n(y)\right) = 0\,, \tag{2.12}$$

(2.11) ჩავსვათ (2.12), მივიღებთ:

$$\left(\sum_{i=1}^{N}\sum_{j=1}^{N} c_{i,j}^{k+1}\left(2I - \tau^2 q_k\Delta\right)\varphi_i(x)\,\varphi_j(y) - g_k(x,y)\,, \varphi_m(x)\,\varphi_n(y)\right) = 0$$

საბოლოოდ,

$$\begin{aligned}&\left(\sum_{i=1}^{N}\sum_{j=1}^{N} c_{i,j}^{k+1}\left(2I - \tau^2 q_k\Delta\right)\varphi_i(x)\,\varphi_j(y), \varphi_m(x)\,\varphi_n(y)\right) = \\ &= \left(g_k(x,y)\,, \varphi_m(x)\,\varphi_n(y)\right),\end{aligned} \tag{2.13}$$

(1.13)-ის თანახმად, ჩვენ მივიღებთ:

$$\left(\varphi_i(x)\,\varphi_j(y)\,, \varphi_m(x)\,\varphi_n(y)\right) = \frac{1}{4}\delta_{im}\delta_{jn}\,, \tag{2.14}$$

ამასთან

$$\Delta\varphi_i(x)\,\varphi_j(y) = -\pi^2\left(i^2 + j^2\right)\varphi_i(x)\,\varphi_j(y)\ , \tag{2.15}$$

(2.15)-ის გათვალისწინებით:

$$\left(2I - \tau^2 q_k\Delta\right)\varphi_i(x)\,\varphi_j(y) = \left(2 + \pi^2\left(i^2 + j^2\right)\tau^2 q_k\right)\varphi_i(x)\,\varphi_j(y)\ , \tag{2.16}$$

თუ გამოვიყენებთ ნაწილობით ინტეგრებას და გავითვალისწინებთ (2.4) – ს მივიღებთ:

$$\int_0^1\int_0^1 \left[\left(\frac{\partial u_k(x,y)}{\partial x}\right)^2 + \left(\frac{\partial u_k(x,y)}{\partial y}\right)^2\right]dxdy = -\int_0^1\int_0^1 \Delta u_k(x,y)u_k(x,y)dxdy\,, \tag{2.17}$$

(2.11)-ის, (2.14), (2.17)-ისა და (2.15)-ის გათვალისწინებით:

$$q_k = \alpha + \left(\frac{\pi}{2}\right)^2\beta\sum_{m=1}^{N}\sum_{n=1}^{N}\left(m^2 + n^2\right)\left(c_{m,n}^{k}\right)^2, \tag{2.18}$$

(2.16)-ის გათვალისწინებით მივიღებთ:

$$\begin{gathered}\left(\sum_{i=1}^{N}\sum_{j=1}^{N}c_{i,j}^{k+1}\left(2I-\tau^2 q_k\Delta\right)\varphi_i\left(x\right)\varphi_j\left(y\right),\varphi_m\left(x\right)\varphi_n\left(y\right)\right)= \\ =\frac{2+\pi^2\left(m^2+n^2\right)\tau^2 q_k}{4}c_{m,n}^{k+1},\end{gathered} \tag{2.19}$$

ახლა გადავიდეთ (2.13) განტოლების მარჯვენა მხარეში შემავალი ფუნქციებისთვის ინტეგრალების გამოთვალზე. $f_k\left(x,y\right)$, $\psi_0\left(x,y\right)$ და $\psi_1\left(x,y\right)$ ფუნქციების გამოყენებულია სიმპსონის განზოგადებული ფორმულა ორჯერადი ინტეგრალებისათვის, რომელიც მოყვანილია დამტკიცების გარეშე. უშუალოდ დამტკიცების იდეა ეყრდნობა სიმპოსნის ფორმულის დამტკიცებას ერთი ცვლადისათვის.

სიმპსონის განზოგადებული ფორმულა

თეორემა 2.1.1 (სიმპსონის განზოგადებული ფორმულა ორმაგი ინტეგრალისათვის). *განვიხილოთ* $z=f\left(x,y\right)$, $R=\{(x,y): a\le x\le b,\, c\le y\le d\}$ *მართკუთხედზე. მოცემული* $[a,b]$ *და* $[c,d]$ *შუალედი დავყოთ შესაბამისად* $2m$, $\{[x_{i-1},x_i]\}_{i=1}^{2m}$ *და* $2n$, $\{[y_{j-1},y_j]\}_{j=1}^{2n}$ *ქვეშუალედებად, რომელთა ბიჯის სიგრძეებია* $h=\frac{b-a}{2m}$ *და* $\ell=\frac{d-c}{2n}$. *კვანძითი წერტილები გამოითვლება შემდეგნაირად* $x_i=x_0+ih$ *და* $y_j=y_0+j\ell$, *სადაც* $i=1,2,\ldots,2m$ *და* $j=1,2,\ldots,2n$.

შედგენილ სიმპსონის ფორმულას აქვს სახე:

$$\int\limits_a^b\int\limits_c^d f\left(x,y\right)dxdy\approx S_{2D}\left(f,h,\ell\right),$$

$$S_{2D}\left(f,h,\ell\right)=\frac{\left(b-a\right)\left(d-c\right)}{36mn}\sum_{i=0}^{2m}\sum_{j=0}^{2n}w_{i+1,j+1}f\left(x_i,y_j\right), \tag{2.20}$$

სადაც

$$W=\begin{pmatrix}1&4&2&4&2&4&\cdots&4&1\\4&16&8&16&8&16&\cdots&16&4\\2&8&4&8&4&8&\cdots&8&2\\4&16&8&16&8&16&\cdots&16&4\\\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\ldots&\vdots&\vdots\\4&16&8&16&8&16&\cdots&16&4\\1&4&2&4&2&4&\cdots&4&1\end{pmatrix}, \tag{2.20.a}$$

შევნიშნოთ, რომ W *მატრიცს აქვს* $(2m+1)$ *სტრიქონი და* $(2n+1)$ *სვეტი.*
ამასთან ცდომილებას აქვს შემდეგი სახე:

$$E_{2D}\left(f,h,\ell\right)=\mathcal{O}\left(h^4\right)+\mathcal{O}\left(\ell^4\right),$$

ახლა დავუბრუნდეთ ჩვენ ამოცანას. დავუშვათ $f_k\left(x,y\right)$, $\psi_0\left(x,y\right)$ და $\psi_1\left(x,y\right)\in C^4\left([0,1]\times[0,1]\right)$. ვიგულისხმოდ, რომ $x\in[0,1]$ და $y\in[0,1]$ დაყოფათა r და s რიცხვები ლუ-

წია. თუ დაყოფათა რიცხვებიდან r ან s კენტია, მაშინ შემოვიღოთ ახალი დაყოფა, რომელიც მიიღება არსებული დაყოფის ერთით გადიდებით (ე. ი. $r_1 = r + 1$ ან $s_1 = s + 1$).

შემოვიღოთ შემდეგი აღნიშვნები:

$$F_{m,n}^{k}(x,y) = f_k(x,y)\,\varphi_m(x)\,\varphi_n(y)\ , \tag{2.21}$$

$$\Psi_{m,n}^{0}(x,y) = \psi_0(x,y)\,\varphi_m(x)\,\varphi_n(y)\ , \tag{2.22}$$

$$\Psi_{m,n}^{1}(x,y) = \psi_1(x,y)\,\varphi_m(x)\,\varphi_n(y)\ , \tag{2.23}$$

(2.21), (2.22) და (2.23) აღნიშვნების გათვალისწინებით გამოვთვალოთ შემდეგი სკალარული ნამრავლები:

$$\begin{aligned} S_{2D}\left(F_{m,n}^{k}\right) &= \left(f_k(x,y)\,,\varphi_m(x)\,\varphi_n(y)\right) = \\ &= \int_0^1\int_0^1 F_{m,n}^{k}(x,y)dxdy \approx \frac{1}{9rs}\sum_{i=0}^{r}\sum_{j=0}^{s} W_{i+1,j+1}F_{m,n}^{k}(x_i,y_j) \end{aligned} \tag{2.24}$$

$$\begin{aligned} S_{2D}\left(\Psi_{m,n}^{0}\right) &= \left(\psi_0(x,y)\,,\varphi_m(x)\,\varphi_n(y)\right) = \\ &= \int_0^1\int_0^1 \Psi_{m,n}^{0}(x,y)dxdy \approx \frac{1}{9rs}\sum_{i=0}^{r}\sum_{j=0}^{s} W_{i+1,j+1}\Psi_{m,n}^{0}(x_i,y_j) \end{aligned} \tag{2.25}$$

$$\begin{aligned} S_{2D}\left(\Psi_{m,n}^{1}\right) &= \left(\psi_1(x,y)\,,\varphi_m(x)\,\varphi_n(y)\right) = \\ &= \int_0^1\int_0^1 \Psi_{m,n}^{1}(x,y)dxdy \approx \frac{1}{9rs}\sum_{i=0}^{r}\sum_{j=0}^{s} W_{i+1,j+1}\Psi_{m,n}^{1}(x_i,y_j) \end{aligned} \tag{2.26}$$

სადაც W არის (2.20.a) მატრიცი $(r + 1)$–ცალი სტრიქონითა და $(s + 1)$–ცალი სვეტით.
(2.6)-ის, (2.11)-ის, (2.14)-ის, (2.16)-ისა და (2.24)-ის გათვალისწინებით ჩვენ მივიღებთ:

$$\left(g_k(x,y)\,,\varphi_m(x)\,\varphi_n(y)\right) = 2\tau^2 S_{2D}\left(F_{m,n}^{k}\right) - \frac{2+\pi^2\left(m^2+n^2\right)\tau^2 q_k}{4}c_{m,n}^{k-1} + c_{m,n}^{k}\,, \tag{2.27}$$

საბოლოოდ (2.19) გავუტოლოთ (2.27)-ს და განვსაზღვროთ $c_{m,n}^{k+1}$ კოეფიციენტი:

$$c_{m,n}^{k+1} = \frac{8\tau^2}{2+\pi^2\left(m^2+n^2\right)\tau^2 q_k}S_{2D}\left(F_{m,n}^{k}\right) - c_{m,n}^{k-1} + \frac{4}{2+\pi^2\left(m^2+n^2\right)\tau^2 q_k}c_{m,n}^{k}\,, \tag{2.28}$$

$k = 1$ მნიშვნელობისათვის (2.28)-ში უცნობი გვექნება $c_{m,n}^{0}$ და $c_{m,n}^{1}$ კოეფიციენტები. ამ კოეფიციენტების საპოვნელად (2.8) და (2.9) გავამრავლოთ $\varphi_m(x)\,\varphi_n(y)$-ზე სკალარულად.

(2.8)-დან (2.11)-ის, (2.14)-სა და (2.25)-ის გათვალისწინებით გვექნება:

$$c_{m,n}^{0} = 4S_{2D}\left(\Psi_{m,n}^{0}\right)\ , \tag{2.29}$$

მარტივად მიიღება შემდეგი ტოლობა:

$$\left(\Delta\psi_0(x,y)\,,\varphi_m(x)\,\varphi_n(y)\right) = -\frac{\pi^2}{4}\left(m^2+n^2\right)c_{m,n}^{0}\,, \tag{2.30}$$

(2.9)-დან (2.11)-ის, (2.24)-ის, (2.25)-ის, (2.26)-ისა და (2.30)-ის გათვალისწინებით მივიღებთ:

$$c_{m,n}^{1} = \left(1 - \frac{1}{2}\pi^2\left(m^2+n^2\right)\tau^2 q_0\right)c_{m,n}^{0} + 2\tau\left(2S_{2D}\left(\Psi_{m,n}^{1}\right) + \tau S_{2D}\left(F_{m,n}^{0}\right)\right), \tag{2.31}$$

საბოლოოდ ყოველ $(k = 1, 2, \ldots, M-1)$ შრეზე გამოვთვლით (2.28) ტოლობას (2.29)-ისა და (2.31)-ის გამოყენებით, შემდგომ ჩავსვამთ (2.11) ტოლობაში.

2.1.2 მიახლოებითი ამონახსნის აგება საბაზისო ფუნქციებად ლეჟანდრის პოლინომების სხვაობების ნამრავლის გამოყენებით

მოცემულია განტოლება:

$$\begin{gathered}\frac{\partial^2 u(x,y,t)}{\partial t^2} - \left(\alpha + \beta\int_{-1}^{1}\int_{-1}^{1}\left[\left(\frac{\partial u(x,y,t)}{\partial x}\right)^2 + \left(\frac{\partial u(x,y,t)}{\partial y}\right)^2\right]dxdy\right)\Delta u(x,y,t) = \\ = f(x,y,t), \quad (x,y,t) \in \left]-1,1\right[^2 \times \left]0,T\right], \end{gathered} \tag{2.32}$$

სადაც, $\alpha > 0$ და $\beta > 0$.
საწყისი პირობები:

$$u(x,y,0) = \psi_0(x,y), \tag{2.33}$$

$$u'_t(x,y,0) = \psi_1(x,y). \tag{2.34}$$

სასაზღვრო პირობები:

$$u(-1,y,t) = u(1,y,t) = 0, \quad u(x,-1,t) = u(x,1,t) = 0. \tag{2.35}$$

ამასთან სრულდება შეთანხმებულობის პირობა:

$$\psi_0(-1,y) = \psi_0(1,y) = 0, \quad \psi_0(x,-1) = \psi_0(x,1) = 0.$$

(2.32)–(2.35) ამოცანისათვის დავწეროთ შესაბამისი ვარიაციულ-სხვაობიანი სქემა.

$t \in [0,T]$ დავყოთ M თანაბარ ნაწილად, რომლის ბიჯის სიგრძეა τ.

$$0 = t_0 < t_1 < \cdots < t_M = T,$$

სადაც,

$$t_k = k\tau, \quad (k = 0, 1, \ldots, M), \quad \tau = \frac{T}{M},$$

ვარიაციულ–სხვაობიან განტოლებას აქვს შემდეგი სახე:

$$\left(2I - \tau^2 q_k \Delta\right)u_{k+1}(x,y) = g_k(x,y). \tag{2.36}$$

სადაც:

$$q_k = \alpha + \beta \int_{-1}^{1}\int_{-1}^{1} \left[\left(\frac{\partial u_k(x,y)}{\partial x} \right)^2 + \left(\frac{\partial u_k(x,y)}{\partial y} \right)^2 \right] dxdy\,, \quad (k = 1, 2, \dots, M-1)\ ,$$
$$g_k(x,y) = 2\tau^2 f_k(x,y) + 4u_k(x,y) - \left(2I - \tau^2 q_k \Delta\right) u_{k-1}(x,y)\ .$$

საძებნი ფუნქციის მნიშვნელობები ნულოვან და პირველ შრეზე განისაზღვრება საწყისი პირობებისა და (2.1) განტოლების საშუალებით:

$$u_0(x,y) = \psi_0(x,y)\ , \tag{2.37}$$
$$u_1(x,y) = \psi_0(x,y) + \tau\psi_1(x,y) + \frac{1}{2}\tau^2 \left(q_0 \Delta \psi_0(x,y) + f_0(x,y)\right)\ . \tag{2.38}$$

სასაზღვრო პირობები:

$$u_k(-1,y) = u_k(1,y) = 0\,, \quad u_k(x,-1) = u_k(x,1) = 0\,. \tag{2.39}$$

(2.36) – (2.39) ამოცანის ამონახსნი ვეძებოთ შემდეგი სახით:

$$u_k(x,y) = \sum_{m=1}^{N}\sum_{n=1}^{N} c_{m,n}^k \varphi_m(x)\varphi_n(y) \tag{2.40}$$

საკოორდინატო ფუნქციებად აღებულია ლეჟანდრის პოლინომების სხვაობები:

$$\varphi_m(x) = A_m \left(P_{m+1}(x) - P_{m-1}(x)\right)\ , \quad \varphi_n(y) = A_n \left(P_{n+1}(y) - P_{n-1}(y)\right)\ ,$$

სადაც,

$$A_i = \frac{1}{\sqrt{2(2i+1)}}\,.$$

ყოველ $(k+1)$-ე შრეზე $c_{m,n}^{k+1}$ $(k = 1, 2, \dots, M-1)$ კოეფიციენტები უნდა ვიპოვნოთ შემდეგი პირობიდან:

$$\left(\left(2I - \tau^2 q_k \Delta\right) u_{k+1}(x,y) - g_k(x,y)\,, \varphi_m(x)\varphi_n(y)\right) = 0\,, \tag{2.41}$$

(2.40) ჩავსვათ (2.41), მივიღებთ:

$$\left(\sum_{i=1}^{N}\sum_{j=1}^{N} c_{i,j}^{k+1} \left(2I - \tau^2 q_k \Delta\right) \varphi_i(x)\varphi_j(y) - g_k(x,y)\,, \varphi_m(x)\varphi_n(y) \right) = 0\,,$$

საბოლოოდ,

$$\left(\sum_{i=1}^{N}\sum_{j=1}^{N}c_{i,j}^{k+1}\left(2I-\tau^{2}q_{k}\Delta\right)\varphi_{i}\left(x\right)\varphi_{j}\left(y\right),\varphi_{m}\left(x\right)\varphi_{n}\left(y\right)\right)= \\ =\left(g_{k}\left(x,y\right),\varphi_{m}\left(x\right)\varphi_{n}\left(y\right)\right), \tag{2.42}$$

შემოვიღოთ შემდეგი აღნიშვნები:

$$\tilde{P}_{i}\left(x\right)=\sqrt{\frac{2i+1}{2}}P_{i}\left(x\right)\ ,\quad \tilde{P}_{j}\left(y\right)=\sqrt{\frac{2j+1}{2}}P_{j}\left(x\right)\ .$$

ადვილი დასანახია, რომ $\tilde{P}_{i}\left(x\right)$ და $\tilde{P}_{j}\left(y\right)$ პოლინომთა სისტემა ორთონორმირებულია. სამართლიანია შემდეგი ინტეგრალური წარმოდგენა:

$$\varphi_{i}\left(x\right)=\int_{-1}^{x}\tilde{P}_{i}\left(t\right)dt\,, \tag{2.43}$$

$$\varphi_{j}\left(y\right)=\int_{-1}^{y}\tilde{P}_{j}\left(t\right)dt\,, \tag{2.44}$$

(2.43)-დან და (2.44)-დან გამომდინარეობს, რომ

$$\varphi'_{i}\left(x\right)=\tilde{P}_{i}\left(x\right)\ , \tag{2.43$'$}$$

$$\varphi'_{j}\left(y\right)=\tilde{P}_{j}\left(y\right)\ , \tag{2.44$'$}$$

შევნიშნოთ, რომ (2.44′) და (2.44′) ტოლობები ასევე მიიღება (1.87) თვისებიდან.

გვაქვს:

$$\left(\varphi_{i}\left(x\right)\varphi_{j}\left(y\right),\varphi_{m}\left(x\right)\varphi_{n}\left(y\right)\right)=\int_{-1}^{1}\varphi_{i}\left(x\right)\varphi_{m}\left(x\right)dx\int_{-1}^{1}\varphi_{j}\left(y\right)\varphi_{n}\left(y\right)dy \tag{2.45}$$

(1.95)-ის თანახმად

$$\int_{-1}^{1}\varphi_{i}\left(x\right)\varphi_{m}\left(x\right)dx=4A_{i}A_{m}\left(A_{i+1}A_{m+1}\delta_{i+1,m+1}-A_{i+1}A_{m-1}\delta_{i+1,m-1}- \\ -A_{i-1}A_{m+1}\delta_{i-1,m+1}+A_{i-1}A_{m-1}\delta_{i-1,m-1}\right)\ , \tag{2.45.1}$$

ანალოგიურად,

$$\int_{-1}^{1}\varphi_{j}\left(y\right)\varphi_{n}\left(y\right)dy=4A_{j}A_{n}\left(A_{j+1}A_{n+1}\delta_{j+1,n+1}-A_{j+1}A_{n-1}\delta_{j+1,n-1}- \\ -A_{j-1}A_{n+1}\delta_{j-1,n+1}+A_{j-1}A_{n-1}\delta_{j-1,n-1}\right)\ , \tag{2.45.2}$$

ადვილად მიიღება შემდეგი ტოლობა:

$$
\begin{aligned}
&\left(-\Delta\varphi_i\left(x\right)\varphi_j\left(y\right),\varphi_m\left(x\right)\varphi_n\left(y\right)\right)=\\
&=-\int\limits_{-1}^{1}\frac{d^2\varphi_i\left(x\right)}{dx^2}\varphi_m\left(x\right)dx\int\limits_{-1}^{1}\varphi_j\left(y\right)\varphi_n\left(y\right)dy-\int\limits_{-1}^{1}\frac{d^2\varphi_j\left(y\right)}{dy^2}\varphi_n\left(y\right)dy\int\limits_{-1}^{1}\varphi_i\left(x\right)\varphi_m\left(x\right)dx\,,
\end{aligned}
\tag{2.46}
$$

(2.46)-ის პირველი და მეორე შესაკრები გამოვთვალოთ ცალ-ცალკე:
(1.94)-ის თანახმად

$$
-\int\limits_{-1}^{1}\frac{d^2\varphi_i\left(x\right)}{dx^2}\varphi_m\left(x\right)dx=\delta_{im}\,,
\tag{2.46.1}
$$

და (2.46.1)-ის ანალოგიურად

$$
-\int\limits_{-1}^{1}\frac{d^2\varphi_j\left(y\right)}{dy^2}\varphi_n\left(y\right)dy=\delta_{jn}\,,
\tag{2.46.2}
$$

ახლა (2.42)-სა და (2.45)-ის გამოყენებით გამოვთვალოთ შემდეგი სკალარული ნამრავლი:

შენიშვნა 2.1.1. (2.45.1)-დან და (2.45.2)-დან გამომდინარე ნულისაგან განსხვავებული წევრები მიიღება i და j ინდექსების შემდეგი მნიშვნელობებისათვის:

$$
i=m-2,\quad i=m,\quad i=m+2,\quad j=n-2,\quad j=n\quad\text{და}\quad j=n+2.
$$

თუ გამოვიყენებთ (1.97) და (1.98) აღნიშვნებს

$$
B_m=4A_{m-1}A_m^2A_{m+1}\,,\qquad B_m=\frac{1}{\left(2m+1\right)\sqrt{\left(2m-1\right)\left(2m+3\right)}}\,,
\tag{1.97}
$$

$$
C_m=4A_m^2\left(A_{m-1}^2+A_{m+1}^2\right)\,,\qquad C_m=\frac{2}{\left(2m-1\right)\left(2m+3\right)}\,,
\tag{1.98}
$$

მივიღებთ:

$$
\begin{aligned}
&\left(u_{k+1}\left(x,y\right),\varphi_m\left(x\right)\varphi_n\left(y\right)\right)=\\
&=B_{m-1}\left(B_{n-1}c_{m-2,n-2}^{k+1}-C_nc_{m-2,n}^{k+1}+B_{n+1}c_{m-2,n+2}^{k+1}\right)-\\
&-C_m\left(B_{n-1}c_{m,n-2}^{k+1}-C_nc_{m,n}^{k+1}+B_{n+1}c_{m,n+2}^{k+1}\right)+\\
&+B_{m+1}\left(B_{n-1}c_{m+2,n-2}^{k+1}-C_nc_{m+2,n}^{k+1}+B_{n+1}c_{m+2,n+2}^{k+1}\right)\,.
\end{aligned}
\tag{2.47}
$$

შენიშვნა 2.1.2. იგივური ოპერატორი გვაძლევს ცხრა კვანძს:

$$
\begin{gathered}
\left(m-2,n-2\right),\quad\left(m-2,n\right),\quad\left(m-2,n+2\right),\\
\left(m,n-2\right),\qquad\left(m,n\right),\qquad\left(m,n+2\right),\\
\left(m+2,n-2\right),\quad\left(m+2,n\right),\quad\left(m+2,n+2\right).
\end{gathered}
$$

(2.42)-ის, (2.45.1)-ის, (2.45.2)-ის, (2.46)-ის, (2.46.1)-ის, (2.46.2)-ის ძალით და ასევე (1.97), (1.98) აღნიშვნების გათვალისწინებით მივიღებთ:

$$
\begin{aligned}
-\left(\Delta u_{k+1}(x,y), \varphi_m(x)\varphi_n(y)\right) = -\left(B_{n-1}c^{k+1}_{m,n-2} - C_n c^{k+1}_{m,n} + B_{n+1}c^{k+1}_{m,n+2}\right) - \\
-\left(B_{m-1}c^{k+1}_{m-2,n} - C_m c^{k+1}_{m,n} + B_{m+1}c^{k+1}_{m+2,n}\right).
\end{aligned} \tag{2.48}
$$

(2.42)-ის, (2.17)-ისა და (2.48)-ის გათვალისწინებით

$$
\begin{aligned}
q_k = \alpha - \beta \sum_{m=1}^{N}\sum_{n=1}^{N} c^k_{m,n}\Big(B_{m-1}c^k_{m-2,n} + B_{n-1}c^k_{m,n-2} - (C_m + C_n)\, c^k_{m,n} + \\
+B_{m+1}c^k_{m+2,n} + B_{n+1}c^k_{m,n+2}\Big),
\end{aligned} \tag{2.49}
$$

სადაც $c^k_{i,j} = 0\,, i$ და j-ის შემდეგ მნიშვნელობებზე: $i \le 0$ ან $i > N$ ან $j \le 0$ ან $j > N$.

გამოვთვალოთ (2.42) განტოლების მარცხენა მხარე. (2.47)-ისა და (2.48)-ის თანახმად, მივიღებთ:

$$
\begin{aligned}
&\left(\left(2I - \tau^2 q_k \Delta\right) u_{k+1}(x,y), \varphi_m(x)\varphi_n(y)\right) = \\
&= 2B_{m-1}B_{n-1}c^{k+1}_{m-2,n-2} - B_{m-1}\left(2C_n + \tau^2 q_k\right) c^{k+1}_{m-2,n} + 2B_{m-1}B_{n+1}c^{k+1}_{m-2,n+2} - \\
&- \left(2C_m + \tau^2 q_k\right) B_{n-1}c^{k+1}_{m,n-2} + \left[2C_mC_n + \tau^2 q_k (C_m + C_n)\right] c^{k+1}_{m,n} - \\
&- \left(2C_m + \tau^2 q_k\right) B_{n+1}c^{k+1}_{m,n+2} + 2B_{m+1}B_{n-1}c^{k+1}_{m+2,n-2} - \\
&- B_{m+1}\left(2C_n + \tau^2 q_k\right) c^{k+1}_{m+2,n} + 2B_{m+1}B_{n+1}c^{k+1}_{m+2,n+2},
\end{aligned} \tag{2.50}
$$

ახლა გამოვთვალოთ (2.42) განტოლების მარჯვენა მხარე, ამისათვის გამოვთვალოთ შემდეგი სკალარული ნამრავლი

$$
\begin{aligned}
\left(g_k(x,y), \varphi_m(x)\varphi_n(y)\right) = 2\tau^2 \left(f_k(x,y), \varphi_m(x)\varphi_n(y)\right) + \\
+ 4\left(u_k(x,y), \varphi_m(x)\varphi_n(y)\right) - \left(\left(2I - \tau^2 q_k \Delta\right) u_{k-1}(x,y), \varphi_m(x)\varphi_n(y)\right),
\end{aligned} \tag{2.51}
$$

შენიშვნა 2.1.3. ამ პარაგრაფშიც $\left(f_k(x,y), \varphi_m(x)\varphi_n(y)\right)$, $\left(\psi_0(x,y), \varphi_m(x)\varphi_n(y)\right)$ და $\left(\psi_1(x,y), \varphi_m(x)\varphi_n(y)\right)$ სკალარული ნამრავლების (ინტეგრალების) გამოსათვლელად გამოვიყენებთ სიმპსონის განზოგადებულ ფორმულას ორჯერადი ინტეგრალისათვის. გამოვიყენებთ პარაგრაფ 2.1.1-ში შემოთანილ აღნიშვნებს.

(2.24)-ის გათვალისწინებით

$$
\left(f_k(x,y), \varphi_m(x)\varphi_n(y)\right) = S_{2D}\left(F^k_{m,n}\right), \tag{2.51.1}
$$

საბოლოოდ თუ (2.51) ტოლობას გამოვთვლით (2.51.1)-ის, (2.47)-სა და (2.50) ფორმულის გამოყენებით და მიღებულს გავუტოლებთ (2.50) მივიღებთ:

$$
\begin{aligned}
&2B_{m-1}B_{n-1}c^{k+1}_{m-2,n-2} - B_{m-1}\left(2C_n + \tau^2 q_k\right) c^{k+1}_{m-2,n} + 2B_{m-1}B_{n+1}c^{k+1}_{m-2,n+2} - \\
&- \left(2C_m + \tau^2 q_k\right) B_{n-1}c^{k+1}_{m,n-2} + \left[2C_mC_n + \tau^2 q_k (C_m + C_n)\right] c^{k+1}_{m,n} - \left(2C_m + \tau^2 q_k\right) B_{n+1}c^{k+1}_{m,n+2} + \\
&+ 2B_{m+1}B_{n-1}c^{k+1}_{m+2,n-2} - B_{m+1}\left(2C_n + \tau^2 q_k\right) c^{k+1}_{m+2,n} + 2B_{m+1}B_{n+1}c^{k+1}_{m+2,n+2} =
\end{aligned}
$$

$$
\begin{aligned}
&= 2\tau^2 S_{2D}\left(F^k_{m,n}\right) + 4B_{m-1}\left(B_{n-1}c^k_{m-2,n-2} - C_n c^k_{m-2,n} + B_{n+1}c^k_{m-2,n+2}\right) - \\
&- 4C_m\left(B_{n-1}c^k_{m,n-2} - C_n c^k_{m,n} + B_{n+1}c^k_{m,n+2}\right) + \\
&+ 4B_{m+1}\left(B_{n-1}c^k_{m+2,n-2} - C_n c^k_{m+2,n} + B_{n+1}c^k_{m+2,n+2}\right) - \\
&- 2B_{m-1}B_{n-1}c^{k-1}_{m-2,n-2} + B_{m-1}\left(2C_n + \tau^2 q_k\right)c^{k-1}_{m-2,n} - 2B_{m-1}B_{n+1}c^{k-1}_{m-2,n+2} + \\
&+ \left(2C_m + \tau^2 q_k\right)B_{n-1}c^{k-1}_{m,n-2} - \left[2C_mC_n + \tau^2 q_k\left(C_m + C_n\right)\right]c^{k-1}_{m,n} + \left(2C_m + \tau^2 q_k\right)B_{n+1}c^{k-1}_{m,n+2} - \\
&- 2B_{m+1}B_{n-1}c^{k-1}_{m+2,n-2} + B_{m+1}\left(2C_n + \tau^2 q_k\right)c^{k-1}_{m+2,n} - 2B_{m+1}B_{n+1}c^{k-1}_{m+2,n+2}\,,
\end{aligned}
\tag{2.52}
$$

ყოველი $k = 1, 2, \ldots, M-1$ შრისათვის ამოვხსნათ (2.52) წრფივ ალგებრულ განტოლებათა სისტემა. $k = 1$ შრისათვის ჩვენ გვექნება უცნობები $c^0_{m,n}$ და $c^1_{m,n}$. ვიპოვნოთ ეს კოეფიციენტები.

(2.8)-ის თანახმად

$$
\left(u_0\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right) = \left(\psi_0\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right)\,, \tag{2.53}
$$

(2.53)-დან (2.25)-ისა და (2.47)-ის გათვალისწინებით მივიღებთ:

$$
\begin{aligned}
&B_{m-1}\left(B_{n-1}c^0_{m-2,n-2} - C_n c^0_{m-2,n} + B_{n+1}c^0_{m-2,n+2}\right) - \\
&- C_m\left(B_{n-1}c^0_{m,n-2} - C_n c^0_{m,n} + B_{n+1}c^0_{m,n+2}\right) + \\
&+ B_{m+1}\left(B_{n-1}c^0_{m+2,n-2} - C_n c^0_{m+2,n} + B_{n+1}c^0_{m+2,n+2}\right) = S_{2D}\left(\Psi^0_{m,n}\right)\,,
\end{aligned}
\tag{2.54}
$$

(2.9)-ისგან მივიღებთ

$$
\begin{aligned}
&\left(u_1\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right) = \left(\psi_0\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right) + \tau\left(\psi_1\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right) + \\
&+ \frac{1}{2}\tau^2\left(q_0\left(\Delta\psi_0\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right) + \left(f_0\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right)\right)\,,
\end{aligned}
\tag{2.55}
$$

(2.25)-სა და (2.26)-ის თანახმად:

$$
\left(\psi_0\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right) = S_{2D}\left(\Psi^0_{m,n}\right)\,, \tag{2.55.1}
$$

$$
\left(\psi_1\left(x,y\right), \varphi_m\left(x\right)\varphi_n\left(y\right)\right) = S_{2D}\left(\Psi^1_{m,n}\right)\,, \tag{2.55.2}
$$

(2.55) ტოლობა საბოლოოდ მიიღებს შემდეგ სახეს:

$$
\begin{aligned}
&B_{m-1}\left(B_{n-1}c^1_{m-2,n-2} - C_n c^1_{m-2,n} + B_{n+1}c^1_{m-2,n+2}\right) - \\
&- C_m\left(B_{n-1}c^1_{m,n-2} - C_n c^1_{m,n} + B_{n+1}c^1_{m,n+2}\right) + \\
&+ B_{m+1}\left(B_{n-1}c^1_{m+2,n-2} - C_n c^1_{m+2,n} + B_{n+1}c^1_{m+2,n+2}\right) = \\
&= S_{2D}\left(\Psi^0_{m,n}\right) + \tau S_{2D}\left(\Psi^1_{m,n}\right) + \\
&+ \frac{1}{2}\tau^2 q_0\left(B_{n-1}c^0_{m,n-2} - C_n c^0_{m,n} + B_{n+1}c^0_{m,n+2}\right) + \\
&+ \frac{1}{2}\tau^2 q_0\left(B_{m-1}c^0_{m-2,n} - C_m c^0_{m,n} + B_{m+1}c^0_{m+2,n}\right) + \frac{1}{2}\tau^2 S_{2D}\left(F^0_{m,n}\right)\,.
\end{aligned}
\tag{2.56}
$$

(2.54) და (2.56) წრფივ ალგებრულ განტოლებათა სისტემიდან ვიპოვნით $c^0_{m,n}$ და $c^1_{m,n}$ კოეფიციენტებს, შემდგომ ყოველი $k = 1, 2, \ldots, M-1$ შრისათვის ამოვხსნით (2.52) სისტემას და

ყოველ შრეზე მიღებულ კოეფიციენტებს ჩავსვამთ (2.11)-ში.

თავი 3

რიცხვითი რეალიზაციების შედეგები

3.1 რიცხვითი რეალიზაციის შედეგები სხვადასხვა ტესტური ამოცანებისათვის

განვიხილოთ კირხოფის არაწრფივი დინამიური განტოლებისათვის ტესტური მაგალითები, რომლის მიახლოებით ამონახსნს ვეძებთ ვარიაციულ–სხვაობიანი მეთოდით.

- განვიხილოთ სივრცით ერთ ცვლადიანი შემთხვევა, სადაც საკოორდინატო ფუნქციებად აღებულია სინუსები. ამასთან $\alpha = \beta = 1$ და $t \in [0; 2]$.

განვიხილოთ შემთხვევა, როდესაც ზუსტ ამონახსნს აქვს შემდეგი სახე:

1. $u\,(x,t) = \sin(3\pi x)(1-t)\,;$

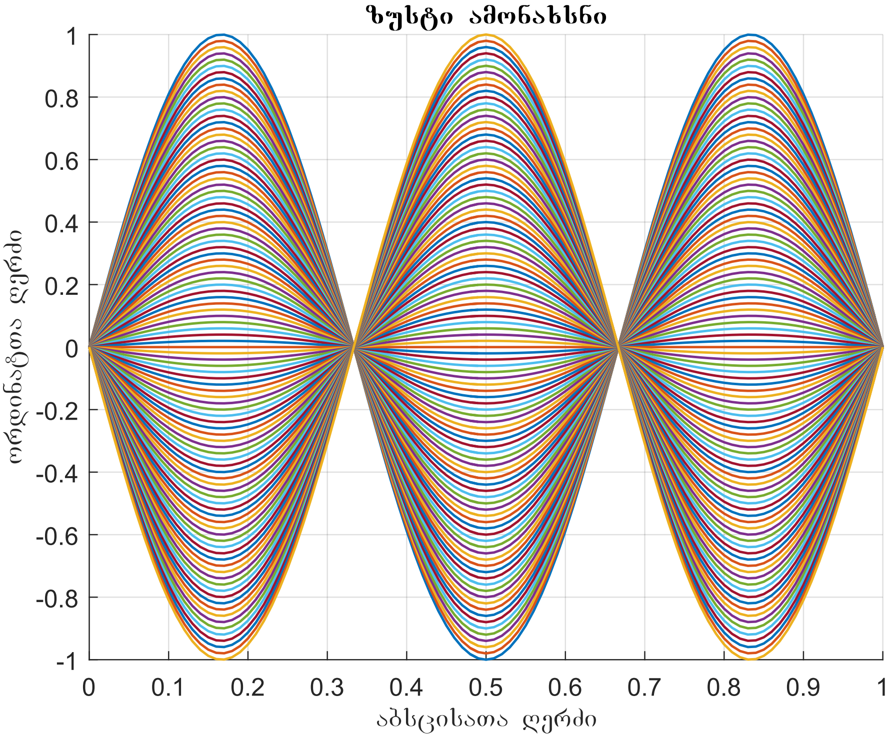
$x \in [0; 1]$ და $t \in [0; 2]$ სეგმენტები დავყოთ შესაბამისად $n = 100$ და $M = 100$ ტოლ ნაწილებად, ე. ი. $h = 0.01$ და $\tau = 0.02\,.$ საკოორდინატო ფუნქციების რაოდენობა $N = 3\,.$
მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}\,(x) = \sum_{m=1}^{3} c_m^{k+1} \sin\,(m\pi x)\,; \quad k = 1, \ldots, 99\,.$$

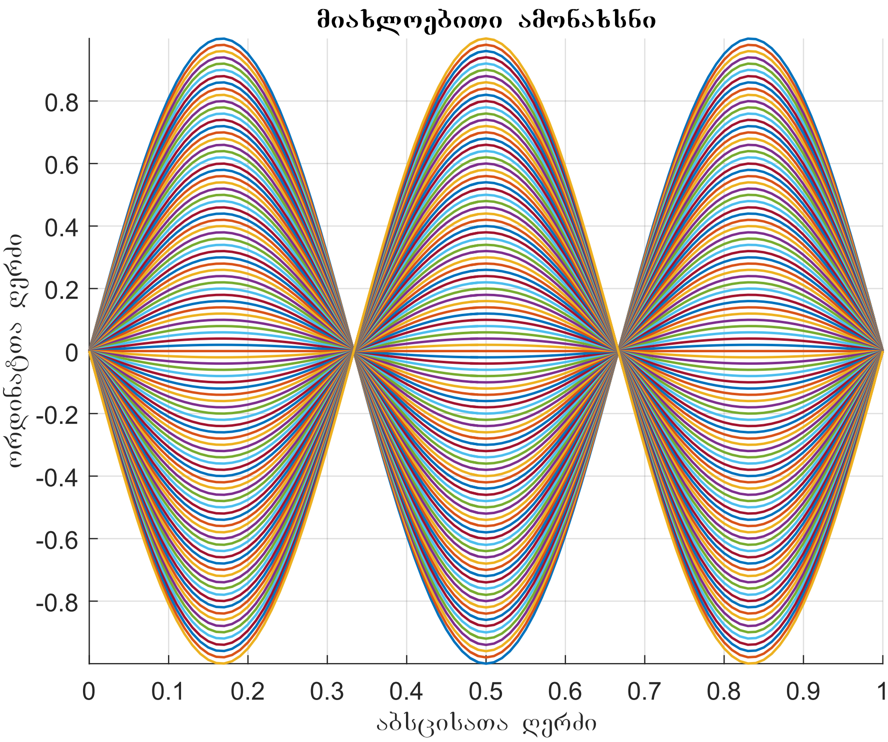
ცდომილება გვექნება:

$$\max\left|\tilde{u}_k\,(x) - u_k\,(x)\right| \approx 2.67 \times 10^{-7}\,;$$

დროის ყოველ შრეზე ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



სურ 3.1



სურ 3.2

2. $u(x,t) = \sin(2\pi x)\sin(3\pi t)$;

$x \in [0;1]$ და $t \in [0;2]$ სეგმენტები დავყოთ შესაბამისად $n = 100$ და $M = 100$ ტოლ ნაწილებად,

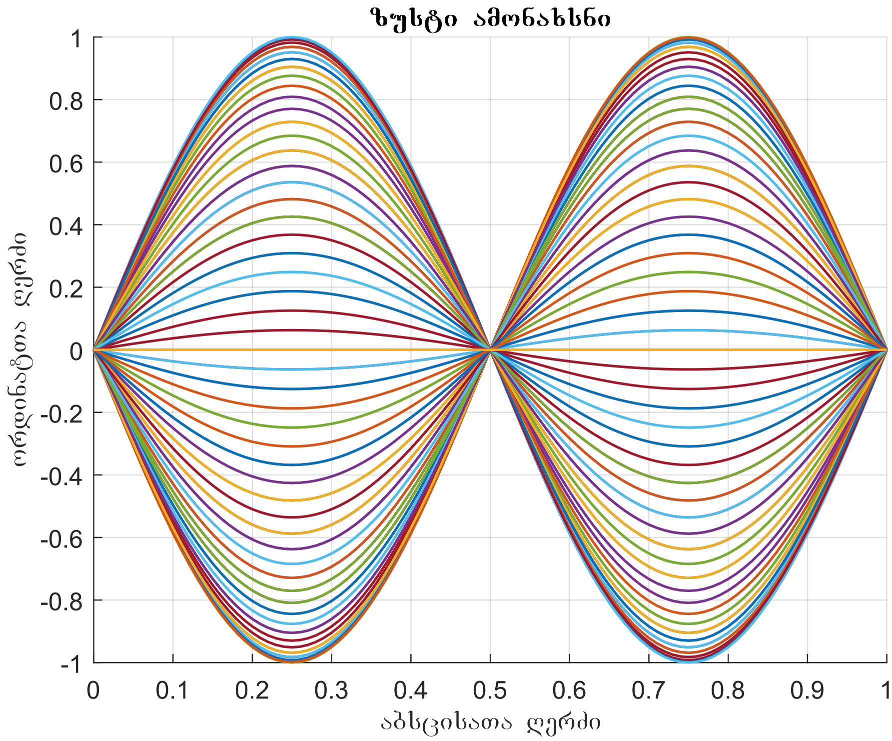
ე. ი. $h = 0.01$ და $\tau = 0.02$. საკოორდინატო ფუნქციების რაოდენობა $N = 2$.
მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}(x) = c_1^{k+1} \sin(\pi x) + c_2^{k+1} \sin(2\pi x) \; ; \quad k = 1, \ldots, 99 \, .$$

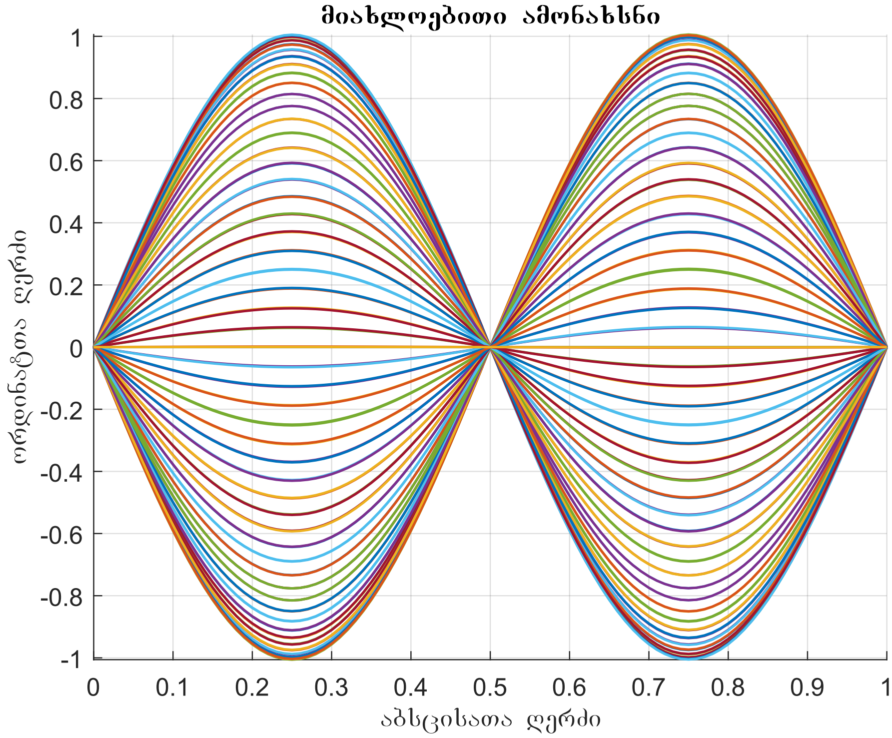
ცდომილება გვექნება:

$$\max |\tilde{u}_k(x) - u_k(x)| \approx 7.21 \times 10^{-3} \, ;$$

დროის ყოველ შრეზე ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



ნახ 3.3



სურ 3.4

3. $u(x,t) = x(1-x)\sin\left(\frac{3}{5}\pi x\right)e^{t}$;

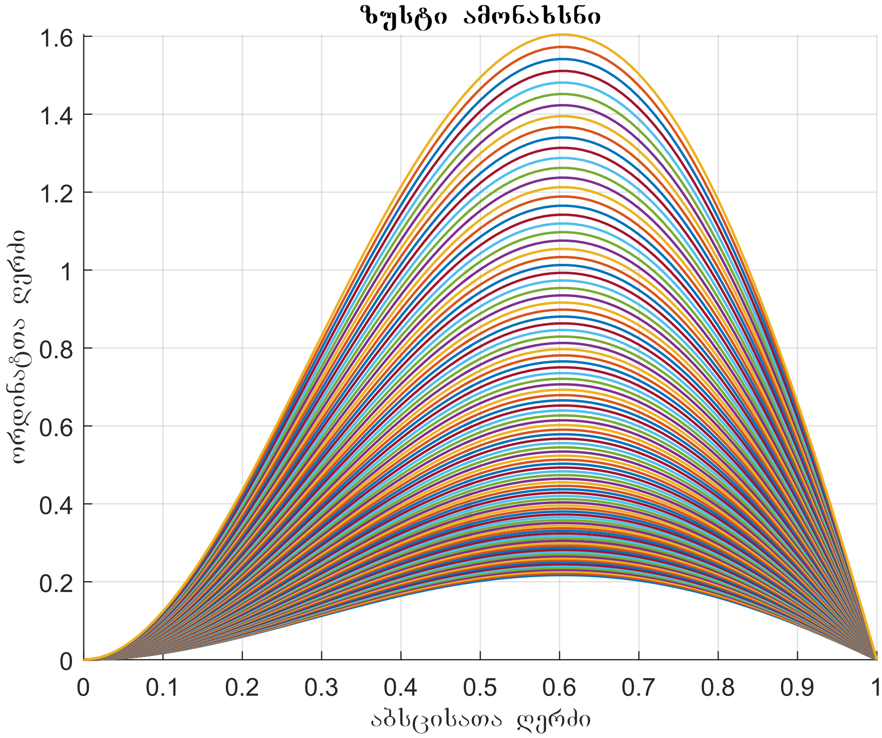
$x \in [0;1]$ და $t \in [0;2]$ სეგმენტები დავყოთ შესაბამისად $n = 100$ და $M = 100$ ტოლ ნაწილებად, ე. ი. $h = 0.01$ და $\tau = 0.02$. საკოორდინატო ფუნქციების რაოდენობა $N = 10$.
მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}(x) = \sum_{m=1}^{10} c_m^{k+1}\sin(m\pi x); \quad k = 1,\ldots,99.$$

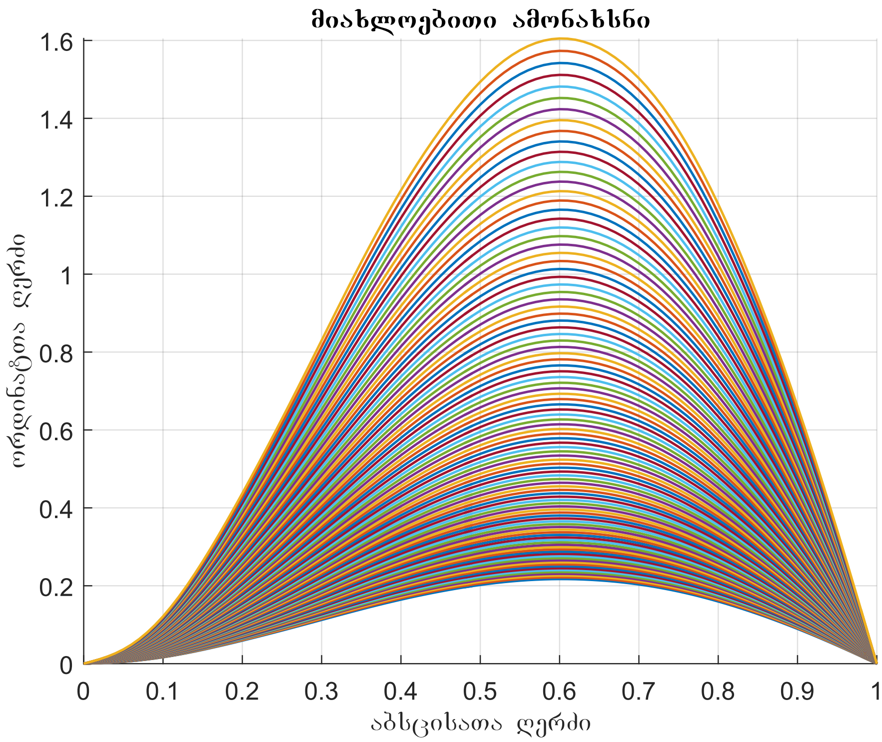
ცდომილება გვექნება:

$$\max\left|\tilde{u}_k(x) - u_k(x)\right| \approx 6.06\times 10^{-3};$$

დროის ყოველ შრეზე ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



სურ 3.5



სურ 3.6

- განვიხილოთ სივრცით ერთ ცვლადიანი შემთხვევა, სადაც საკოორდინატო ფუნქციებად აღებულია ლეჟანდრის პოლინომების სხვაობა. ამასთან $\alpha = \beta = 1$ და $t \in [0; 2]$.

განვიხილოთ შემთხვევა, როდესაც ზუსტ ამონახსნს აქვს შემდეგი სახე:

1. $u\left(x,t\right)=\varphi_3\left(x\right)\left(1-t\right)=\frac{1}{\sqrt{14}}\left(P_4\left(x\right)-P_2\left(x\right)\right)\left(1-t\right)$;

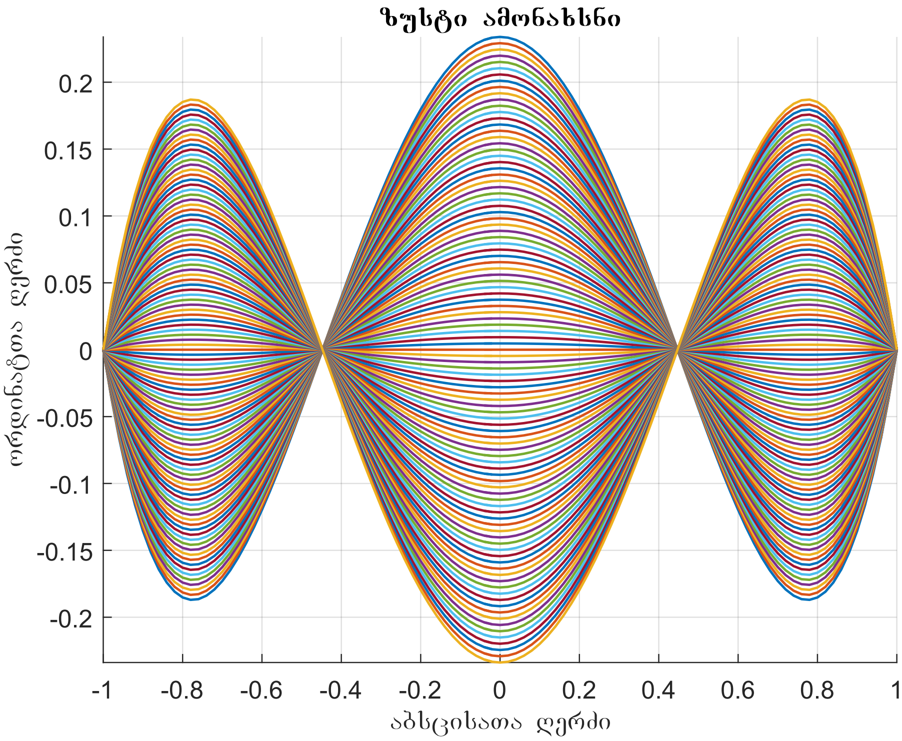
$x\in\left[-1;1\right]$ და $t\in\left[0;2\right]$ სეგმენტები დავყოთ შესაბამისად $n=100$ და $M=100$ ტოლ ნაწილებად, ე. ი. $h=0.01$ და $\tau=0.02$. საკოორდინატო ფუნქციების რაოდენობა $N=5$.
მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}\left(x\right)=\sum_{m=1}^{5}c_m^{k+1}\frac{1}{\sqrt{2\left(2m+1\right)}}\left(P_{m+1}\left(x\right)-P_{m-1}\left(x\right)\right);\quad k=1,\ldots,99\,.$$

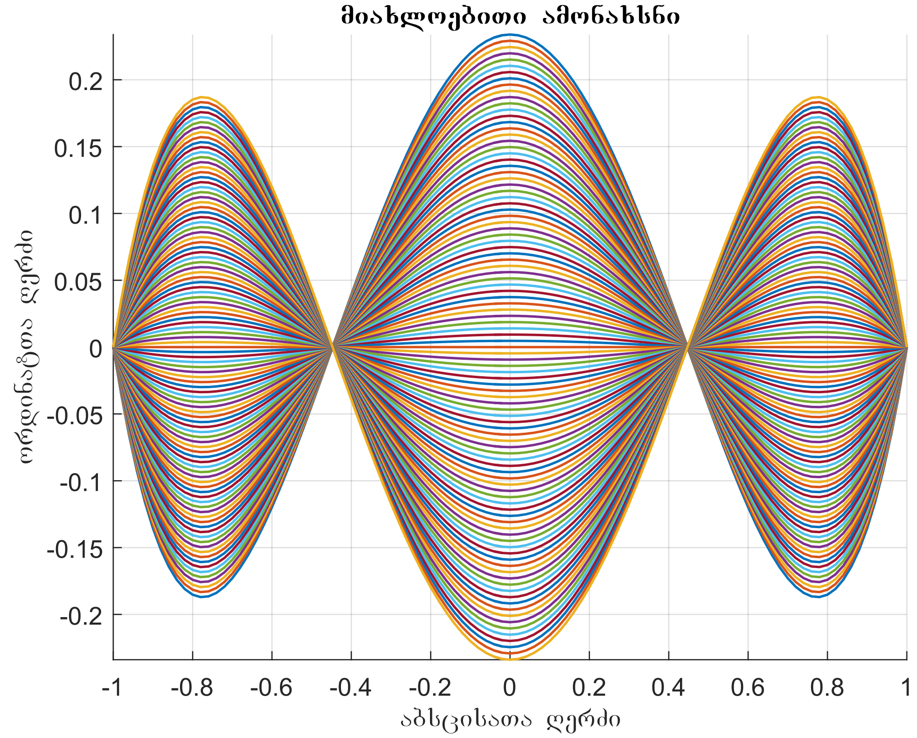
ცდომილება გვექნება:

$$\max\left|\tilde{u}_k\left(x\right)-u_k\left(x\right)\right|\approx 1.17\times 10^{-8}\,;$$

დროის ყოველ შრეზე ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



სურ 3.7



სურ 3.8

2. $u(x,t) = \varphi_2(x)\sin(3\pi t) = \frac{1}{\sqrt{10}}\left(P_3(x) - P_1(x)\right)\sin(3\pi t)$;

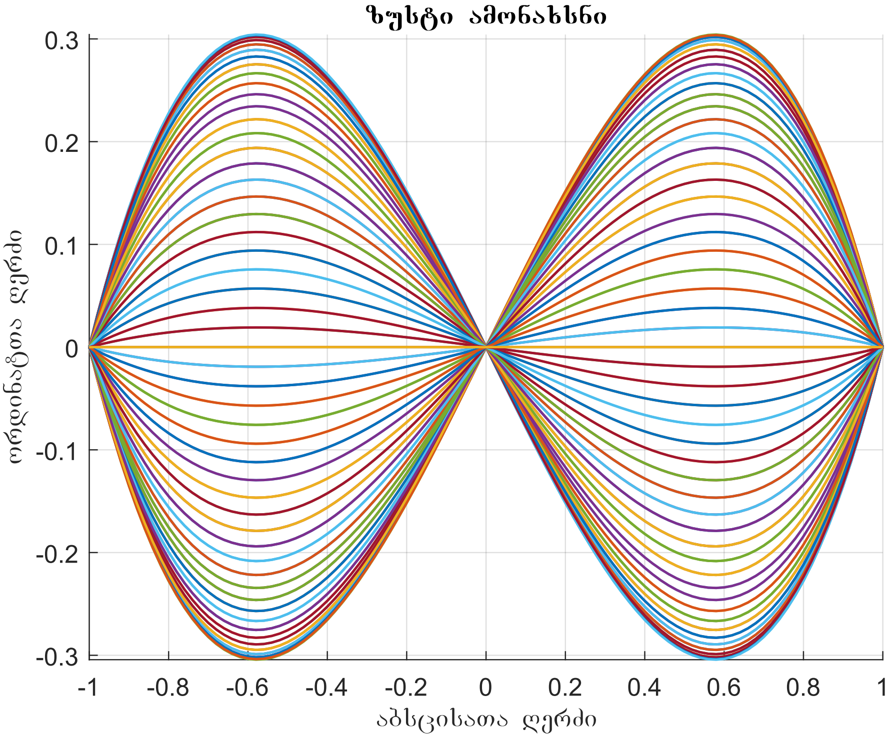
$x \in [-1;1]$ და $t \in [0;2]$ სეგმენტები დავყოთ შესაბამისად $n = 100$ და $M = 100$ ტოლ ნაწილებად, ე. ი. $h = 0.01$ და $\tau = 0.02$. საკოორდინატო ფუნქციების რაოდენობა $N = 5$.
მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}(x) = \sum_{m=1}^{5} c_m^{k+1} \frac{1}{\sqrt{2(2m+1)}} \left(P_{m+1}(x) - P_{m-1}(x)\right); \quad k = 1, \ldots, 99.$$

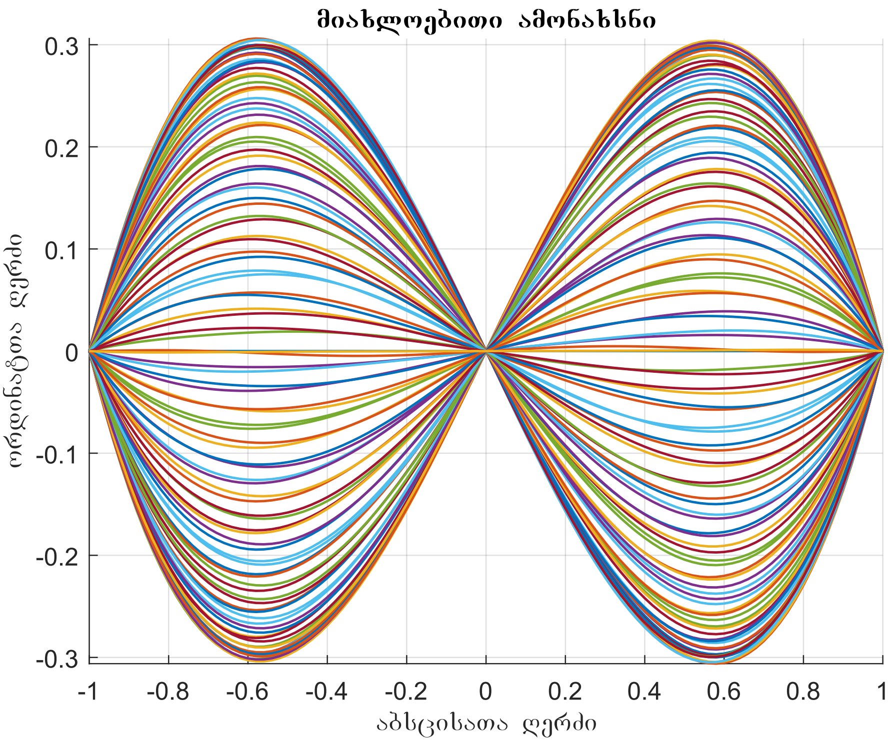
ცდომილება გვექნება:

$$\max\left|\tilde{u}_k(x) - u_k(x)\right| \approx 7.63 \times 10^{-3};$$

დროის ყოველ შრეზე ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



სურ 3.9



სურ 3.10

3. $u\left(x,t\right)=\frac{1}{4}\left(1+x\right)\left(1-x\right)\sin\left(\frac{3}{10}\pi\left(1+x\right)\right)e^{t}$;

$x\in[-1;1]$ და $t\in[0;2]$ სეგმენტები დავყოთ შესაბამისად $n=100$ და $M=100$ ტოლ ნაწილე-

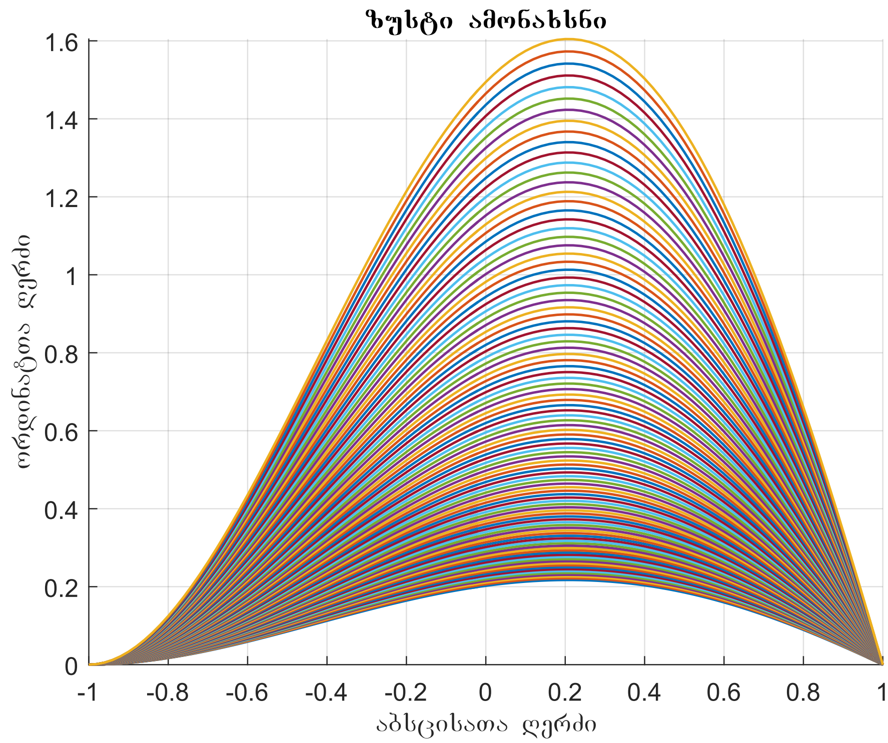
ბად, ე. ი. $h = 0.01$ და $\tau = 0.02$. საკოორდინატო ფუნქციების რაოდენობა $N = 10$. მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}(x) = \sum_{m=1}^{10} c_m^{k+1} \frac{1}{\sqrt{2(2m+1)}} (P_{m+1}(x) - P_{m-1}(x)); \quad k = 1, \ldots, 99.$$

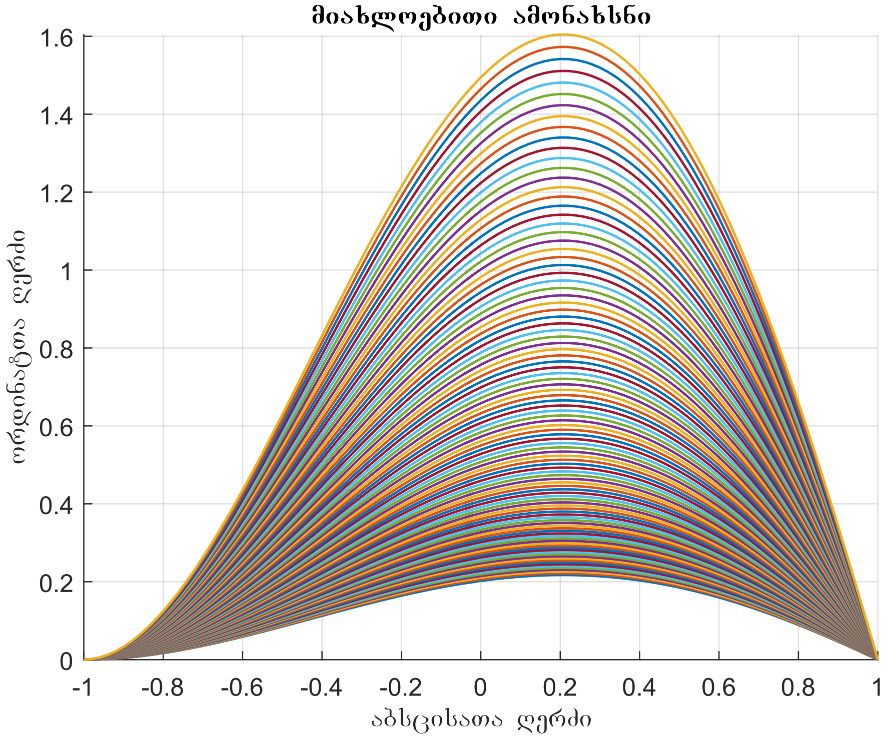
ცდომილება გვექნება:

$$\max |\tilde{u}_k(x) - u_k(x)| \approx 1.08 \times 10^{-4};$$

დროის ყოველ შრეზე ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



სურ 3.11



სურ 3.12

- განვიხილოთ სივრცით ორ ცვლადიანი შემთხვევა, სადაც საკოორდინატო ფუნქციებად აღებულია სინუსები. ამასთან $\alpha = \beta = 1$ და $t \in [0;2]$.

განვიხილოთ შემთხვევა, როდესაც ზუსტ ამონახსნს აქვს შემდეგი სახე:

1. $u\left(x,y,t\right) = \sin\left(3\pi x\right)\sin\left(2\pi y\right)\left(1-t\right)$;

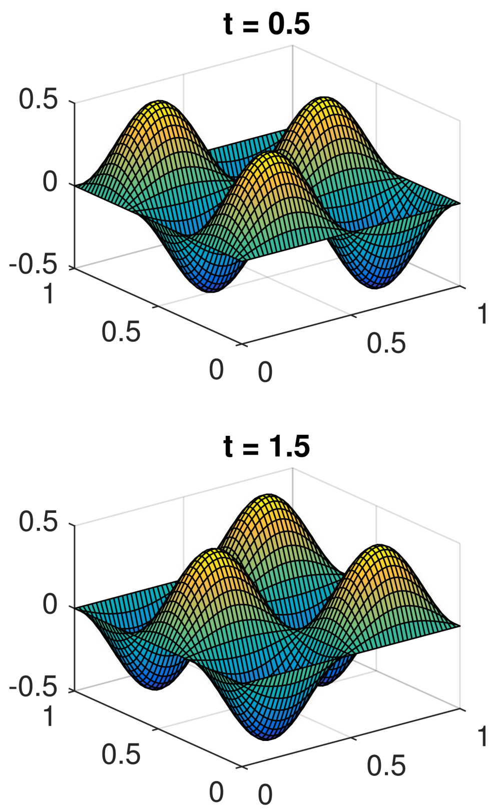
$x \in [0;1]$ და $y \in [0;1]$ სეგმენტები დაყოფილია $r = s = 50$ ტოლ ნაწილებად. $t \in [0;2]$ სეგმენტები დავყოთ $M = 100$ ტოლ ნაწილებად, ე. ი. $h_1 = h_2 = 0.02$ და $\tau = 0.02$. საკოორდინატო ფუნქციების რაოდენობა $N = 3$.
მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}\left(x,y\right) = \sum_{m=1}^{3}\sum_{n=1}^{3} c_{m,n}^{k+1}\sin\left(m\pi x\right)\sin\left(n\pi y\right); \quad k = 1,\ldots,99\,.$$

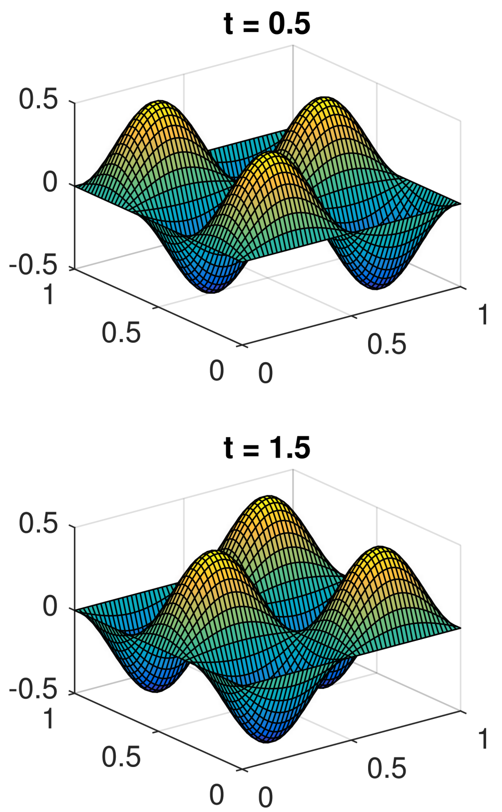
ცდომილება გვექნება:

$$\max\left|\tilde{u}_k\left(x,y\right) - u_k\left(x,y\right)\right| \approx 6.90\times 10^{-15}\,;$$

ავაგოთ $t = 0.5$, $t = 1$, $t = 1.5$ და $t = 2$ წერტილებში ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



სურ 3.13: ზუსტი ამონახსნი



სურ 3.14: მიახლოებითი ამონახსნი

2. $u(x,y,t) = \sin(\pi x)\sin(3\pi y)\sin(3\pi t)$;

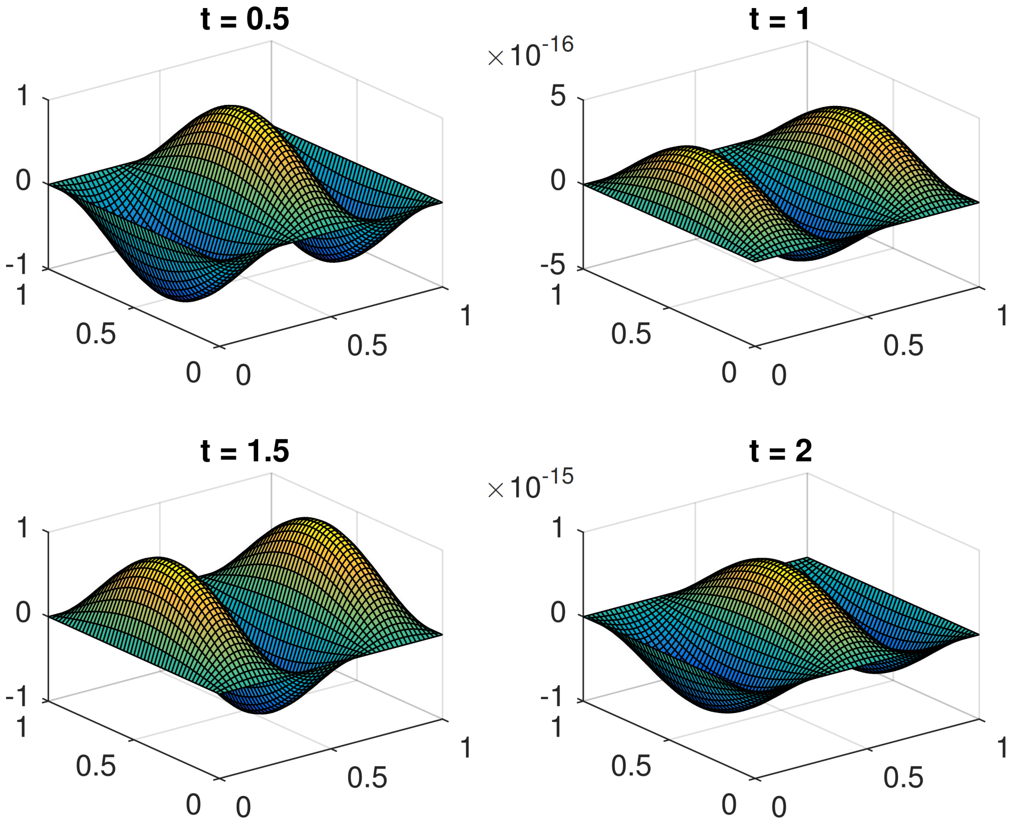
$x \in [0;1]$ და $y \in [0;1]$ სეგმენტები დაყოფილია $r = s = 50$ ტოლ ნაწილებად. $t \in [0;2]$ სეგმენტები დავყოთ $M = 100$ ტოლ ნაწილებად, ე. ი. $h_1 = h_2 = 0.02$ და $\tau = 0.02$. საკოორდინატო ფუნქციების რაოდენობა $N = 3$.
მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}(x,y) = \sum_{m=1}^{3}\sum_{n=1}^{3} c_{m,n}^{k+1}\sin(m\pi x)\sin(n\pi y)\,; \quad k = 1,\ldots,99\,.$$

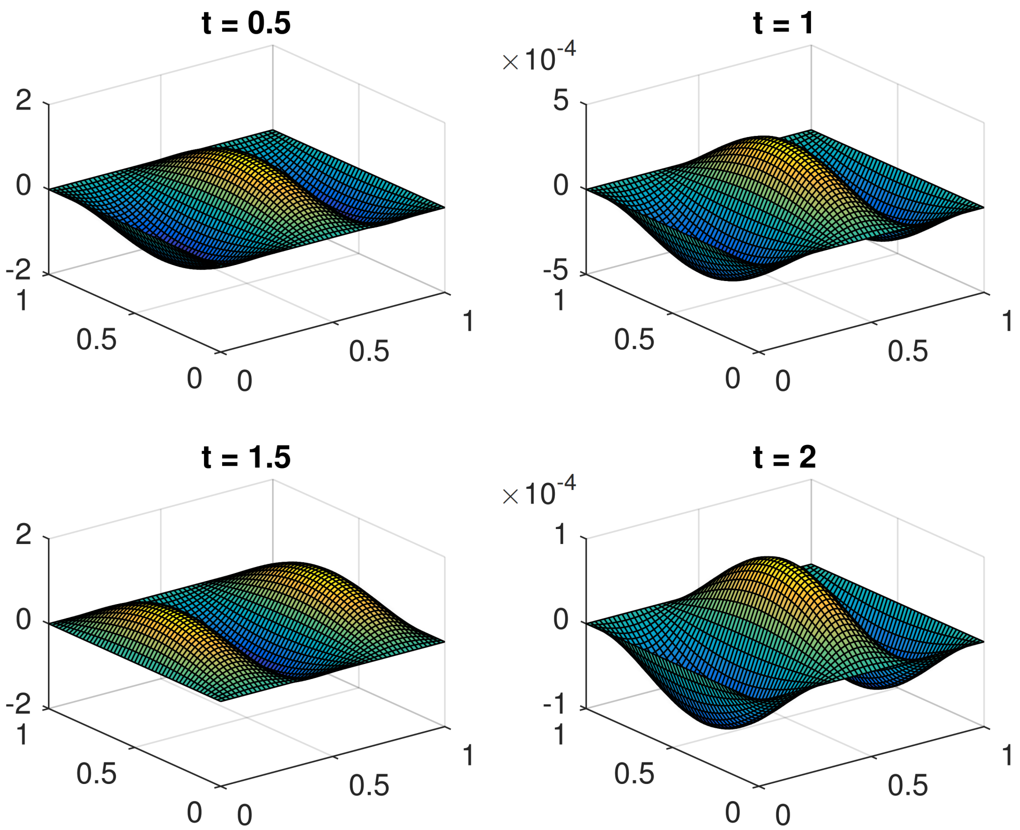
ცდომილება გვექნება:

$$\max\left|\tilde{u}_k(x,y) - u_k(x,y)\right| \approx 6.69\times 10^{-3}\,;$$

ავაგოთ $t = 0.5$, $t = 1$, $t = 1.5$ და $t = 2$ წერტილებში ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



სურ 3.15: ზუსტი ამონახსნი



სურ 3.16: მიახლოებითი ამონახსნი

3. $u(x,y,t) = x(1-x)\sin\left(\frac{2}{5}\pi x\right) y(1-y)\sin\left(\frac{3}{5}\pi y\right) e^t$;

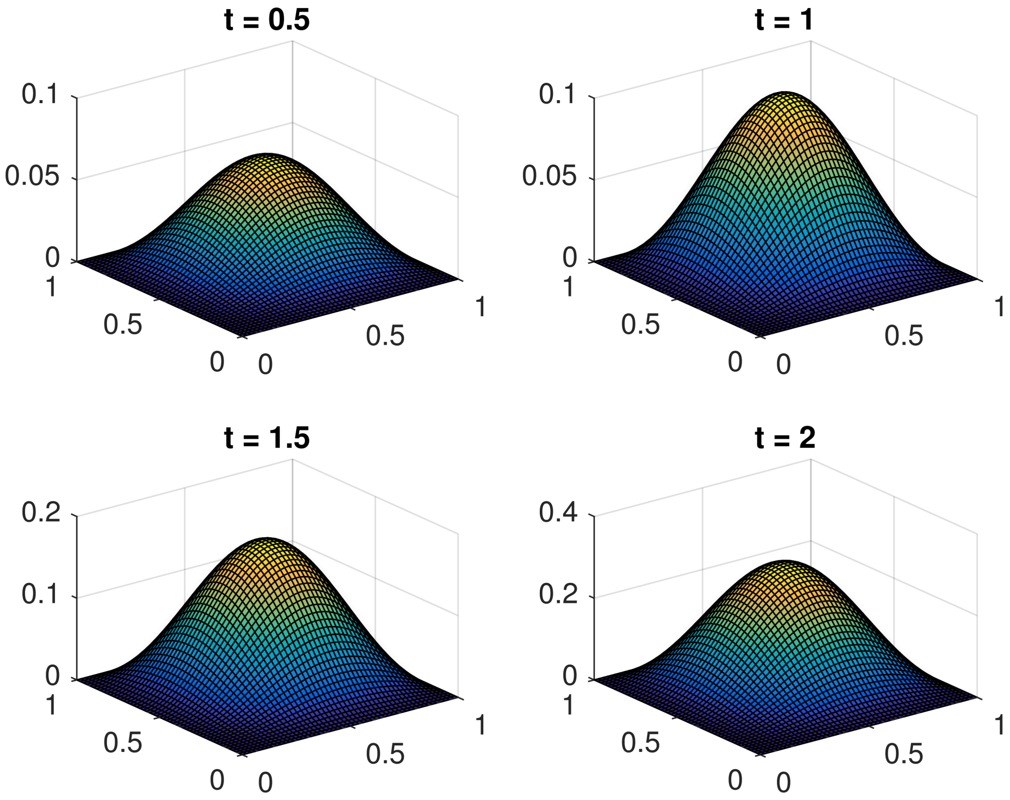
$x \in [0;1]$ და $y \in [0;1]$ სეგმენტები დაყოფილია $r = s = 50$ ტოლ ნაწილებად. $t \in [0;2]$ სეგმენტები დავყოთ $M = 100$ ტოლ ნაწილებად, ე. ი. $h_1 = h_2 = 0.02$ და $\tau = 0.02$. საკოორდინატო ფუნქციების რაოდენობა $N = 5$.
მიახლოებით ამონახსნს ვეძებთ შემდეგი სახით:

$$\tilde{u}_{k+1}(x,y) = \sum_{m=1}^{5}\sum_{n=1}^{5} c_{m,n}^{k+1} \sin(m\pi x)\sin(n\pi y); \quad k = 1,\ldots,99.$$

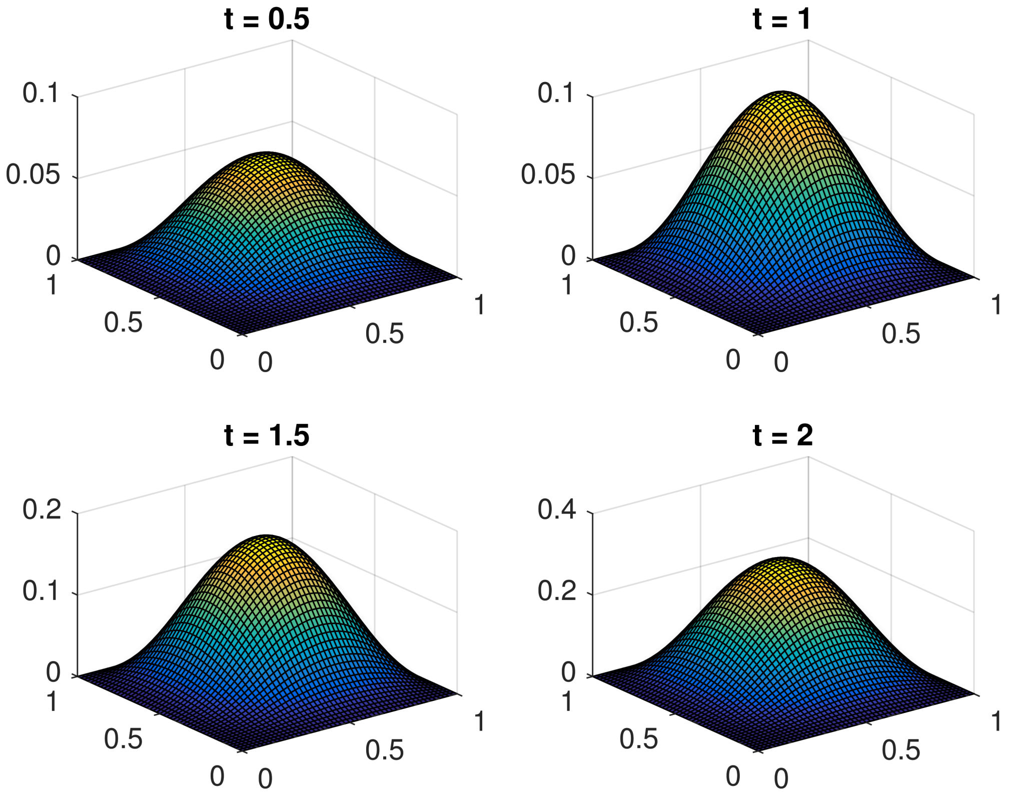
ცდომილება გვექნება:

$$\max|\tilde{u}_k(x,y) - u_k(x,y)| \approx 4.44 \times 10^{-3};$$

ავაგოთ $t = 0.5$, $t = 1$, $t = 1.5$ და $t = 2$ წერტილებში ზუსტი და მიახლოებითი ამონახსნების გრაფიკები:



სურ 3.17: ზუსტი ამონახსნი



სურ 3.18: მიახლოებითი ამონახსნი

დასკვნა

ნაშრომში სიმისთვის კირხოფის კლასიკური განტოლებისთვის და მისი ორგანზომილებიანი განზოგადებისთვის აგებულია ლოკალურად წრფივი, ნახევრადდისკრეტული სამშრიანი სქემა დროითი ცვლადის მიხედვით. სივრცითი ცვლადების მიხედვით გამოყენებულია ვარიაციული მეთოდი. საკოოედინატო ფუნქციებად აღებულია სინუსები და ლეჟანდრის პოლინომების სხვაობა.

კირხოფის სივრცით ერთგანზომილებიანი განტოლების შესაბამისი წრფივი ვარიაციული ამოცანისთვის შეფასებულია მიახლოებითი ამონახსნის ცდომილება და დადგენილია კრებადობის რიგი საკოორდინატო ფუნქციების რიცხვის მიხედვით. ზოგადი ოპერატორული განტოლებისთვის, სიმეტრიული ოპერატორით, დამტკიცებულია ვარიაციული სისტემის შესაბამისი მატრიცის დადებითად განსაზღვრულობა, როცა საკოორდინატო ფუნქციები აკმაყოფილებენ გარკვეულ ბუნებრივ პირობებს. კირხოფის სივრცით ორგანზომილებიანი განტოლებისთვის საწყის-სასაზღვრო ამოცანის მიახლოებითი ამოხსნის ვარიაციული-სხვაობიანი სქემა, იმ შემთხვევისათვის, როცა საკორდინატო ფუნქციებად აღებულია ლეჟანდრის პოლინომების სხვაობა წარმოდგენილია სარეალიზაციო სახით, ხოლო იმ შემთხვევისათვის, როცა საკორდინატო ფუნქციებად აღებულია სინუსები, განხორციელებულია შესაბამისი სქემის კომპიუტერული რეალიზაცია.

აგებული ვარიაციული-სხვაობიანი სქემის საფუძველზე შეიქმნა რიცხვითი რეალიზაციის პროგრამა შესაბამისი ინტერფეისით. ჩატარდა რიცხვითი გათვლები სხვადასხვა ტიპის მოდელური ამოცანებისათვის, როგორც ერთგანზომილებიანი ასევე ორგანზომილებიანი შემთხვევისათვის. მიღებულ თეორიულ შედეგებზე და რიცხვით გათვლებზე დაყრდნობით გაკეთდა პრაქტიკული დასკვნები მეთოდის მდგრადობისა და კრებადობის შესახებ.

ლიტერატურა

[1] J. H. Ahlberg, E. N. Nilson, and J. L. Walsh (Eds.). *The Theory of Splines and Their Applications*. Mathematics in Science and Engineering 38. Elsevier, Academic Press, first edition, 1967. 8

[2] A. Arosio and S. Panizzi. On the well-posedness of the kirchhoff string. *Trans. Amer. Math. Soc.*, 348(1):305 – 330, 1996. iii

[3] S. Bernstein. Sur une classe d'équations fonctionnelles aux dérivées partielles. *Bull. Acad. Sci. URSS Sér. Math. (Izv. Akad. Nauk SSSR)*, 4:17 – 26, 1940. iii

[4] L. Berselli and R. Manfrin. Linear perturbations of the kirchhoff equation. *Comput. Appl. Math.*, 19(2):157 – 178, 2000. iii

[5] P Biler. Remark on the decay for damped string and beam equations. *Non-Linear Anal.*, 10(9):839–842, 1986. iii

[6] E.H. Brito. Estimates for the generalized damped extensible string and beam equations. *Non-Linear Anal.*, 8. iii

[7] Steven Chapra. *Applied Numerical Methods with MATLAB: For Engineers and Scientists, 3rd Edition*. McGraw-Hill Science/Engineering/Math, 3rd edition edition, 2011. 74

[8] I. Christie and J. M. Sanz-Serna. A galerkin method for a nonlinear integro-differential wave system. *Computer Methods in Applied Mechanics and Engineering*, 44(2):229 – 237, 1984. iii

[9] Piero D'Ancona and Sergio Spagnolo. *On an Abstract Weakly Hyperbolic Equation Modelling the Nonlinear Vibrating String*, pages 27–32. Springer US, Boston, MA, 1992. iii

[10] Piero D'Ancona and Sergio Spagnolo. A class of nonlinear hyperbolic problems with global solutions. *Archive for Rational Mechanics and Analysis*, 124(3):201–219, Nov 1993. iii

[11] Orin J. Farrell and Bertram Ross. *Solved Problems in Analysis: As Applied to Gamma, Beta, Legendre and Bessel Functions*. Dover Books on Mathematics. Dover Publications, reprint 1991 edition, 2013. 17

[12] Dunham Jackson. *Fourier Series and Orthogonal Polynomials*. MAA. Dover Publications, dover ed edition, 2004. 17

[13] V. I. Lebedev. *An Introduction to Functional Analysis in Computational Mathematics*. Birkhäuser Basel, 1st edition, 1996. 21

[14] I-Shih Liu and M.A. Rincon. Effect of moving boundaries on the vibrating elastic string. *Applied Numerical Mathematics*, 47(2):159 – 172, 2003. Second International Workshop on Numerical Linear Algebra - Numerical Methods for Partial Differential Equations and Optimization. iii

[15] R. Manfrin. Global solvability to the kirchhoff equation for a new class of initial data. *Portugaliae Mathematica. Nova Série*, 59(1):91–109, 2002. iii

[16] Marivaldo Pereira Matos. Mathematical analysis of the nonlinear model for the vibrations of a string. *Nonlinear Analysis: Theory, Methods & Applications*, 17(12):1125 – 1137, 1991. iii

[17] Luiz Adauto Medeiros. On a new class of nonlinear wave equations. *Journal of Mathematical Analysis and Applications*, 69(1):252 – 262, 1979. iii

[18] Kenji Nishihara. On a global solution of some quasilinear hyperbolic equation. *Tokyo Journal of Mathematics*, 7(2):437–459, 1984. iii

[19] Stephen R. Otto and James P. Denier. *An Introduction to Programming and Numerical Methods in MATLAB*. Springer, 1st edition, 2005. 74

[20] Stefano Panizzi. Low regularity global solutions for nonlinear evolution equations of kirchhoff type. *Journal of Mathematical Analysis and Applications*, 332(2):1195 – 1215, 2007. iii

[21] A. Papukashvili, J. Peradze, and J. Rogava. An approximate algorithm for a kirchhoff nonlinear dynamic beam equation. *Reports of Enlarged Session of the Seminar of I. Vekua Institute of Applied Mathematics*, 23:84 – 86, 2009. iii

[22] A. Papukashvili, J. Rogava, and Z. Vashakidze. On one numerical method of research of the stress-deformed condition of some multystructures with difficult geometry. *TICSSAM-2015*. iii

[23] A. Papukashvili, J. Rogava, and Z. Vashakidze. On the numerical solution of contact problem for poissons and kirchhoff equation system. *V Annual Meeting of the Georgian Mechanical Union*, Book of Abstracts:59 – 60. iii

[24] Jemal Peradze. A numerical algorithm for the nonlinear kirchhoff string equation. *Numerische Mathematik*, 102(2):311–342, 2005. iii

[25] Andy H. Register. *A Guide to MATLAB Object-Oriented Programming*. Chapman & Hall/CRC, SCITECH Pub, 2007. 74

[26] K. Rektorys. *Variational Methods in Mathematics, Science and Engineering*. Springer, 2nd edition, 1980. 1, 22, 32

[27] J. Rogava and M. Tsiklauri. Three-layer semidiscrete scheme for generalized kirchhoff equation. In *Proceedings of the 2Nd WSEAS International Conference on Finite Differences, Finite Elements, Finite Volumes, Boundary Elements*, FANDB'09, pages 193–199, 2009. iii

[28] J. Rogava and M. Tsiklauri. Convergence of a semi-discrete scheme for an abstract nonlinear second order evolution equation. *Applied Numerical Mathematics*, 75:22 – 36, 2014. 10th IMACS International Symposium on Iterative Methods in Scientific Computing. iii

[29] William E. Schiesser and Graham W. Griffiths. *A Compendium of Partial Differential Equation Models with MATLAB*. Cambridge University Press, 1st edition, 2009. 74

[30] Gábor Szegö. *Orthogonal Polynomials*. Colloquium publications (American Mathematical Society), v. 23. American Mathematical Society, 4th edition, 1939-1975. 17

[31] თამაზ ვაშაყმაძე. *რიცხვითი ანალიზი I*. რიცხვითი ანალიზი და გამოთვლითი ტექნოლოგიები. თბილისის უნივერსიტეტის გამომცემლობა, 2009. 8

[32] ჯემალ როგავა. *ფუნქციონალური ანალიზის მეთოდები გამოთვლით მათემატიკაში (ლექციების კურსი)*. რიცხვითი ანალიზი და გამოთვლითი ტექნოლოგიები. თბილისი, 2016. 1, 21

დანართი A

პროგრამები

A.1 პროგრამული კოდი სივრცით ერთგანზომილებიანი ამოცანისათვის. საკოორდინატო ფუნქციებად აღებულია სინუსები

პროგრამები დაწერილია MATLAB 8.6 (R2015b) ვერსიაში. ამ დარგში აღსანიშნავია შემდეგი ლიტერატურა [7], [19], [25] და [29].

ძირითადი პროგრამა შედგება ქვეპროგრამებისაგან. ძირითადი პროგრამის გაშვების შემდგომ ხდება მათი გამოძახება. აღვწეროთ თითოეული მათგანი:

(1.59), (1.60), (1.61) და (1.62) ინტეგრალების ქვეპროგრამები:

1. integr_I0.m

```
function [ I0 ] = integr_I0( x,m,h )

I0 = 2/(m*pi)*sin(m*(x - 0.5*h)*pi)*sin(0.5*m*h*pi);

end
```

2. integr_I1.m

```
function [ I1 ] = integr_I1( x,m,h )

I1 = (x - h)*integr_I0( x,m,h ) + 1/(m*pi)*(2/(m*pi)*sin(0.5*m*h*pi)*...
    cos(m*(x - 0.5*h)*pi) - h*cos(m*pi*x));

end
```

3. integr_I2.m

```
function [ I2 ] = integr_I2( x,m,h )

I2 = ((x - h)*(x - h) - 2/(m*m*pi*pi))*integr_I0( x,m,h ) + ...
    2/(m*m*pi*pi)*(2*(x - h)*sin(0.5*m*h*pi)*cos(m*(x - 0.5*h)*pi) + ...
    h*sin(m*pi*x)) - h*(2*x - h)/(m*pi)*cos(m*pi*x);

end
```

4. integr_I3.m

```
function [ I3 ] = integr_I3( x,m,h )

I3 = (x - h)*(x - h)*(x - h)*integr_I0( x,m,h ) + ...
    1/(m*pi)*(3/(m*pi)*h*(2*x - h)*sin(m*pi*x) - h*(3*x*(x - h) + ...
    h*h)*cos(m*pi*x)) + 6/(m*m*pi*pi)*((x - h)*(x - h)*...
    sin(0.5*m*h*pi)*cos(m*(x - 0.5*h)*pi) - integr_I1( x,m,h ));

end
```

სპლაინ აპროქსიმაციას ვახდენთ შემდეგი ქვეპროგრამით:

spline_function_approximation.m

```
d       = zeros(n - 2,1);
d       = [d1;d;dn];
clear d1 dn

lambda  = zeros(n,1);
mu      = zeros(n,1);

for i = 2:1:(n - 1)
    lambda(i) = 0.5;
    mu(i)     = 0.5;
    d(i)      = 3*(y(i + 1) - 2*y(i) + y(i - 1))/(h*h);
end

clear i

lambda(1)      = 0; %%% piroba lambda_0-Si          %%%
mu(n)          = 0; %%% piroba mu_n-Si              %%%

clear i

b          = 2*ones(n,1); %%% samdiagonaluri matricis mTavari diagonali %%%
a          = mu;          %%%% samdiagonaluri matricis qveda diagonali %%%%
c          = lambda;      %%%%% samdiagonaluri matricis zeda diagonali %%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%% Tridiagonal matrix algorithm %%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

q = zeros(n + 1,1);
p = zeros(n,1);
u1 = zeros(n + 1,1);
M1 = zeros(n,1);

for s = 2:1:(n + 1)
    p(s - 1) = a(s - 1)*q(s - 1) + b(s - 1);
    q(s)     = -c(s - 1)/p(s - 1);
```

```
    u1(s)    = (d(s - 1) - a(s - 1)*u1(s - 1))/p(s - 1);
end

M1(n) = u1(n + 1);

clear s

for s = (n - 1):-1:1
    M1(s) = q(s + 1)*M1(s + 1) + u1(s + 1);
end

clear i j s p q u1 a b c lambda mu
```

ძირითად პროგრამას აქვს შემდეგი სახე (ეს პროგრამა იძახებს ზემოთ მოყვანილ ქვეპროგრამებს):

`sine_basis_for_1d_spline.m`

```
clc
clear all
format longG

%% funqciebis Setana

syms psi0(x) d2psi0(x) psi1(x) d2psi1(x) f(x,t) d2f(x,t) U(x,t) ...
    diff_t_U(x,t) diff_tt_U(x,t) diff_x_U(x,t) diff_xx_U(x,t)

%% alpha and beta
alpha     = 1;
beta      = 1;

%% zusti amonaxsni

U(x,t)     = x*(1 - x)*sin(0.6*pi*x)*exp(t);
diff_t_U(x,t)  = diff(U,t,1);
diff_tt_U(x,t) = diff(U,t,2);
diff_x_U(x,t)  = diff(U,x,1);
diff_xx_U(x,t) = diff(U,x,2);

%% gantolebis marjvena mxare

f(x,t)     = diff_tt_U(x,t) - ...
    (alpha + beta*int((diff_x_U(x,t)*diff_x_U(x,t)),x,0,1))*diff_xx_U(x,t);

%% sawyisi pirobebi

psi0(x)    = U(x,0);
psi1(x)    = diff_t_U(x,0);

%% sasazRvro pirobebi kuburi splainisaTvis

d2psi0(x) = diff(psi0,x,2);
```

```
d2psi1(x) = diff(psi1,x,2);
d2f(x,t)  = diff(f,x,2);
psi0      = matlabFunction(psi0);
psi1      = matlabFunction(psi1);
f         = matlabFunction(f);
d2psi0    = matlabFunction(d2psi0);
d2psi1    = matlabFunction(d2psi1);
d2f       = matlabFunction(d2f);
U         = matlabFunction(U);
diff_t_U  = matlabFunction(diff_t_U);
diff_tt_U = matlabFunction(diff_tt_U);
diff_x_U  = matlabFunction(diff_x_U);
diff_xx_U = matlabFunction(diff_xx_U);

clear x t

%% x da t RerZebis dayofa tol qveSualedebad h da tau bijad

n         = 101;                        %%% x RerZis dayofaTa ricxvi %%%
T         = 2;                             %%% t drois bolo wertili %%%
M         = 101;                        %%% t RerZis dayofaTa ricxvi %%%
h         = 1/(n - 1);                         %%% x--is bijebis ricxvi %%%
tau       = T/(M - 1);                                  %%% t droiTi biji %%%
x         = (0:h:1)';                                          %%% x RerZi %%%
t         = (0:tau:T)';                                 %%% t droiTi RerZi %%%

%% bazisebis generacia

N         = 10;                  %%% sakoordinato funqciebis raodenoba %%%
basis     = zeros(n,N);                           %%% bazisebis masivi %%%

for i = 1:n
    for m = 1:N
        basis(i,m) = sin(x(i)*pi*m);
    end
end

%% c koeficientebi
c1 = zeros(N,1);
c2 = zeros(N,1);
c3 = zeros(N,1);
%% kuburi splainisaTvis saWiro koeficientebi
A = zeros(n,1);
B = zeros(n,1);
C = zeros(n,1);
D = zeros(n,1);

%% integralebi
I_f = zeros(N,M-1);                            %%% f(x,t)--dan integrali %%%

u       = zeros(n,M);                      %%% yoveli SrisaTvis funqciebi %%%
```

```
u_zusti = zeros(n,M);  %%% zusti mniSvnelobebi yoveli x--i da t--saTvis %%%
y       = zeros(n,1);

%% k = SrisaTvis f-is integralis daTvla

for k = 1:(M - 1)
    for i = 1:n
        y(i) = f(x(i),t(k));
    end
    d1    = 2*d2f(x(1),t(k));
    dn    = 2*d2f(x(n),t(k));
    spline_function_approximation;
    for j = 2:n
        A(j) = 1/(6*h)*(M1(j) - M1(j-1));
        B(j) = 0.5*M1(j) - 3*x(j)*A(j);
        C(j) = -h*h*A(j) - x(j)*B(j) - 0.5*x(j - 1)*M1(j) + ...
            1/h*(y(j) - y(j - 1));
        D(j) = -x(j)*(x(j)*x(j) + h*h)*A(j) - (x(j)*x(j) + ...
            1/3*h*h)*B(j) - x(j)*C(j) + 1/6*h*h*M1(j) + y(j);
    end
    for m = 1:N
        for j = 2:n
            I_f(m,k) = I_f(m,k) + A(j)*integr_I3( x(j),m,h ) + ...
                B(j)*integr_I2( x(j),m,h ) + ...
                C(j)*integr_I1( x(j),m,h ) + ...
                D(j)*integr_I0( x(j),m,h );
        end
    end
end

clear i j m k

k  = 1;                                   %%% pirveli (nulovani) Sre %%%

while k <= M
    switch k
        case 1
            I    = zeros(N,1);                        %%% integrali %%%
            for i = 1:n
                y(i) = psi0(x(i));
            end
            d1    = 2*d2psi0(x(1));
            dn    = 2*d2psi0(x(n));
            spline_function_approximation;
            for j = 2:n
                A(j) = 1/(6*h)*(M1(j) - M1(j-1));
                B(j) = 0.5*M1(j) - 3*x(j)*A(j);
                C(j) = -h*h*A(j) - x(j)*B(j) - 0.5*x(j - 1)*M1(j) + ...
                    1/h*(y(j) - y(j - 1));
                D(j) = -x(j)*(x(j)*x(j) + h*h)*A(j) - ...
                    (x(j)*x(j) + 1/3*h*h)*B(j) - ...
```

```
                x(j)*C(j) + 1/6*h*h*M1(j) + y(j);
        end
        for m = 1:N
            for j = 2:n
                I(m) = I(m) + A(j)*integr_I3( x(j),m,h ) + ...
                    B(j)*integr_I2( x(j),m,h ) + ...
                    C(j)*integr_I1( x(j),m,h ) + ...
                    D(j)*integr_I0( x(j),m,h );
            end
        end
        c1 = 2*I;
        q1 = 0;
        for m = 1:N
            q1 = q1 + m*m*c1(m)*c1(m);
        end
        q1 = alpha + 0.5*pi*pi*beta*q1;
        % disp(q1);
        u(:,k)       = basis*c1;
        u_zusti(:,k) = U(x,t(k));
    case 2
        I    = zeros(N,1);                             %%% integrali %%%
        for i = 1:n
            y(i) = psi1(x(i));
        end
        d1     = 2*d2psi1(x(1));
        dn     = 2*d2psi1(x(n));
        spline_function_approximation;
        for j = 2:n
            A(j) = 1/(6*h)*(M1(j) - M1(j-1));
            B(j) = 0.5*M1(j) - 3*x(j)*A(j);
            C(j) = -h*h*A(j) - x(j)*B(j) - 0.5*x(j - 1)*M1(j) + ...
                1/h*(y(j) - y(j - 1));
            D(j) = -x(j)*(x(j)*x(j) + h*h)*A(j) - ...
                (x(j)*x(j) + 1/3*h*h)*B(j) - ...
                x(j)*C(j) + 1/6*h*h*M1(j) + y(j);
        end
        for m = 1:N
            for j = 2:n
                I(m) = I(m) + A(j)*integr_I3( x(j),m,h ) + ...
                    B(j)*integr_I2( x(j),m,h ) + ...
                    C(j)*integr_I1( x(j),m,h ) + ...
                    D(j)*integr_I0( x(j),m,h );
            end
        end
        for m = 1:N
            c2(m) = (1 - 0.5*m*m*pi*pi*tau*tau*q1)*c1(m) + ...
                tau*(2*I(m) + tau*I_f(m,k - 1));
        end
        q2 = 0;
        for m = 1:N
            q2 = q2 + m*m*c2(m)*c2(m);
```

```
            end
            q2 = alpha + 0.5*pi*pi*beta*q2;
            % disp(q2);
            u(:,k)        = basis*c2;
            u_zusti(:,k) = U(x,t(k));
        otherwise
            for m = 1:N
                c3(m) = (4*tau*tau)/(2 + m*m*pi*pi*tau*tau*q2)*...
                    I_f(m,k - 1) - c1(m) + ...
                    4/(2 + m*m*pi*pi*tau*tau*q2)*c2(m);
            end
            q3 = 0;
            for m = 1:N
                q3 = q3 + m*m*c3(m)*c3(m);
            end
            q3 = alpha + 0.5*pi*pi*beta*q3;
            % disp(q1);
            u(:,k)        = basis*c3;
            u_zusti(:,k) = U(x,t(k));

            c1 = c2;
            c2 = c3;
            q2 = q3;
    end

    k = k + 1;

end

error = max(max(abs(u - u_zusti)));
fprintf('max(max(abs(u - u_zusti))) = %.16f\n', error);
```

ზუსტი და მიახლოებითი ამონახსნის აგება დროის ყველა შრეზე:

1. ზუსტი ამონახსნი

   exact_solution.m

```
figure,hold on
grid on
title('zusti amonaxsni','fontname','acadnusx','fontsize',16);
xlabel('abscisaTa RerZi','fontname','acadnusx','fontsize',14);
ylabel('ordinatTa RerZi','fontname','acadnusx','fontsize',14);
axis([0 1 min(min(u_zusti)) max(max(u_zusti))])

for k = 1:M
    plot(x,u_zusti(:,k),'LineWidth',1);
    pause(0.001)
end
```

2. მიახლოებითი ამონახსნი

   approx_solution.m

```
figure,hold on
grid on
title('miaxloebiTi amonaxsni','fontname','acadnusx','fontsize',16);
xlabel('abscisaTa RerZi','fontname','acadnusx','fontsize',14);
ylabel('ordinatTa RerZi','fontname','acadnusx','fontsize',14);
axis([0 1 min(min(u)) max(max(u))])

for k = 1:M
    plot(x,u(:,k),'LineWidth',1);
    pause(0.001)
end
```

ზუსტი და მიახლოებითი ამონახსნის ანიმაცია

`animation_of_exact_and_approx_solution.m`

```
figure(1)
filename = 'exact_and_approx_animation_sine_basis.gif';
for k = 1:M
      y1 = u_zusti(:,k);
      y2 = u(:,k);
      plot(x,y1,'b',x,y2,'r','LineWidth',2)
      axis([0 1 min(min(min(u_zusti)),min(min(u))) ...
          max(max(max(u_zusti)),max(max(u)))])
      drawnow
      frame = getframe(1);
      im = frame2im(frame);
      [imind,cm] = rgb2ind(im,256);
      if k == 1;
          imwrite(imind,cm,filename,'gif', 'Loopcount',inf);
      else
          imwrite(imind,cm,filename,'gif','WriteMode','append');
      end
end
```

A.2 პროგრამული კოდი სივრცით ერთგანზომილებიანი ამოცანისათვის. საკოორდინატო ფუნქციებად აღებულია ლეჟანდრის პოლინომების სხვაობა

სანამ ძირითად პროგრამაზე გადავალთ აღვწეროთ საჭირო ქვეპროგრამები. (1.86) ფორმულის გამოყენებით ლეჟანდრის პოლინომების ($P_m(x)$-ების) აგება ფიქსირებული m-სა და ფიქსირებული x-სათვის.

`Leg_Pol.m`

```
function [ Pm ] = Leg_Pol( m, x )

for i = 0:m
switch i
case 0
```

```
P0 = 1;
Pm = P0;
case 1
P1 = x;
Pm = P1;
otherwise
Pm = ((2*i - 1)*x*P1 - (i - 1)*P0)/i;
P0 = P1;
P1 = Pm;
end
end
end
```

(1.83)–ის, (1.97)–ისა და (1.98)–ის გამოთვლა:

- coeff_A.m

```
function [ coeff1 ] = coeff_A( m )

coeff1 = 1/sqrt(2*(2*m + 1));

end
```

- coeff_B.m

```
function [ coeff2 ] = coeff_B( m )

coeff2 = 1/((2*m + 1)*sqrt((2*m - 1)*(2*m + 3)));

end
```

- coeff_C.m

```
function [ coeff3 ] = coeff_C( m )

coeff3 = 2/((2*m - 1)*(2*m + 3));

end
```

- coordinate_func.m

```
function [ psi ] = coordinate_func( m, x )

psi = coeff_A( m )*(Leg_Pol( m + 1, x ) - Leg_Pol( m - 1, x ));

end
```

(1.102) და (1.103) გამოსათვლელი ქვეპროგრამები:

1. integr_I0.m

```
function [ I0 ] = integr_I0( x,m,h )

```

```
switch m
case 0
I0 = h;
case 1
I0 = 0.5*(x*x - (x - h)*(x - h));
otherwise
I0 = 2*coeff_A( m )*coeff_A( m )*...
(Leg_Pol( m + 1, x ) - Leg_Pol( m - 1, x ) - Leg_Pol( m + 1, x - h ) + ...
Leg_Pol( m - 1, x - h ));
end
end
```

2. `integr_I1.m`

```
function [ I1 ] = integr_I1( x,m,h )

switch m
case 0
I1 = integr_I0( x,m + 1,h );
case 1
I1 = (x*x*x - (x - h)*(x - h)*(x - h))/3;
otherwise
I1 = 2*coeff_A( m )*coeff_A( m )*...
((m + 1)*integr_I0( x,m + 1,h ) + m*integr_I0( x,m - 1,h ));
end
end
```

3. `integr_I2.m`

```
function [ I2 ] = integr_I2( x,m,h )

switch m
case 0
I2 = integr_I1( x,m + 1,h );
case 1
I2 = 0.25*(x*x*x*x - (x - h)*(x - h)*(x - h)*(x - h));
otherwise
I2 = 2*coeff_A( m )*coeff_A( m )*...
((m + 1)*integr_I1( x,m + 1,h ) + m*integr_I1( x,m - 1,h ));
end
end
```

4. `integr_I3.m`

```
function [ I3 ] = integr_I3( x,m,h )

switch m
case 0
I3 = integr_I2( x,m + 1,h );
case 1
I3 = 0.2*(x*x*x*x*x - (x - h)*(x - h)*(x - h)*(x - h)*(x - h));
otherwise
```

```
I3 = 2*coeff_A( m )*coeff_A( m )*...
((m + 1)*integr_I2( x,m + 1,h ) + m*integr_I2( x,m - 1,h ));
end
end
```

$E^s_{j,m} = I^s_{j,m+1} - I^s_{j,m-1}\,, \quad (s = 0, 1, 2, 3)$ გამოსათვლელი ქვეპროგრამები:

1. `integr_basis0.m`

```
function [ E0 ] = integr_basis0( x,m,h )

E0 = integr_I0( x,m + 1,h ) - integr_I0( x,m - 1,h );

end
```

2. `integr_basis1.m`

```
function [ E1 ] = integr_basis1( x,m,h )

E1 = integr_I1( x,m + 1,h ) - integr_I1( x,m - 1,h );

end
```

3. `integr_basis2.m`

```
function [ E2 ] = integr_basis2( x,m,h )

E2 = integr_I2( x,m + 1,h ) - integr_I2( x,m - 1,h );

end
```

4. `integr_basis3.m`

```
function [ E3 ] = integr_basis3( x,m,h )

E3 = integr_I3( x,m + 1,h ) - integr_I3( x,m - 1,h );

end
```

სპლაინ აპროქსიმაციის ქვეპროგრამა

`spline_function_approximation.m`

```
d_s       = zeros(n - 2,1);
d_s       = [d1;d_s;dn];
clear d1 dn

lambda  = zeros(n,1);
mu      = zeros(n,1);

for i = 2:1:(n - 1)
    lambda(i) = 0.5;
    mu(i)     = 0.5;
```

```
    d_s(i)      = 3*(y_s(i + 1) - 2*y_s(i) + y_s(i - 1))/(h*h);
end

clear i

lambda(1)      = 0; %%% piroba lambda_0-Si           %%%
mu(n)          = 0; %%% piroba mu_n-Si               %%%

clear i

b_s          = 2*ones(n,1);%%% samdiagonaluri matricis mTavari diagonali %%%
a_s          = mu;          %%% samdiagonaluri matricis qveda diagonali %%%
c_s          = lambda;      %%%% samdiagonaluri matricis zeda diagonali %%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%% Tridiagonal matrix algorithm %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

q_s = zeros(n + 1,1);
p_s = zeros(n,1);
u1 = zeros(n + 1,1);
M1 = zeros(n,1);

for s = 2:1:(n + 1)
    p_s(s - 1) = a_s(s - 1)*q_s(s - 1) + b_s(s - 1);
    q_s(s)      = -c_s(s - 1)/p_s(s - 1);
    u1(s)     = (d_s(s - 1) - a_s(s - 1)*u1(s - 1))/p_s(s - 1);
end

M1(n) = u1(n + 1);

clear s

for s = (n - 1):-1:1
    M1(s) = q_s(s + 1)*M1(s + 1) + u1(s + 1);
end

clear i j s p_s q_s u1 a_s b_s c_s lambda mu
```

ძირითად პროგრამას, რომელიც იყენებს ზემოთ მოცემულ ქვეპროგრამებს აქვს შემდეგი სახე:

`kirchhoff_1d_with_legendre_polynomials_spline_approx.m`

```
clc
clear all
format longG

%% funqciebis Setana

syms psi0(x) d2psi0(x) psi1(x) d2psi1(x) f(x,t) d2f(x,t) U(x,t) ...
    diff_t_U(x,t) diff_tt_U(x,t) diff_x_U(x,t) diff_xx_U(x,t)
```

```
%% alpha da beta

alpha = 1;
beta  = 1;

%% zusti amonaxsni

U(x,t)     = 0.25*(1 + x)*(1 - x)*sin(0.3*pi*(x + 1))*exp(t);
diff_t_U(x,t)  = diff(U,t,1);
diff_tt_U(x,t) = diff(U,t,2);
diff_x_U(x,t)  = diff(U,x,1);
diff_xx_U(x,t) = diff(U,x,2);

%% gantolebis marjvena mxare

f(x,t)     = diff_tt_U(x,t) - ...
    (alpha + beta*int((diff_x_U(x,t)*diff_x_U(x,t)),x,-1,1))...
    *diff_xx_U(x,t);

diff_tt_U = matlabFunction(diff_tt_U);
diff_x_U  = matlabFunction(diff_x_U);
diff_xx_U = matlabFunction(diff_xx_U);

%% sawyisi piroba

psi0(x)    = U(x,0);
psi1(x)    = diff_t_U(x,0);

%% meore rigis warmoebulebi sawyisi pirobebis splainebisaTvis

d2psi0(x) = diff(psi0,x,2);
d2psi1(x) = diff(psi1,x,2);

%% meore rigis warmoebuli gantolebis marjvena mxaris splainebisaTvis

d2f(x,t)  = diff(f,x,2);

%% simboluri cvladebis funqciebis Function Handle-ebad gadayvana

U          = matlabFunction(U);
diff_t_U   = matlabFunction(diff_t_U);
psi0       = matlabFunction(psi0);
psi1       = matlabFunction(psi1);
f          = matlabFunction(f);
d2psi0     = matlabFunction(d2psi0);
d2psi1     = matlabFunction(d2psi1);
d2f        = matlabFunction(d2f);

clear x t

```

```
%% x da t RerZebis dayofa tol qveSualedebad h da tau bijad

n         = 101;                          %%% x RerZis dayofaTa ricxvi %%%
T         = 2;                               %%% t drois bolo wertili %%%
M         = 101;                          %%% t RerZis dayofaTa ricxvi %%%
h         = 2/(n - 1);                          %%% x--is bijebis ricxvi %%%
tau       = T/(M - 1);                                 %%% t droiTi biji %%%
x         = (-1:h:1)';                                      %%% x RerZi %%%
t         = (0:tau:T)';                              %%% t droiTi RerZi %%%

%% sakoordinato funqciebis generacia

N         = 10;                  %%% sakoordinato funqciebis raodenoba %%%
basis     = zeros(n,N);                             %%% bazisebis masivi %%%

for i = 1:n
    for m = 1:N
        basis(i,m) = coordinate_func( m, x(i) );
    end
end

%% c koeficientebi
c1 = zeros(N,1);
c2 = zeros(N,1);
c3 = zeros(N,1);
z  = zeros(N,1);
y  = zeros(N,1);

%% kuburi splainisaTvis saWiro koeficientebi
A_s = zeros(n,1);
B_s = zeros(n,1);
C_s = zeros(n,1);
D_s = zeros(n,1);

%% integralebi
I_f = zeros(N,M-1);                           %%% f(x,t)--dan integrali %%%

u       = zeros(n,M);                    %%% yoveli SrisaTvis funqciebi %%%
u_zusti = zeros(n,M);  %%% zusti mniSvnelobebi yoveli x--i da t--saTvis %%%

%% k -- SrisaTvis f-is sakoordinato funqciebze namravlis integralis daTvla

y_s = zeros(n,1);
d1  = 0;
dn  = 0;

for k = 1:(M - 1)
    for i = 1:n
        y_s(i) = f(x(i),t(k));
    end
    d1    = 2*d2f(x(1),t(k));
```

```matlab
    dn     = 2*d2f(x(n),t(k));
    spline_function_approximation;
    for j = 2:n
        A_s(j) = 1/(6*h)*(M1(j) - M1(j-1));
        B_s(j) = 0.5*M1(j) - 3*x(j)*A_s(j);
        C_s(j) = -h*h*A_s(j) - x(j)*B_s(j) - 0.5*x(j - 1)*M1(j) + ...
            1/h*(y_s(j) - y_s(j - 1));
        D_s(j) = -x(j)*(x(j)*x(j) + h*h)*A_s(j) - (x(j)*x(j) + ...
            1/3*h*h)*B_s(j) - x(j)*C_s(j) + 1/6*h*h*M1(j) + y_s(j);
    end
    for m = 1:N
        for j = 2:n
            I_f(m,k) = I_f(m,k) + A_s(j)*integr_basis3( x(j),m,h ) + ...
                B_s(j)*integr_basis2( x(j),m,h ) + ...
                C_s(j)*integr_basis1( x(j),m,h ) + ...
                D_s(j)*integr_basis0( x(j),m,h );
        end
        I_f(m,k) = coeff_A( m )*I_f(m,k);
    end
end

%% sawyisi pirobebis sakoordinato funqciebze namravlis integrali [-1,1]

int0 = zeros(N,1);
int1 = zeros(N,1);

%% integrali psi0*phi(m,x)--ze [-1,1]

for i = 1:n
    y_s(i) = psi0(x(i));
end
d1     = 2*d2psi0(x(1));
dn     = 2*d2psi0(x(n));
spline_function_approximation;
for j = 2:n
    A_s(j) = 1/(6*h)*(M1(j) - M1(j-1));
    B_s(j) = 0.5*M1(j) - 3*x(j)*A_s(j);
    C_s(j) = -h*h*A_s(j) - x(j)*B_s(j) - 0.5*x(j - 1)*M1(j) + ...
        1/h*(y_s(j) - y_s(j - 1));
    D_s(j) = -x(j)*(x(j)*x(j) + h*h)*A_s(j) - (x(j)*x(j) + ...
        1/3*h*h)*B_s(j) - x(j)*C_s(j) + 1/6*h*h*M1(j) + y_s(j);
end
for m = 1:N
    for j = 2:n
        int0(m) = int0(m) + A_s(j)*integr_basis3( x(j),m,h ) + ...
            B_s(j)*integr_basis2( x(j),m,h ) + ...
            C_s(j)*integr_basis1( x(j),m,h ) + ...
            D_s(j)*integr_basis0( x(j),m,h );
    end
    int0(m) = coeff_A( m )*int0(m);
end
```

```
%% integrali psi1*phi(m,x)-ze [-1,1]

for i = 1:n
    y_s(i) = psi1(x(i));
end
d1     = 2*d2psi1(x(1));
dn     = 2*d2psi1(x(n));
spline_function_approximation;
for j = 2:n
    A_s(j) = 1/(6*h)*(M1(j) - M1(j-1));
    B_s(j) = 0.5*M1(j) - 3*x(j)*A_s(j);
    C_s(j) = -h*h*A_s(j) - x(j)*B_s(j) - 0.5*x(j - 1)*M1(j) + ...
        1/h*(y_s(j) - y_s(j - 1));
    D_s(j) = -x(j)*(x(j)*x(j) + h*h)*A_s(j) - (x(j)*x(j) + ...
        1/3*h*h)*B_s(j) - x(j)*C_s(j) + 1/6*h*h*M1(j) + y_s(j);
end
for m = 1:N
    for j = 2:n
        int1(m) = int1(m) + A_s(j)*integr_basis3( x(j),m,h ) + ...
            B_s(j)*integr_basis2( x(j),m,h ) + ...
            C_s(j)*integr_basis1( x(j),m,h ) + ...
            D_s(j)*integr_basis0( x(j),m,h );
    end
    int1(m) = coeff_A( m )*int1(m);
end

clear i j m k

%% cholesky-is dekompoziciis algoriTmi yoveli k SrisaTvis

d = zeros(N,1);

k = 1;

b = int0;

while k <= M
    switch k
        case 1
            d(1) = coeff_C( 1 );
            z(1) = b(1);
            d(2) = coeff_C( 2 );
            z(2) = b(2);
            for m = 3:N
                d(m) = coeff_C( m ) - ...
                    coeff_B( m - 1 )*coeff_B( m - 1 )/d(m - 2);
                z(m) = b(m) + coeff_B( m - 1 )/d(m - 2)*z(m - 2);
            end
            y = z./d;
            c1(N)      = y(N);
```

```
            c1(N - 1) = y(N - 1);
            for m = (N - 2):-1:1
                c1(m) = y(m) + coeff_B( m + 1 )/d(m)*c1(m + 2);
            end
            q1 = alpha + beta*c1'*c1;
            u(:,k)        = basis*c1;
            u_zusti(:,k) = U(x,t(k));
        case 2
            b = int0 + tau*int1 - ...
                0.5*tau*tau*(q1*c1 - I_f(:,k - 1));
            z(1) = b(1);
            z(2) = b(2);
            for m = 3:N
                z(m) = b(m) + coeff_B( m - 1 )/d(m - 2)*z(m - 2);
            end
            y = z./d;
            c2(N)      = y(N);
            c2(N - 1) = y(N - 1);
            for m = (N - 2):-1:1
                c2(m) = y(m) + coeff_B( m + 1 )/d(m)*c2(m + 2);
            end
            q2 = alpha + beta*c2'*c2;
            u(:,k)        = basis*c2;
            u_zusti(:,k) = U(x,t(k));
        otherwise
            b(1)      = -2*coeff_C( 1 )*(c1(1) - 2*c2(1)) - ...
                tau*tau*(q2*c1(1) - 2*I_f(1,k - 1)) + ...
                2*coeff_B( 2 )*(c1(3) - 2*c2(3));
            b(2)      = -2*coeff_C( 2 )*(c1(2) - 2*c2(2)) - ...
                tau*tau*(q2*c1(2) - 2*I_f(2,k - 1)) + ...
                2*coeff_B( 3 )*(c1(4) - 2*c2(4));
            b(N - 1) = 2*coeff_B( N - 2 )*(c1(N - 3) - 2*c2(N - 3)) - ...
                2*coeff_C( N - 1 )*(c1(N - 1) - 2*c2(N - 1)) - ...
                tau*tau*(q2*c1(N - 1) - 2*I_f(N - 1,k - 1));
            b(N)      = 2*coeff_B( N - 1 )*(c1(N - 2) - 2*c2(N - 2)) - ...
                2*coeff_C( N )*(c1(N) - 2*c2(N)) - ...
                tau*tau*(q2*c1(N) - 2*I_f(N,k - 1));
            for m = 3:(N - 2)
                b(m) = 2*coeff_B( m - 1 )*(c1(m - 2) - 2*c2(m - 2)) - ...
                    2*coeff_C( m )*(c1(m) - 2*c2(m)) - ...
                    tau*tau*(q2*c1(m) - 2*I_f(m,k - 1)) + ...
                    2*coeff_B( m + 1 )*(c1(m + 2) - 2*c2(m + 2));
            end
            d(1) = 2*coeff_C( 1 ) + tau*tau*q2;
            z(1) = b(1);
            d(2) = 2*coeff_C( 2 ) + tau*tau*q2;
            z(2) = b(2);
            for m = 3:N
                d(m) = 2*coeff_C( m ) + tau*tau*q2 - ...
                    4*coeff_B( m - 1 )*coeff_B( m - 1 )/d(m - 2);
                z(m) = b(m) + 2*coeff_B( m - 1 )/d(m - 2)*z(m - 2);
```

```
            end
            y = z./d;
            c3(N) = y(N);
            c3(N - 1) = y(N - 1);
            for m = (N - 2):-1:1
                c3(m) = y(m) + 2*coeff_B( m + 1 )/d(m)*c3(m + 2);
            end
            q3 = alpha + beta*c3'*c3;
            u(:,k)        = basis*c3;
            u_zusti(:,k) = U(x,t(k));

            c1 = c2;
            c2 = c3;
            q2 = q3;
    end
    k = k + 1;
end

error = max(max(abs(u - u_zusti)));
fprintf('max(max(abs(u - u_zusti))) = %.16f\n', error);
```

ზუსტი და მიახლოებითი ამონახსნის აგება დროის ყველა შრეზე:

1. ზუსტი ამონახსნი

   exact_solution.m

```
figure,hold on
grid on
title('zusti amonaxsni','fontname','acadnusx','fontsize',16);
xlabel('abscisaTa RerZi','fontname','acadnusx','fontsize',14);
ylabel('ordinatTa RerZi','fontname','acadnusx','fontsize',14);
axis([-1 1 min(min(u_zusti)) max(max(u_zusti))])

for k = 1:M
    plot(x,u_zusti(:,k),'LineWidth',1)
    pause(0.001)
end
```

2. მიახლოებითი ამონახსნი

   approx_solution.m

```
figure,hold on
grid on
title('miaxloebiTi amonaxsni','fontname','acadnusx','fontsize',16);
xlabel('abscisaTa RerZi','fontname','acadnusx','fontsize',14);
ylabel('ordinatTa RerZi','fontname','acadnusx','fontsize',14);
axis([-1 1 min(min(u)) max(max(u))])

for k = 1:M
    plot(x,u(:,k),'LineWidth',1)
    pause(0.001)
end
```

ზუსტი და მიახლოებითი ამონახსნის ანიმაცია

animation_of_exact_and_approx_solution.m

```
figure(1)
filename = 'exact_and_approx_animation_legendre_basis.gif';
for k = 1:M
      y1 = u_zusti(:,k);
      y2 = u(:,k);
      plot(x,y1,'b',x,y2,'r','LineWidth',2)
      axis([-1 1 min(min(min(u_zusti)),min(min(u))) ...
          max(max(max(u_zusti)),max(max(u)))])
      drawnow
      frame = getframe(1);
      im = frame2im(frame);
      [imind,cm] = rgb2ind(im,256);
      if k == 1;
          imwrite(imind,cm,filename,'gif', 'Loopcount',inf);
      else
          imwrite(imind,cm,filename,'gif','WriteMode','append');
      end
end
```

A.3 პროგრამული კოდი სივრცით ორთგანზომილებიანი ამოცანისათვის. საკოორდინატო ფუნქციებად აღებულია სინუსები

პროგრამას აქვს შემდეგი სახე:

sine_basis_simpsons_rule_for_2d.m

```
clc
clear all
format longG

%% funqciebis Setana

syms psi0(x,y) psi1(x,y) f(x,y,t) x y m n U(x,y,t) diff_t_U(x,y,t) ...
    diff_tt_U(x,y,t) diff_x_U(x,y,t) diff_y_U(x,y,t) diff_xx_U(x,y,t) ...
    diff_yy_U(x,y,t) F(x,y,t,m,n) Psi0(x,y,m,n) Psi1(x,y,m,n)

%% alpha and beta

alpha = 1;
beta  = 1;

%% zusti amonaxsni

U(x,y,t) = x*(1 - x)*sin(0.4*pi*x)*y*(1 - y)*sin(0.6*pi*y)*exp(t);
diff_t_U(x,y,t)  = diff(U,t,1);
diff_tt_U(x,y,t) = diff(U,t,2);
diff_x_U(x,y,t)  = diff(U,x,1);
diff_y_U(x,y,t)  = diff(U,y,1);
```

```
diff_xx_U(x,y,t) = diff(U,x,2);
diff_yy_U(x,y,t) = diff(U,y,2);
U                = matlabFunction(U);

%% gantolebis marjvena mxare

f(x,y,t) = diff_tt_U(x,y,t) - (alpha + beta*...
    (int(int(diff_x_U(x,y,t)*diff_x_U(x,y,t),x,0,1),y,0,1) + ...
    int(int(diff_y_U(x,y,t)*diff_y_U(x,y,t),x,0,1),y,0,1)))*...
    (diff_xx_U(x,y,t) + diff_yy_U(x,y,t));

%% sawyisi pirobebi

psi0(x,y) = U(x,y,0);
psi1(x,y) = diff_t_U(x,y,0);

%% integralqveSa funqciebi

F(x,y,t,m,n)   = f(x,y,t)*sin(m*pi*x)*sin(n*pi*y);
Psi0(x,y,m,n) = psi0(x,y)*sin(m*pi*x)*sin(n*pi*y);
Psi1(x,y,m,n) = psi1(x,y)*sin(m*pi*x)*sin(n*pi*y);

%% simbluri funqciebis gadayvana "Handle" funqciebad

diff_t_U   = matlabFunction(diff_t_U);
diff_tt_U = matlabFunction(diff_tt_U);
diff_x_U   = matlabFunction(diff_x_U);
diff_y_U   = matlabFunction(diff_y_U);
diff_xx_U = matlabFunction(diff_xx_U);
diff_yy_U = matlabFunction(diff_yy_U);

f          = matlabFunction(f);
psi0       = matlabFunction(psi0);
psi1       = matlabFunction(psi1);

F          = matlabFunction(F);
Psi0       = matlabFunction(Psi0);
Psi1       = matlabFunction(Psi1);

clear x y t m n

%% t RerZebis dayofa tol qveSualedebad h da tau bijad

T   = 2;
M   = 101;
tau = T/(M - 1);
t   = (0:tau:T)';

%% x da y RerZebis dayofa tol Sualedebad
r = 51;
s = r;
```

```
h1 = 1/(r - 1);
h2 = 1/(s - 1);
x = (0:h1:1)';
y = (0:h2:1)';

%% sakoordinato funqciebis raodenoba

N = 5;

%% matricis dagenerireba simpsonis formulisaTvis

s1 = ones(r,1);
s2 = ones(1,s);

for i = 2:(s - 1)
    if mod(i,2) == 0
        s1(i) = 4;
    else
        s1(i) = 2;
    end
end

for i = 2:(r - 1)
    if mod(i,2) == 0
        s2(i) = 4;
    else
        s2(i) = 2;
    end
end

W  = s1*s2;

clear s1 s2

%% integralebis gamoTvla simpsonis formuliT

simp_coeff = h1*h2/9;

int0 = zeros(N,N);
int1 = zeros(N,N);
I_f  = zeros(N,N,M - 1);

for m = 1:N
    for n = 1:N
        for i = 1:r
            for j = 1:s
                int0(m,n) = int0(m,n) + W(i,j)*Psi0(x(i),y(j),m,n);
                int1(m,n) = int1(m,n) + W(i,j)*Psi1(x(i),y(j),m,n);
                for k = 1:(M - 1)
                    I_f(m,n,k) = I_f(m,n,k) + W(i,j)*F(x(i),y(j),t(k),m,n);
                end
```

```
            end
        end
    end
end

int0 = simp_coeff*int0;
int1 = simp_coeff*int1;
I_f  = simp_coeff*I_f;

%% bazisebis generacia

basis2 = zeros(N,s);

for n = 1:N
    for i = 1:s
        basis2(n,i) = sin(n*pi*y(i));
    end
end

basis1 = basis2';

%% c koeficientebi

c1 = zeros(N,N);
c2 = zeros(N,N);
c3 = zeros(N,N);

%% miaxloebiTi amonaxsni

u = zeros(r,s,k);

%% zusti amonaxsni

u_zusti = zeros(r,s,k);

for k = 1:M
    switch k
        case 1
            c1 = 4*int0;
            q1 = 0;
            for m = 1:N
                for n = 1:N
                    q1 = q1 + (m*m + n*n)*c1(m,n)*c1(m,n);
                end
            end
            q1 = alpha + 0.25*pi*pi*beta*q1;
            for i = 1:r
                for j = 1:s
                    u(i,j,k)       = basis1(i,:)*c1*basis2(:,j);
                    u_zusti(i,j,k) = U(x(i),y(j),t(k));
                end
```

```
            end
        case 2
            q2 = 0;
            for m = 1:N
                for n = 1:N
                    c2(m,n) = (1 - 0.5*pi*pi*(m*m + n*n)*tau*tau*q1)*...
                        c1(m,n) + 2*tau*(2*int1(m,n) + ...
                        tau*I_f(m,n,k - 1));

                    q2 = q2 + (m*m + n*n)*c2(m,n)*c2(m,n);
                end
            end
            q2 = alpha + 0.25*pi*pi*beta*q2;
            for i = 1:r
                for j = 1:s
                    u(i,j,k)       = basis1(i,:)*c2*basis2(:,j);
                    u_zusti(i,j,k) = U(x(i),y(j),t(k));
                end
            end
        otherwise
            q3 = 0;
            for m = 1:N
                for n = 1:N
                    c3(m,n) = 8*tau*tau/(2 + pi*pi*(m*m + n*n)*tau*tau*...
                        q2)*I_f(m,n,k - 1) - c1(m,n) + ...
                        4/(2 + pi*pi*(m*m + n*n)*tau*tau*q2)*c2(m,n);

                    q3 = q3 + (m*m + n*n)*c3(m,n)*c3(m,n);
                end
            end
            q3 = alpha + 0.25*pi*pi*beta*q3;
            for i = 1:r
                for j = 1:s
                    u(i,j,k)       = basis1(i,:)*c3*basis2(:,j);
                    u_zusti(i,j,k) = U(x(i),y(j),t(k));
                end
            end
            q2 = q3;
            c1 = c2;
            c2 = c3;
    end
end

error = max(max(max(abs(u - u_zusti))));
fprintf('max(max(max(abs(u - u_zusti)))) = %.16f\n', error);
```

ზუსტი და მიახლოებითი ამონახსნის გრაფიკის ბეჭდვა დროის კონკრეტულ შრეზე (შრეებზე):

1. ზუსტი ამონახსნი

   plot_layer_exact_solution_2d.m

```
z1 = u_zusti(:,:,26);
```

```
z2 = u_zusti(:,:,51);
z3 = u_zusti(:,:,76);
z4 = u_zusti(:,:,101);

figure
grid on
subplot(2,2,1)
axis([0 1 0 1 min(min(min(u_zusti))) max(max(max(u_zusti)))])
surf(x,y,z1);
title('t = 0.5')

subplot(2,2,2)
axis([0 1 0 1 min(min(min(u_zusti))) max(max(max(u_zusti)))])
surf(x,y,z2);
title('t = 1')

subplot(2,2,3)
axis([0 1 0 1 min(min(min(u_zusti))) max(max(max(u_zusti)))])
surf(x,y,z3);
title('t = 1.5')

subplot(2,2,4)
axis([0 1 0 1 min(min(min(u_zusti))) max(max(max(u_zusti)))])
surf(x,y,z4);
title('t = 2')
```

2. მიახლოებითი ამონახსნი

plot_layer_approx_solution_2d.m

```
z1 = u(:,:,26);
z2 = u(:,:,51);
z3 = u(:,:,76);
z4 = u(:,:,101);

figure
grid on
subplot(2,2,1)
axis([0 1 0 1 min(min(min(u))) max(max(max(u)))])
surf(x,y,z1);
title('t = 0.5')

subplot(2,2,2)
axis([0 1 0 1 min(min(min(u))) max(max(max(u)))])
surf(x,y,z2);
title('t = 1')

subplot(2,2,3)
axis([0 1 0 1 min(min(min(u))) max(max(max(u)))])
surf(x,y,z3);
title('t = 1.5')

```

```
subplot(2,2,4)
axis([0 1 0 1 min(min(min(u))) max(max(max(u)))])
surf(x,y,z4);
title('t = 2')
```

ზუსტი და მიახლოებითი ამონახსნის ანიმაცია:

1. ზუსტი ამონახსნი

   animation_of_exact_solution_2d.m

```
figure(1)
filename = 'animation_sine_basis_exact_2d.gif';
for k = 1:M
      Z = u_zusti(:,:,k);
      surf(x,y,Z);
      axis([0 1 0 1 min(min(min(u_zusti))) max(max(max(u_zusti)))]);
      drawnow
      frame = getframe(1);
      im = frame2im(frame);
      [imind,cm] = rgb2ind(im,256);
      if k == 1;
          imwrite(imind,cm,filename,'gif', 'Loopcount',inf);
      else
          imwrite(imind,cm,filename,'gif','WriteMode','append');
      end

      pause(0.024)

end
```

2. მიახლოებითი ამონახსნი

   animation_of_approx_solution_2d.m

```
figure(1)
filename = 'animation_sine_basis_approx_2d.gif';
for k = 1:M
      Z = u(:,:,k);
      surf(x,y,Z);
      axis([0 1 0 1 min(min(min(u))) max(max(max(u)))]);
      drawnow
      frame = getframe(1);
      im = frame2im(frame);
      [imind,cm] = rgb2ind(im,256);
      if k == 1;
          imwrite(imind,cm,filename,'gif', 'Loopcount',inf);
      else
          imwrite(imind,cm,filename,'gif','WriteMode','append');
      end

      pause(0.024)

end
```