

Primljen / Received: 12.6.2017.

Ispravljen / Corrected: 6.10.2017.

Prihvaćen / Accepted: 15.11.2017.

Dostupno online / Available online: 10.1.2018.

Increasing efficiency of iterative application of the force density method

Authors:



Elizabeta Šamec, MCE
University of Zagreb
Faculty of Civil Engineering
Department for Engineering of Mechanics
esamec@grad.hr



Prof. **Krešimir Fresl**, PhD. CE
University of Zagreb
Faculty of Civil Engineering
Department for Engineering of Mechanics
fresl@grad.hr



Maja Baniček, MCE
University of Zagreb
Faculty of Civil Engineering
Department for Engineering of Mechanics
mbanicek@grad.hr

Preliminary report

Elizabeta Šamec, Krešimir Fresl, Maja Baniček

Increasing efficiency of iterative application of the force density method

The method for reducing total computational time in iterative application of the force density method, intended to attain prescribed force values in cables or prescribed lengths of cable segments, is described in the paper. In each step of the iterative procedure, linear systems are solved with the accuracy that takes into account differences between the calculated and required values. The rule that prevents excessively fast increase in accuracy, while maintaining it high enough not to compromise the convergence of the iterative process, is proposed.

Key words:

cable net structure, form finding, force density method, generalized minimal net, iterative method, accuracy

Prethodno priopćenje

Elizabeta Šamec, Krešimir Fresl, Maja Baniček

Povećanje učinkovitosti iteracijske primjene metode gustoće sila

U radu je opisan postupak skraćivanja trajanja proračuna pri iteracijskoj primjeni metode gustoća sila kojom se u fazi nalaženja oblika prednapete konstrukcije od užadi postižu tražene vrijednosti sila ili duljine odsječaka kabela. U svakom se iteracijskom koraku linearni sustavi jednadžbi rješavaju s točnošću koja u obzir uzima razlike izračunanih i traženih vrijednosti. Predloženo je pravilo koje sprečava prebrzo povećavanje točnosti, održavajući je istodobno dovoljno velikom da se konvergencija ne naruši.

Ključne riječi:

prednapete konstrukcije od užadi, nalaženje oblika, metoda gustoća sila, iteracijski postupak, točnost

Vorherige Mitteilung

Elizabeta Šamec, Krešimir Fresl, Maja Baniček

Steigerung der Effizienz der iterativen Anwendung der Kraftdichtemethode

In der Arbeit wird das Verfahren zur Reduzierung der gesamten Berechnungszeit bei der iterativen Anwendung der Kraftdichtemethode beschrieben, mit welcher in der Phase der Formfindung von vorgespannten Seilkonstruktionen die geforderten Kraftwerte oder Kabelabschnittslängen erreicht werden. In jedem Iterationsschritt werden die linearen Gleichungssysteme mit einer Genauigkeit gelöst, welche die Differenz zwischen den errechneten und geforderten Werten berücksichtigt. Es wurde eine Regel vorgeschlagen, die einen zu schnellen Anstieg der Genauigkeit verhindert, sie aber auf einer solchen Ebene aufrechterhält, dass die Konvergenz nicht beeinträchtigt wird.

Schlüsselwörter:

vorgespannte Seilkonstruktionen, Formfindung, Kraftdichtemethode, Iterationsverfahren, Genauigkeit

1. Introduction

Cables have negligible flexural stiffness. Consequently, when subjected to forces perpendicular to their axes, they change their configuration to enable development of equilibrating internal tensile forces. The shape of a flexible cable structure is maintained by arranging cables into a net that "describes" a surface with a negative Gaussian curvature. Additionally, cables should be prestressed for tensile forces to be preserved at all load combinations. The shape of cable net structures is determined by the laws of statics: designing the structural form is a search for the system of forces in equilibrium [1, 2]. The form finding process is therefore the first stage in the design of prestressed cable nets (Section 2.1). It is a non-standard, inverse task, since in a "standard" design the shape of a structure is already specified so that "only" a compatible distribution of internal forces needs to be determined.

Furthermore, computational model equations are highly nonlinear (Section 2.2). In the force density method, developed in the early 1970s for the design of the Olympic Games facilities in Munich, the problem was linearized [3-5] (Section 3.1). However, the concept of force density, as a ratio of one static and one geometric magnitude, has arisen from formal manipulation of mathematical expressions. Hence it is often difficult to select a distribution of force densities that complies with aesthetic, functional and structural requirements.

Using the nonlinear force density method, based on the least squares method and also introduced in article [3], some structural requirements can be satisfied in a systematic way: attainment of prescribed force values in selected cables or cable segments, attainment of prescribed cable segment lengths in the equilibrium net configuration, and attainment of prescribed unstrained cable segment lengths. An extension to the nonlinear force density method, enabling attainment of the prescribed support reactions, and indirectly assigning position of selected (free) nodes, is given in reference [6]. Another extension, described in [7], enables assignment of coordinates (all three, two, or just one) of some or all nodes.

As indicated in references [1, 2], an iterative application of the linear force density method enables either partial or complete attainment of prescribed force values in cables, while in [8] the iterative procedure is extended in order to attain prescribed lengths of the equilibrium configuration segments (Sections 2.3 and 3.2). An experienced designer can satisfy other, usually non-structural, requirements in a series of attempts through interactive work. An iterative application of the force density method sometimes demands time-consuming calculations and can therefore be inappropriate for interactive work. The objective of our work is to reduce duration of iterative computations (Section 3.3, with programming implementation in Section 4, and examples in Section 5). Initial results of this research were presented at the 40th Solid Mechanics Conference [9].

2. Mathematical model for form finding

2.1. Form finding

The notion of form finding for a prestressed cable net implies determination of its initial equilibrium configuration, before application of service loads and without taking self weight into account. The notion of configuration involves both the geometric shape and distribution of prestressing forces in cables; in [10] the authors use the term "initial equilibrium problem" instead of the "form finding problem". A comprehensive review of various cable net form finding methods is given in reference [11]. An interesting idea involving application of graphical statics and reciprocal diagrams, which presupposes interactive work, is described in reference [12].

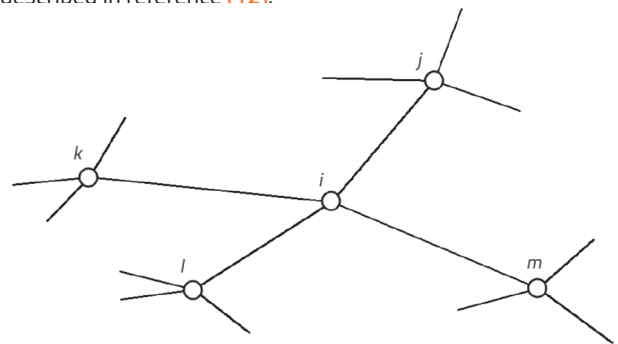


Figure 1. Computational model of a cable net

Taking into account some simplifying assumptions (explained in more detail in [8]), cable nets will be modeled as space trusses [1,3]: the points in which cables cross each other and where they are connected to the ground or to rigid structural elements are spherical joints, while cable segments between the crossing points are bars pinned at their ends (Figure 1). Bar connecting joints i and j will be denoted $\{i, j\}$. Joints that are not supports will be called free nodes.

Basic variables in the cable net form finding are: disposition of cables, positions of their crossing points in space, disposition and spatial position of supports, and values of prestressing forces in cables or their ratios (adapted from [10] for cable nets). The choice of the form finding procedure will depend on which of the variables are given and which are unknown. The disposition of cables, which determines the disposition and connectivity of their crossing points, *i.e.* the topology of the system of pinned bars in the computational model, is almost always given in advance. As the shape of the net is an equilibrium configuration of prestressing forces in the system of bars, the coordinates of free joints are the principal unknowns. Prestressing force values can be either given in advance, or obtained as unknowns by the conditions of nodal equilibrium or, possibly, by satisfying additional kinematic constraints. The shape of the net can be varied by changing the ratios of force values. The shape of the net can also be influenced by adding/removing supports or by altering their positions.

2.2. Nodal equilibrium equations

Mathematical formulation of the cable net form finding is very simple: it all comes to equilibrium equations of free nodes that are acted upon only by prestressing forces in connected bars. The set of bars connected to the node i will be denoted β_i . For each free node i , three scalar equilibrium equations, representing the vanishing of force projection sums on three coordinate axes, can be written as follows:

$$\sum_{\{i,j\} \in \beta_i} S_{i,j} \frac{x_j - x_i}{\ell_{i,j}} = 0, \quad \sum_{\{i,j\} \in \beta_i} S_{i,j} \frac{y_j - y_i}{\ell_{i,j}} = 0, \quad \sum_{\{i,j\} \in \beta_i} S_{i,j} \frac{z_j - z_i}{\ell_{i,j}} = 0 \quad (1)$$

where S_{ij} denotes the value of the force in connected bar $\{i, j\}$ and $\ell_{i,j}$ is the length of that bar. Since

$$\ell_{i,j} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2} \quad (2)$$

these equations are nonlinear.

If the net contains n free nodes, the obtained system will have $3n$ equations of the form shown in (1). If b is the number of bar elements, the number of potential unknowns will be $3n+b$, since every set consisting of n coordinate triples (x_i, y_i, z_i) and b force values S_{ij} and satisfying these equations, forms an equilibrium configuration. Since equations (1) do not contain coefficients that express constitutive relations between cable extensions and force values, the form finding problem is a pure static problem.

2.3. Generalized minimal nets and kinematic constrains

Equilibrium equations (1) can be interpreted as a condition for the minimum of the function ε

$$\varepsilon(\{x_k, y_k, z_k\}_{k \in N}) = \sum_{\{i,j\} \in \beta} S_{i,j} \ell_{i,j} \quad (3)$$

where N and β denote set of free nodes and set of bars, respectively. The nets with specified prestressing force values, which satisfy that condition, can be called generalized minimal nets. Namely, if the values of all forces are equal, $S_{ij} = S$, the solution of equations (1) is the shape for which the sum of cable lengths is smaller than in any other shape of the net with the same topology [1, 2, 8, 13].

The possibility of assigning different force values in different cables increases the number of possible cable net forms. For example, if forces in other cables stay fixed, the selected cable will tighten as its force increases and its length will decrease. At the same time, the spatial polygonal line of the cable axis straightens and approaches the rectilinear connection of its ends (example in Sections 5.1 and 5.2).

To enable the (generalized) minimal configuration with specified force values, the sliding of cables one over another during the prestressing procedure should not be prevented (Figure 2) [1, 2].

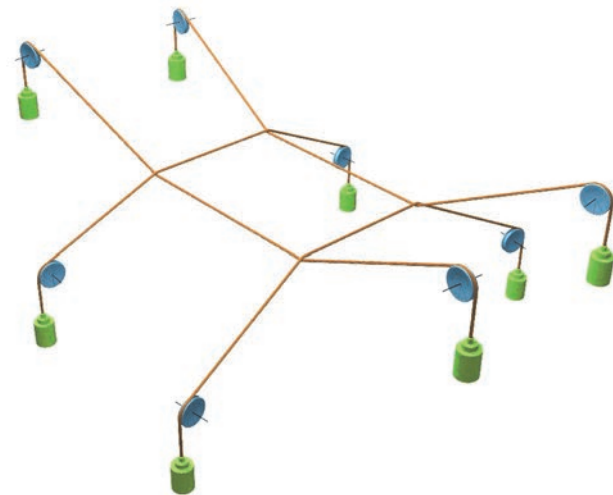


Figure 2. Generalized minimal net

However, it may happen that two or more nodes slide along a cable into a single point, regardless of the values of forces. The unconfined approaching and merging of nodes cannot be prevented by increasing the force value in that cable, compared to forces in sliding cables.

In our computational model, the sliding along a cable can be prevented by specifying lengths of segments between the crossing points, i.e. the lengths of bars into which a cable is divided. But, if we specify the bar lengths, the force values required to accomplish them become the unknowns [8]. Using Lagrange multipliers $\lambda_{i,j}$ the specified bar lengths can be expressed as kinematic constraints, and so the function ε becomes

$$\varepsilon_c(\{x_k, y_k, z_k\}_{k \in N}, \{\lambda_{i,j}\}_{\{i,j\} \in \beta_c}) = \sum_{\{i,j\} \in \beta} S_{i,j} \ell_{i,j} + \sum_{\{i,j\} \in \beta_c} \lambda_{i,j} (\ell_{i,j} - \bar{\ell}_{i,j}) \quad (4)$$

where β_c is the set of bars with specified length $\bar{\ell}_{i,j}$. Since the number of Lagrange multipliers $\lambda_{i,j}$ is equal to the number of unknown force values, the condition for stationary points of the function ε_c is a system with an appropriate number of equations.

The application of Newton–Krylov methods in solving the system of equations (1) (function ε minimum condition), and the system of stationarity conditions of function ε_c , is analysed in reference [8]. In the first case, the analysis confirmed the well-known fact already described in [2, 14, 15]: the duration of calculation depends upon the assumed initial form, and the domain of convergence ("basin of attraction") exhibits a highly irregular, fractal shape. On the other hand, the problem of stationarity of function ε_c is a saddle point problem, and so the duration of calculation is significantly extended, and the domain of convergence is reduced. Furthermore, unlike the minimum of the convex function ε , the saddle point is not unique, and so the calculation, if it converges, can converge to the solution the realisation of which is physically impossible.

3. Force density method

3.1. Initial version of the linear force density

By specifying ratios

$$q_{i,j} = \frac{S_{i,j}}{\ell_{i,j}} \tag{5}$$

for each bar nonlinear equations (1) are linearized:

$$\sum_{\{i,j\} \in \beta} q_{i,j}(x_j - x_i) = 0, \quad \sum_{\{i,j\} \in \beta} q_{i,j}(y_j - y_i) = 0, \quad \sum_{\{i,j\} \in \beta} q_{i,j}(z_j - z_i) = 0 \tag{6}$$

The force-to-length ratios q_{ij} are called force densities [3-5]. If instead of force values S_{ij} force densities q_{ij} are specified, unknowns are force values and free node coordinates, which gives $b+3n$ unknowns. Unknown force values and unknown nodal coordinates are connected by additional b equations (5). But the specified values q_{ij} are constant coefficients in equations (6), and so the system of $3n$ equilibrium equations is decomposed into three independent systems with n unknowns. The unknowns of the first, second and third systems are $\{x_k\}_{k \in N'}$, $\{y_k\}_{k \in N'}$ and $\{z_k\}_{k \in N'}$ respectively. The systems have the same system matrix, but they differ in vectors on the right-hand side, since their components are $q_{i,s}x_s$, $q_{i,s}y_s$ and $q_{i,s}z_s$ for $s \in S$, where S is a set of supports. By obtaining free node coordinates, we can calculate bar lengths ℓ_{ij} and thereafter force values, since (5) gives $S_{ij} = q_{ij}\ell_{ij}$.

3.2. Iterated force density method

Each solution of the system (6), for any one of ω^b possible distributions of force densities in bars of the net of a given topology, is an equilibrium configuration. However, it is difficult to predict the force density distribution that will give the imagined net shape or the required distribution of prestressing forces.

An iterative application of the force density method will be called the *iterated force density method* (IFDM). In that procedure, the linear force density method is applied in each step, while force densities are determined in a given step based on the requirements and results from the preceding step.

To attain the force value $\bar{S}_{i,j}$ in the bar $\{i,j\}$, the force density in the bar is calculated in the k -th iteration step according to the expression

$$q_{i,j}^{(k)} = q_{i,j}^{(k-1)} \frac{\bar{S}_{i,j}}{S_{i,j}^{(k-1)}} \quad \text{or} \quad q_{i,j}^{(k)} = \frac{\bar{S}_{i,j}}{\ell_{i,j}^{(k-1)}} \tag{7}$$

where $S_{i,j}^{(k-1)}$ is the force value calculated in the preceding step. Force values can be different in different bars [1, 2]. A special case are minimal or geodetic nets where force values are distributed uniformly [13]: $\bar{S}_{i,j} = \bar{S}$.

Analogously, the required length $\bar{\ell}_{i,j}$ of the bar $\{i,j\}$ and the required distance $\bar{\ell}_{i,j}$ of nodes i and j can be attained in the equilibrium configuration of a net by calculating the force density in the k -th step according to one of the following expressions

$$q_{i,j}^{(k)} = q_{i,j}^{(k-1)} \frac{\ell_{i,j}^{(k-1)}}{\bar{\ell}_{i,j}} \quad \text{or} \quad q_{i,j}^{(k)} = \frac{S_{i,j}^{(k-1)}}{\bar{\ell}_{i,j}} \tag{8}$$

where $\ell_{i,j}^{(k-1)}$ is the bar length/node distance calculated in the preceding step [8].

Both the force value in the bar and the bar length cannot be assigned at the same time, but it is possible to assign the approximate force value and then, after few iteration steps, switch over to obtaining the length.

The iteration is terminated when

$$e_S^{(k)} = \max_{\{i,j\} \in \beta} (|S_{i,j}^{(k)} - \bar{S}_{i,j}|) < \tau_S \quad \text{and} \quad e_\ell^{(k)} = \max_{\{i,j\} \in \beta} (|\ell_{i,j}^{(k)} - \bar{\ell}_{i,j}|) < \tau_\ell \tag{9}$$

where τ_S and τ_ℓ are the prescribed tolerances, which can differ in different bars.

Since every solution of equations (6) is an equilibrium configuration, the described iterative procedure converges towards the target solution through a sequence of equilibrium configurations. If the iterative procedure is terminated before the required tolerance is achieved, or even if it cannot be achieved, the net obtained is still in equilibrium. Another advantage of iterative procedure is that in addition to the required force values, the required lengths of cable segments can also be obtained without introduction of Lagrange multipliers. However, numerical experiments have shown that the examples in which force values and lengths of cable segments are given demand more iteration steps than the examples with an approximately equal number of segments in which only force values are prescribed. The differences are however not so large compared to Newton-Krylov methods.

It is stated in reference [8] that the described procedure is very rapid. This claim needs to be rewritten to be more accurate. The procedure is faster than the Newton-Krylov methods, but in the initial version, for form finding of cable nets with a large number of segments in real time, it takes quite a long time to complete if a higher accuracy is needed. A possible solution for speeding up this process is shown in the next section.

3.3. Reduction of computation time

Three linear equation systems with equivalent system matrices are solved in each step. The LU decomposition, which enables simultaneous back-substitution of several right-hand side vectors, is therefore a particularly suitable solution method [16]. Since these matrices are sparse, a sparse variant of the LU decomposition should be used for larger systems [17]. Nevertheless, the limited fill-in of the matrices will still occur, and our numerical experiments have shown that it can be significant.

Iterative procedures for solving linear systems do not require any fill-in of system matrices. The group of Krylov procedures is very efficient [18] although three systems must be solved separately when these procedures are applied. The entire procedure of iterative application of the force density method will be called the outer loop, while the iterative procedure for solving the linear system will be called the inner loop.

The tolerance τ_{eq} needs to be specified to enable iterative solution of linear systems. This tolerance depends on tolerances for required bar lengths and bar forces, τ_l and τ_s , respectively, contained in the criteria for the outer loop termination. Lengths and force values are functions of nodal coordinates. The error expected in the determination of these coordinates is greater than τ_{eq} . Namely, in the condition for termination of an iterative procedure, the tolerance τ_{eq} is compared with the norm of the residual, $\|r^{(k)}\| < \tau_{eq}$, where κ is a step of the inner loop. Since the error of the solution and the norm of the residual are related by $\|e^{(k)}\| = \|A^{-1}r^{(k)}\| \leq \|A^{-1}\| \|r^{(k)}\|$, the error of the solution can be up to $\|A^{-1}\|$ times larger than the residual. In addition, the length calculation error can be up to $2\sqrt{3}$ times larger than the error with which nodal coordinates are determined. Systems (6) should therefore be solved at least with the tolerance of

$$\tau_{eq} = \min(\tau_S / \alpha, \tau_l / \alpha) \quad (10)$$

where α is an estimate of A^{-1} (as such estimates are rather crude, there is no need for the factor $2\sqrt{3}$).

The application of Krylov methods reduces duration of the outer loop significantly. First of all, in an outer loop step, the solutions obtained in the preceding step are taken as the first approximations of solutions of linear systems. And, more importantly, the idea of the Newton-Krylov or, more evocatively called, inexact Newton methods, is borrowed. This provides a balance between accuracy of the linear system solutions and the amount of computation done in a single step of the outer loop [19]. Namely, if the computed force values or bar lengths are far from those required, it makes sense to solve linear systems (6) only approximately, by reducing accuracy with an increase of the deviation from the required value. It should be noted that the aforementioned advantage of the iterative application of the force density method is thereby sacrificed: since linear systems are in almost all steps solved only approximately, the obtained configurations, except for those in the last steps, are not in an equilibrium.

Therefore the tolerance with which the systems (6) are solved is gradually decrease, from the relatively loose initial tolerance τ_{max} to the specified tight tolerance τ_{eq} that provides the final equilibrium configuration. If we wish to solve the system in the k -th step of the outer loop with the tolerance that reflects "distances" of calculated force values and lengths from the required ones, $\tau^{(k)}$ needs to depend on errors $e_S^{(k)}$ and $e_l^{(k)}$. Since these errors are not known before the systems are solved, we will use values from the preceding step, $e_S^{(k-1)}$ and $e_l^{(k-1)}$. The dependences can be expressed as

$$\bar{\tau}_{e_S}^{(k)} = f_S(e_S^{(k-1)}) \quad \text{and} \quad \bar{\tau}_{e_l}^{(k)} = f_l(e_l^{(k-1)}) \quad (11)$$

Let's assume, to begin with, that functions f_S and f_l are linear:

$$\bar{\tau}_{e_S}^{(k)} = \beta_S e_S^{(k-1)} \quad \text{and} \quad \bar{\tau}_{e_l}^{(k)} = \beta_l e_l^{(k-1)} \quad (12)$$

Also, we will take that the relationship between $\bar{\tau}_{e_S}^{(k)}$ and $e_S^{(k-1)}$ is equal to the relationship between τ_{eq} and $e_S^{(n)}$ in the final step of the outer loop and, similarly, that the relationship between $\bar{\tau}_{e_l}^{(k)}$ and $e_l^{(k-1)}$ is equal to the relationship between τ_{eq} and $e_l^{(n)}$:

$$\frac{\bar{\tau}_{e_S}^{(k)}}{e_S^{(k-1)}} = \frac{\tau_{eq}}{e_S^{(n)}} \quad \text{and} \quad \frac{\bar{\tau}_{e_l}^{(k)}}{e_l^{(k-1)}} = \frac{\tau_{eq}}{e_l^{(n)}} \quad (13)$$

Since $e_S^{(n)} \approx \tau_S$ and $e_l^{(n)} \approx \tau_l$, we can write

$$\bar{\tau}_{e_S}^{(k)} = \frac{\tau_{eq}}{\tau_S} e_S^{(k-1)} \quad \text{and} \quad \bar{\tau}_{e_l}^{(k)} = \frac{\tau_{eq}}{\tau_l} e_l^{(k-1)} \quad (14)$$

At the end of the outer loop, when $e_S^{(k-1)}$ and $e_l^{(k-1)}$ reach τ_S and τ_l , it should be $\bar{\tau}_{e_S}^{(n)} = \tau_{eq}$ and $\bar{\tau}_{e_l}^{(n)} = \tau_{eq}$. But, it will never be $e_S^{(k-1)} = \tau_S$ or $e_l^{(k-1)} = \tau_l$, errors will be either slightly higher or slightly lower than the specified tolerances. If errors are smaller than the specified tolerances, the condition (9) for the termination of the outer loop is satisfied and the system (6) will never be solved with the tolerance τ_{eq} . Therefore, slightly smaller values will be used:

$$\bar{\tau}_{e_S}^{(k)} = \frac{\tau_{eq}(1-\sqrt{\tau_S})}{\tau_S} e_S^{(k-1)} \quad \text{and} \quad \bar{\tau}_{e_l}^{(k)} = \frac{\tau_{eq}(1-\sqrt{\tau_l})}{\tau_l} e_l^{(k-1)} \quad (15)$$

The analysis of examples has shown that the tolerance is increasing too rapidly. Therefore, quadratic functions are taken:

$$\bar{\tau}_{e_S}^{(k)} = \frac{\tau_{eq}(1-\sqrt{\tau_S})}{\tau_S^2} (e_S^{(k-1)})^2 \quad \text{and} \quad \bar{\tau}_{e_l}^{(k)} = \frac{\tau_{eq}(1-\sqrt{\tau_l})}{\tau_l^2} (e_l^{(k-1)})^2 \quad (16)$$

The reduction rate of errors $e_S^{(k)}$ and $e_l^{(k)}$ will also be taken into account by considering the ratio of these values in two consecutive steps [20]:

$$\bar{\tau}_{e_S}^{(k)} = \eta \left(\frac{e_S^{(k-1)}}{e_S^{(k-2)}} \right)^2 e_S^{(k-1)} \quad \text{and} \quad \bar{\tau}_{e_l}^{(k)} = \eta \left(\frac{e_l^{(k-1)}}{e_l^{(k-2)}} \right)^2 e_l^{(k-1)} \quad (17)$$

where η is the "dumping" that provides additional control.

The tighter of the tolerances $\bar{\tau}_{e_S}^{(k)}$ and $\bar{\tau}_{e_l}^{(k)}$ should be taken and therefore

$$\begin{aligned} \tau_{e_S}^{(k)} &= \min \left(\frac{\tau_{eq}(1-\sqrt{\tau_S})}{\tau_S^2} (e_S^{(k-1)})^2, \eta \frac{(e_S^{(k-1)})^3}{(e_S^{(k-2)})^2} \right) \\ \tau_{e_l}^{(k)} &= \min \left(\frac{\tau_{eq}(1-\sqrt{\tau_l})}{\tau_l^2} (e_l^{(k-1)})^2, \eta \frac{(e_l^{(k-1)})^3}{(e_l^{(k-2)})^2} \right) \end{aligned} \quad (18)$$

In the "inexact" procedure, the tolerance of the step is determined by the greater of the errors $e_S^{(k-1)}$ and $e_l^{(k-1)}$. Besides,

the tolerance should not be smaller than τ_{eq} . Therefore, the constraint is introduced as follows:

$$\tau_e^{(k)} = \max(\tau_{e_S}^{(k)}, \tau_{e_\ell}^{(k)}, \tau_{eq}) \tag{19}$$

We have noticed that in some examples the convergence of the outer loop is not uniform: it can happen that $e_S^{(k)} > e_S^{(k-1)}$ or $e_\ell^{(k)} > e_\ell^{(k-1)}$. That is why it should be confirmed that the tolerance in k -th step is equal to or smaller than the tolerance of the previous step:

$$\tau^{(k)} = \min(\tau^{(k-1)}, \tau_e^{(k)}) \tag{20}$$

4. About the computer programme

The genuine programme code in which the calculations were conducted is written in programming package *SageMath* [21]. The flow chart of the programme is shown in Figure 3. Linear equation systems are solved by functions **splu()** and **cg()** contained within the programming library *SciPy* [22] that is included in *SageMath*. Sparse matrices are represented using the **csc_matrix** class (Compressed Sparse Column Matrix) from the *SciPy* library.

5. Examples

5.1. Minimal net with rigid supports

The first example is the net with "rigid" supports: all cables have fixed nodes at both ends. The net is spread over a ground-plan area $[-a, a]^2$. The heights of supports are given by the expression

$$z = a \ln \frac{\cos(x/a)}{\cos(y/a)} \tag{21}$$

The net has two families of 23 cables, which gives $23 \cdot 23 = 529$ inner nodes, $4 \cdot 23 = 92$ boundary nodes, and $46 \cdot 24 = 1104$ bars. Inner nodes are free, and so three systems of 529 equations with 529 unknowns need to be solved in each iteration step. Unit force densities are assigned to all bars in the first step. The resulting net is shown in Figures 4.a and 4.c. Force values in bars range from 1,668 to 2,903.

The minimal net, determined by iterative application of the force density method, is shown in Figures 4.b and 4.d. Force densities in each step of iterative procedures are calculated according to

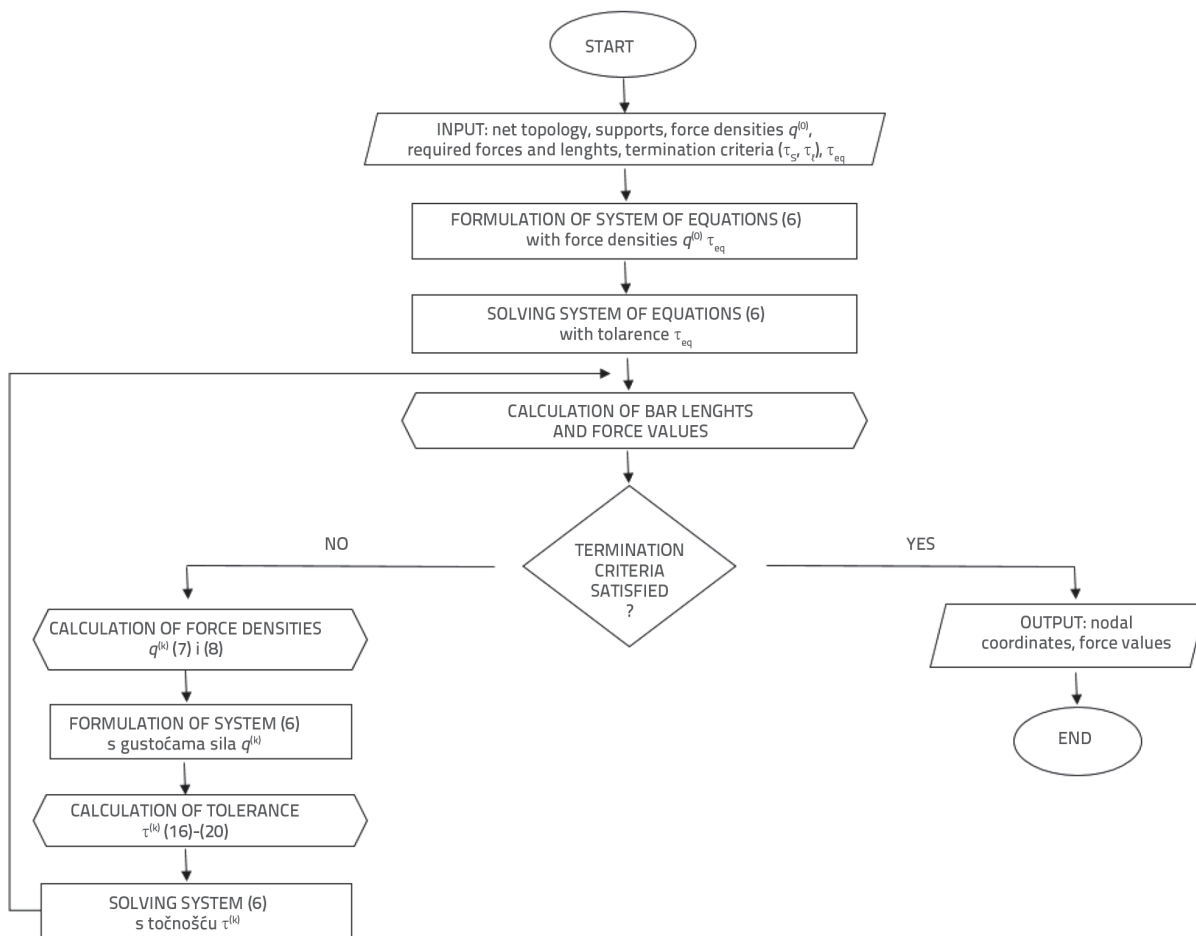


Figure 3. Flow chart for the proposed iterative procedure

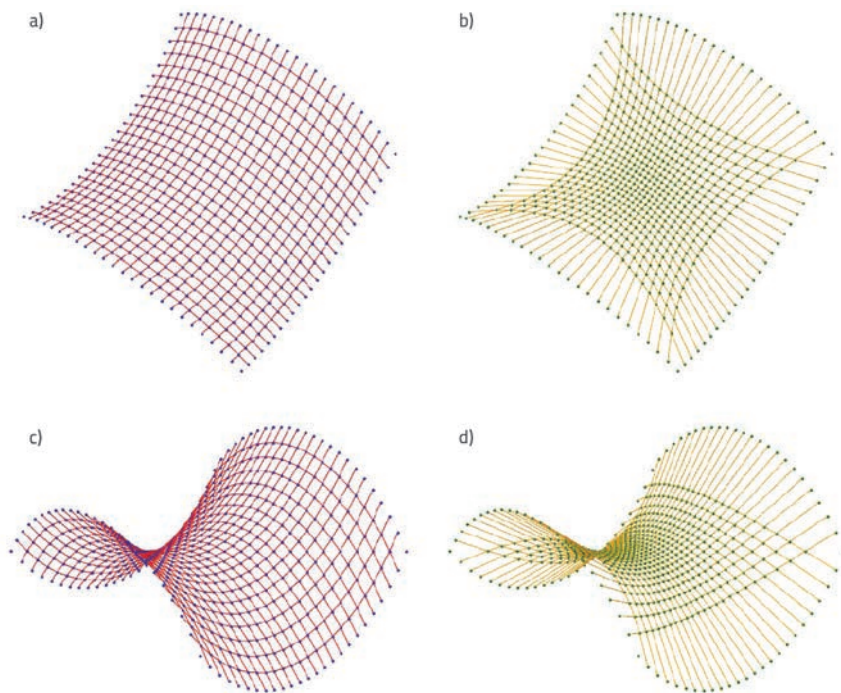


Figure 4. Net with "rigid" supports: a) and c) with equal force densities in all bars; b) and d) minimal net

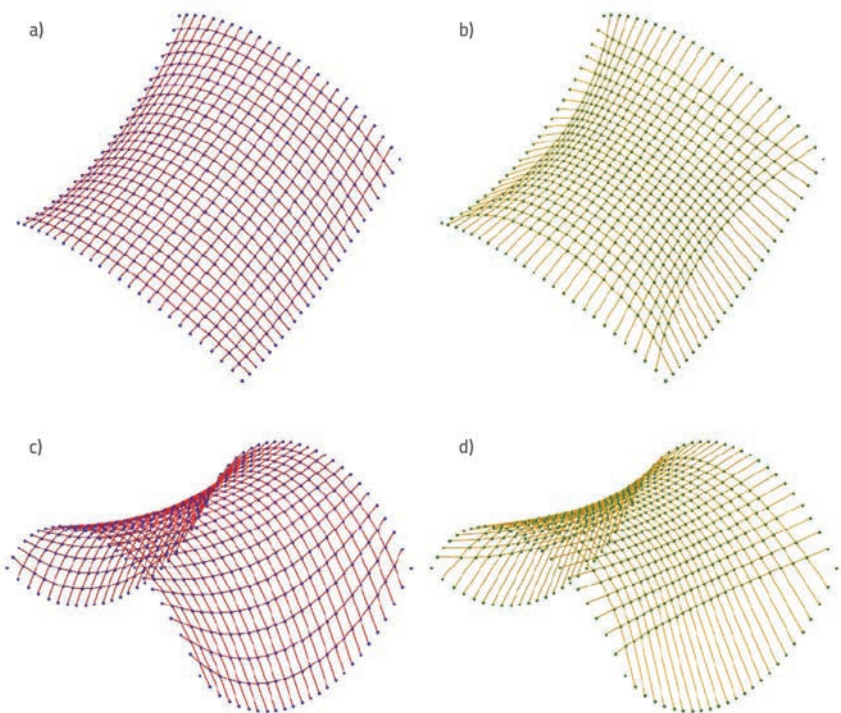


Figure 5. Net with "rigid" supports: a) and c) with equal force densities in bars along individual cables; b) and d) generalized minimal net

expression (7), where the unit value of \bar{S}_{ij} is taken for all $\{i, j\}$ (since the shape of the net does not depend on force values, but only on their ratios, the same shape is obtained for any positive

\bar{S}). Final force densities range from 0,090 to 1,197.

When solving the linear system using the sparse LU decomposition, the required tolerance $\tau_s = 10^{-4}$ is achieved after 576 steps. The same number of steps of the outer loops is required to solve the system using the conjugate gradient method (CG). The tolerance with which linear systems have to be solved is specified as $\tau_{eq} = 5 \times 10^{-7}$. If zero vectors ($\mathbf{x}_0^{(k)} = \mathbf{0}$) are taken as initial approximations of solutions, the total number of inner steps for all three systems is 78258. However, if solutions from preceding steps ($\mathbf{x}_0^{(k)} = \mathbf{x}_n^{(k-1)}$) are taken as initial approximations, the total number of inner steps is reduced to 34505. In the proposed "inexact" procedure, linear systems are also solved using conjugate gradients (ICG). The tolerance is achieved after 557 steps of outer loop with the total of 16201 steps of inner loops. In this and all other examples, $\eta = 0.025$ in (8). Results for this example (Ex. 1), as well as for the other three examples, are concisely shown in Table 1 at the end of this section.

5.2. Different cable forces

The shape of the net can be influenced by changing the ratio of force values in individual cables. The net with the same layout disposition and the same topology is shown in Figure 5. In the net in Figures 5.a and 5.c, unit force densities are assigned to all bars that belong to the convex family of cables, while the force densities in bars of the concave family of cables are equal to one half of the height of their supports. In the generalized minimal net in Figures 5 b) and d), force values in all bars of the concave family of cables are equal to one half of the height of their supports, while bars of the convex family have unit forces. Force densities are calculated according to expression (7) in the iterative procedure of the form finding of the generalized minimal net.

Since the topology of the net is equal to the one given in the previous example, the number of equations is the same.

The steps required to achieve tolerance $\tau_s = 10^{-4}$ in various computational procedures, with $\tau_{eq} = 5 \times 10^{-7}$ in CG and in final steps of "inexact" procedure, are listed in Table 1, rows Ex. 2.

5.3. Net with edge cables

Cable net structures are often made with edge cables. Inner cables of the net are connected to edge cables, and edge cables are, at their end points, connected to "rigid" structural elements. The only known coordinates are the coordinates of these few supports and so the form finding of edge cables becomes a part of the form finding of the entire net.

The net shown in Figure 6 has four edge cables stretched between four corner points. Three corners are in the same horizontal plane and the fourth one is raised above it. There are 78 inner cables. The net has 837 free nodes; 76 of them are at the edge cables. Systems contain 837 equations with 837 unknowns.

The force values in edge cables are significantly larger than the values in inner ones. Still, if the sliding of end joints of inner cables along edge cables is not prevented, the inner cables will crowd together. The problem of joint sliding can be prevented by prescribing the lengths of bars into which edge cables are divided.

Unit force densities in bars of inner cables and several times larger force densities in bars of edge cables are specified in the first step (Figures 6.a and 6.c). Since the force values are proportional to force densities, the resulting values at edge cable bars will also be a few times greater than those in inner bars.

Then the iterative calculation is carried out with specified force values in bars of inner cables, and with specified lengths of bars of edge cables. These lengths are arithmetic means of the lengths of bars of each cable obtained in the first step (the lengths of 80 bars are specified, while the unit force values are given in 1600 bars).

Force densities in inner bars are calculated according to expression (7), while those in edge cable bars are calculated using expression (8). To terminate the loop, both conditions (9) need to be satisfied, where $\tau_s = \tau_l = 10^{-4}$. The inner loop tolerance is $\tau_{eq} = 5 \times 10^{-7}$. The number of steps required to achieve given tolerances in different computational procedures is listed in Table 1, rows Ex. 3. The resulting net is shown in Figures 6.b and 6.d.

5.4. Net over octagon

In the final example, the net has nine supports: eight on the edges and one inner "high point". Edge cables are stretched between the edge supports. The "ridge" and "valley" cables are stretched between the "high point" and edge supports. In addition to the lengths of edge cable bars, the lengths of bars of "ridge" and "valley" cables are also specified. The net contains 90 cables in total.

As in the previous example, specified bar lengths are arithmetic means of the lengths obtained in the first step, in which force densities in the corresponding bars were several times larger than in other bars (Figures 7.a and 7.b). Lengths are specified for

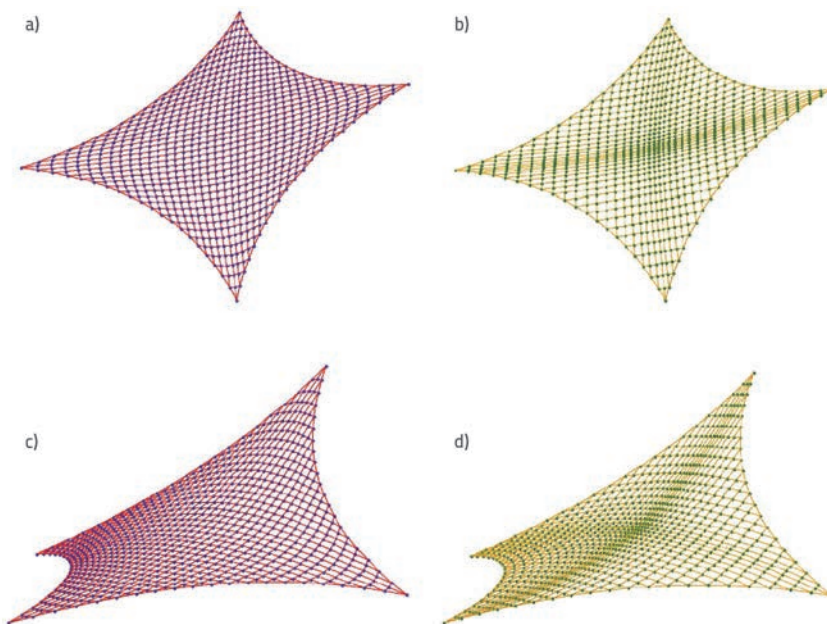


Figure 6. Net with edge cables: a) and c) specified force densities; b) and d) specified force values in bars of inner cables and specified lengths of bars of edge cables

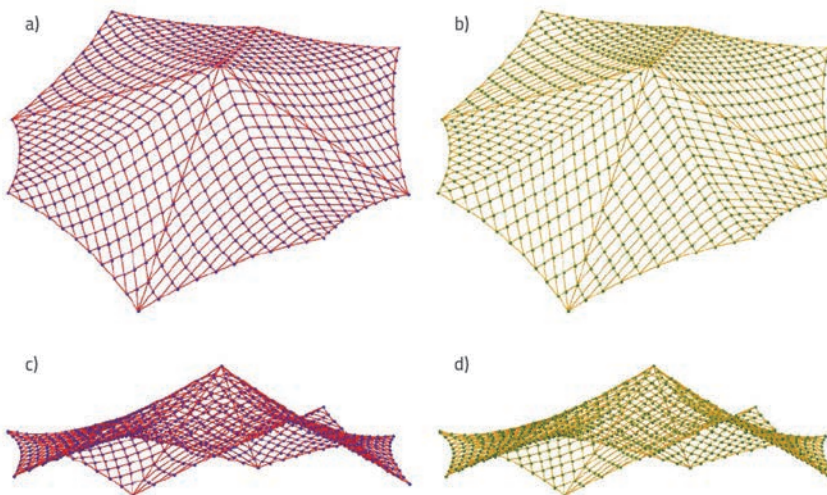


Figure 7. Net over octagon: a) and c)) specified force densities; b) and d)) specified force values in bars of inner cables and specified lengths of edge, valley and ridge cables

Table 1. Number of required steps in inner and outer loops of IFDM

		LU	CG		ICG
			$\mathbf{x}_0^{(k)} = 0$	$\mathbf{x}_0^{(k)} = \mathbf{x}_n^{(k-1)}$	
Example 1	Outer loop	576	576	576	557
	Inner loop		78.240	34 489	16.201
Example 2	Outer loop	2923	2920	2923	2768
	Inner loop		634.607	99.143	26.400
Example 3	Outer loop	1081	1114	1078	1137
	Inner loop		740.449	44.176	17.130
Example 4	Outer loop	298	299	300	300
	Inner loop		146.527	20.982	6793

200 bars, while unit force values are given for other 1520 bars. Tolerances that control outer loop are $\tau_s = 10^{-4}$ and $\tau_l = 10^{-3}$. The net has 832 free nodes and, therefore, three systems, each containing 832 equations, need to be solved. The tolerance in iterative procedures is once again $\tau_{eq} = 5 \times 10^{-7}$. The resulting net is shown in Figures 7 b) and d), while required steps are listed in Table 1, rows Ex. 4.

6. Conclusion

The form finding of prestressed cable structures is usually conducted in a series of attempts in which the designer tries to meet a set of architectural and structural requirements. The described iterative application of the force density method facilitates attainment of specified force values in bars or specified bar lengths, but in some cases the computation can be time-consuming and therefore unsuitable for interactive work. We believe that the proposed method for reducing the computation time will enable integration of

the force density method and its iterative application in an interactive environment for form finding, where the designer can change boundary conditions and force values in cables, as well as mark the cables whose segment lengths must not be changed.

Extensive numerical experiments show that the proposed method is almost always efficient and robust, although there are cases in which the efficiency strongly depends on constants in the proposed termination rule that keeps the accuracy of linear solutions high enough not to compromise the convergence of the iterative process, but prevents it from quickly becoming unnecessarily high. This rule is not the only possibility, which leaves ample room for further research.

Acknowledgements

This work was fully supported by Croatian Science Foundation (HRZZ) under the project IP-2014-09-2899. Figures 1 and 2 were made by Romana Vrančić, M.Eng. (Civ.Eng.).

REFERENCES

- [1] Dvornik, J., Lazarević, D.: Prednapregnute gipke konstrukcije od užadi i tkanine, *GRAĐEVINAR*, 47 (1995) 4, pp. 185-199.
- [2] Dvornik, J., Lazarević, D.: Viseće konstrukcije od platna i užadi, Građevni godišnjak '97, HDGI, Zagreb, 1997., pp. 239-297.
- [3] Schek, H.J.: The force density method for form finding and computation of general networks, *Computer Methods in Applied Mechanics and Engineering*, 3 (1974) 1, pp. 115-134.
- [4] Linkwitz, K.: Formfinding by the direct Approach" and pertinent strategies for the conceptual design of prestressed and hanging structures, *International Journal of Space Structures*, 14 (1999) 2, pp. 73-87.
- [5] Linkwitz, K.: About formfinding of double-curved structures, *Engineering Structures*, 21 (1999) 8, pp. 709-718.
- [6] Malerba, P.G., Patelli, M., Quagliaroli, M.: An extended force density method for the form finding of cable systems with new forms, *Structural Engineering and Mechanics*, 42 (2012) 2, pp. 191-200.
- [7] Aboul-Nasr, G., Mourad, S.A.: An extended force density method for form finding of constrained cable nets, *Case Studies in Structural Engineering*, 3 (2015), pp. 19-32, <https://doi.org/10.1016/j.csse.2015.02.001>
- [8] Fresl, K., Gidak, P., Vrančić, R.: Generalized minimal nets in form finding of prestressed cable nets, *GRAĐEVINAR*, 65 (2013) 8, pp. 707-720.
- [9] Šamec, E., Fresl, K., Baniček, M.: Iterative application of the force density method, 40th Solid Mechanics Conference, PO45, Warszawa, 2016., <http://solmech2016.ippt.pan.pl/Abstracts/0045.pdf>, pristupljeno 7. 6. 2017.

- [10] Haber, R.B., Abel, J.F.: Initial equilibrium solution methods for cable reinforced membranes. Part I – Formulations, *Computer Methods in Applied Mechanics and Engineering*, 30 (1982) 3, pp. 263-284.
- [11] Veenendaal, D., Block, P.: An overview and comparison of structural form finding methods for general networks, *International Journal of Solids and Structures*, 49 (2012) 26, pp. 3741-3753.
- [12] Enrique, L., Block, P., Schwartz, J.: Form-finding Method for Prestressed Cable Networks Using Graphic Statics, *Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium 2015, Amsterdam, 2015.*, na CD-u: IASS2015-524506.pdf.
- [13] Maurin, M., Motro, R.: Investigation of minimal forms with conjugate gradient method, *International Journal of Solids and Structures*, 38 (2001) 14, pp. 2387-2399.
- [14] Dvornik, J., Lazarević, D.: Fractals and formfinding - magic with real numbers, *Engineering Modelling*, 16 (2003) 1-2, pp. 1-11.
- [15] Dvornik, J., Lazarević, D., Fresl, K.: The Fractal Nature of the Form Finding Equations, *Computational mechanics in the UK - 11th Annual Conference of the Association for Computational Mechanics in Engineering (United Kingdom), Glasgow, 2003.*, pp. 201-204.
- [16] Golub, G.H., Van Loan, C.F.: *Matrix Computations*, Second Edition, The John Hopkins University Press, Baltimore, 1989.
- [17] Duff, I.S., Erisman, A.M., Reid, J.K.: *Direct Methods for Sparse Matrices*, Clarendon Press, Oxford, 1986.
- [18] Kelley, C.T.: *Iterative Methods for Linear and Nonlinear Equations*, Society for Industrial and Applied Mathematics, Philadelphia, 1995, <https://doi.org/10.1137/1.9781611970944>
- [19] Dembo, R.S., Eisenstat, S.C., Steihaug, T.: Inexact Newton methods, *SIAM Journal on Numerical Analysis*, 19 (1982) 2, pp. 400-408.
- [20] Eisenstat, S.C., Walker, H.F.: Choosing the Forcing Terms in an Inexact Newton Method, *SIAM Journal on Scientific Computing*, 17 (1996) 1, pp. 16-32.
- [21] SageMath, the Sage Mathematics Software System (Version 7.6), The Sage Developers, 2016, <http://www.sagemath.org>.
- [22] Jones, E., Oliphant, T., Peterson, P. et al.: *SciPy: Open source scientific tools for Python*, 2001, <http://www.scipy.org/>, pristupljeno 7. 6. 2017.