

IZRADA PYTHON PROGRAMA ZA RJEŠAVANJE OPĆEG RAVNINSKOG PROBLEMA OKVIRNIH NOSAČA

Topić, Toma

Graduate thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Split / Sveučilište u Splitu**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:228:326203>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-12-22**



Repository / Repozitorij:

[Repository of University Department of Professional Studies](#)



UNIVERSITY OF SPLIT



SVEUČILIŠTE U SPLITU
SVEUČILIŠNI ODJEL ZA STRUČNE STUDIJE

Stručni diplomski studij Strojarsvo

Toma Topić

ZAVRŠNI RAD

**Izrada Python programa za rješavanje općeg ravninskog
problema okvirnih nosača**

Split, lipanj 2024

SVEUČILIŠTE U SPLITU
SVEUČILIŠNI ODJEL ZA STRUČNE STUDIJE

Stručni diplomski studij Strojarsvo

Predmet: Python u strojarstvu

ZAVRŠNI RAD

Kandidat: Toma Topić

Naslov rada: Izrada Python programa za rješavanje
općeg ravninskog problema okvirnih nosača

Mentor: Igor Gabrić, v.pred.

Split, lipanj 2024.

REPUBLIKA HRVATSKA
SVEUČILIŠTE U SPLITU
Sveučilišni odjel za stručne studije

Studij: Specijalistički diplomski stručni studij Strojarsstvo
Predmet: Python u strojarstvu
Nastavnik: Igor Gabrić, v.pred.

ZADATAK

Kandidat: Topić Toma

Zadatak: Izrada Python programa za rješavanje općeg ravninskog problema okvirnih nosača

U okviru završnog rada potrebno je:

- uvodno prikazati teorijsku podlogu za rješavanje općeg problema ravninskih okvirnih nosača primjenom metode konačnih elemenata (FEM) za statički određene i neodređene slučajeve opterećenja
- detaljno, analitički, opisati postupak rješavanja problema na jednostavnom statički neodređenom problemu ravninskog okvira primjenom FEM metode
- Izraditi i prikazati plan za izradu programa podjelom problema na logičke cjeline
- izraditi temeljni i detaljne dijagrame toka svih potrebnih funkcionalnosti programa popraćene svim potrebnim pojašnjenjima
- organizirati unos podataka korištenjem eksternih tabličnih datoteke (xlsx)
- izrađena aplikacija treba odrediti reakcije u osloncima te pomake i zakrete svih čvorova
- na temelju dobivenih podataka, program treba generirati dijagrame momenata i poprečnih sila te prikazati iste u obliku slike na zaslonu računala
- odrediti veličinu i poziciju maksimalnog momenta savijanja
- dobivene rezultate program treba tablično ispisati na zaslonu te pohraniti u vanjsku datoteku (xlsx).
- programski kôd treba biti podijeljen na funkcionalne cjeline (paket, moduli, funkcije) uz detaljna tekstualna pojašnjenja svih funkcionalnosti, programskih blokova i varijabli
- testirati izrađeni program na barem jednom primjeru koji treba riješiti primjenom izrađenog programskog kôda i na alternativan način – analitički ili primjenom renomiranog računalnog programa (način i rezultate testiranja prikazati u radu)
- iznijeti zaključke
- navesti literaturu i izvore podataka.

Sažetak

U današnjem inženjerskom okruženju, metode numeričke analize postaju sve važnije za rješavanje složenih problema. Metoda Konačnih Elemenata (MKE) je jedan od najčešće korištenih pristupa za detaljnu analizu i simulaciju ponašanja konstrukcija pod različitim opterećenjima. Cilj ovog završnog rada je izrada programa za rješavanje problema ravninskih okvirnih nosača primjenom MKE. Ulazni podaci koji opisuju geometriju i karakteristike materijala okvira bit će učitani iz Excel datoteke. Program će na temelju tih podataka izračunati reakcije u osloncima, pomake i nagibe svih čvorova okvira. Metoda konačnih elemenata omogućava precizno modeliranje složenih geometrija i svojstava materijala te analizu učinaka različitih opterećenja. U radu će biti detaljno prikazan proračunski postupak MKE, algoritam, te rad programskog rješenja uključujući učitavanje podataka, implementaciju MKE algoritma, interpretaciju i vizualizaciju rezultata. Dobivena rješenja bit će verificirana usporedbom s rezultatima javno dostupne aplikacije.

Ključne riječi: Okvirni nosač, FEM analiza, Python programiranje, ...

Summary (Development of a Python program for solving General planar frame structure problems)

In today's engineering environment, numerical analysis methods are increasingly important for solving complex problems. The Finite Element Method (FEM) is one of the most commonly used approaches for detailed analysis and simulation of structural behavior under various loads. The aim of this thesis is to develop a program for solving planar frame structures using FEM. Input data describing the geometry and material properties of the frame will be loaded from an Excel file. Based on this data, the program will calculate support reactions, node displacements, and node rotations. The Finite Element Method allows for precise modeling of complex geometries and material properties, as well as analysis of the effects of different loads. The thesis will detail the computational procedure of FEM, the algorithm steps, and the functionality of the program, including data loading, FEM algorithm implementation, and result interpretation and visualization. The obtained solutions will be verified by comparison with results from a publicly available application that performs similar calculations.

Key words: Frame structure, FEM analysis, Python programming

SADRŽAJ

Sažetak	ii
Summary (Development of a Python program for solving General planar frame structure problems).....	ii
1. Uvod.....	1
2. Opisivanje proračuna metodom konačnih elemenata za zadani okvirni nosač.....	2
2.1. Određivanje lokalnih i globalne matrice krutosti	4
2.2. Lokalni vektor efektivnih sila.....	8
2.3. Određivanje globalnih efektivnih vektora sila elemenata	12
2.4. Ulančavanje globalnih vektora efektivnih vektora sila	15
2.5. Primjena rubnih uvjeta i rješavanje jednadžbe	21
2.6. Određivanje sila u elementima	26
3. Dijagrami toka	27
3.1. Unos podataka	28
3.2. Provjera 1.....	29
3.3. Provjera 2.....	30
3.4. Proračun (glavni program).....	31
3.5. Proračun duljine i kuta nagiba elemenata.....	32
3.6. Generiranje lokalnih i globalnih matrica krutosti.....	33
3.7. Kreiranje globalnih matrica krutosti elemenata u formatu globalne matrice krutosti okvira 34	
3.8. Generiranje globalne matrice krutosti okvira	35
3.9. Generiranje lokalnih vektora ukupnog opterećenja okvira.....	36
3.10. Generiranje globalnih vektora ekvivalentnog opterećenja elemenata	37
3.11. Generiranje globalnih vektora opterećenja okvira.....	38
3.12. Određivanje pomaka slobodnih čvorova primjenom rubnih uvjeta.....	39
3.13. Određivanje reakcija u osloncima primjenom rubnih uvjeta.....	40

3.14.	Generiranje vektora svih pomaka čvorova, te lista naziva pomaka i sila	41
3.15.	Pomaci svih čvorova okvira	42
3.16.	Određivanje unutrašnjih sila(momenta, uzdužna i poprečna sila) u elementima ..	43
3.17.	Crtanje dijagrama za deformiran i nedeformiran okvirni nosač	44
3.18.	Crtanje dijagrama momenata, poprečnih i uzdužnih sila.....	45
3.19.	Ispis podataka.....	46
4.	Lista varijabli i funkcija	47
5.	Opis programa.....	50
5.1.	Glavni kôd koji se izvršava	50
5.2.	Unos podataka	55
5.3.	Provjera broja	56
5.3.1.	Provjera je li broj pozitivan.....	57
5.4.	Duljina i nagiba elementa	58
5.5.	Generiranje lokalnih matrica krutosti elemenata.....	59
5.6.	Kreiranje globalnih matrica krutosti elemenata u formatu globalne matrice krutosti okvira	59
5.7.	Generiranje globalne matrice krutosti sustava.....	61
5.8.	Generiranje lokalnih vektora ekvivalentnog opterećenja	61
5.9.	Generiranje globalni vektor ekvivalentnog opterećenja okvira.-popravit	63
5.10.	Odredi reakcije u osloncima	64
5.11.	Pomaci po čvorovima okvira	65
5.12.	Određivanje sila u elementima.....	66
5.13.	Crtanje dijagrama deformirane i ne deformirane okvirnog nosača	68
5.14.	Crtanje dijagrama momenta, poprečnih i uzdužnih sila.....	69
6.	Testiranje kôda.....	72
6.1.	Prikaz provjere ispravnosti funkcija je li broj i je li broj pozitivan.....	72
6.2.	Ispis i usporedba rezultata	73

7. Zaključak.....	78
8. Reference	79

Popis slika

Slika 2.1 Prikaz zadanog okvirnog nosača.....	
Slika 2.2 Prikaz djelovanja kontinuiranog opterećenja na element	8
Slika 2.3 Prikaz djelovanje koncentriranog opterećenja na element.....	9
Slika 2.4 Prikaz globalnog koordinatnog sustava na zadanom okvirnom nosaču	12
Slika 3.1 Opći dijagram toka.....	27
Slika 3.2 Dijagram toka za unos podataka	28
Slika 3.3 Dijagram toka provjera 1	29
Slika 3.4 Dijagram toka provjera 2	30
Slika 3.5 Prikaz dijagrama toka vršenje proračuna	31
Slika 3.6 dijagram toka određivanje duljine i nagiba elementa	32
Slika 3.7 dijagram toka generiranje lokalnih i globalnih matrica krutosti	33
Slika 3.8 Kreiranje globalnih matrica krutosti elemenata u formatu globalne matrice krutosti okvira	34
Slika 3.9 Dijagram toka generiranje globalnih matrica krutosti okvira	35
Slika 3.10 Dijagram toka generiranje lokalnih vektora ukupnog opterećenja okvira	36
Slika 3.11 Dijagram toka generiranje globalnih vektora ekvivalentnog opterećenja elemenata	37
Slika 3.12 Dijagram toka generiranje globalnih vektora opterećenja okvira.....	38
Slika 3.13 dijagram toka pomaci slobodnih čvorova	39
Slika 3.14 Dijagram reakcije u osloncima	40
Slika 3.15 Dijagram toka pomaci svih čvorova	41
Slika 3.16 dijagram toka pomaci po svim čvorovima.....	42
Slika 3.17 Dijagram toka za određivanje unutrašnjih sila(momenta, uzdužna i poprečna sila) u elementima.....	43
Slika 3.18 dijagram toka za crtanje dijagrama za deformiran i nedeformiran okvirni nosač ..	44
Slika 3.19 Prikaz dijagrama toka za crtanje dijagrama momenata, poprečnih uzdužnih sila ..	45
Slika 3.20 dijagram toka ispis podatka.....	46
Slika 5.1 Prikaz okvirnog nosača	
Slika 5.2 Unos podataka.....	51
Slika 5.3 Vršenje proračuna	52
Slika 5.4 Prikaz funkcija za pripremu podatka i pozivanje funkcije za crtanje dijagrama momenta uzdužnih i poprečnih sila	53

Slika 5.5 Prikaz crtanja okvira	53
Slika 5.6 Ispis rezultata	54
Slika 5.7 Prikaz čitanja uvoz podatak, provjera podataka je li je broj i je li veći od nule	55
Slika 5.8 Prikaz povratak podataka iz funkcije	56
Slika 5.9 Provjera je li element broj	57
Slika 5.10 Prikaz provjerili je li broj pozitivama	58
Slika 5.11 Izračun duljine i nagiba elementa	58
Slika 5.12 Generiranje lokalnih matrica krutosti elemenata	59
Slika 5.13 Kreiranje globalnih matrica krutosti elemenata u formatu globalne matrice krutosti okvira	60
Slika 5.14 Generiranje globalnih matrica krutosti elemenata	61
Slika 5.15 Generiranje globalnih vektora ekvivalentnog opterećenja.....	62
Slika 5.16 generiranje globalni vektor ekvivalentnog opterećenja okvira.....	63
Slika 5.17 Odredi reakcije u osloncima	65
Slika 5.18 Pomaci po čvorovima okvira.....	66
Slika 5.19 Sile u elementima.....	67
Slika 5.20 Crtanje dijagram.....	69
Slika 5.21 Prikaz naredbi za pripreme podataka za crtanje dijagrama momenta, poprečnih i uzdužnih sila	70
Slika 5.22 Prikaz naredbi za crtanje dijagrama momenta, uzdužnih i poprečnih sila.....	71
Slika 6.1 Prikaz krivog unosa za površine	72
Slika 6.2 Ispis na konzoli koji upućuje na pogrešni unos broja u listi površina A	72
Slika 6.3 Prikaz negativnog unosa	72
Slika 6.4 Ispis konzole za negativni broj.....	73
Slika 6.5 Prikaz ispisa podataka na konzoli	73
Slika 6.6 Prikaz okvirnog nosača sa reakcijama u osloncima.....	75
Slika 6.7 Usporedba dijagrama deformiranog i nedeformiranog okvirnog nosača.....	76
Slika 6.8 Usporedba dijagrama uzdužnih sila	77
Slika 6.9 Usporedba dijagrama poprečnih sila.....	77
Slika 6.10 Usporedba dijagrama momenata.....	77

Popis tablica

Tablica 2.1 Ulazni podaci za elemente i djelovanje opterećenja na elementima.....	3
Tablica 2.2 Ulazni podaci za čvorove i djelovanje sila i moneta na čvorovima	3
Tablica 5.1 Prikaz unosa podataka za čvorove	51
Tablica 5.2 Prikaz unosa podataka za elemente	51
Tablica 6.1 Pomaci po čvorovima.....	74
Tablica 6.2 Rezultati pomaka čvorova iz aplikacije SW FEA 2D FRAME	74
Tablica 6.3 Rezultati reakcije u oslancima	75
Tablica 6.4 Prikaz rezultata sile u čvorovima	76

TABLICA 2.1 ULAZNI PODACI ZA ELEMENTE I DJELOVANJE OPTEREĆENJA NA ELEMENTIMA.....	11
TABLICA 2.2 ULAZNI PODACI ZA ČVOROVE I DJELOVANJE SILA I MONETA NA ČVOROVIMA	11
TABLICA 5.1 PRIKAZ UNOSA PODATAKA ZA ČVOROVE.....	59
TABLICA 5.2 PRIKAZ UNOSA PODATAKA ZA ELEMENTE	59
<i>TABLICA 6.1 POMACI PO ČVOROVIMA</i>	<i>82</i>
TABLICA 6.2 REZULTATI POMAKA ČVOROVA IZ APLIKACIJE SW FEA 2D FRAME.....	82
TABLICA 6.3 REZULTATI REAKCIJE U OSLONCIMA	83
<i>TABLICA 6.4 PRIKAZ REZULTATA SILE U ČVOROVIMA.....</i>	<i>84</i>

1. Uvod

Današnjem inženjerskom okruženju, metode numeričke analize postaju sve važnije za rješavanje složenih problema u svim strukama. Jedan od najčešće korištenih pristupa je Metoda Konačnih Elemenata (MKE), koja omogućava detaljnu analizu i simulaciju ponašanja konstrukcija pod različitim opterećenjima. Cilj ovog završnog rada je izrada programa koji će omogućiti rješavanje općeg problema ravninskih okvirnih nosača primjenom MKE.

Ulazni podaci, koji opisuju geometriju i karakteristike materijala okvira, bit će učitani iz datoteke u odgovarajućem Excel formatu. Ovi podaci uključuju informacije o čvorovima, elementima, materijalnim svojstvima te opterećenjima koja djeluju na okvir. Na temelju ulaznih podataka, program će primijeniti matematički postupak rješavanja okvira metodom konačnih elemenata kako bi izračunao reakcije u osloncima, pomake svih čvorova i nagibe svih čvorova na okviru.

Metoda konačnih elemenata je numerička tehnika koja se koristi za pronalaženje aproksimativnih rješenja u složenim problemima inženjerske analize. Postupak uključuje diskretizaciju kontinuiranog problema u konačan broj manjih, jednostavnijih problema, koji se zatim rješavaju simultano [1]. Prednost ove metode leži u sposobnosti da precizno modelira složene geometrije i svojstva materijala, kao i da analizira učinke različitih opterećenja na konstrukciju [2].

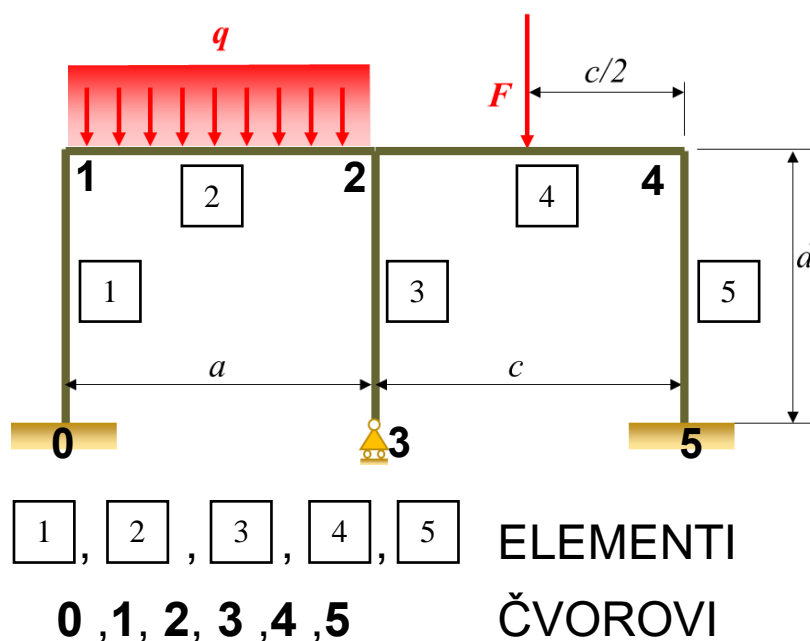
U ovom radu, detaljno će biti prikazan proračunski postupak metode konačnih elemenata, uz priložene dijagrame toka koji ilustriraju korake algoritma. Također će biti prikazan rad samog programskog rješenja, uključujući postupak učitavanja podataka, implementaciju algoritma MKE, te interpretaciju i vizualizaciju rezultata. Dodatno, dobivena rješenja biti će uspoređena s rezultatima javno dostupne aplikacije koja vrši slične proračune kako bi se verificirala točnost i pouzdanost razvijenog programa.

2. Opisivanje proračuna metodom konačnih elemenata za zadani okvirni nosač

Prije početka proračuna, važno je provjeriti ispravnost geometrije i numeracije elemenata i čvorova. Svaki element mora biti pravilno definiran sa svojim početnim i krajnjim čvorom, te odgovarajućim svojstvima materijala. Na ovaj način osiguravamo da će ulazni podaci biti točni i da će rezultati analize biti pouzdani.

Nakon definiranja elemenata i čvorova, može se pristupiti formiranju globalne matrice krutosti, što je prvi korak u primjeni metode konačnih elemenata. Globalna matrica krutosti sastavljena je od lokalnih matrica krutosti pojedinih elemenata, koje su transformirane i objedinjene u jedinstvenu matricu koja opisuje cjelokupnu konstrukciju [1].

Formiranje globalne matrice krutosti ključan je korak u analizi, jer ona sadrži sve informacije o međusobnim interakcijama elemenata unutar okvirnog nosača. Kada je globalna matrica formirana, primjenjuju se rubni uvjeti kako bi se modelirala stvarna ograničenja konstrukcije, kao što su oslonci i veze koje sprečavaju određene pomake i rotacije [2]. Prikaz elemenata i čvorova nalazi se na slici 2.1.



Slika 2.1 Prikaz zadanog okvirnog nosača

Uzeti podaci za proračun nalaze se u tablicama 2.1 i 2.2.

Tablica 2.1 Ulazni podaci za elemente i djelovanje opterećenja na elementima

Element broj	Površina A, mm ²	Moment tromosti I _z , mm ⁴	q _{opt} , N/m	Sila, F; N	Youngov modul, E GPa
1	204	10132	0	0	210
2	204	10132	1000	0	210
3	204	10132	0	0	210
4	204	10132	0	1000	210
5	204	10132	0		210

Tablica 2.2 Ulazni podaci za čvorove i djelovanje sila i momenta na čvorovima

Čvor broj	Cvor X (mm)	Cvor Y (mm)	Sila X (N)	Sila Y (N)	Moment Z (Nm)
0	0	0	0	0	0
1	0	2000	0	0	0
2	2000	2000	0	0	0
3	2000	0	0	0	0
4	4000	2000	0	0	0
5	4000	0	0	0	0

Objasnjen značajki iz tablice 2.1 i 2.2 su :

A	površina, mm ²
I _z	moment tromosti, mm ⁴
q _{opt}	kontinuirano opterećenje N/m
E	Youngov modul, GPa
Sila F	Sila koja djeluje na element, N
Sila X	Uzdužne sile u čvoru, N
Sila Y	Poprečne sile u čvoru, N
Moment Z	Moment koji djeluje na čvor Nm

2.1. Određivanje lokalnih i globalne matrice krutosti

Matrica krutosti je ključna komponenta u metodi konačnih elemenata (FEM), koja se koristi za analizu konstrukcija pod opterećenjem. U osnovi, matrica krutosti povezuje sile i pomake u strukturi. Svaki element strukture pridonosi globalnoj matrici krutosti, koja opisuje cjelokupno ponašanje konstrukcije.

Za proračun potrebno je prvo odrediti lokalnu matricu krutosti za svaki element. Izraz za lokalnu matricu krutosti okvirnih nosača glasi [3]:

$$[\mathbf{K}] = \frac{E}{L^3} \cdot \begin{bmatrix} A \cdot L^2 & 0 & 0 & -A \cdot L^2 & 0 & 0 \\ 0 & 12 \cdot I_z & 6 \cdot I_z \cdot L & 0 & -12 \cdot I_z & 6 \cdot I_z \cdot L \\ 0 & 6 \cdot I_z \cdot L & 4 \cdot I_z \cdot L^2 & 0 & -6 \cdot I_z \cdot L & 2 \cdot I_z \cdot L^2 \\ -A \cdot L^2 & 0 & 0 & A \cdot L^2 & 0 & 0 \\ 0 & -12 \cdot I_z & -6 \cdot I_z \cdot L & 0 & 12 \cdot I_z & -6 \cdot I_z \cdot L \\ 0 & 6 \cdot I_z \cdot L & 2 \cdot I_z \cdot L^2 & 0 & -6 \cdot I_z \cdot L & 4 \cdot I_z \cdot L^2 \end{bmatrix} \quad (2.1)$$

Gdje su simboli :

- $[\mathbf{K}]$ – matrica krutosti elementa u lokalnim koordinatnom sustavu
- E – modul elastičnosti (Pa)
- L – dužina svakog elementa (m)
- I_z – moment tromosti (m^4)
- A – površina (m^2)

Da bi se transformirala lokalna matrica krutosti u globalni koordinatni sustav, koristi se matrica transformacije. Matrica transformacije ovisi o orijentaciji elementa u globalnom koordinatnom sustavu koja se izražava kutom θ [3].

Za dvodimenzionalni problem, matrica transformacije \mathbf{T} može se izraziti kao [3]:

$$[\mathbf{T}] = [\mathbf{T}]^{glob \rightarrow lok} = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & \sin\theta & 0 \\ 0 & 0 & 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

$$[\mathbf{T}]^T = [\mathbf{T}]^{lok \rightarrow glob} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & -\sin\theta & 0 \\ 0 & 0 & 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

gdje su :

- $[\mathbf{T}]$ – transformacijske matrice
- θ – kut usmjerenja elementa

Matrica krutosti elementa u globalnom koordinatnom sustavu dobiva se primjenom izraza koji glasi [3]:

$$[\bar{\mathbf{K}}] = [\mathbf{T}]^T \cdot [\mathbf{K}] \cdot [\mathbf{T}] \quad (2.4)$$

Za dobivanje globalne matrice krutosti potrebno je izvršiti "ulančavanje" matrica krutosti elemenata. U globalnom koordinatnom sustavu ulančavanje (preklapanje) matrica je ključni korak u analizi metodom konačnih elemenata (FEM) koji omogućuje stvaranje globalne matrice krutosti strukture. Ovaj proces integrira lokalne matrice krutosti svakog pojedinačnog elementa u jednu globalnu matricu koja opisuje ponašanje cijele strukture [3].

Koraci pri izvođenju procesa „ulančavanja“ su [3]:

1. Identifikacija čvorova: Prvo se identificiraju čvorovi (čvorovi su točke u strukturi) koji dijele elementi (tj. odgovarajuće matrice krutosti). To su čvorovi na kojima su elementi međusobno pričvršćeni.
2. Spajanje čvorova: Nakon identifikacije čvorova, matrice krutosti elemenata u globalnom koordinatnom sustavu se spajaju zajedno na način da se koriste njihovi zajednički čvorovi. To znači da se odgovarajući članovi matrica krutosti elemenata dodaju na odgovarajuće pozicije globalne matrice krutosti.
3. Formiranje globalne matrice krutosti: Kada su sve matrice krutosti elemenata međusobno povezane, dobiva se globalna matrica krutosti koja opisuje ponašanje cijele strukture. Ova globalna matrica krutosti može se koristiti za daljnju analizu ponašanja strukture pod različitim opterećenjima i uvjetima.
4. Rješavanje sustava jednažbi: Globalna matrica krutosti se može koristiti za rješavanje sustava jednažbi koji opisuju ponašanje strukture pod određenim uvjetima opterećenja. To omogućuje pronalaženje pomaka, reakcija u osloncima, deformacija, naprezanja i drugih karakteristika strukture.

Kako se radi o velikim matrica krutosti kod ovog primjera taj proces će bit prikazan kasnije u radu kod globalnih efektivnih vektora sila u izrazima 2.19 i 2.20 .

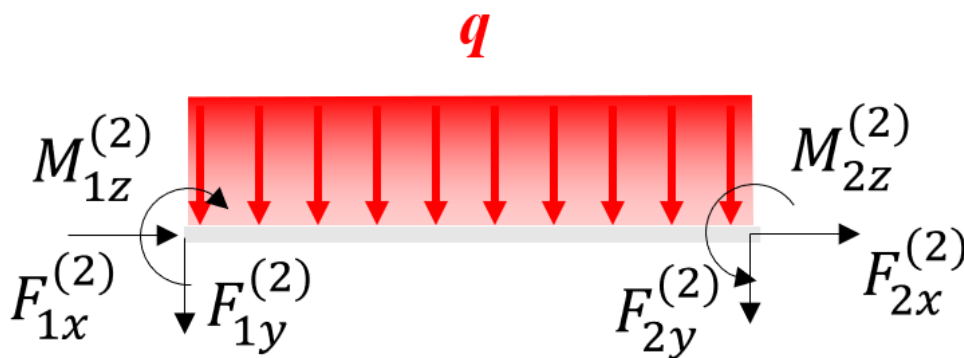
U nastavku se nalazi globalna matrica krutosti cijelog sustava s proračunatim vrijednostima:

$$[\bar{K}] = \begin{pmatrix} 3191.58 & 0 & -3191.58 & -3191.58 & 0 & -3191.58 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2,14E+10 & 0 & 0 & -2,14E+10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -3191.58 & 0 & 4255.44 & 3191.58 & 0 & 2127.72 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -3191.58 & 0 & 3191.58 & 2,14E+12 & 0 & 3191.58 & -2,14E+10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2,14E+10 & 0 & 0 & 2,14E+12 & 3191.58 & 0 & -3191.58 & 3191.58 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -3191.58 & 0 & 2127.72 & 3191.58 & 3191.58 & 8510.88 & 0 & -3191.58 & 2127.72 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2,14E+10 & 0 & 0 & 4,28E+12 & 0 & 3191.58 & -3191.58 & 0 & 3191.58 & -2,14E+10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -3191.58 & -3191.58 & 0 & 2,14E+12 & 0 & 0 & -2,14E+10 & 0 & 0 & -3191.58 & 3191.58 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3191.58 & 2127.72 & 3191.58 & 0 & 12766.3 & -3191.58 & 0 & 2127.72 & 0 & -3191.58 & 2127.72 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -3191.58 & 0 & -3191.58 & 3191.58 & 0 & -3191.58 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2,14E+10 & 0 & 0 & 0 & 2,14E+10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3191.58 & 0 & 2127.72 & -3191.58 & 0 & 4255.44 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2,14E+10 & 0 & 0 & 0 & 0 & 0 & 2,14E+12 & 0 & 3191.58 & -3191.58 & 0 & 3191.58 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3191.58 & -3191.58 & 0 & 0 & 0 & 0 & 2,14E+12 & -3191.58 & 0 & -2,14E+10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3191.58 & 2127.72 & 0 & 0 & 0 & 3191.58 & -3191.58 & 8510.88 & -3191.58 & 0 & 2127.72 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3191.58 & 0 & -3191.58 & 3191.58 & 0 & -3191.58 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2,14E+10 & 0 & 0 & 2,14E+10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3191.58 & 0 & 2127.72 & -3191.58 & 0 & 4255.44 \end{pmatrix} \quad (2.5)$$

2.2. Lokalni vektor efektivnih sila

Za proračun je bitno odrediti lokalne vektore efektivnih sila u elementima. Opterećenja se mogu podijeliti na kontinuirana, koncentrirana opterećenja i opterećenja sustava. Svaka od ovih kategorija opterećenja ima svoj specifičan utjecaj na elemente strukture i moraju se uzeti u obzir prilikom proračuna kako bi se osigurala točna analiza. U slučaju kontinuiranih opterećenja ista je potrebno zamijeniti koncentriranim opterećenjima (silama i momentima) u susjednim čvorovima, a koja imaju ekvivalentno djelovanje kontinuiranom opterećenju.

Efektivne sile koje djeluje na element 2 uslijed kontinuiranog opterećenja prikazane su na slici 2.2.

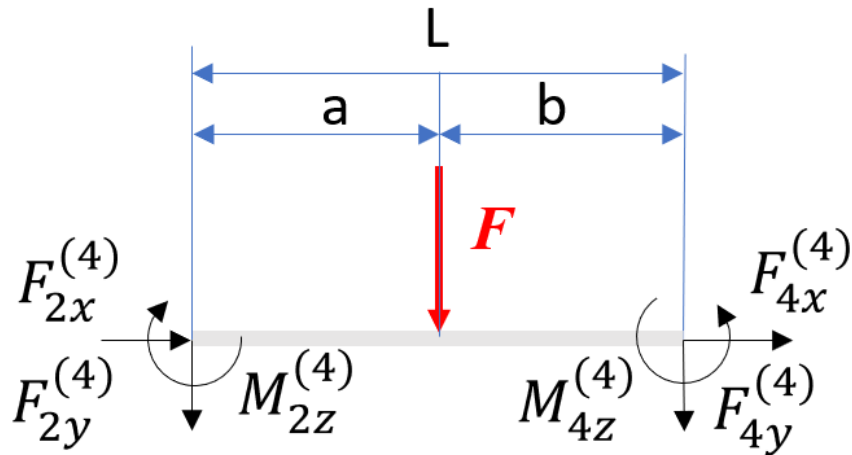


Slika 2.2 Prikaz djelovanja kontinuiranog opterećenja na element

Vektor ekvivalentnih sila uslijed djelovanja kontinuiranog opterećenja je [3] :

$$\{\hat{f}\}_{q,ekv}^{(2)} = \begin{Bmatrix} F_{1x} \\ F_{1y} \\ M_{1z} \\ F_{2x} \\ F_{2y} \\ M_{2z} \end{Bmatrix} = \begin{Bmatrix} 0 \\ -\frac{q \cdot L_2}{2} \\ -\frac{q \cdot L_2^2}{12} \\ 0 \\ -\frac{q \cdot L_2}{2} \\ \frac{q \cdot L_2^2}{12} \end{Bmatrix} \quad (2.6)$$

Određivanje ekvivalentnih sila u čvorovima za slučaj kada na elementu djeluje koncentrirano opterećenje prikazano je na slici 2.3. Koncentrirano opterećenje u ovom primjeru djeluje na element 4 [3].



Slika 2.3 Prikaz djelovanje koncentriranog opterećenja na element

Vektor ekvivalentnih sila uslijed djelovanja koncentriranog opterećenja glasi [3] :

$$\{\hat{f}\}_{k,ekv}^{(4)} = \begin{Bmatrix} F_{2x} \\ F_{2y} \\ M_{2z} \\ F_{4x} \\ F_{4y} \\ M_{4z} \end{Bmatrix} = \begin{Bmatrix} 0 \\ -\frac{F \cdot b^2(3a + b)}{L^3} \\ \frac{F \cdot a \cdot b^2}{L^2} \\ 0 \\ -\frac{F \cdot a^2(3b - a)}{L^3} \\ \frac{F \cdot b \cdot a^2}{L^2} \end{Bmatrix} \quad (2.7)$$

Kad su određeni vektori ekvivalentnih sila mogu se odrediti lokalni vektori efektivnih sila koji su svaki element dani izrazima:

Izraz za element 1:

$$\{f\}_{ef}^{(1)} = \{f\}_{ekv}^{(1)} + \{f\}^{(1)} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} + \begin{Bmatrix} f_{0x}^{(1)} \\ f_{0y}^{(1)} \\ m_{0z}^{(1)} \\ f_{1x}^{(1)} \\ f_{1y}^{(1)} \\ m_{1z}^{(1)} \end{Bmatrix} = \begin{Bmatrix} f_{0x}^{(1)} \\ f_{0y}^{(1)} \\ m_{0z}^{(1)} \\ f_{1x}^{(1)} \\ f_{1y}^{(1)} \\ m_{1z}^{(1)} \end{Bmatrix} \quad (2.8)$$

Izraz za element 2 :

$$\{f\}_{ef}^{(2)} = \{f\}_{ekv}^{(2)} + \{f\}^{(2)} = \begin{Bmatrix} 0 \\ -\frac{q \cdot L}{2} \\ \frac{q \cdot L^2}{12} \\ 0 \\ -\frac{q \cdot L}{2} \\ \frac{q \cdot L^2}{12} \end{Bmatrix} + \begin{Bmatrix} f_{1x}^{(2)} \\ f_{1y}^{(2)} \\ m_{1z}^{(2)} \\ f_{2x}^{(2)} \\ f_{2y}^{(2)} \\ m_{2z}^{(2)} \end{Bmatrix} = \begin{Bmatrix} f_{1x}^{(2)} \\ -\frac{q \cdot L}{2} + f_{1y}^{(2)} \\ -\frac{q \cdot L^2}{12} + m_{1z}^{(2)} \\ f_{2x}^{(2)} \\ -\frac{q \cdot L}{2} + f_{2y}^{(2)} \\ \frac{q \cdot L^2}{12} + m_{2z}^{(2)} \end{Bmatrix} \quad (2.9)$$

Izraz za element 3:

$$\{f\}_{ef}^{(3)} = \{f\}_{ekv}^{(3)} + \{f\}^{(3)} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} + \begin{Bmatrix} f_{2x}^{(1)} \\ f_{2y}^{(1)} \\ m_{2z}^{(1)} \\ f_{3x}^{(1)} \\ f_{3y}^{(1)} \\ m_{3z}^{(1)} \end{Bmatrix} = \begin{Bmatrix} f_{2x}^{(1)} \\ f_{2y}^{(1)} \\ m_{2z}^{(1)} \\ f_{3x}^{(1)} \\ f_{3y}^{(1)} \\ m_{3z}^{(1)} \end{Bmatrix} \quad (2.10)$$

Izraz za element 4 :

$$\{f\}_{ef}^{(4)} = \{f\}_{ekv}^{(4)} + \{f\}^{(4)} = \begin{pmatrix} 0 \\ -\frac{F \cdot b^2(3a+b)}{L^3} \\ \frac{F \cdot a \cdot b^2}{L^2} \\ 0 \\ -\frac{F \cdot a^2(3b-a)}{L^3} \\ \frac{F \cdot b \cdot a^2}{L^2} \end{pmatrix} + \begin{pmatrix} f_{2x}^{(4)} \\ f_{2y}^{(4)} \\ m_{2z}^{(4)} \\ f_{4x}^{(4)} \\ f_{4y}^{(4)} \\ m_{4z}^{(4)} \end{pmatrix} =$$

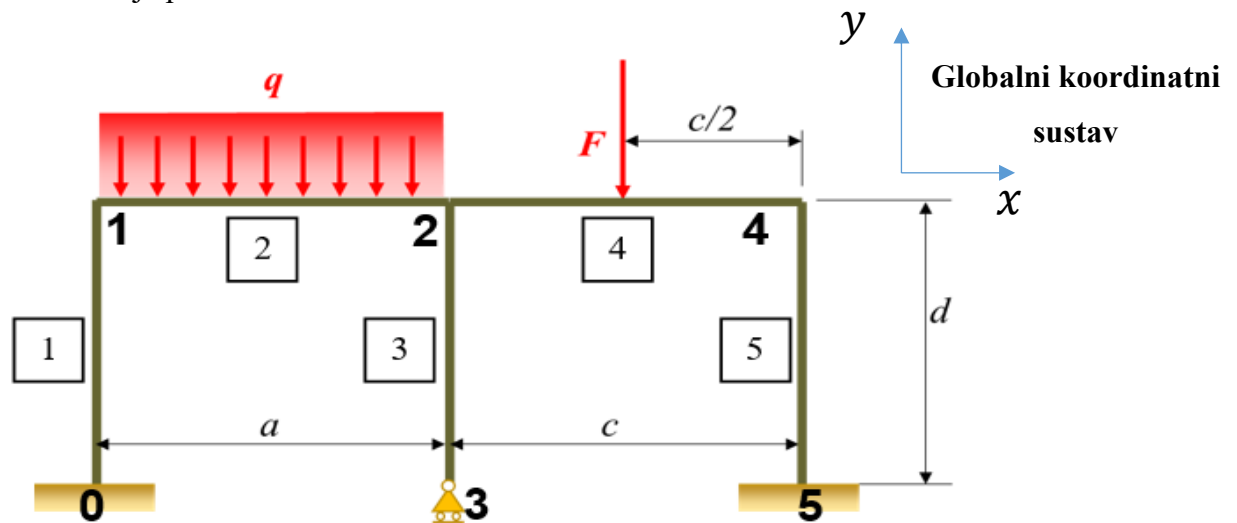
$$= \begin{pmatrix} f_{2x}^{(4)} \\ -\frac{F \cdot b^2(3a+b)}{L^3} + f_{2y}^{(4)} \\ -\frac{F \cdot a \cdot b^2}{L^2} + m_{2z}^{(4)} \\ f_{4x}^{(4)} \\ -\frac{F \cdot a^2(3b-a)}{L^3} + f_{4y}^{(4)} \\ \frac{F \cdot b \cdot a^2}{L^2} + m_{4z}^{(4)} \end{pmatrix} \quad (2.11)$$

Izraz za element 5:

$$\{f\}_{ef}^{(5)} = \{f\}_{ekv}^{(5)} + \{f\}^{(5)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} f_{4x}^{(5)} \\ f_{4y}^{(5)} \\ m_{4z}^{(5)} \\ f_{5x}^{(5)} \\ f_{5y}^{(5)} \\ m_{5z}^{(5)} \end{pmatrix} = \begin{pmatrix} f_{4x}^{(5)} \\ f_{4y}^{(5)} \\ m_{4z}^{(5)} \\ f_{5x}^{(5)} \\ f_{5y}^{(5)} \\ m_{5z}^{(5)} \end{pmatrix} \quad (2.12)$$

2.3. Određivanje globalnih efektivnih vektora sila elemenata

Za određivanje globalnih efektivnih sila potrebno je odrediti globalni koordinatni sustav okvirnog nosača što je prikazano na slici 2.5:



Slika 2.4 Prikaz globalnog koordinatnog sustava na zadanom okvirnom nosaču

Izraz za određivanje globalnih efektivnih vektora sila elemenata glasi [3] :

$$\{\bar{f}\}_{ef}^{(a)} = [T]^T \cdot \{f\}_{ef}^{(a)} \quad (2.13)$$

Gdje su :

- $\{\bar{f}\}_{ef}^{(a)}$ - globalni vektor sila, gdje „a“ označava broj elementa
- $[T]^T$ - matrica transformacije
- $\{f\}_{ef}^{(a)}$ - lokalni vektor sila, gdje „a“ označava broj elementa

Globalni vektori ekvivalentnog opterećenja za svaki element dani su izrazima:

Izraz za element 1:

$$\{\bar{f}\}_{ef}^{(1)} = \begin{Bmatrix} f_{0x}^{(1)} \\ f_{0y}^{(1)} \\ m_{0z}^{(1)} \\ f_{1x}^{(1)} \\ f_{1y}^{(1)} \\ m_{1z}^{(1)} \end{Bmatrix} \quad (2.14)$$

Izraz za element 2:

$$\{\bar{f}\}_{ef}^{(2)} = \begin{Bmatrix} \lambda \cdot \frac{q \cdot L}{2} \\ -\mu \cdot \frac{q \cdot L}{2} \\ -\frac{q \cdot L^2}{12} \\ \lambda \cdot \frac{q \cdot L}{2} \\ -\mu \cdot \frac{q \cdot L}{2} \\ \frac{q \cdot L^2}{12} \end{Bmatrix} + \begin{Bmatrix} f_{1x}^{(2)} \\ f_{1y}^{(2)} \\ m_{1z}^{(2)} \\ f_{2x}^{(2)} \\ f_{2y}^{(2)} \\ m_{2z}^{(2)} \end{Bmatrix} \quad (2.15)$$

Izraz za element 3:

$$\{\bar{f}\}_{ef}^{(3)} = \begin{Bmatrix} f_{2x}^{(3)} \\ f_{2y}^{(3)} \\ m_{2z}^{(3)} \\ f_{3x}^{(3)} \\ f_{3y}^{(3)} \\ m_{3z}^{(3)} \end{Bmatrix} \quad (2.16)$$

Izraz za element 4:

$$\{\bar{f}\}_{ef}^{(4)} = \left\{ \begin{array}{c} \lambda \cdot 0 \\ -\mu \cdot \frac{F \cdot b^2 (3a + b)}{L^3} \\ -\frac{F \cdot a \cdot b^2}{L^2} \\ \lambda \cdot 0 \\ -\lambda \cdot \frac{F \cdot a^2 (3b - a)}{L^3} \\ \frac{F \cdot b \cdot a^2}{L^2} \end{array} \right\} + \left\{ \begin{array}{c} f_{2x}^{(4)} \\ f_{2y}^{(4)} \\ m_{2z}^{(4)} \\ f_{4x}^{(4)} \\ f_{4y}^{(4)} \\ m_{4z}^{(4)} \end{array} \right\} \quad (2.17)$$

Izraz za element 5:

$$\{\bar{f}\}_{ef}^{(5)} = \left\{ \begin{array}{c} f_{4x}^{(5)} \\ f_{4y}^{(5)} \\ m_{4z}^{(5)} \\ f_{5x}^{(5)} \\ f_{5y}^{(5)} \\ m_{5z}^{(5)} \end{array} \right\} \quad (2.18)$$

Gdje su :

- $\mu = \cos\theta$ – član matrice transformacije
- $\lambda = \sin\theta$ – član matrice transformacije

2.4. Ulančavanje globalnih vektora efektivnih vektora sila

Za ulančavanje globalnih efektivnih vektora sila za kontinuirano opterećenje. Prvo će se preklopiti globalni vektori za element 1 i 2 čiji izrazi glase :

Globalni vektori za elementi 1 :

$$\{\bar{f}\}_{ef,kon}^{(1)} = \begin{Bmatrix} f_{0x}^{(1)} \\ f_{0y}^{(1)} \\ m_{\theta z}^{(1)} \\ f_{1x}^{(1)} \\ f_{1y}^{(1)} \\ m_{1z}^{(1)} \end{Bmatrix} \quad (2.19)$$

Globalni vektori za element 2 :

$$\{\bar{f}\}_{ef,kon}^{(2)} = \begin{Bmatrix} f_{1x}^{(2)} \\ -\frac{q \cdot L}{2} + f_{1y}^{(2)} \\ -\frac{q \cdot L^2}{12} + m_{1z}^{(2)} \\ f_{2x}^{(2)} \\ -\frac{q \cdot L}{2} + f_{2y}^{(2)} \\ \frac{q \cdot L^2}{12} + m_{2z}^{(2)} \end{Bmatrix} \quad (2.20)$$

Vektori označeni plavim kvadratom imaju istu vrijednost i preko njih se vrši ulančavanje vektora te novi ulančani vektor glasi:

$$\{\bar{f}\}_{ef,kon}^{(1-2)} = \left\{ \begin{array}{c} f_{0x} \\ f_{0y} \\ m_{0z} \\ f_{1x} \\ -\frac{q \cdot L}{2} + f_{1y} \\ \frac{q \cdot L^2}{12} + m_{1z} \\ f_{2x} \\ -\frac{q \cdot L}{2} + f_{2y} \\ \frac{q \cdot L^2}{12} + m_{2z} \end{array} \right\} \quad (2.21)$$

Za element 2,3,4 potrebno je napraviti isti postupak ulančavanja te vektor nakon preklapanja ima oblik:

$$\{\bar{f}\}_{ef,kon}^{(2-3-4)} = \left\{ \begin{array}{c} f_{1x} \\ -\frac{q \cdot L}{2} + f_{1y} \\ \frac{q \cdot L^2}{12} + m_{1z} \\ f_{2x} \\ -\frac{q \cdot L}{2} + f_{2y} \\ \frac{q \cdot L^2}{12} + m_{2z} \\ f_{3x} \\ f_{3y} \\ m_{3z} \\ f_{4x} \\ f_{4y} \\ m_{4z} \end{array} \right\} \quad (2.22)$$

Ulančani globalni vektor elementa 4-5 je:

$$\{\bar{f}\}_{ef,kon}^{(4-5)} = \begin{pmatrix} f_{2x} \\ -\frac{q \cdot L}{2} + f_{2y} \\ \frac{q \cdot L^2}{12} + m_{2z} \\ f_{3x} \\ f_{3y} \\ m_{3z} \\ f_{4x} \\ f_{4y}^{(4)} \\ m_{4z} \\ f_{5x} \\ f_{5y} \\ m_{5z} \end{pmatrix} \quad (2.23)$$

Potrebno je ulančati sve vektore nakon čega se dobije globalni vektor kontinuiranog opterećenja:

$$\{\bar{f}\}_{ef,kon} = \begin{pmatrix} f_{0x} \\ f_{0y} \\ m_{0z} \\ f_{1x} \\ -\frac{q \cdot L}{2} + f_{1y} \\ -\frac{q \cdot L^2}{12} + m_{1z} \\ f_{2x} \\ -\frac{q \cdot L}{2} + f_{2y} \\ \frac{q \cdot L^2}{12} + m_{2z} \\ f_{3x} \\ f_{3y} \\ m_{3z} \\ f_{4x} \\ f_{4y} \\ m_{4z} \\ f_{5x} \\ f_{5y} \\ m_{5z} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\frac{q \cdot L}{2} \\ -\frac{q \cdot L^2}{12} \\ 0 \\ -\frac{q \cdot L}{2} \\ \frac{q \cdot L^2}{12} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (2.24)$$

Istim postupkom dobivamo dobije se globalni vektor koncentriranog opterećenja koji je:

$$\{\bar{f}\}_{ef,konc} = \begin{pmatrix} f_{0x} \\ f_{0y} \\ m_{0z} \\ f_{1x} \\ f_{1y} \\ m_{1z} \\ f_{2x} \\ -\frac{F \cdot b^2(3a+b)}{L^3} + f_{2y} \\ -\frac{F \cdot a \cdot b^2}{L^2} + m_{2z} \\ f_{3x} \\ f_{3y} \\ m_{3z} \\ f_{4x} \\ -\frac{F \cdot a^2(3b-a)}{L^3} + f_{4y} \\ \frac{F \cdot b \cdot a^2}{L^2} + m_{4z} \\ f_{5x} \\ f_{5y} \\ m_{5z} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{F \cdot b^2(3a+b)}{L^3} \\ -\frac{F \cdot a \cdot b^2}{L^2} \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{F \cdot a}{L} \\ \frac{F \cdot b \cdot a^2}{L^2} \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (2.25)$$

Globalni vektor ekvivalentnih sila dobije se oduzimanjem globalnog vektora ekvivalentnih sila za kontinuirano i koncentrirano opterećenje. Nakon toga treba pribrojiti reakcije oslonaca.

Izraz za oduzimanje globalnih vektora ekvivalentnih sila kontinuiranog i koncentriranog opterećenja[3]:

$$\{\bar{f}\}_{ef} = \{\bar{f}\}_{ef,kon} + \{\bar{f}\}_{ef,konc} = \quad (2.26)$$

$$= \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\frac{q \cdot L}{2} \\ -\frac{q \cdot L^2}{12} \\ 0 \\ \frac{q \cdot L}{2} \\ \frac{q \cdot L^2}{12} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{F \cdot b^2(3a+b)}{L^3} \\ -\frac{F \cdot a \cdot b^2}{L^2} \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{F \cdot a^2(3b-a)}{L^3} \\ \frac{F \cdot b \cdot a^2}{L^2} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\frac{q \cdot L}{2} \\ -\frac{q \cdot L^2}{12} \\ 0 \\ -\frac{q \cdot L}{2} - \frac{F \cdot b^2(3a+b)}{L^3} \\ \frac{q \cdot L^2}{12} - \frac{F \cdot a \cdot b^2}{L^2} \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{F \cdot a^2(3b-a)}{L^3} \\ \frac{F \cdot b \cdot a^2}{L^2} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -1000 \\ -333,34 \\ 0 \\ -1500 \\ 83,334 \\ 0 \\ 0 \\ 0 \\ 0 \\ -500 \\ 250 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (2.27)$$

Zbrajanjem globalnih vektora efektivnih sila s reakcijama u osloncima dobije se globalni vektor sila:

$$\{f\}_{ef} = \{f\}_{ekv} + \{f\} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -1000 \\ -333,34 \\ 0 \\ -1500 \\ 83,334 \\ 0 \\ 0 \\ 0 \\ 0 \\ -500 \\ 250 \\ 0 \\ 0 \\ 0 \end{Bmatrix} + \begin{Bmatrix} f_{0x} \\ f_{0y} \\ m_{0z} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ f_{3y} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ f_{5x} \\ f_{5y} \\ m_{5z} \end{Bmatrix} = \begin{Bmatrix} f_{0x} \\ f_{0y} \\ m_{0z} \\ 0 \\ -1000 \\ -333,34 \\ 0 \\ -1500 \\ 83,334 \\ 0 \\ f_{3y} \\ 0 \\ 0 \\ -500 \\ 250 \\ f_{5x} \\ f_{5y} \\ m_{5z} \end{Bmatrix} \quad (2.28)$$

2.5. Primjena rubnih uvjeta i rješavanje jednadžbe

Rubni uvjeti u analizi metodom konačnih elemenata određuju uvjete na granicama simulirane strukture. To uključuje ograničavanje pomaka, primjenu vanjskih sila, definiranje kontakata s drugim strukturama, ograničenje naprezanja te primjenu simetrije ako je primjenjiva. Ti uvjeti su ključni za preciznu simulaciju ponašanja strukture pod različitim opterećenjima.

Potrebno je odrediti rubne uvjete:

- Čvor 0 je krutom zahvatu te nema pomak u x i y osi i zakret čvora
- Čvor 3 je na pomičnom osloncu te nema pomak u smjeru y osi
- Čvor 5 je krutom zahvatu i isto nema pomak y i x osi i zakret u čvoru

Prikaz matrice pomaka u čvorovima [3]:

$$\{\mathbf{v}\} = \begin{Bmatrix} v_{0x} = 0 \\ v_{0y} = 0 \\ \theta_0 = 0 \\ v_{1x} \\ v_{1y} \\ \theta_1 \\ v_{2x} \\ v_{2y} \\ \theta_2 \\ v_{3x} \\ v_{3y} = 0 \\ \theta_3 \\ v_{4x} \\ v_{4y} \\ \theta_4 \\ v_{5x} = 0 \\ v_{5y} = 0 \\ \theta_5 = 0 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ v_{1x} \\ v_{1y} \\ \theta_1 \\ v_{2x} \\ v_{2y} \\ \theta_2 \\ v_{3x} \\ 0 \\ \theta_3 \\ v_{4x} \\ v_{4y} \\ \theta_4 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad (2.29)$$

Izraz za određivanje pomaka prikazan je u nastavku, gdje su linijom precrtani svi elementi koji nemaju pomak [3]:

$$[K] \cdot [v] = \{f\}_{ef} \tag{2.30}$$

$$\begin{pmatrix}
 3191.58 & 0 & -3191.58 & -3191.58 & 0 & -3.19E+21 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 2.14E+10 & 0 & 0 & -2.14E+10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -3191.58 & 0 & 4266.44 & 3191.58 & 0 & 2.13E+21 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -3191.58 & 0 & 3191.58 & 2.14E+10 & 0 & 3191.58 & -2.14E+10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & -2.14E+10 & 0 & 0 & 2.14E+10 & 3191.58 & 0 & -3191.58 & 3191.58 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -3191.58 & 0 & 2127.72 & 3191.58 & 3191.58 & 8510.88 & 0 & -3191.58 & 2127.72 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & -2.14E+10 & 0 & 0 & 4.28E+12 & 0 & 3191.58 & -3191.58 & 0 & 3191.58 & -2.14E+10 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & -3191.58 & -3191.58 & 0 & 2.14E+10 & 0 & 0 & -2.14E+10 & 0 & 0 & -3191.58 & 3191.58 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 3191.58 & 2127.72 & 3191.58 & 0 & 12766.3 & -3191.58 & 0 & 2127.72 & 0 & -3191.58 & 2127.72 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & -3191.58 & 0 & -3191.58 & 3191.58 & 0 & -3191.58 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2.14E+10 & 0 & 0 & 2.14E+10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 3191.58 & 0 & 2127.72 & -3191.58 & 0 & 4255.44 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & -2127.72 & 0 & 0 & 0 & 0 & 2.14E+12 & 0 & 3191.58 & -3191.58 & 0 & 3191.58 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3191.58 & -3191.58 & 0 & 0 & 0 & 2.14E+12 & -3191.58 & 0 & -2.14E+10 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3191.58 & 2127.7 & 0 & 0 & 0 & 3191.58 & -3191.58 & 8510.88 & -3191.58 & 0 & 2127.72 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3191.58 & 0 & -3191.58 & 3191.58 & 0 & -3191.58 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2.14E+25 & 0 & 0 & 2.14E+10 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3191.58 & 0 & 2127.72 & -3191.58 & 0 & 4255.44
 \end{pmatrix} \cdot \begin{pmatrix}
 0 \\
 0 \\
 0 \\
 v_{1x} \\
 v_{1y} \\
 \theta_1 \\
 v_{2x} \\
 v_{2y} \\
 \theta_2 \\
 v_{3x} \\
 \theta_3 \\
 v_{4x} \\
 v_{4y} \\
 \theta_4 \\
 0 \\
 0 \\
 0 \\
 0
 \end{pmatrix} = \begin{pmatrix}
 0 \\
 0 \\
 0 \\
 f_{1x} \\
 f_{1y} - 1000 \\
 m_{1z} - 333,34 \\
 f_{2x} \\
 -1500 \\
 m_{2z} + 83,334 \\
 f_{3x} \\
 \theta \\
 m_{3z} \\
 f_{4x} \\
 f_{4y} - 500 \\
 m_{4z} + 250 \\
 0 \\
 0 \\
 0 \\
 0
 \end{pmatrix} \tag{2.31}$$

U nastavku je prikazana reducirana matrica pomaka:

$$\begin{pmatrix} v_{1x} & v_{1y} & \theta_1 & v_{2x} & v_{2y} & \theta_2 & v_{3x} & \theta_3 & v_{4x} & v_{4y} & \theta_4 \\ -3191.58 & 0 & -3191.58 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2,14E+10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3191.58 & 0 & 2127.72 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2,14E+10 & 0 & 3191.58 & -2,14E+10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2,14E+12 & 3191.58 & 0 & -3191.58 & 3191.58 & 0 & 0 & 0 & 0 & 0 \\ 3191.58 & 3191.58 & 8519.88 & 0 & -3191.58 & 2172.72 & 0 & 0 & 0 & 0 & 0 \\ -2,14E+10 & 0 & 0 & 4,28E+12 & 0 & 3191.58 & -3191.58 & 3191.58 & -2,14E+10 & 0 & 0 \\ 0 & -3191.58 & -3191.58 & 0 & 2,14E+12 & 0 & 0 & 0 & 0 & -3191.58 & 3191.58 \\ 0 & 3191.58 & 2127.72 & 3191.58 & 0 & 12766.3 & -3191.58 & 2172.27 & 0 & -3191.58 & 2127.72 \\ 0 & 0 & 0 & -3191.58 & 0 & -3191.58 & 3191.58 & -3191.58 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2,14E+10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3191.58 & 0 & 2127.72 & -3191.58 & 4255.44 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2,14E+10 & 0 & 0 & 0 & 0 & 2,14E+12 & 0 & 3191.58 \\ 0 & 0 & 0 & 0 & -3191.58 & -3191.58 & 0 & 0 & 0 & 2,14E+12 & -3191.58 \\ 0 & 0 & 0 & 0 & 3191.58 & 2127.72 & 0 & 0 & -3191.58 & -3191.58 & 8510,88 \end{pmatrix} \cdot \begin{pmatrix} v_{0x} \\ v_{0y} \\ \theta_0 \\ v_{1x} \\ v_{1y} \\ \theta_1 \\ v_{2x} \\ v_{2y} \\ \theta_2 \\ v_{3x} \\ \theta_3 \\ v_{4x} \\ v_{4y} \\ \theta_4 \end{pmatrix} = \begin{pmatrix} f_{1x} \\ f_{1y} - 1000 \\ m_{1z} - 333,34 \\ f_{2x} \\ -1500 \\ m_{2z} + 83,334 \\ f_{3x} \\ m_{3z} \\ f_{4x} \\ f_{4y} - 500 \\ m_{4z} + 250 \end{pmatrix} \quad (2.32)$$

U nastavku je prikazan izraz sa kojim se određuje reakcije u čvorovima. Svi podcrtani elementi ne utječu na određivanje reakcije u čvorovima[3]:

$$[\bar{K}] \cdot \{v\} = \{f\}_{ef} \tag{2.33}$$

v_{0x}	v_{0y}	θ_0	v_{1x}	v_{1y}	θ_1	v_{2x}	v_{2y}	θ_2	v_{3x}	v_{3y}	θ_3	v_{4x}	v_{4y}	θ_4	v_{5x}	v_{5y}	θ_5
3191.58	0	-3191.58	-3191.58	0	-3,19E+21	0	0	0	0	0	0	0	0	0	0	0	0
0	2,14E+10	0	0	-2,14E+10	0	0	0	0	0	0	0	0	0	0	0	0	0
-3191.58	0	4266.44	3191.58	0	2,13E+21	0	0	0	0	0	0	0	0	0	0	0	0
-3191.58	0	3191.58	2,14E+10	0	3191.58	-2,14E+10	0	0	0	0	0	0	0	0	0	0	0
0	-2,14E+10	0	0	2,14E+10	3191.58	0	-3191.58	3191.58	0	0	0	0	0	0	0	0	0
-3191.58	0	2127.72	3191.58	3191.58	8510.88	0	-3191.58	2127.72	0	0	0	0	0	0	0	0	0
0	0	0	-2,14E+25	0	0	4,28E+12	0	3191.58	-3191.58	0	3191.58	-2,14E+10	0	0	0	0	0
0	0	0	0	-3191.58	-3191.58	0	2,14E+10	0	0	-2,14E+10	0	0	-3191.58	3191.58	0	0	0
0	0	0	0	3191.58	2127.72	3191.58	0	12766.3	-3191.58	0	2127.72	0	-3191.58	2127.72	0	0	0
0	0	0	0	0	0	-3191.58	0	-3191.58	3191.58	0	-3191.58	0	0	0	0	0	0
0	0	0	0	0	0	0	-2,14E+10	0	0	2,14E+10	0	0	0	0	0	0	0
0	0	0	0	0	0	3191.58	0	2127.72	-3191.58	0	4255.44	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	2,14E+12	0	3191.58	-3191.58	3191.58
0	0	0	0	0	0	0	-3191.58	-3191.58	0	0	0	0	2,14E+12	-3191.58	0	-2,14E+10	0
0	0	0	0	0	0	0	3191.58	2127.7	0	0	0	3191.58	-3191.58	8510.88	-3191.58	0	2127.72
0	0	0	0	0	0	0	0	0	0	0	0	-3191.58	0	-3191.58	3191.58	0	-3191.58
0	0	0	0	0	0	0	0	0	0	0	0	0	-2,14E+10	0	0	2,14E+10	0
0	0	0	0	0	0	0	0	0	0	0	0	3191.58	0	2127.72	-3191.58	0	4255.44

v_{0x}

v_{0y}

θ_0

$m\theta_{0z}$

v_{1x}

v_{1y}

θ_1

v_{2x}

v_{2y}

θ_2

v_{3x}

v_{3y}

θ_3

v_{4x}

v_{4y}

θ_4

v_{5x}

v_{5y}

θ_5

·

θ_{0x}

θ_{0y}

$m\theta_{0z}$

θ

θ

θ

θ

θ

θ

θ

v_{3y}

θ

θ

θ

θ

v_{5x}

v_{5y}

θ_{5z}

=

f_{0x}

f_{0y}

m_{0z}

θ

-1000

$-333,34$

θ

-1500

$83,334$

θ

f_{3y}

θ

θ

-500

250

f_{5x}

f_{5y}

m_{5z}

$$(2.34)$$

Prikaz reduciranog izraza:

$$\begin{matrix}
 v_{0x} & v_{0y} & \theta_0 & v_{3y} & v_{5x} & v_{5y} & \theta_5 \\
 \left(\begin{array}{ccccccc}
 3191.58 & 0 & -3191.58 & 0 & 0 & 0 & 0 \\
 0 & 2,14E+10 & 0 & 0 & 0 & 0 & 0 \\
 -3191.58 & 0 & 4266.44 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 2,14E+10 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 3191.58 & 0 & -3191.58 \\
 0 & 0 & 0 & 0 & 0 & 2,14E+10 & 0 \\
 0 & 0 & 0 & 0 & -3191.58 & 0 & 4255.44
 \end{array} \right) & \cdot & \begin{Bmatrix} v_{0x} \\ v_{0y} \\ \theta_0 \\ v_{3y} \\ v_{5x} \\ v_{5y} \\ \theta_5 \end{Bmatrix} & = & \begin{Bmatrix} f_{0x} \\ f_{0y} \\ m_{0z} \\ f_{3y} \\ f_{5x} \\ f_{5y} \\ m_{5z} \end{Bmatrix} & (2.33)
 \end{matrix}$$

Rješavanjem jednadžbe:

$$[\bar{K}] \cdot \{v\} = \{f\}_{ef} \quad (2.34)$$

dobiju se pomaci u čvorovima.

$$\{v\} = [\bar{K}]^{-1} \cdot \{f\}_{ef} \quad (2.35)$$

2.6. Određivanje sila u elementima

Da bi se odredile sile (uzdužne, poprečne i momenti) u čvorovima konačnog elementa za okvirne konstrukcije, potrebno je primijeniti ravnotežu na samom elementu. Matrica krutosti elementa se koristi u lokalnom koordinatnom sustavu. Izraz glasi[4]:

$$\{\mathbf{v}\}^{(a)} = [\mathbf{T}] \cdot \{\bar{\mathbf{v}}\}^{(a)} \quad (2.36)$$

Gdje su:

- $\{\mathbf{v}\}^{(a)}$ – lokalni pomaci čvorova u elementima „a“ označava broj elementa
- $[\mathbf{T}]$ – transformacijske matrice
- $\{\bar{\mathbf{v}}\}^{(a)}$ – globalni pomaci čvorova u elementima „a“ označava broj elementa

Nakon toga, vektor pomaka čvorova tog elementa se transformira iz globalnog u lokalni koordinatni sustav. Tek tada se sile u elementima mogu se izračunati koristeći odgovarajuće izraze. Taj izraz glasi[4]:

$$\mathbf{S}^{(a)} = [\mathbf{K}]^{(a)} \cdot \{\mathbf{v}\}^{(a)} - \{\mathbf{f}\}_{ef}^{(a)} \quad (2.37)$$

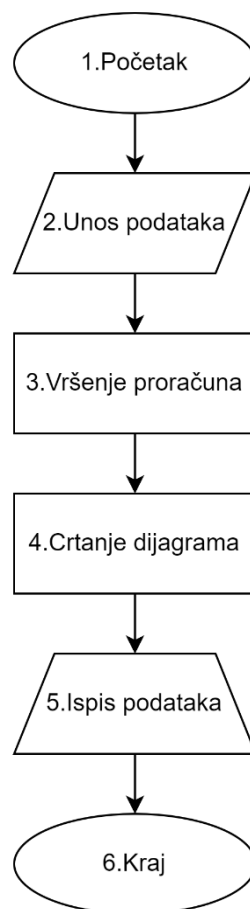
Gdje su:

- $\mathbf{S}^{(a)}$ – sile u elementima, „a“ označava broj elementa
- $\{\mathbf{f}\}_{ef}^{(a)}$ – lokalni vektor sila, gdje „a“ označava broj elementa

lokalni vektori efektivnih sila

3. Dijagrami toka

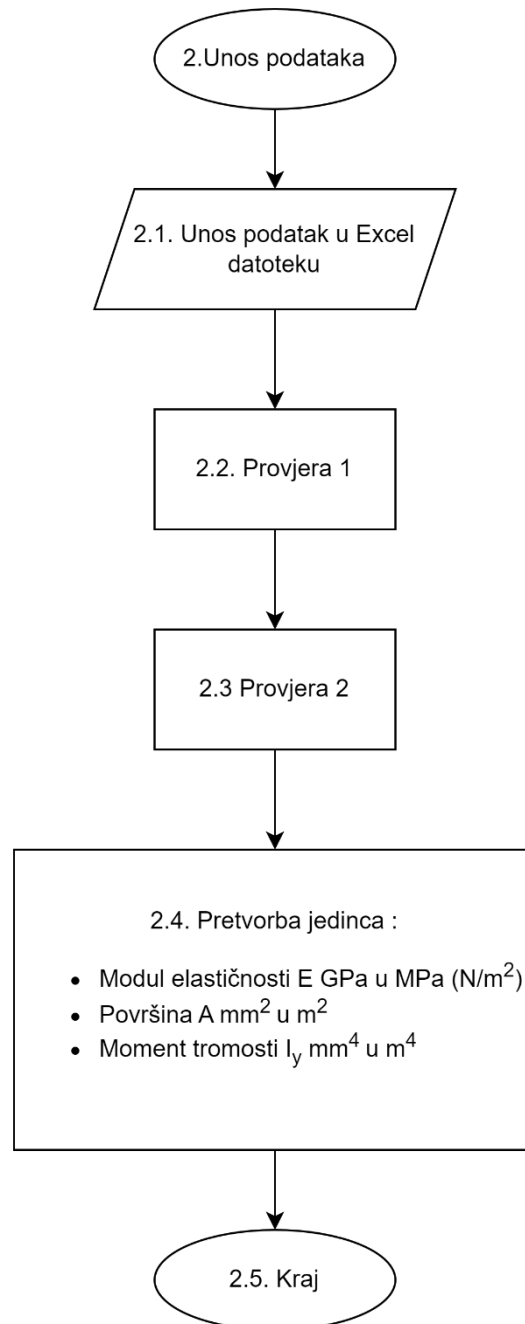
Za početak programiranja moraju se odrediti varijable programa što je prikazano u dijagramima toka. Prije izrade programskog rješenja izrađen je osnovni dijagrama toka koji prikazuje izvođenje svih radnji. Na slici 3.1 je osnovni dijagram toka. Svi dijagrami toka su napravljeni pomoću online aplikacije „*draw.io*“.



Slika 3.1 Opći dijagram toka

3.1. Unos podataka

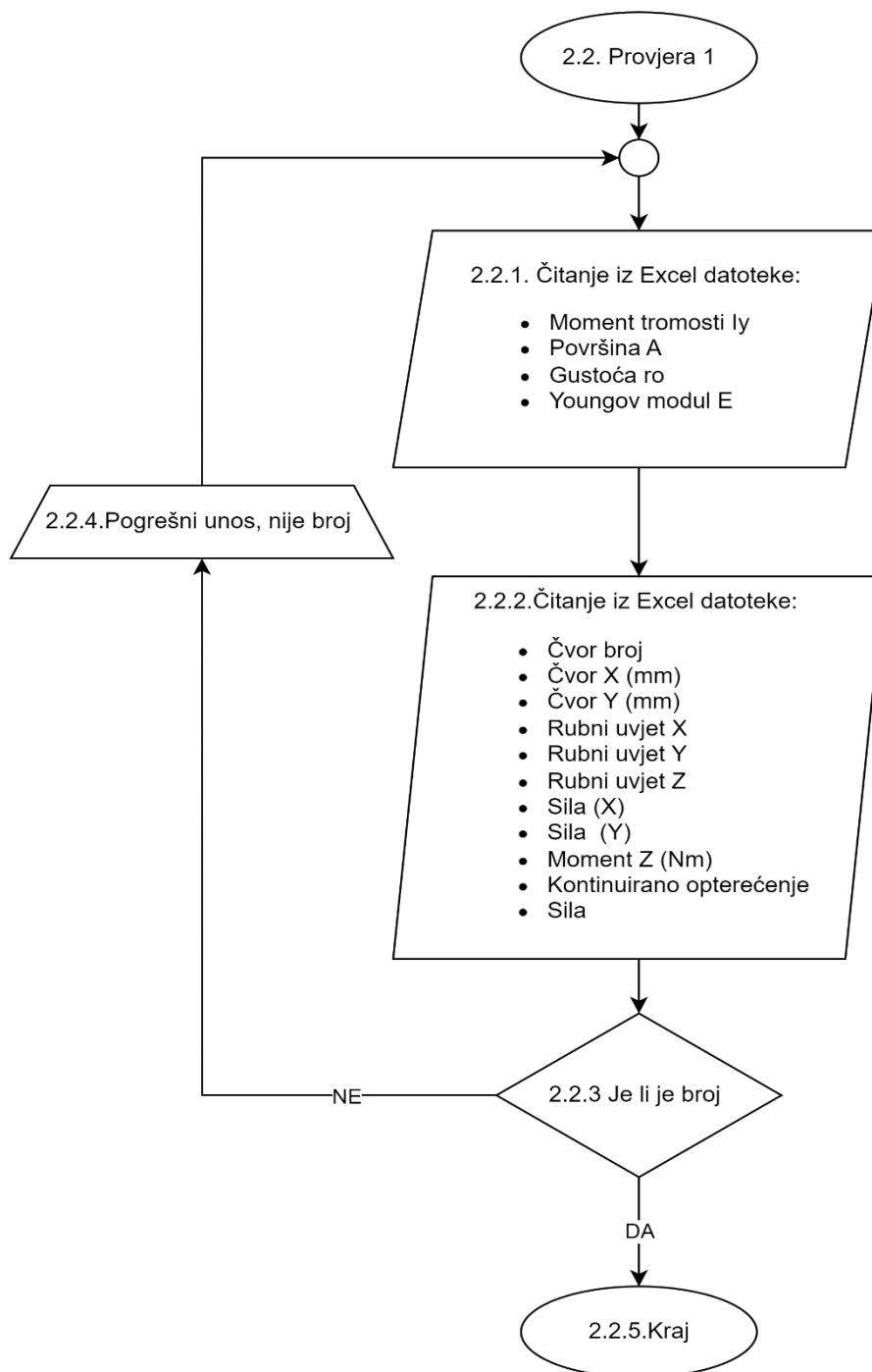
U ovom dijagramu toka pod nazivom unos podataka prikaza uvoz podatka iz excel datoteke. Još se vrši provjera unosa podataka pod provjera 1 i provjera 2. Dijagram toka je prikazan na slici 3.2.



Slika 3.2 Dijagram toka za unos podataka

3.2. Provjera 1

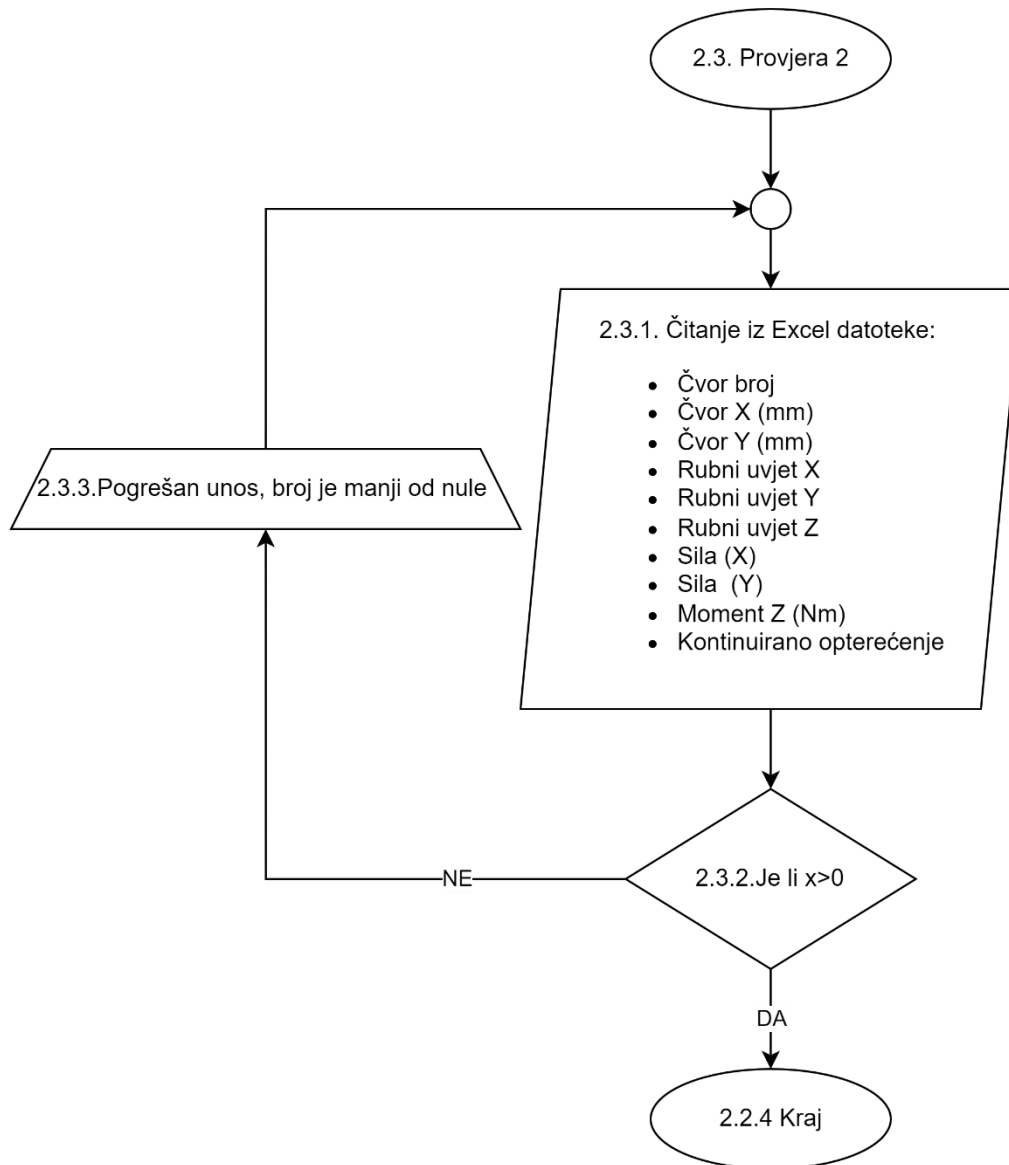
U dijagramu toka provjera 1 se provjera dali li su unesi podaci iz excel datoteka. U slučaju pogrešnog unosa program završava rad i ispisuje poruku pogrešan, pogrešan unos podataka u excel datoteci“. Na slici 3.3 je prikazan dijagram toka „provjera 1“.



Slika 3.3 Dijagram toka provjera 1

3.3. Provjera 2

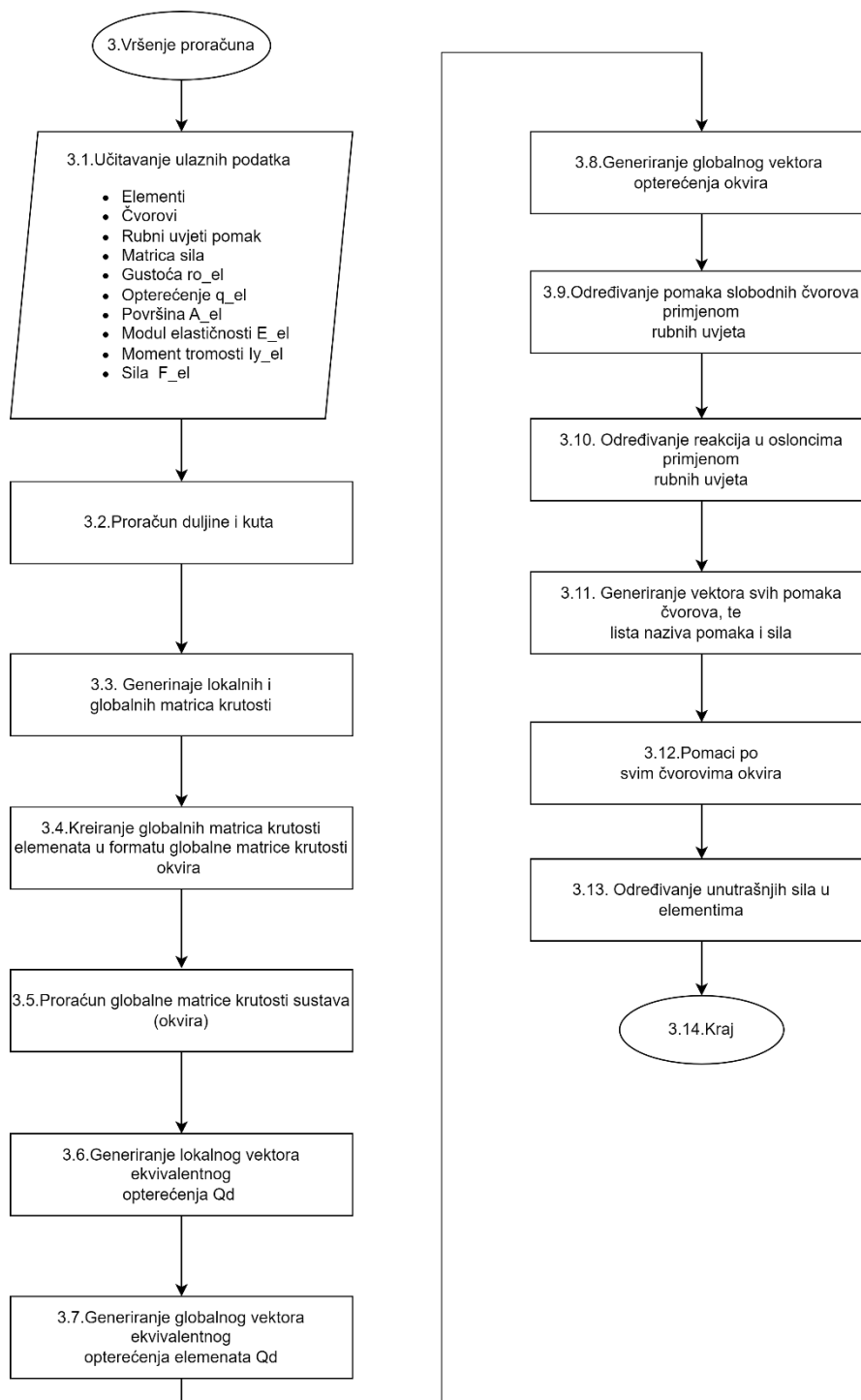
U dijagramu toka vrši se provjera podataka dali je broj veći od nule, ako je pogrešan unos uvraća poruku „pogrešan pogrešan unos, broj je manji ili jednak nuli“ i program završava sa radom. Na slici je prikazn dijgram toka provjera 2.



Slika 3.4 Dijagram toka provjera 2

3.4. Proračun (glavni program)

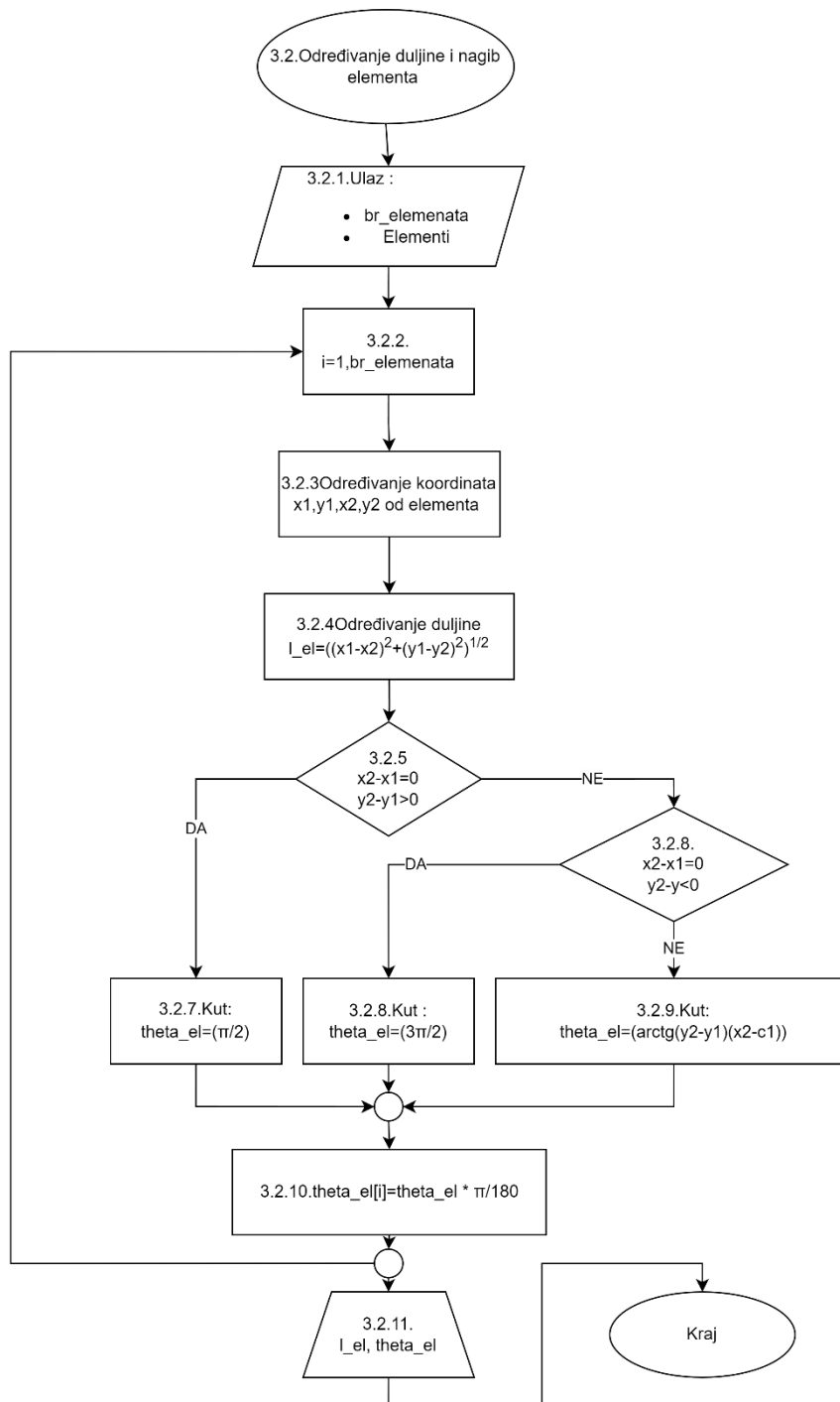
U dijagramu toka "Proračun" je prikaz tok izvršavanja glavnog programa (slika 3.5).



Slika 3.5 Prikaz dijagrama toka vršenje proračuna

3.5. Proračun duljine i kuta nagiba elemenata

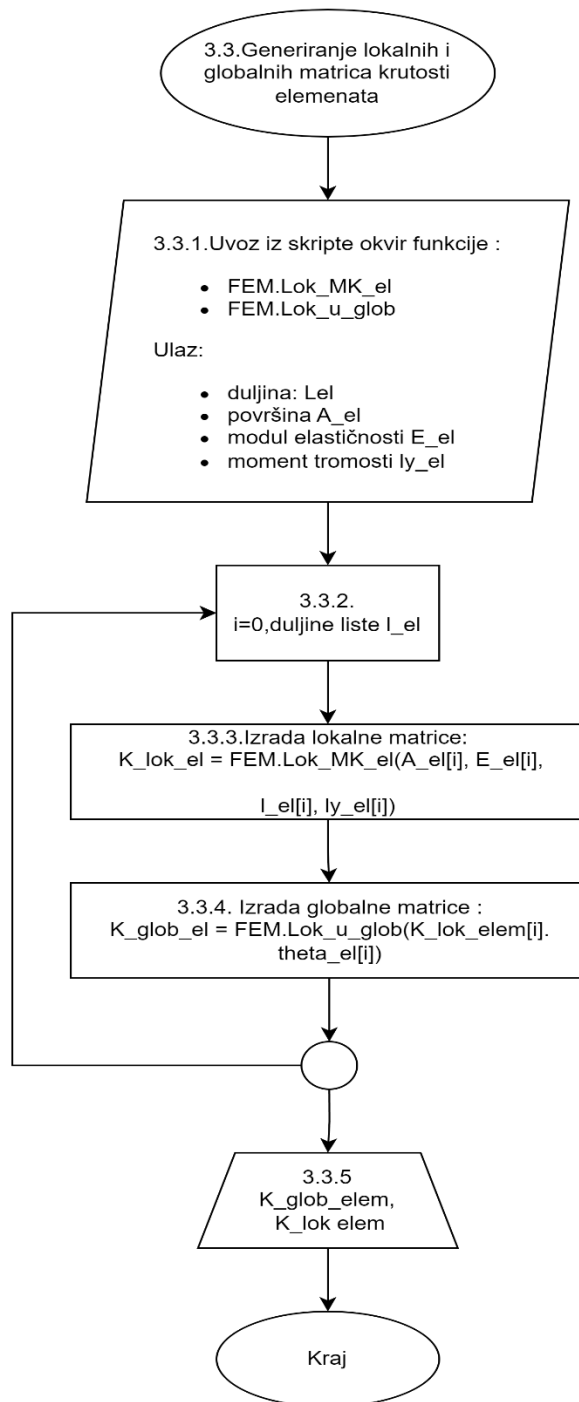
Cilj ovog dijagrama toka je prikazati kako bi se trebali odrediti duljine i nagib elemenata. Dijagram toka je prikazan na slici 3.6.



Slika 3.6 dijagram toka određivanje duljine i nagiba elementa

3.6. Generiranje lokalnih i globalnih matrica krutosti

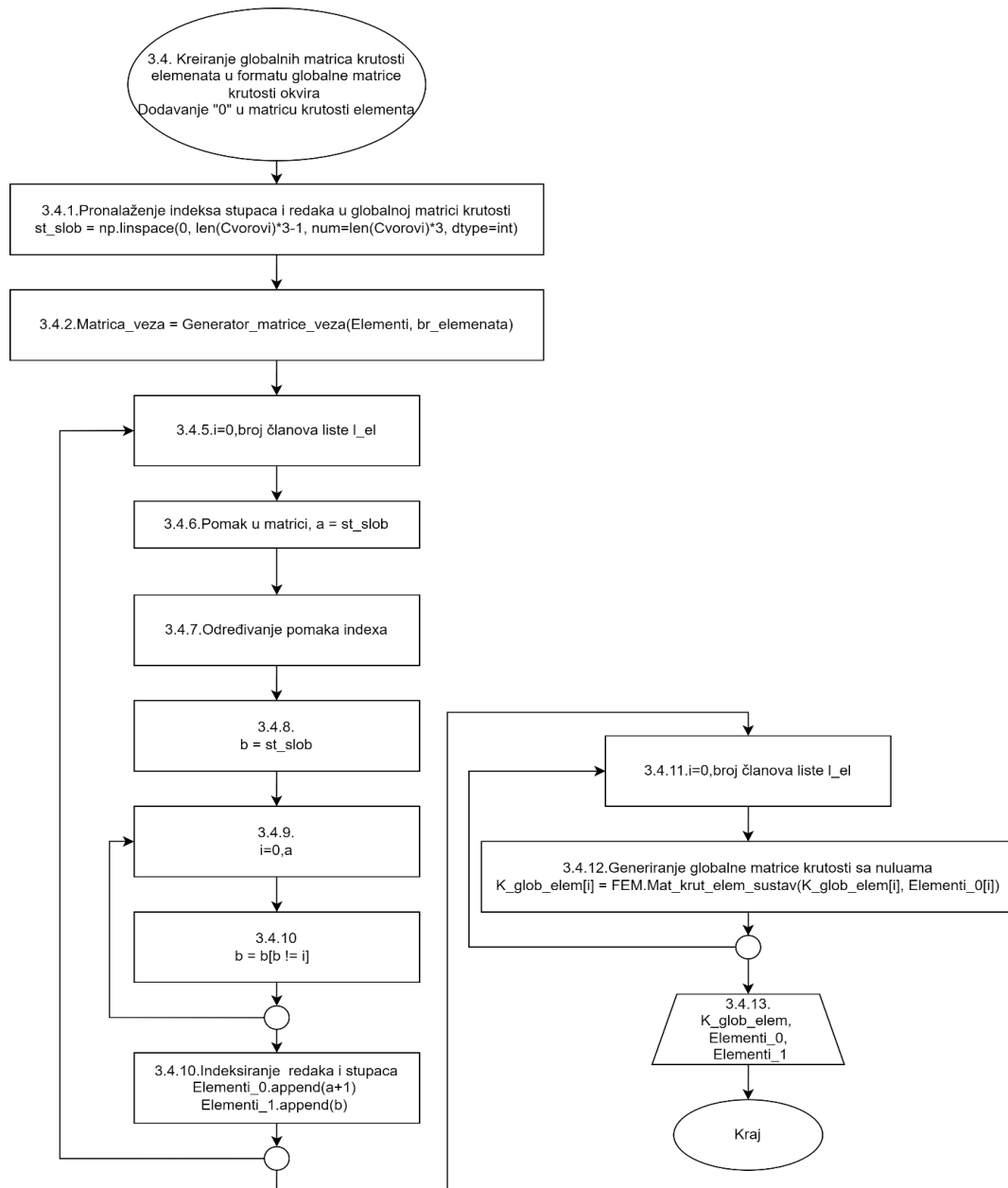
Na slici 3.7 prikazan je dijagram toka za generiranje lokalnih i globalnih matrica krutosti elemenata



Slika 3.7 dijagram toka generiranje lokalnih i globalnih matrica krutosti

3.7. Kreiranje globalnih matrica krutosti elemenata u formatu globalne matrice krutosti okvira

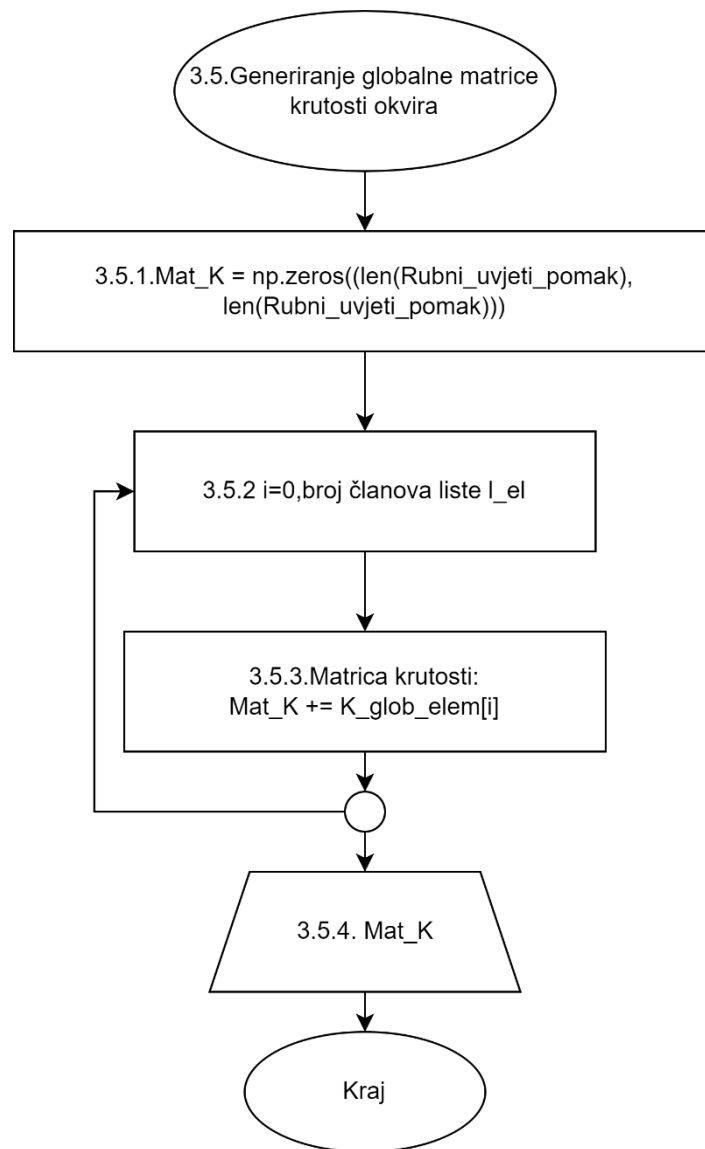
Cilj ovog dijagrama toka je prikaz dodavanja nula u postojeće globalne matrice krutosti elemenata kako bi iste dobile dimenziju globalne matrice krutosti sustava. Dijagram toka prikazan je na slici 3.8.



Slika 3.8 Kreiranje globalnih matrica krutosti elemenata u formatu globalne matrice krutosti okvira

3.8. Generiranje globalne matrice krutosti okvira

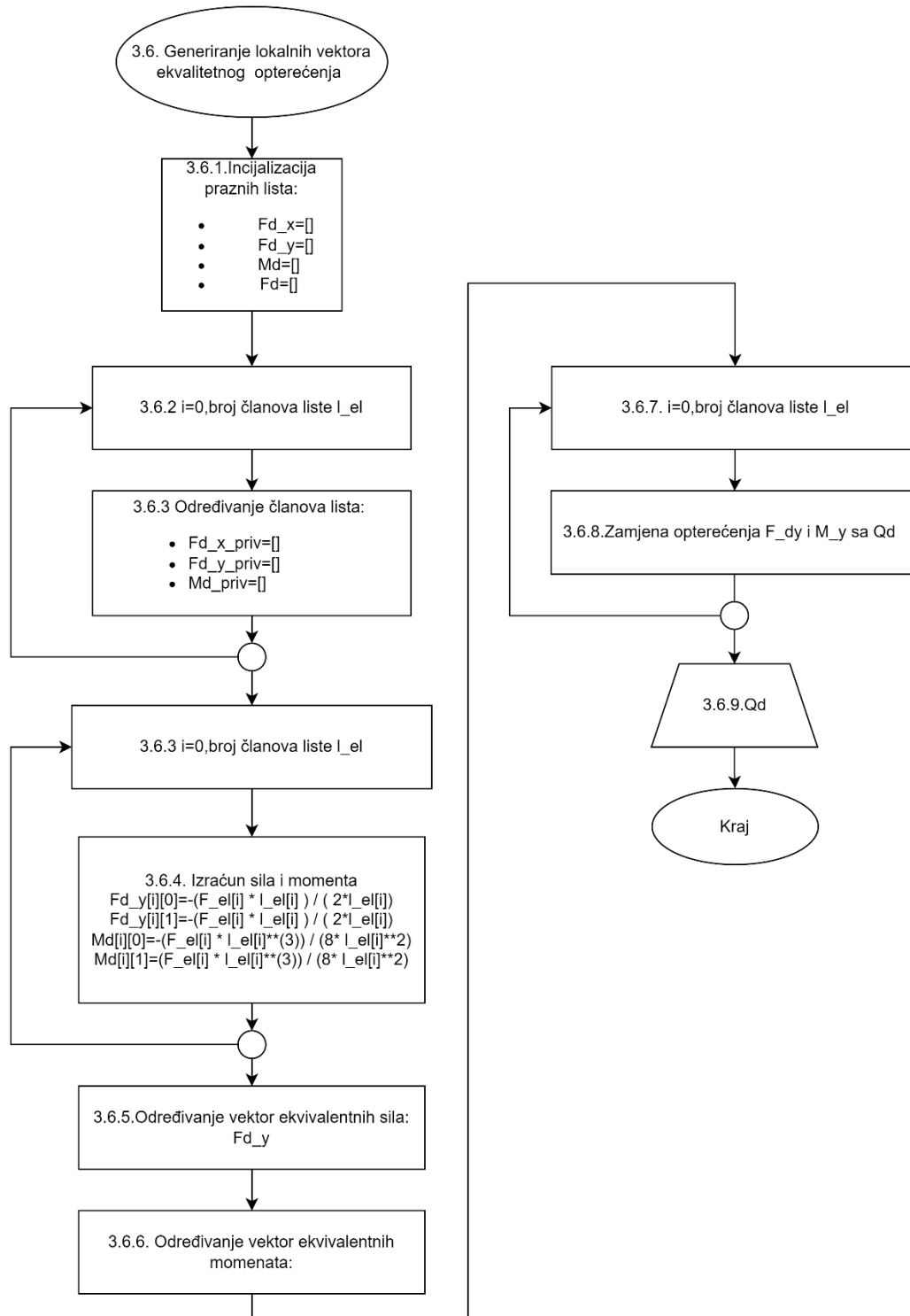
U ovom dijagramu toka je prikazano ulančavanje globalnih matrica krutosti elemenata s ciljem dobivanja globalne matrice krutosti sustava što je prikazano na slici 3.9.



Slika 3.9 Dijagram toka generiranje globalnih matrica krutosti okvira

3.9. Generiranje lokalnih vektora ukupnog opterećenja okvira

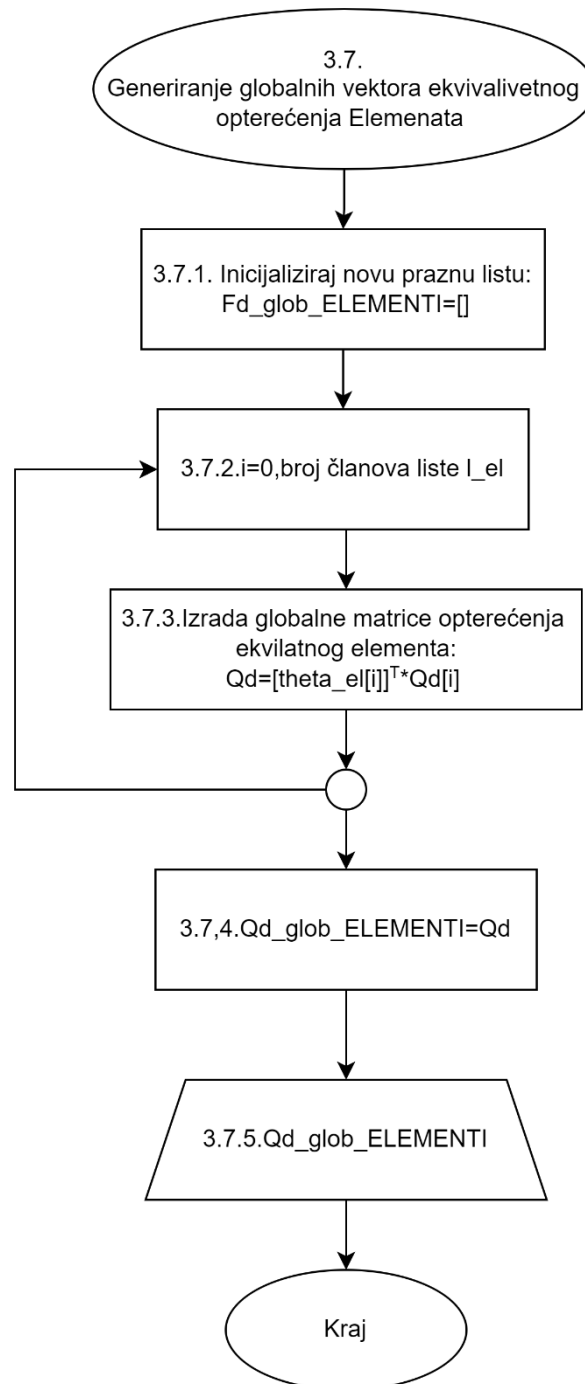
U ovom dijagramu toka je opisano generiranje lokalnih vektora opterećenja okvira, a njegov prikaz se nalazi na slici 3.10.



Slika 3.10 Dijagram toka generiranje lokalnih vektora ukupnog opterećenja okvira

3.10. Generiranje globalnih vektora ekvivalentnog opterećenja elemenata

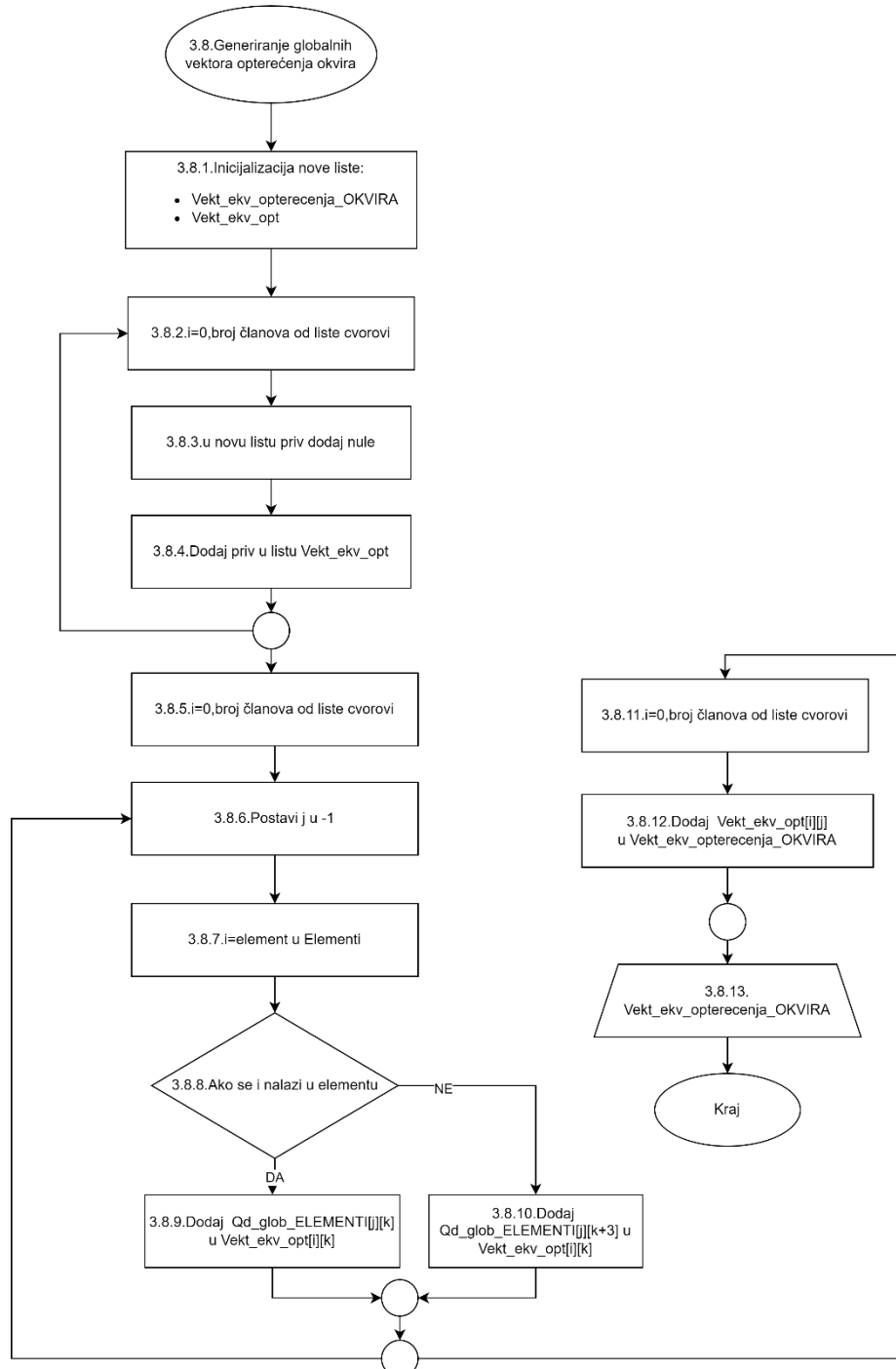
U ovom dijagramu toka je opisano generiranje globalnih vektora ekvivalentnog opterećenja elemenata, a njegov prikaz se nalazi na slici 3.11.



Slika 3.11 Dijagram toka generiranje globalnih vektora ekvivalentnog opterećenja elemenata

3.11. Generiranje globalnih vektora opterećenja okvira

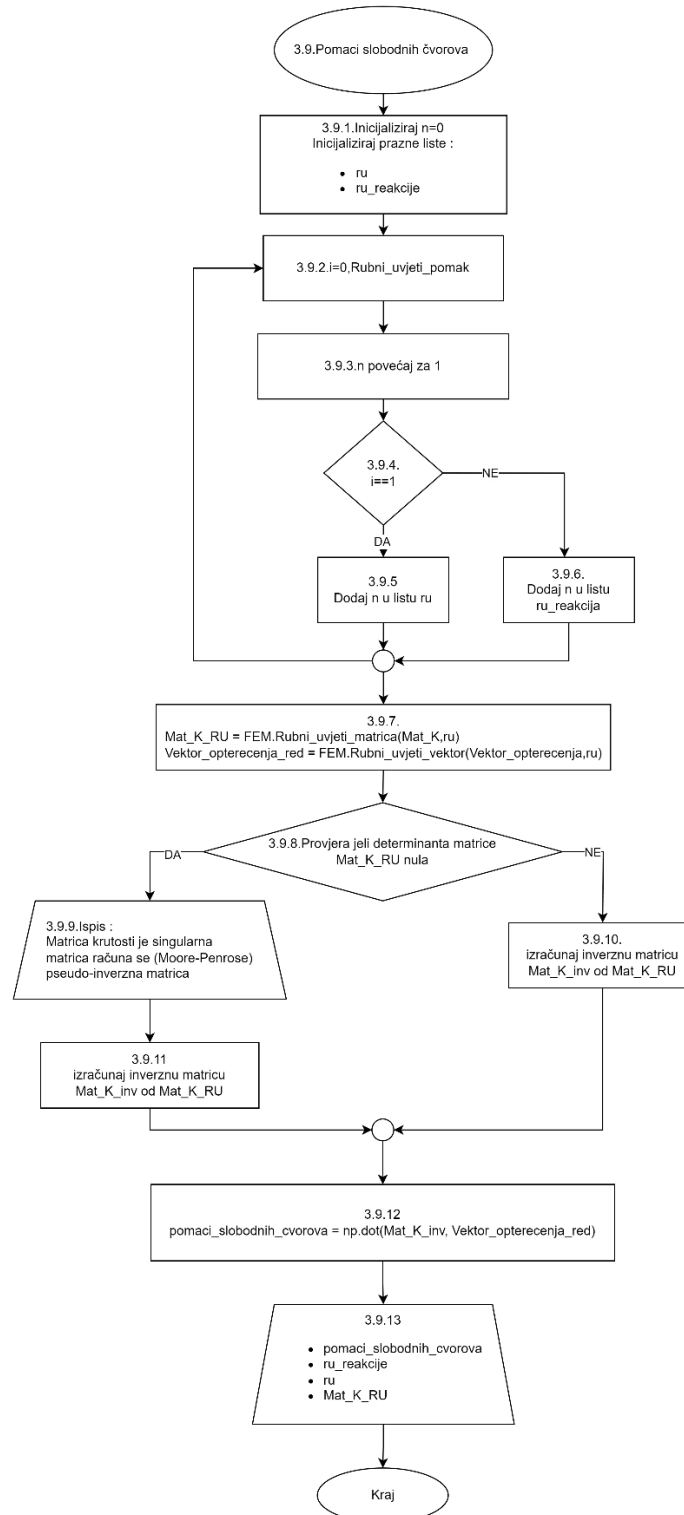
U ovom dijagramu toka je opisano generiranje globalnih vektora opterećenja okvira, a njegov prikaz se nalazi na slici 3.12.



Slika 3.12 Dijagram toka generiranje globalnih vektora opterećenja okvira

3.12. Određivanje pomaka slobodnih čvorova primjenom rubnih uvjeta

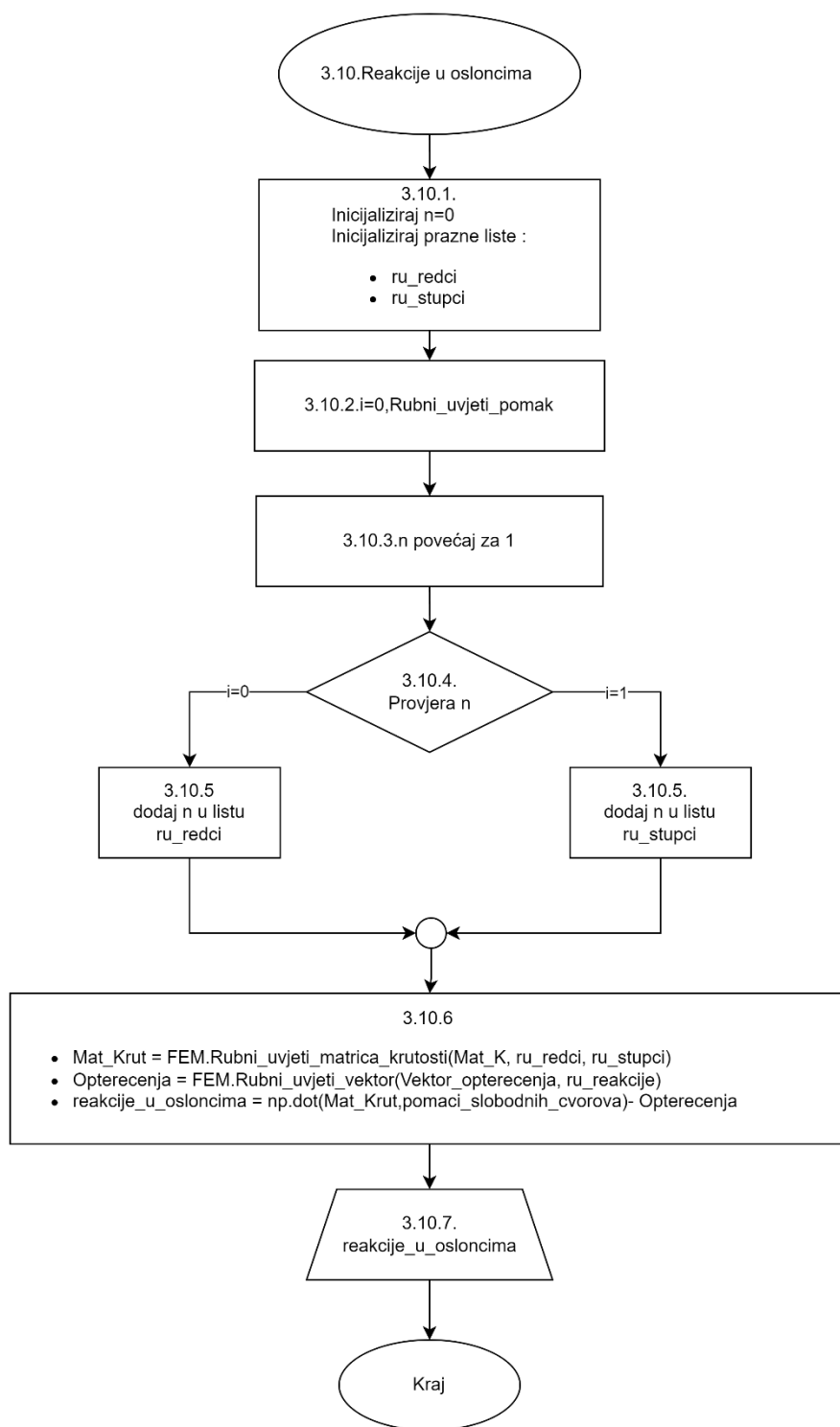
Određivanje pomaka slobodnih čvorova primjenom rubnih uvjeta prikazano je u dijagramu toka koji se nalazi na slici 3.13.



Slika 3.13 dijagram toka pomaci slobodnih čvorova

3.13. Određivanje reakcija u osloncima primjenom rubnih uvjeta

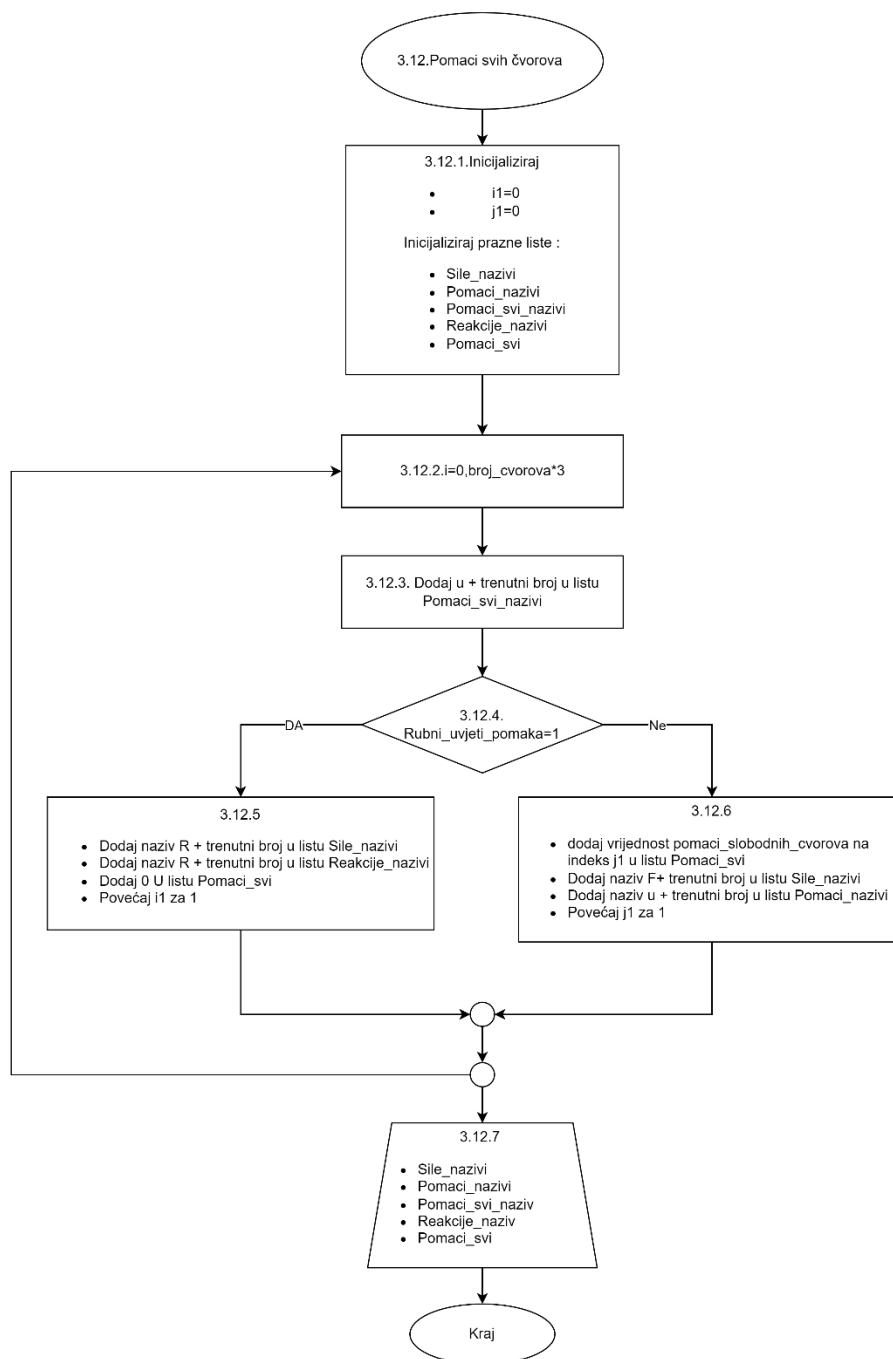
U ovom dijagramu toka određene su sve reakcije na osloncima primjenom rubnih uvjeta (slika 3.14).



Slika 3.14 Dijagram reakcije u osloncima

3.14. Generiranje vektora svih pomaka čvorova, te lista naziva pomaka i sila

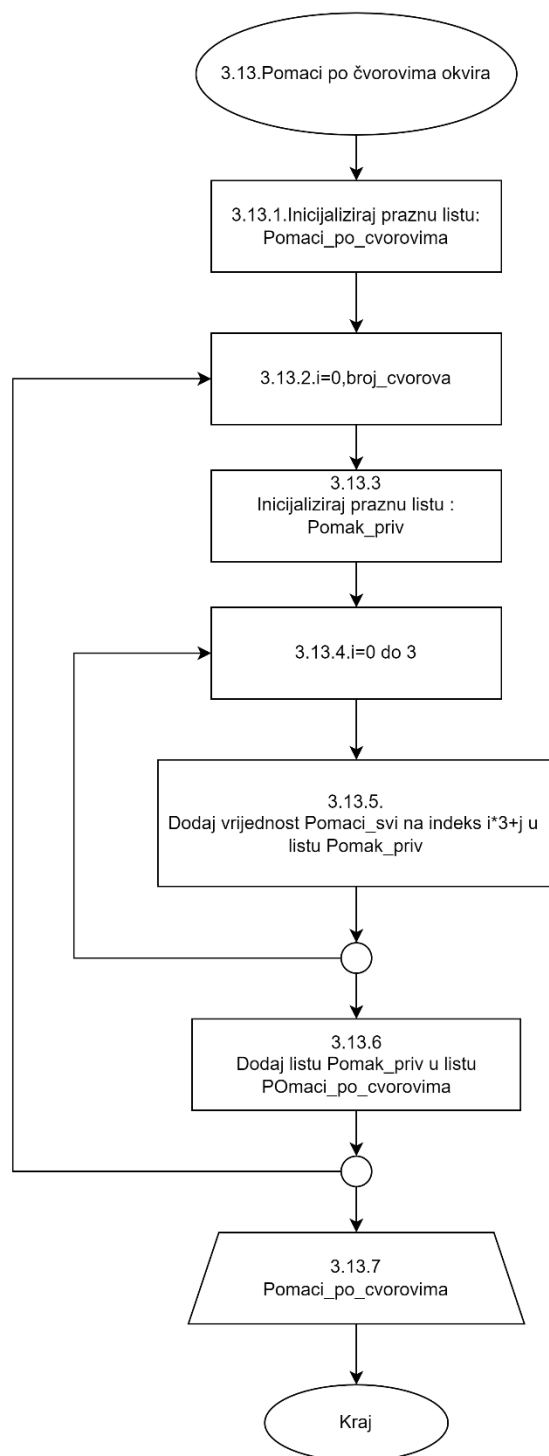
U dijagramu toka pod nazivom „Pomaci slobodnih čvorova“ su određeni vektori svih pomaka čvorova te izrađene liste naziva pomaka i sila. Dijagram toka je prikazan na slici 3.13.



Slika 3.15 Dijagram toka pomaci svih čvorova

3.15. Pomaci svih čvorova okvira

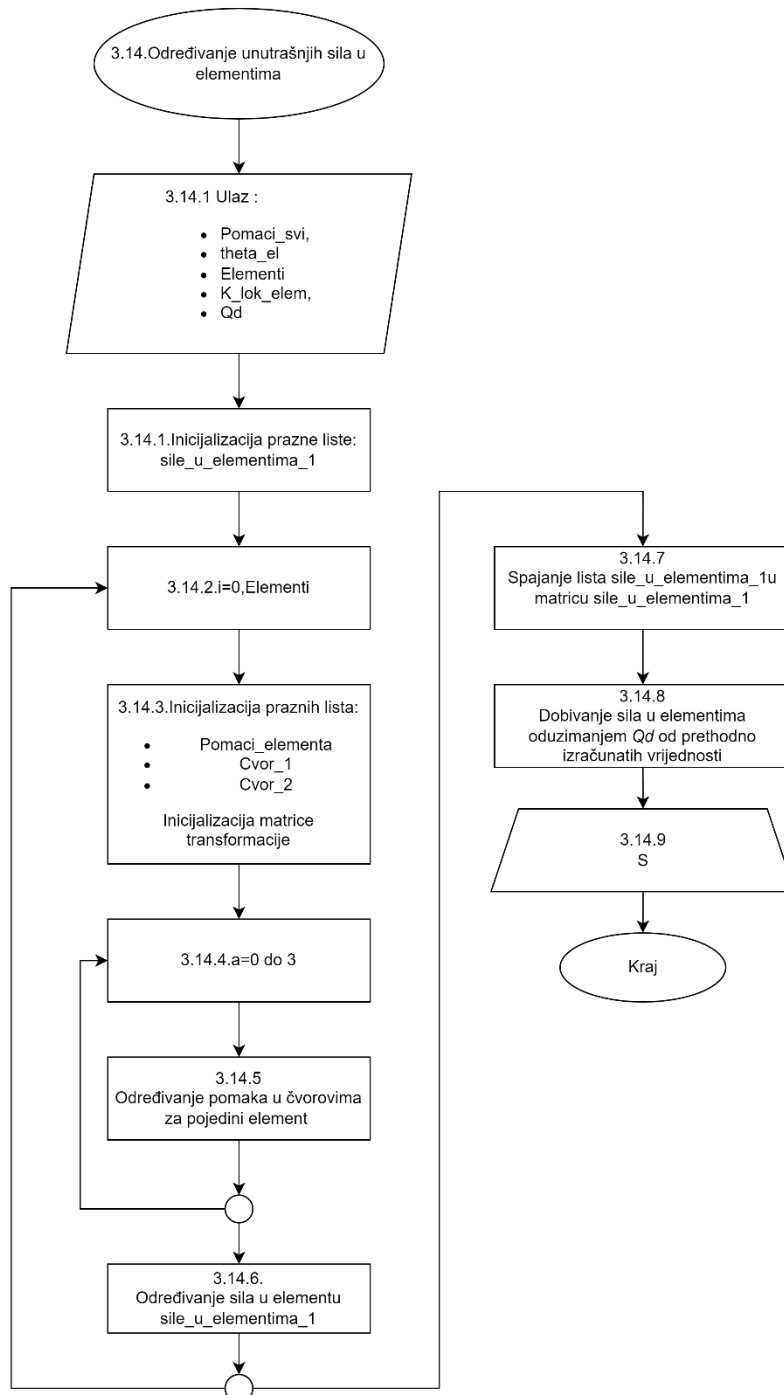
Dijagram toka pomaka svih čvorova okvira prikazan je na slici 3.16.



Slika 3.16 dijagram toka pomaci po svim čvorovima

3.16. Određivanje unutrašnjih sila(momenta, uzdužna i poprečna sila) u elementima

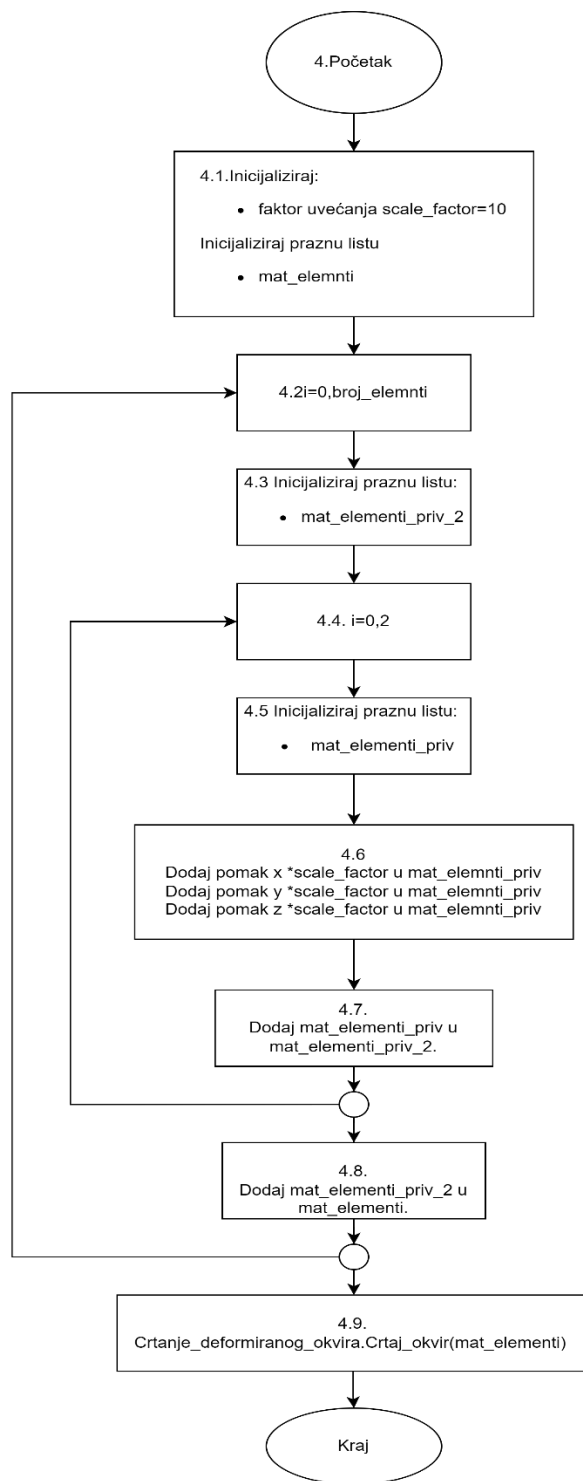
U ovom poglavlju će biti prikaz dijagram toka za određivanja unutrašnjih sila u elementima. Na slici 3.17 je prikazan dijagram toka za određivanje unutrašnjih sila u elementima.



Slika 3.17 Dijagram toka za određivanje unutrašnjih sila(momenta, uzdužna i poprečna sila) u elementima

3.17. Crtanje dijagrama za deformiran i nedeformiran okvirni nosač

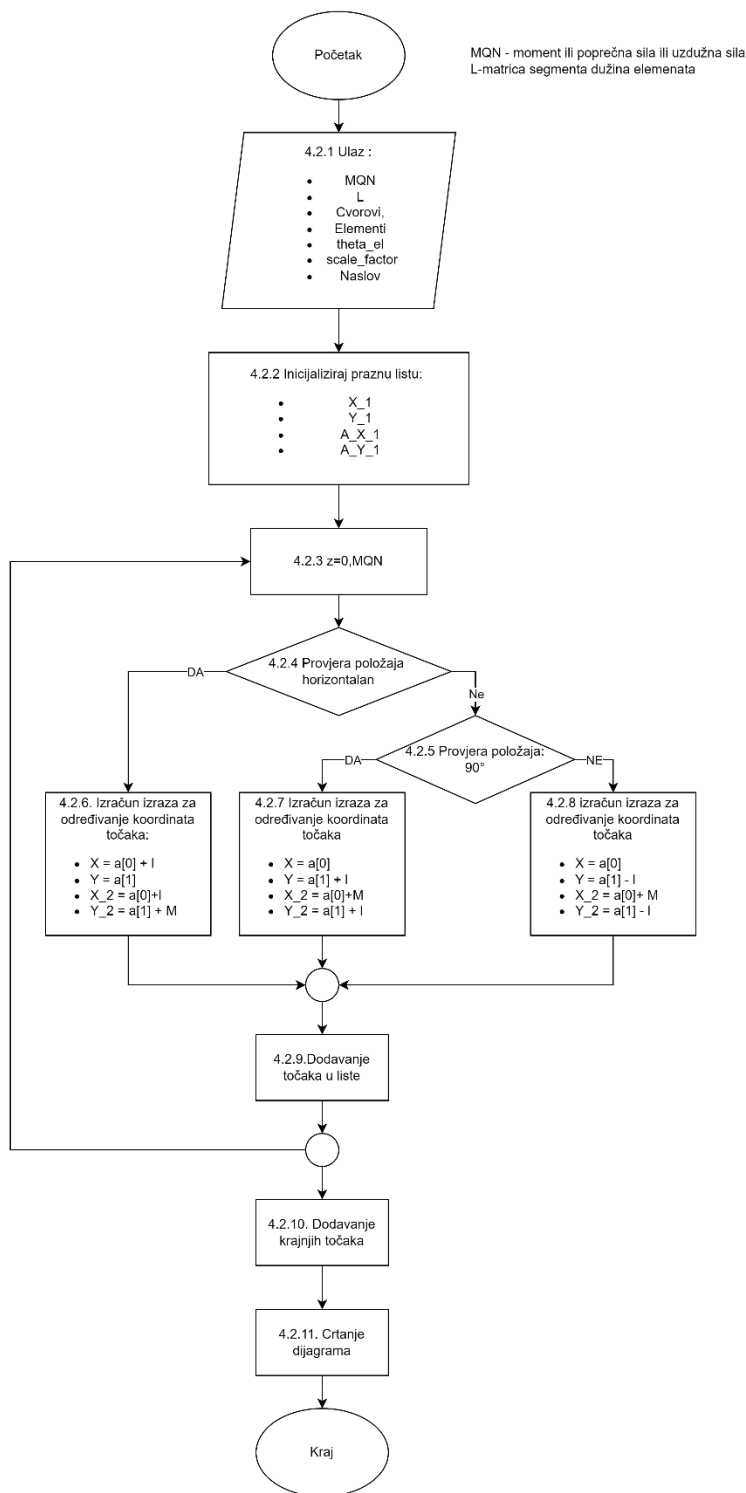
Na slici 3.18 je prikazan dijagram toka za crtanje dijagrama za deformiran i nedeformiran okvirni nosač.



Slika 3.18 dijagram toka za crtanje dijagrama za deformiran i nedeformiran okvirni nosač

3.18. Crtanje dijagrama momenata, poprečnih i uzdužnih sila

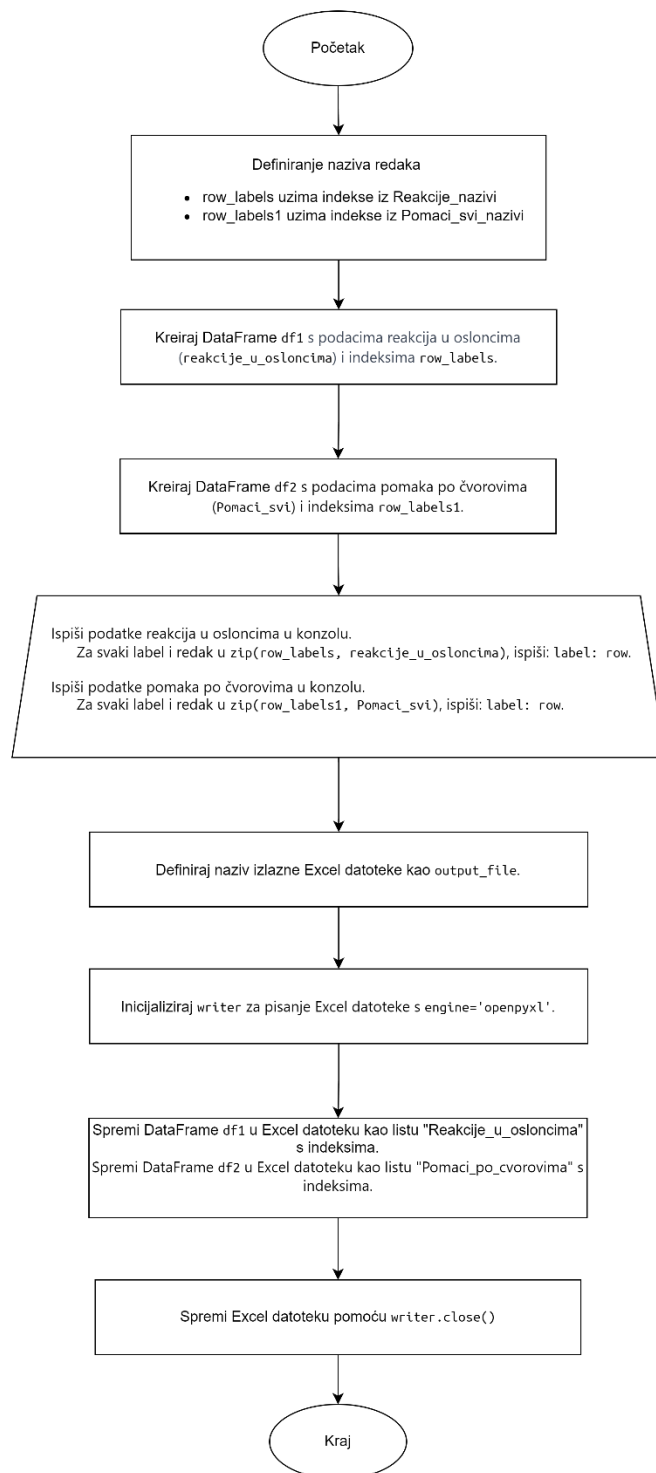
Slika 3.19 daje prikaz dijagrama toka za crtanje dijagrama momenata, poprečnih i uzdužnih sila.



Slika 3.19 Prikaz dijagrama toka za crtanje dijagrama momenata, poprečnih uzdužnih sila

3.19. Ispis podataka

U ovom dijagram toka (slika 3.16) je prikazan način ispisa podatka na konzoli i u excel datoteku.



Slika 3.20 dijagram toka ispis podatka

4. Lista varijabli i funkcija

A_el - varijabla za površinu

Crtanje_deformiranog_okvira – funkcija za crtanje deformiranog okvira

Crtanje_okvira - funkcija za crtanje nedeformiranog okvira

Cvorovi – lista sa koordinatama čvorova

Dodaj_0_u_mat_krut_elementa – funkcija za dodavanje nula u globalnu matricu krutosti

Duljine_nagibi_elementa – funkcija za određivanje nagiba i duljine elementa

E_el – lista modula elastičnosti

Elementi – lista koja definira čvorove elementa

Elementi_0 – matrica indeksa redaka i stupac = 0

Elementi_1 – matrica indeksa redaka i stupaca $\neq 0$

Generiraj_glob_mat_krut_sustava – funkcija za generiranje matrice krutosti sustava

Generiraj_globalni_vektor_ekv_opt_ELEMENATA – funkcija za generiranje globalnih vektora ekvivalentnog kontinuiranog opterećenja elementa

Generiraj_globalni_vektor_ekv_opt_OKVIRA – funkcija za generiranje globalnih vektora ekvivalentnog kontinuiranog opterećenja elementa okvira

Generiraj_lokalni_vektor_ekv_op – funkcija za generiranje globalnih vektora ekvivalentnog kontinuiranog opterećenja lokalnih elementa

Generiraj_lokalni_vektor_konc_opt – funkcija za generiranje globalnih vektora ekvivalentnog kontinuiranog opterećenja lokalnih elementa

Globalne_i_lokalne_matrice_krutosti_elementa – funkcija za generiranje globalnih i lokalnih matrica krutosti elementa

Iy_el – lista momenata tromosti elementa

K_glob_elem – lokalne matrice krutosti elementa

K_lok_elem – lokalne matrice krutosti elementa

Mat_K – matrica krutosti

Mat_K_RU – reducirana matrica krutosti

Matrica_sila – matrica sila

Odredi_reakcije_u_osloncima – funkcija za određivanje reakcije u osloncima

Pomaci_nazivi – lista naziva za pomake

Pomaci_po_cvorovima – matrica pomaka čvorova i nagiba čvorova

Pomaci_slobodnih_cvorova – matrica pomaka slobodnih čvorova

Pomaci_po_cvorovima_okvira – matrica pomaka čvorova po okviru

Pomaci_svi – lista svih pomaka čvorova

Pomaci_svi_nazivi – lista naziva svih pomaka čvorova

Pomaci_svih_cvorova – matrica svih pomaka čvorova

Qd – matrica kontinuiranih opterećenja elementa

Qd_glob_ELEMENTI – matrica globalnih kontinuiranih opterećenja

Reakcije_nazivi – lista naziva reakcija

Sile_nazivi – lista naziva sila

Uvoz_podataka – funkcija za uvoz podataka

Vekt_ekv_opterecenja_OKVIRA – matrica vektora ekvivalentnog opterećenja okvira

br_elementata – lista broja elementa

l_el – lista dužina elemenata

pomaci_slobodnih_cvorova – lista pomaka svih slobodnih čvorova

provjera_broja – funkcija za provjeru jesu li uvezeni podaci broj

provjera_je_li_broj_pozitivan – funkcija za provjeru jesu li uvezeni podaci pozitivni broj

q_el – lista kontinuiranih opterećenja

reakcije_u_osloncima – funkcija za dobivanje reakcije u osloncima

ro_el – lista gustoća elementa

ru – lista učvršćenih čvorova po smjeru

ru_reakcije – lista slobodnih čvorova

theta_el – lista nagiba elementa

odredi_sile_momenta_u_elementima – funkcija za određivanje sila i momenta u čvorovima elemenata

S – lista sila u elementima

izracunaj_Q – funkcija za određivanje poprečnih sila u segmentima elementa

Q_el – lista poprečnih sila u segmentima elemenata

Q_l – lista segmenata elemenata

izracunaj_N – funkcija za određivanje uzdužnih sila u segmentima elementa

N_el – lista uzdužnih sila u segmentima elemenata

N_l – lista segmenata elemenata

izracunaj_M – funkcija za određivanje momenta u segmentima elementa

M_el – lista momenata u segmentima elemenata

l_M – lista segmenata elemenata

plot_diagrams – funkcija za crtanje dijagrama momenata uzdužnih i poprečnih sila u elementima

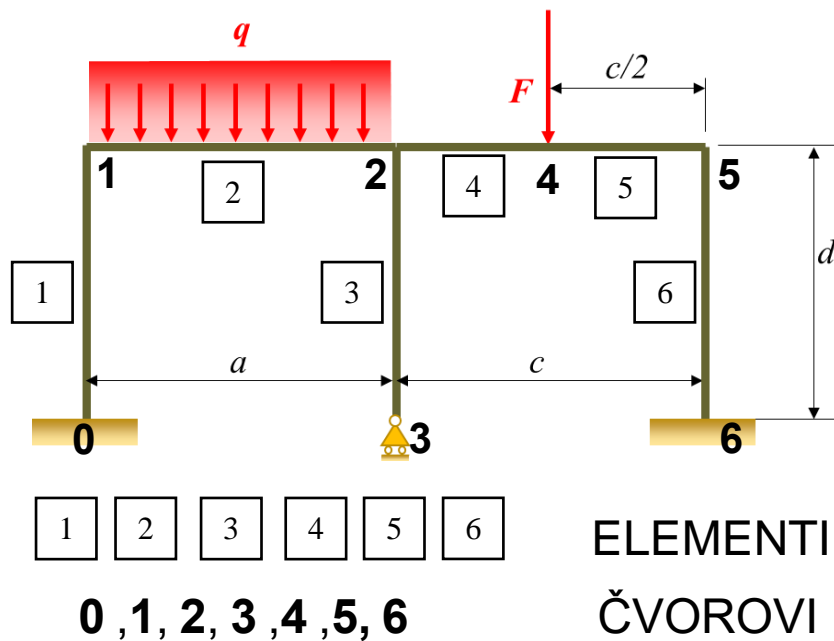
5. Opis programa

U ovom poglavlju su prikazani dijelovi Python kôda i dana su objašnjenja za navedeni kôd.

5.1. Glavni kôd koji se izvršava

Glavni programa možemo podijeliti u četiri skupine, unos podataka, vršenje proračuna, izvoz i ispis podatka te crtanje dijagrama.

Prije samog pokretanja programa potrebno je odrediti oblik okvirnog nosača u ovom radu je prikazan primjer okvirnog nosača za koji će se vršiti proračun u *Python* programu pomoću implementirane *FEM* analize. Na slici 5.1 je prikaz okvirnog nosača s pripadajućim čvorovima i elementima gdje je element na koji djeluje sila zbog jednostavnijeg implementiranja programa podijeljen na dva elementa i fiktivni čvor je postavljen na točnu poziciju na koju djeluje sila.



Slika 5.1 Prikaz okvirnog nosača

Na slici 5.2 možemo vidjeti poziv funkcije *Uvoz_podataka* koja kao ulazni argument ima *excel* datoteku (*filename*). Funkcija dodjeljuje svakoj navedenoj varijabli odgovarajuće vrijednosti iz *excel* listova.

```
749. if __name__ == "__main__":
750. # UČITAVANJE ULAZNIH PODATAKA
751. Elementi, Cvorovi, Rubni_uvjeti_pomak, Matrica_sila, q_el, A_el, E_el, Iy_el, =
Uvoz_podataka(filename)
```

Slika 5.2 Unos podataka

U tablicama 5.1 i 5.2 je prikaz unosa podataka koji se radi u *excel* programu za elemente i čvove s pripadajući podaci.

Tablica 5.1 Prikaz unosa podataka za čvorove

Čvor broj	Cvor X (mm)	Cvor Y (mm)	Rubni uvjeti X	Rubni uvjeti Y	Rubni uvjeti Z	Sila X (N)	Sila Y (N)	Moment Z (Nm)
0	0	0	1	1	1	0	0	0
1	0	2000	0	0	0	0	0	0
2	2000	2000	0	0	0	0	0	0
3	2000	0	0	1	0	0	0	0
4	3000	2000	0	0	0	0	-1000	0
5	4000	2000	0	0	0	0	0	0
6	4000	0	1	1	1	0	0	0

Tablica 5.2 Prikaz unosa podataka za elemente

Element broj	A, mm ²	Iy, mm ⁴	Element Cvor A	Element Cvor B	q_opt, N/m	Youngov modul, GPa
1	204	10132	0	1	0	210
2	204	10132	1	2	1000	210
3	204	10132	2	3	0	210
4	204	10132	2	4	0	210
5	204	10132	4	5	0	210
6	204	10132	5	6	0	210

Cilj izvođenja proračuna je rješavanje problema ravninskih okvirnih nosača uz pomoć metode konačnih elementa kako bi dobile reakcije u osloncima, pomaci u čvorovima i sile koje djeluju u elementima. Na slici 5.3 je prikaz poziva glavne funkcija za proračun koji se izvršava.

```

760. if __name__ == "__main__":
761.     # UČITAVANJE ULAZNIH PODATAKA
762.     Elementi, Cvorovi, Rubni_uvjeti_pomak, Matrica_sila, q_el, A_el, E_el, Iy_el, =
Uvoz_podataka(filename)
763.     # =====
764.     # ODREĐIVANJE REAKCIJA U OSLONCIMA TE POMAKA I ZAKRETA ČVOROVA
765.     # =====
766.     # ODREĐIVANJE BROJA KONAČNIH ELEMENATA
767.     br_elementata= np.array(Elementi).shape[0]
768.     # PRORAČUN DULJINE I KUTA NAGIBA ELEMENATA
769.     l_el, theta_el = Duljine_nagibi_elementata(br_elementata, Elementi)
770.     # GENERIRANJE LOKALNIH I GLOBALNIH MATRICA KRUTOSTI ELEMENATA
771.     K_glob_elem, K_lok_elem = Globalne_i_lokalne_matrice_krutosti_elementata()
772.     # KREIRANJE GLOBALNIH MATRICA KRUTOSTI ELEMENATA U FORMATU GLOBALNE MATRICE
773.     # KRUTOSTI OKVIRA
774.     K_glob_elem, Elementi_0, Elementi_1 = Dodaj_0_u_mat_krut_elementata()
775.     # GENERIRANJE GLOBALNE MATRICE KRUTOSTI SUSTAVA (OKVIRA)
776.     Mat_K = Generiraj_glob_mat_krut_sustava()
777.     # GENERIRANJE LOKALNOG VEKTORA EKVALENTNOG OPTEREĆENJA Qd
778.     Qd = Generiraj_lokalni_vektor_ekv_opt()
779.     # GENERIRANJE GLOBALNOG VEKTORA EKVALENTNOG OPTEREĆENJA ELEMENATA Qd
780.     Qd_glob_ELEMENTI = Generiraj_globalni_vektor_ekv_opt_ELEMENTATA()
781.     # GENERIRANJE GLOBALNOG VEKTORA EKVALIVETNOG OPTEREĆENJA OKVIRA
782.     Vekt_ekv_opterecenja_OKVIRA = Generiraj_globalni_vektor_ekv_opt_OKVIRA()
783.     # GENERIRANJE GLOBALNOG VEKTORA UKUPNOG OPTEREĆENJA OKVIRA
784.     #Vektor_opterecenja = Vekt_ekv_opterecenja_OKVIRA + Matrica_sila
785.     Vektor_opterecenja = Vekt_ekv_opterecenja_OKVIRA + Matrica_sila
786.     # ODREĐIVANJE POMAKA SLOBODNIH ČVOROVA PRIMJENOM RUBNIH UVJETA
787.     pomaci_slobodnih_cvorova, ru_reakcije, ru, Mat_K_RU = Pomaci_slobodnih_cvorova()
788.     # ODREĐIVANJE REAKCIJA U OSLONCIMA PRIMJENOM RUBNIH UVJETA
789.     reakcije_u_osloncima = Odredi_reakcije_u_osloncima()
790.     # GENERIRANJE VEKTORA SVIH POMAKA ČVOROVA, TE LISTA NAZIVA POMAKA I SILA
791.     Sile_nazivi, Pomaci_nazivi, Pomaci_svi_nazivi, Reakcije_nazivi, Pomaci_svi =
Pomaci_svih_cvorova()
792.     # POMACI PO SVIM ČVOROVIMA OKVIRA
793.     Pomaci_po_cvorovima = Pomaci_po_cvorovima_okvira()
794.     #ODREĐIVANJA SILE U ELEMNTIMA
795.     S = odredi_sile_momenta_u_elementima(Pomaci_svi, theta_el, Elementi, K_lok_elem, Qd)

```

Slika 5.3 Vršenje proračuna

Nakon izvršenja proračuna radi se vizualizacija dobivenih rezultata u obliku dijagrama. Predstavljeni su dijagrami deformiranog i nedeformiranog okvira, dijagram momenta, poprečnih i uzdužnih sila. Za dijagram momenta, poprečnih i uzdužnih sila je potrebno prvo obraditi podatka te se tada može vršit crtanje. Na slici 5.4 su prikazane funkcije obrade podatka pod imenima *izracunaj_Q*, *izracunaj_N* i *izracunaj_M*, a pod imenom *plot_digram* se crtaju dijagrami momenta uzdužnih i poprečnih sila.

```

796. #ODREĐIVAJE POPREČNIH SILA NA ELEMNTIMA
797. Q_el, Q_l = izracunaj_Q(l_el, S, q_el)
798. MQN = Q_el
799. L = Q_l
800. scale_factor=1000
801. Naslov='Q'
802. # Pozivanje funkcije za crtanje dijagrama
803. plot_diagrams(MQN, L, Cvorovi, Elementi, theta_el, scale_factor,Naslov)
804. #ODREĐIVANJE UZDUŽNIH SILA NA ELEMNTIMA
805. N_el, N_l = izracunaj_N(l_el, S)
806. MQN=N_el
807. L=N_l
808. scale_factor=2000
809. Naslov='N'
810. # Pozivanje funkcije za crtanje dijagrama
811. plot_diagrams(MQN, L, Cvorovi, Elementi, theta_el, scale_factor,Naslov)
812. #ODREĐIVANJE MOMENTA U ELEMNTIMA
813. M_el, l_M = izracunaj_M(l_el, S, q_el,theta_el)
814. MQN=M_el
815. L=l_M
816. scale_factor=1000
817. Naslov='Mz'
818. # Pozivanje funkcije za crtanje dijagrama
819. plot_diagrams(MQN, L, Cvorovi, Elementi, theta_el, scale_factor, Naslov)

```

Slika 5.4 Prikaz funkcija za pripremu podatka i pozivanje funkcije za crtanje dijagrama momenta uzdužnih i poprečnih sila

Na slici 5.5 prikazano je pozivanje funkcije za crtanje deformiranog i nedeformiranog okvira.

```

677. # Call the function to plot the frame
678. # Calculate the displacements of all nodes (Pomaci_svi) and store them in a variable
679.
680. # Call the function to plot the frame with displacements
681. # plot_frame(Elementi, Cvorovi, Pomaci_svi, scale_factor=500)
682. # CRTANJE DEFORMIRANOG OKVIRA
683. """ ***** DODANO ***** """
684. Crtanje_okvira()

```

Slika 5.5 Prikaz crtanja okvira

Program ispisuje krajnje rezultate na konzoli i u *excel* datoteci pod imenom *Rezultati_grede.xlsx* s listovima *Reakcije_u_osloncima*, *Pomaci_po_cvorovima* i *Sile_u_cvorovima*. Poziv funkcija prikazan je na slici 5.6. Pri radu s *excel* datotekom korištena je skripta *pandas*.

```
#ODREĐIVANJE TIPOVA PODTAKA KOJE JE POTREBNO ISPISATI(REAKCIJE U OSLONCIMA I POMACI PO CVOROCIMA
)
828.     df1 = pd.DataFrame({'Reakcije_u_osloncima': reakcije_u_osloncima}, index=row_labels)
829.     df2 = pd.DataFrame(Pomaci_svi, index=row_labels1, columns=['Pomaci_po_cvorovima'])
830.     df3 = pd.DataFrame(S, columns=["Fx", "Fy", "M", "Fx", "Fy", "M"])
831.
832.     #ISPIS PODTAKA U KONZOLI
833.     print('reakcije_u_osloncima:')
834.     for label, row in zip(row_labels, reakcije_u_osloncima):
835.         print(f'{label}: {row}')
836.
837.     print("\n", 'Pomaci_po_cvorovima:')
838.     for label, row in zip(row_labels1, Pomaci_svi):
839.         print(f'{label}: {row}')
840.
841.     # KREIRANJE EXCEL TABLICE U PYTHON DATOTECI SA PANDASOM
842.     output_file = 'Rezultati_okvirnog_nosaca.xlsx'
843.     writer = pd.ExcelWriter(output_file, engine='openpyxl')
844.
845.     # ISPISIVANJE POJEDINAČNIH PODATAKA U EXCEL
846.     df1.to_excel(writer, sheet_name='Reakcije_u_osloncima', index=True)
847.     df2.to_excel(writer, sheet_name='Pomaci_po_cvorovima', index=True)
848.     df3.to_excel(writer, sheet_name='Sile_u_cvorovima', index=True)
849.
850.     # SPREMANJE PODTAKAU U EXCEL TABLICU
851.     writer.close()
```

Slika 5.6 Ispis rezultata

5.2. Unos podataka

Funkcija *Uvoz_podataka* služi za učitavanje podataka iz Excel datoteke. Prvo se učitavaju dva lista iz Excel datoteke s nazivima "Elementi" i "Cvorovi". Nakon toga, podaci iz tih lista se spremaju u različite varijable. Na primjer, podaci iz stupaca "Element Cvor A" i "Element Cvor B" se spremaju u varijable *CvoroviA* i *CvoroviB*. Slično se radi za ostale podatke kao što su koordinate čvorova, rubni uvjeti, sile, momenti, itd.

Nakon što su podaci učitani, provjerava se ispravnost brojeva korištenjem funkcije *provjera_broja*. Zatim se provjerava je li svaki broj pozitivan korištenjem funkcije *provjera_jeli_broj_pozitivan*.

Na slikama 5.7 i 5.8 je prikazan opisani kôd.

```

54. def Uvoz_podataka(filename):
55.     """
56.     UČITAVANJE ULAZNIH PODATAKA IZ EXCEL DATOTEKE
57.     """
58.     elementi = pd.read_excel (filename, sheet_name='Elementi')
59.     cvorovi = pd.read_excel (filename, sheet_name='Cvorovi')
60.
61.     CvoroviA = elementi['Element Cvor A'].values[:]
62.     CvoroviB = elementi['Element Cvor B'].values[:]
63.     Cvorovi_X = cvorovi['Cvor X (mm)'].values[:]
64.     Cvorovi_Y = cvorovi['Cvor Y (mm)'].values[:]
65.     Rubni_uvjeti_X = cvorovi['Rubni uvjeti X'].values[:]
66.     Rubni_uvjeti_Y = cvorovi['Rubni uvjeti Y'].values[:]
67.     Rubni_uvjeti_Z = cvorovi['Rubni uvjeti Z'].values[:]
68.     Sile_X = cvorovi['Sila X (N)'].values[:]
69.     Sile_Y = cvorovi['Sila Y (N)'].values[:]
70.     Momenti_Z = cvorovi['Moment Z (Nm)'].values[:]
71.
72.     q_el = elementi['q_opt, N/m '].values[:]
73.     A_el = elementi['A, mm^2'].values[:]
74.     Iy_el = elementi['Iy, mm^4'].values[:]
75.     E_el = elementi['Youngov modul, GPa'].values[:]
76.
77.
78.
79.     provjera_broja(
80.         CvoroviA, CvoroviB, Cvorovi_X, Cvorovi_Y,
81.         Rubni_uvjeti_X, Rubni_uvjeti_Y, Rubni_uvjeti_Z,
82.         Sile_X, Sile_Y, Momenti_Z, q_el,
83.         A_el, Iy_el, E_el
84.     )
85.
86.
87.     provjera_je_li_broj_pozitivan( A_el, Iy_el, E_el)
88.
89.
90.     ro_el = np.array(ro_el) # kg/m3
91.     q_el = np.array(q_el) # N/m
92.     A_el = np.array(A_el)/1e6 # m2
93.     E_el = np.array(E_el)*1e9 # N/m2
94.     Iy_el = np.array(Iy_el)/1e12 # m4

```

Slika 5.7 Prikaz čitanja uvoz podataka, provjera podataka je li je broj i je li veći od nule

```
96.     Elementi = []
97.     for i in range(len(A_el)):
98.         Priv = []
99.         Priv.append(CvoroviA[i])
100.        Priv.append(CvoroviB[i])
101.        Elementi.append(Priv)
102.
103.        Cvorovi = []
104.        Rubni_uvjeti_pomak = []
105.        Matrica_sila = []
106.
107.        for i in range(len(Cvorovi_X)):
108.            Priv = []
109.            Priv.append(Cvorovi_X[i])
110.            Priv.append(Cvorovi_Y[i])
111.            Cvorovi.append(Priv)
112.            Rubni_uvjeti_pomak.append(Rubni_uvjeti_X[i])
113.            Rubni_uvjeti_pomak.append(Rubni_uvjeti_Y[i])
114.            Rubni_uvjeti_pomak.append(Rubni_uvjeti_Z[i])
115.            Matrica_sila.append(Sile_X[i])
116.            Matrica_sila.append(Sile_Y[i])
117.            Matrica_sila.append(Momenti_Z[i])
118.
119.        Elementi = np.array(Elementi)
120.        Cvorovi = np.array(Cvorovi)/1000
121.        Rubni_uvjeti_pomak = np.array(Rubni_uvjeti_pomak)
122.        Matrica_sila = np.array(Matrica_sila)
123.        q_el = np.array(q_el)
124.        Matrica_sila = np.array(Matrica_sila)
125.        return Elementi, Cvorovi, Rubni_uvjeti_pomak, Matrica_sila, q_el, A_el, E_el, Iy_el,
```

Slika 5.8 Prikaz povratak podataka iz funkcije

5.3. Provjera broja

Funkcija *provjera_broja* provjerava jesu li uneseni podaci brojevi. Ova provjera se vrši pomoću bloka *try-except*.

Unutar bloka *try*, petlja prolazi kroz liste podataka koje su proslijeđene funkciji. Za svaki element u listi se provjerava je li moguće pretvoriti ga u broj. Ako se za bilo koji element iz liste ne može izvršiti pretvorba u broj ili nije broj, generira se iznimka.

U bloku *except*, hvataju se iznimke tipova *ValueError* i *TypeError*, koje se javljaju ako pretvorba u broj nije moguća ili se uneseni podatak ne može smatrati brojem. Kada se uhvati iznimka, ispisuje se poruka koja ukazuje na to da unos u određenom polju nije broj. U poruci se koristi varijabla *name* koja sadrži naziv polja ili stupca u kojem se dogodila greška.

Na slici 5.9 je prikazan funkcija provjera broja.

```
17. def provjera_broja(*lists):
18.     """
19.         PROVJERA ULAZNIH PODATAKA JESU LI BROJ
20.     """
21.     list_names = [
22.         'Element Cvor A', 'Element Cvor B', 'Cvor X (mm)', 'Cvor Y (mm)',
23.         'Rubni uvjeti X', 'Rubni uvjeti_Y', 'Rubni uvjeti_Z',
24.         'Sila X (N)', 'Sila Y (N)', 'Moment Z (Nm)', 'q_opt, N/m ',
25.         'A, mm^2', 'Iy, mm^4', 'Youngov modul, GPa',
26.     ]
27.
28.     for i, (lst, name) in enumerate(zip(lists, list_names), 1):
29.         try:
30.             all(isinstance(float(num), (int, float)) for num in lst)
31.
32.         except (ValueError, TypeError):
33.             print(f"Unos u polju {name}. nije broj")
```

Slika 5.9 Provjera je li element broj

5.3.1. Provjera je li broj pozitivan

Funkcija `provjera_je_li_broj_pozitivan` provjerava jesu li uneseni podaci pozitivni brojevi. Također koristi blok `try-except` mehanizam za hvatanje iznimki. U bloku `try`, petlja prolazi kroz liste podataka koje su prosljeđene funkciji. Za svaki element u listi se provjerava je li broj veći od 0 koristeći izraz `float(num) > 0`. Ako za bilo koji element iz liste izraz nije zadovoljen, generira se iznimka `ValueError`.

U bloku `except`, hvataju se iznimke tipova `ValueError` i `TypeError`, koje se javljaju ako pretvorba u broj nije moguća ili se uneseni podatak ne može smatrati brojem. Kada se uhvati iznimka, ispisuje se poruka koja ukazuje na to da u listi određenog naziva broj mora biti veći od 0. U poruci se koristi varijabla `name` koja sadrži naziv liste ili skupa podataka u kojem se dogodila greška.

Na slici 5.10 prikaza je kôd.

```
36. def provjera_je_li_broj_pozitivan(*lists):
37.     """
38.     PROVJERA ULAZNIH PODATAKA JESU LI MANJI OD NULE
39.     """
40.     list_names = [
41.         'A, mm^2', 'Iy, mm^4', 'Youngov modul, GPa'
42.     ]
43.
44.     for i, (lst, name) in enumerate(zip(lists, list_names), 1):
45.         try:
46.             if not all(float(num) > 0 for num in lst):
47.                 raise ValueError
48.
49.         except (ValueError, TypeError):
50.             print(f"U listi {name} broj mora biti veći od 0.")
```

Slika 5.10 Prikaz provjerili je li broj pozitivama

5.4. Duljina i nagiba elementa

Ovaj program računa duljinu i kut nagiba elemenata. Ulazni parametri su broj elemenata i lista elemenata. Program prolazi kroz svaki element i izračunava duljinu temeljem koordinata njegovih čvorova. Također, izračunava kut nagiba elementa na temelju razlike koordinata čvorova. Rezultati se vraćaju kao liste duljina i kutova nagiba elemenata. Na slici 5.11 je prikazan kôd za određivanje duljine i nagiba elementa .

```
def Duljine_nagibi_elementa(br_elementa, Elementi):
129.     """
130.     Proračun duljine i kuta nagiba elementa
131.     """
132.     l_el = []
133.     theta_el = []
134.     for i in range(br_elementa):
135.         x1 = Cvorovi[Elementi[i][0]][0]
136.         y1 = Cvorovi[Elementi[i][0]][1]
137.         x2 = Cvorovi[Elementi[i][1]][0]
138.         y2 = Cvorovi[Elementi[i][1]][1]
139.         l_el.append(np.sqrt(abs(x1-x2)**2+abs(y1-y2)**2))
140.         if (x2-x1) == 0 and (y2-y1)>0:
141.             theta_el.append(np.pi/2)
142.         elif (x2-x1) == 0 and (y2-y1)<0:
143.             theta_el.append(3*np.pi/2)
144.         else:
145.             theta_el.append(np.arctan((y2-y1)/(x2-x1)))
146.
147.     theta_el = np.array(theta_el)*180/np.pi
148.     l_el = list(l_el)
149.     return l_el, theta_el
```

Slika 5.11 Izračun duljine i nagiba elementa

5.5. Generiranje lokalnih matrica krutosti elemenata

Ova funkcija generira lokalne i globalne matrice krutosti elemenata. Prvo se inicijaliziraju prazne liste K_{lok_elem} i K_{glob_elem} . Zatim se prolazi kroz sve elemente i za svaki element se generira lokalna matrica krutosti K_{lok_el} koristeći parametre kao što su površina A_{el} , modul elastičnosti E_{el} , duljina l_{el} i inercijski moment Iy_{el} . Lokalna matrica krutosti se dodaje u listu K_{lok_elem} . Nakon toga, lokalna matrica krutosti se transformira u globalnu matricu krutosti K_{glob_el} pomoću kuta nagiba $theta_{el}$ koristeći funkciju Lok_u_glob . Globalna matrica krutosti se dodaje u listu K_{glob_elem} . Na kraju, funkcija vraća liste K_{glob_elem} i K_{lok_elem} koje sadrže generirane matrice. Na slici 5.12 je prikazano generiranje lokalnih matrica krutosti elemenata.

```
152. def Globalne_i_lokalne_matrice_krutosti_elementa():
153.     # GENERIRANJE LOKALNIH I GLOBALNIH MATRICA KRUTOSTI ELEMENATA
154.     K_lok_elem = []
155.     K_glob_elem = []
156.     for i in range(len(l_el)):
157.         K_lok_el = FEM.Lok_MK_el(A_el[i], E_el[i], l_el[i], Iy_el[i])
158.         K_lok_elem.append(K_lok_el)
159.         K_glob_el = FEM.Lok_u_glob(K_lok_elem[i], theta_el[i])
160.         K_glob_elem.append(K_glob_el)
161.     return K_glob_elem, K_lok_elem
```

Slika 5.12 Generiranje lokalnih matrica krutosti elemenata

5.6. Kreiranje globalnih matrica krutosti elemenata u formatu globalne matrice krutosti okvira

Ova funkcija kreira globalne matrice krutosti elemenata u formatu globalne matrice krutosti okvira. Dodaje "0" stupce i retke u globalne matrice krutosti elemenata na mjestima čvorova koji ne pripadaju elementu. Također generira globalne matrice krutosti elemenata s dodanim "0". Funkcija vraća globalnu matricu krutosti elemenata, indekse "0" redaka i stupaca te indekse redaka i stupaca koji nisu "0".

Na slici 5.13 je prikazan je kreiranje globalnih matrica krutosti elementa u formatu globalnih matrica krutosti okvira

```

165. def Dodaj_0_u_mat_krut_elemenata():
166.     """
167.     KREIRANJE GLOBALNIH MATRICA KRUTOSTI ELEMENATA U FORMATU GLOBALNE MATRICE
168.     KRUTOSTI OKVIRA
169.     DODAVANJE "0" STUPACA I REDAKA U GLOBALNIM MATRICAMA KRUTOSTI ELEMENATA NA
170.     MJESTIMA ČVOROVA KOJI NE PRIPADAJU ELEMENTU
171.     GENERIRANJE GLOBALNE MATRICA KRUTOSTI ELEMENATA S DODANIM "0"
172.     Returns
173.     K_glob_elem : TYPE
174.     Globalna matrica krutosti elemenata u formatu globalne matrice krutosti
175.     sustava
176.     """
177.     # PRONALAZENJE INDEKSA "0" STUPACA I REDAKA U GLOBALNIM MATRICAMA KRUTOSTI
178.     # ELEMENATA
179.
180.     st_slob = np.linspace(0, len(Cvorovi)*3-1, num=len(Cvorovi)*3, dtype=int)
181.     # Matrica_veza = Generator_matrice_veza(Elementi, br_elemenata)
182.     Elementi_0 = []
183.     Elementi_1 = []
184.     for i in range(len(l_el)):
185.         a = st_slob
186.         pomak_cv1_1 = Elementi[i][0]*3 ; pomak_cv1_2 = Elementi[i][0]*3+1
187.         pomak_cv1_3 = Elementi[i][0]*3+2 ; pomak_cv2_1 = Elementi[i][1]*3
188.         pomak_cv2_2 = Elementi[i][1]*3+1 ; pomak_cv2_3 = Elementi[i][1]*3+2
189.         a = a[a!=pomak_cv1_1]; a = a[a!=pomak_cv1_2]; a = a[a!=pomak_cv1_3]
190.         a = a[a!=pomak_cv2_1]; a = a[a!=pomak_cv2_2]; a = a[a!=pomak_cv2_3]
191.
192.         b = st_slob
193.         for i in a:
194.             b = b[b != i]
195.             Elementi_0.append(a+1) # Indeksi pozicija "0" redaka i stupaca
196.             Elementi_1.append(b) # Indeksi redaka i stupaca "<>0"
197.     Elementi_0 = np.array(Elementi_0)
198.     Elementi_1 = np.array(Elementi_1)
199.     # GENERIRANJE GLOBALNE MATRICA KRUTOSTI ELEMENATA S DODANIM "0"
200.     for i in range(len(l_el)):
201.         K_glob_elem[i] = FEM.Mat_krut_elem_sustav(K_glob_elem[i], Elementi_0[i])
202.
203.     return K_glob_elem, Elementi_0, Elementi_1
204.

```

Slika 5.13 Kreiranje globalnih matrica krutosti elemenata u formatu globalne matrice krutosti okvira

5.7. Generiranje globalne matrice krutosti sustava

Ova funkcija generira globalnu matricu krutosti sustava (okvira). Prvo se inicijalizira nula-matrica Mat_K dimenzija koja odgovara broju rubnih uvjeta pomaka. Zatim se prolazi kroz sve elemente i za svaki element se dodaje njegova globalna matrica krutosti $K_{glob_elem}[i]$ u matricu Mat_K pomoću operacije zbrajanja. Na kraju, funkcija vraća generiranu globalnu matricu krutosti sustava Mat_K . na slici je prikazan kôd funkcije generiranje matrice krutosti sustava.

```

207. def Generiraj_glob_mat_krut_sustava():
208.     """
209.     GENERIRANJE GLOBALNE MATRICE KRUTOSTI SUSTAVA (OKVIRA)
210.
211.     Returns
212.     -----
213.     Mat_K : TYPE numpy ndarrays
214.         Globalna matrica krutosti sustava
215.     """
216.     Mat_K = np.zeros((len(Rubni_uvjeti_pomak), len(Rubni_uvjeti_pomak)))
217.
218.     for i in range(len(l_el)):
219.         Mat_K += K_glob_elem[i] # Globalna matrica krutosti sustava
220.     return Mat_K

```

Slika 5.14 Generiranje globalnih matrica krutosti elemenata

5.8. Generiranje lokalnih vektora ekvivalentnog opterećenja

Ova funkcija generira lokalni vektor ekvivalentnog opterećenja Qd za svaki element. Prvo se inicijaliziraju prazne liste Fd_x , Fd_y , Md i Fd . Zatim se prolazi kroz sve elemente i za svaki element se inicijaliziraju privremene liste Fd_x_priv , Fd_y_priv i Md_priv . Nakon toga, te privremene liste se dodaju u liste Fd_x , Fd_y i Md . U sljedećem koraku se računaju vrijednosti ekvivalentnih sila Fd_y i momenata Md za svaki element na temelju ulaznih podataka kao što su sila Q_el i duljina l_el .

Za svaki element se inicijalizira privremena lista Qd_priv , koja sadrži vrijednosti opterećenja u x i y smjeru te momente. Ta privremena lista se dodaje u vektor Qd za svaki element. Na kraju se vektor Qd vraća kao rezultat funkcije.

Na slici 5.15 je prikaza generiranje lokalnih vektora ekvivalentnog opterećenja.

```
223. def Generiraj_lokalni_vektor_ekv_opt():
224.     """
225.     GENERIRANJE LOKALNOG VEKTORA
226.     EKVIVALENTNOG OPTEREĆENJA Qd
227.     Returns
228.     -----
229.     Qd : TYPE numpy ndarrays
230.         DESCRIPTION.
231.     """
232.     Fd_x=[]
233.     Fd_y=[]
234.     Md=[]
235.     Qd=[]
236.     for i in range(len(l_el)):
237.         Fd_x_priv=[]
238.         Fd_y_priv=[]
239.         Md_priv=[]
240.         for j in range(2):
241.             Fd_x_priv.append(0)
242.             Fd_y_priv.append(0)
243.             Md_priv.append(0)
244.             Fd_x.append(Fd_x_priv)
245.             Fd_y.append(Fd_y_priv)
246.             Md.append(Md_priv)
247.         for i in range(len(l_el)):
248.             Fd_y[i][0]=-(q_el[i] * l_el[i] / 2)
249.             Fd_y[i][1]=-(q_el[i] * l_el[i] / 2)
250.             Md[i][0]=-(q_el[i] * l_el[i] ** 2 / 12)
251.             Md[i][1]=(q_el[i] * l_el[i] ** 2 / 12)
252.         # Vektor ekvivalentnih sila
253.         Fd_y = np.around(np.array(Fd_y), decimals=6)
254.         # Vektor ekvivalentnih momenata
255.         Md = np.around(np.array(Md), decimals=6)
256.         # VEKTOR EKVIVALENTNOG OPTEREĆENJA Qd
257.         # (zamjenjuje kontinuirano opterećenje)
258.         for i in range(len(l_el)):
259.             Qd_priv = []
260.             Qd_priv.append(0)
261.             Qd_priv.append(Fd_y[i][0])
262.             Qd_priv.append(Md[i][0])
263.             Qd_priv.append(0)
264.             Qd_priv.append(Fd_y[i][1])
265.             Qd_priv.append(Md[i][1])
266.             Qd.append(Qd_priv)
267.         Qd = np.array(Qd)
268.     return Qd
```

Slika 5.15 Generiranje globalnih vektora ekvivalentnog opterećenja

5.9. Generiranje globalni vektor ekvivalentnog opterećenja okvira.-popravit

Na početku se inicijaliziraju prazne liste *Vekt_konc_o_ekvivalentnog_OKVIRA* i *Vekt_ekv_opt*. Zatim se prolazi kroz svaki čvor pomoću petlje for. Unutar petlje se inicijalizira privremena prazna lista *priv* s tri elementa za svaki čvor, koja se dodaje u listu *Vekt_ekc_opt*. Nakon toga, prolazi se kroz svaki element okvira pomoću petlje for, ako je čvor *i* sadržan u elementu, provjerava se je li taj čvor prvi čvor u elementu, ako jest, dodaju se vrijednosti ekvivalentnog opterećenja *Qd_glob_ELEMENTI* u prva tri elementa vektora *Vekt_ekv_opt[i]*. Ako čvor *i* nije prvi čvor u elementu, dodaju se vrijednosti *Qd_glob_ELEMENTI* u druga tri elementa vektora *Vekt_ekv_opt[i]*. Nakon toga, prolazi se kroz svaki čvor *i* i svaku komponentu vektora *Vekt_ekv_opt* i dodaju se u listu *Vekt_ekv_opterecenja_OKVIRA*.

Na slici 5.16 je prikazan funkcija generiranje globalni vektor ekvivalentnog opterećenja okvira.

```
def Generiraj_globalni_vektor_ekv_opt_OKVIRA():
295.     """
296.     GENERIRANJE GLOBALNOG VEKTORA EKVIVALENTNOG
297.     OPTEREĆENJA OKVIRA
298.     Returns
299.     -----
300.     Vekt_ekv_opterecenja_OKVIRA :
301.     TYPE numpy ndarray
302.
303.     """
304.     # Vektor ekvivalentnog opterećenja OKVIRA
305.     # (za kontinuirana opterećenja)
306.     Vekt_ekv_opterecenja_OKVIRA = []
307.     Vekt_ekv_opt = []
308.
309.     for i in range(len(Cvorovi)):
310.         priv = []
311.         for j in range(3):
312.             priv.append(0)
313.         Vekt_ekv_opt.append(priv)
314.     for i in range(len(Cvorovi)):
315.         j = -1
316.     for i in range(len(Cvorovi)):
317.         j = -1
318.         for element in Elementi:
319.             j += 1
320.             if i in element:
321.                 if element[0] == i:
322.                     for k in range(3):
323.                         Vekt_ekv_opt[i][k] += Qd_glob_ELEMENTI[j][k]
324.                 else:
325.                     for k in range(3):
326.                         Vekt_ekv_opt[i][k] += Qd_glob_ELEMENTI[j][k+3]
327.
328.     for i in range(len(Cvorovi)):
329.         for j in range(3):
330.             Vekt_ekv_opterecenja_OKVIRA.append(Vekt_ekv_opt[i][j])
331.     Vekt_ekv_opterecenja_OKVIRA = np.array(Vekt_ekv_opterecenja_OKVIRA)
332.     return Vekt_ekv_opterecenja_OKVIRA
```

Slika 5.16 generiranje globalni vektor ekvivalentnog opterećenja okvira.

5.10. Odredi reakcije u osloncima

Ova funkcija određuje reakcije u osloncima primjenom rubnih uvjeta. Na početku se inicijaliziraju varijable n , ru_redci i ru_stupci . Varijabla n se koristi kao brojač za identifikaciju redaka i stupaca u matrici. Liste ru_redci i ru_stupci služe za pohranjivanje redaka i stupaca s rubnim uvjetima pomaka.

Zatim se prolazi kroz rubne uvjete pomaka $Rubni_uvjeti_pomak$ pomoću petlje `for` ako je rubni uvjet pomaka jednak 0, dodaje se odgovarajući redak (n) u listu ru_redci ako je rubni uvjet pomaka jednak 1, dodaje se odgovarajući stupac (n) u listu ru_stupci .

Nakon toga se koristi funkcija $Rubni_uvjeti_matrica_krutosti$ iz modula FEM kako bi se dobila matrica krutosti Mat_Krut s primijenjenim rubnim uvjetima. Ova funkcija uzima globalnu matricu krutosti Mat_K i liste ru_redci i ru_stupci kako bi generirala novu matricu krutosti s primijenjenim rubnim uvjetima.

Također se koristi funkcija $Rubni_uvjeti_vektor$ iz modula FEM kako bi se dobila vektor opterećenja s primijenjenim rubnim uvjetima. Ova funkcija uzima $Vektor_opterecenja$ i listu $ru_reakcije$ kako bi generirala novi vektor opterećenja s primijenjenim rubnim uvjetima.

Matrično množenje između matrice krutosti Mat_Krut , vektora pomaka slobodnih čvorova $pomaci_slobodnih_cvorova$ i oduzimanje vektora opterećenja rezultira vektorom $reakcije_u_osloncima$. Funkcija vraća vektor reakcija u osloncima. Na slici je prikazan funkcija odredi reakcije u osloncima.

Na slici 5.17 je prikazana funkcija određivanje reakcije u osloncima

```

377. def Odredi_reakcije_u_osloncima():
378.     """
379.     ODREĐIVANJE REAKCIJA U OSLONCIMA PRIMJENOM RUBNIH UVJETA
380.     Returns
381.     -----
382.     reakcije_u_osloncima : TYPE numpy ndarray
383.     """
384.     n = 0
385.     ru_redci = []
386.     ru_stupci = []
387.     for i in Rubni_uvjeti_pomak:
388.         n += 1
389.         if i == 0:
390.             ru_redci.append(n)
391.         if i == 1:
392.             ru_stupci.append(n)
393.
394.     Mat_Krut = FEM.Rubni_uvjeti_matrica_krutosti(Mat_K, ru_redci, ru_stupci)
395.
396.     Opterecenja = FEM.Rubni_uvjeti_vektor(Vektor_opterecenja, ru_reakcije)
397.
398.     reakcije_u_osloncima = np.dot(Mat_Krut, pomaci_slobodnih_cvorova) - Opterecenja
399.
400.
401.     return reakcije_u_osloncima

```

Slika 5.17 Odredi reakcije u osloncima

5.11. Pomaci po čvorovima okvira

Ova funkcija računa pomake po svim čvorovima okvira u matričnom zapisu. Na početku se inicijalizira prazna lista *Pomaci_po_cvorovima*. Zatim se koristi petlja for za prolazak kroz sve čvorove u listi *Cvorovi*. Za svaki čvor, stvara se prazna lista *Pomak_priv* koja će sadržavati pomake po pojedinim osima (x, y, z). Unutar unutarnje petlje for, za svaku os (x, y, z), dodaje se *Pomaci_svi[i*3+j]* na listu *Pomak_priv*. Indeksiranje *Pomaci_svi[i*3+j]* se koristi za dobivanje odgovarajućeg pomaka iz vektora svih pomaka *Pomaci_svi*. Nakon toga lista *Pomak_priv* se dodaje u listu *Pomaci_po_cvorovima*, čime se generira lista pomaka po čvorovima u matričnom zapisu i vraća kao rezultat funkcije.

Funkcija omogućuje dobivanje matrice pomaka po čvorovima okvira, gdje svaki redak predstavlja pomake po osima (x, y, z) za određeni čvor.

Na slici 5.18 je prikazan funkcija pomaci po čvorovima okvira.

```
433. def Pomaci_po_cvorovima_okvira():
434.     """
435.     POMACI PO SVIM ČVOROVIMA OKVIRA (matrični zapis)
436.
437.     Returns
438.     -----
439.     Pomaci_po_cvorovima : TYPE numpy ndarray
440.
441.     """
442.
443.     Pomaci_po_cvorovima = []
444.     for i in range(len(Cvorovi)):
445.         Pomak_priv = []
446.         for j in range(3):
447.             Pomak_priv.append(Pomaci_svi[i*3+j])
448.         Pomaci_po_cvorovima.append(Pomak_priv)
449.     Pomaci_po_cvorovima = np.array(Pomaci_po_cvorovima)
450.     return Pomaci_po_cvorovima
```

Slika 5.18 Pomaci po čvorovima okvira

5.12. Određivanje sila u elementima

Za funkciju određivanje sila elementima prvo se inicijalizira prazna lista *sile_u_elementima_1*, koja će sadržavati sile za svaki element konstrukcije. Zatim, petlja prolazi kroz sve elemente definirane u matrici *Elementi*. Za svaki element dohvaćaju se indeksi čvorova i pohranjuju u varijable *idx1* i *idx2*. Inicijaliziraju se liste *Pomaci_elementa*, *Cvor_1*, i *Cvor_2* za spremanje pomaka čvorova elementa. Izračunavaju se sinus i kosinus kuta elementa *theta_el*, te se definira matrica transformacije *T*. Unutarnja petlja dohvaća pomake za svaki čvor iz liste *Pomaci_svi* i pohranjuje ih u odgovarajuće liste, koje se zatim dodaju u listu *Pomaci_elementa*.

Pomaci se transformiraju iz globalnih u lokalne koordinate pomoću matrice *T* i pohranjuju u varijablu *f_lokalno*. Sile u lokalnim koordinatama računaju se množenjem lokalnih pomaka s lokalnom matricom krutosti *K_lok_elem* za trenutni element i pohranjuju u varijablu *sila*. Ta varijabla se dodaje u listu *sile_u_elementima_1*. Sve sile se spajaju u jednu *numpy* matricu koristeći *np.vstack*. Konačno, od tih sila oduzimaju se vanjska opterećenja *Qd* kako bi se dobile konačne sile u elementima *S*, koje se vraćaju kao rezultat funkcije. Ovaj proces omogućava izračunavanje sila koje djeluju na elemente konstrukcije, uzimajući u obzir pomake čvorova i lokalne matrice krutosti.

Na slici 5.19 je prikazana funkcija sile u elementima.

```
483. #Funkcija za dobivanje sile u elementima
484. def odredi_sile_momenta_u_elementima(Pomaci_svi, theta_el, Elementi, K_lok_elem, Qd):
485.     """
486.     ODREĐIVANJE SILA I MOMENTA U ČVOROVIMA ELEMENATA
487.
488.     Returns
489.     -----
490.     S : TYPE numpy ndarray
491.         sile u elementima
492.     """
493.
494.
495.     sile_u_elementima_1 = []
496.     # Petlja kroz svaki element i računanje sila
497.     for i in range(len(Elementi)):
498.         # Dohvaćanje lokalne matrice krutosti za ovaj element
499.
500.         idx1 = int(Elementi[i, 0])
501.         idx2 = int(Elementi[i, 1])
502.         Pomaci_elementa = []
503.         Cvor_1 = []
504.         Cvor_2 = []
505.
506.         sin = np.sin(np.radians(theta_el[i]))
507.         cos = np.cos(np.radians(theta_el[i]))
508.         T = [[cos, sin, 0, 0, 0, 0],
509.             [-sin, cos, 0, 0, 0, 0],
510.             [0, 0, 1, 0, 0, 0],
511.             [0, 0, 0, cos, sin, 0],
512.             [0, 0, 0, -sin, cos, 0],
513.             [0, 0, 0, 0, 0, 1]]
514.
515.         for a in range(3):
516.             Cvor_1.append(Pomaci_svi[idx1*3+a])
517.             Cvor_2.append(Pomaci_svi[idx2*3+a])
518.         Pomaci_elementa.extend(Cvor_1)
519.         Pomaci_elementa.extend(Cvor_2)
520.
521.
522.         # Računanje sila u lokalnim koordinatama
523.         f_lokalno = np.dot(Pomaci_elementa, T)
524.         sila=np.dot(f_lokalno, K_lok_elem[i])
525.         sila=sila.reshape(1,6)
526.         sile_u_elementima_1.append(sila)
527.
528.     sile_u_elementima_1=np.vstack(sile_u_elementima_1)
529.
530.     S= sile_u_elementima_1 - Qd
531.
532.     return S
```

Slika 5.19 Sile u elementima

5.13. Crtanje dijagrama deformirane i ne deformirane okvirnog nosača

Ova funkcija priprema podatke za crtanje deformiranog okvira. Na početku se postavlja faktor skaliranja *scale_factor* koji određuje koliko će se pomaci u odnosu na stvarne koordinate povećati radi bolje vidljivosti na crtežu. Zatim se inicijalizira prazna lista *mat_elementi* koja će sadržavati informacije o elementima okvira potrebnim za crtanje. Unutar petlje for, prolazi se kroz sve elemente u listi *Elementi*. Za svaki element, stvara se prazna lista *mat_elementi_priv_2* koja će sadržavati informacije o čvorovima tog elementa. Unutar unutarnje petlje for, za svaki čvor ($j = 0, 1$) u elementu, stvara se prazna lista *mat_elementi_priv* koja će sadržavati informacije o koordinatama čvora i deformacijama:

- Prvo se dodaju koordinate čvora iz liste *Cvorovi* na pozicije 0 i 1 u *mat_elementi_priv*.
- Zatim se izračunavaju nove koordinate čvora dodavanjem odgovarajućeg pomaka iz *Pomaci_po_cvorovima* na osi x i y, pomnoženih sa *scale_factor*.
- Pomak čvora na osi z se također dodaje na poziciju 2 u *mat_elementi_priv*.

Nakon toga, *mat_elementi_priv* se dodaje u *mat_elementi_priv_2*. Kada su prikupljeni svi podaci za pojedini element, *mat_elementi_priv_2* se dodaje u *mat_elementi*. Nakon završetka petlje, poziva se funkcija *Crtanje_deformiranog_okvira*. *Crtaj_okvir()* s *mat_elementi* kao argumentom. Ta funkcija će izvršiti crtanje deformiranog okvira na temelju prikupljenih podataka.

Na slici 5.20 je prikazan funkcija crtanje dijagram.

```
454. def Crtanje_okvira():
455.     """
456.     MODUL ZA PRIPREMU PODATAKA ZA CRTANJE DEFORMIRANG OKVIRA
457.
458.     Returns
459.     -----
460.     None.
461.
462.     """
463.     scale_factor = 10
464.     mat_elementi = []
465.     for i in range(len(Elementi)):
466.         mat_elementi_priv_2 = []
467.         for j in range(2):
468.             mat_elementi_priv = []
469.             mat_elementi_priv.append(Cvorovi[Elementi[i][j]][0])
470.             mat_elementi_priv.append(Cvorovi[Elementi[i][j]][1])
471.             a = mat_elementi_priv[0]
472.             b = mat_elementi_priv[1]
473.             mat_elementi_priv.append(a + Pomaci_po_cvorovima[Elementi[i][j]][0] *
scale_factor)
474.             mat_elementi_priv.append(b + Pomaci_po_cvorovima[Elementi[i][j]][1] *
scale_factor)
475.             mat_elementi_priv.append(Pomaci_po_cvorovima[Elementi[i][j]][2] * scale_factor)
476.             mat_elementi_priv_2.append(mat_elementi_priv)
477.         mat_elementi.append(mat_elementi_priv_2)
478.
479.     Crtanje_deformiranog_okvira.Crtaj_okvir(mat_elementi)
```

Slika 5.20 Crtanje dijagram

5.14. Crtanje dijagrama momenta, poprečnih i uzdužnih sila

Funkcija *plot_diagrams* omogućava grafički prikaz momenata, uzdužnih i poprečnih sila koji djeluju na svaki element konstrukcije. Koristeći *matplotlib* biblioteku, ova funkcija prima podatke izračunate iz prethodnih koraka analize, kao što su vrijednosti momenata, duljine segmenata, koordinate čvorova i kutovi elemenata.

Kroz nekoliko koraka, funkcija prvo inicijalizira prazne liste za pohranu koordinata (*mat_X*, *mat_Y*, *A_X*, *A_Y*), zatim prolazi kroz svaki element konstrukcije kako bi definirala početne točke i izračunala koordinate točaka potrebnih za crtanje dijagrama. Nakon toga, funkcija kreira novi dijagram koristeći *matplotlib*, gdje crta linije koje predstavljaju elemente i dijagrame sila. Osim toga, dodaje oznake za vrijednosti sila na dijagramu kako bi korisnicima olakšala interpretaciju rezultata.

Na slici 5.21 je prikazana priprema podataka za crtanje dijagrama momenta, poprečnih i uzdužnih sila.


```

676. def plot_diagrams(MQN, L, Cvorovi, Elementi, theta_el, scale_factor, Naslov ):
677.     """
678.
679.     """
680.     # Inicijalizacija praznih lista
681.     mat_X = []
682.     mat_Y = []
683.     A_X = []
684.     A_Y = []
685.
686.     for i in range(len(Elementi)):
687.         a = Cvorovi[Elementi[i], 0]
688.         b = Cvorovi[Elementi[i], 1]
689.
690.         X_1 = []
691.         Y_1 = []
692.         A_X_1 = []
693.         A_Y_1 = []
694.
695.         ## Ubacivanje početnih točaka
696.
697.         A_X_1.append(a[0])
698.         A_Y_1.append(a[1])
699.
700.         for z in range(len(MQN[0])):
701.             l = L[i, z]
702.             M = MQN[i, z]
703.             sin_theta = np.sin(np.radians(theta_el[i]))
704.
705.             if sin_theta == 0:
706.                 X = a[0] + l
707.                 Y = a[1]
708.                 X_2 = a[0] + l
709.                 Y_2 = a[1] + M / scale_factor
710.             elif sin_theta > 0:
711.                 X = a[0]
712.                 Y = a[1] + l
713.                 X_2 = a[0] + M / scale_factor
714.                 Y_2 = a[1] + l
715.             else:
716.                 X = a[0]
717.                 Y = a[1] - l
718.                 X_2 = a[0] + M / scale_factor
719.                 Y_2 = a[1] - l
720.
721.
722.             # Dodavanje točaka u listu
723.             X_1.append(X)
724.             Y_1.append(Y)
725.             A_X_1.append(X_2)
726.             A_Y_1.append(Y_2)
727.
728.             # Dodavanje početnih i krajnjih točaka elementa na kraj dijagrama
729.             A_X_1.append(b[0])
730.             A_Y_1.append(b[1])
731.
732.             mat_X.append(X_1)
733.             mat_Y.append(Y_1)
734.             A_X.append(A_X_1)
735.             A_Y.append(A_Y_1)
736.             #IZRADA NOVOG DIJAGRAMA
737.             plt.figure(figsize=(10, 6))

```

Slika 5.21 Prikaz naredbi za pripreme podataka za crtanje dijagrama momenta, poprečnih i uzdužnih sila

Na slici 5.22 je prikazan izvršavanje crtanja navedenih dijagrama.

```
738.
739.     # Plotting
740.     for i in range(len(Elementi)):
741.         # Crtanje crte između točaka prvog grafa (okvirni nosač)
742.         plt.plot(mat_X[i], mat_Y[i], "k-", linewidth=2) # Debljina crte može se
kontrolirati pomoću parametra linewidth
743.         # Crtanje iscrtane linije drugog grafa (dijagram momenta)
744.         plt.plot(A_X[i], A_Y[i], "b--")
745.
746.         # Dodavanje crte između svih točaka
747.         for j in range(len(mat_X[i]) - 1):
748.             plt.plot([mat_X[i][j], A_X[i][j + 1]], [mat_Y[i][j], A_Y[i][j + 1]], "b-",
linewidth=1) # Crna linija
749.
750.             # Dodavanje oznaka za vrijednosti momenta na dijagram
751.             # plt.text(A_X[i][j + 1], A_Y[i][j + 1], f"MQN[i, j]:.2f)", color='blue',
fontsize=8, ha='center', va='bottom')
752.
753.
754.     plt.title(Naslov)
755.     plt.grid(True)
756.     plt.show()
757.
```

Slika 5.22 Prikaz naredbi za crtanje dijagrama momenta, uzdužnih i poprečnih sila

6. Testiranje kôda

U sljedećim koracima bit će prikazani ispravnost rada funkcija provjera je li broj i je li broj pozitivan te će biti dani rezultati proračunati pomoću izrađenog programa i javnog dostupnog alata structural-analyser.com koji su bazirani na metodi konačnih elemenata za okvirne nosače. Također će se prikazati izlazni podaci programskog kôda u *excel* tablici i ispis na konzoli.

6.1. Prikaz provjere ispravnosti funkcija je li broj i je li broj pozitivan

Za funkciju provjeru je li broj unesena je kriva vrijednost i prikazan je na slici 6.1 u tablici *okvir_primjer* na listu elementi.

Element broj	A, mm ²	Iy, mm ⁴	Element Cvor A	Element Cvor B	q, o
1	204	10132	0	1	1
2	204	10132	1	2	2
3	204	10132	2	3	3
4	204	10132	2	4	4
5	204	10132	4	5	5
6	204	10132	5	6	6

Slika 6.1 Prikaz krivog unosa za površine

Na sljedećoj slici 6.2 možemo vidjeti rezultat provjere koja korisniku ukazuje na pogrešku.

```
In [7]: runfile('C:/Users/Toma/Desktop/okvirni nosač_finalna verzija 2.py', wdir='C:/Users/Toma/Desktop')
Unos u polju A, mm^2. nije broj
U listi A, mm^2 broj mora biti veći od 0.
Traceback (most recent call last):
```

Slika 6.2 Ispis na konzoli koji upućuje na pogrešni unos broja u listi površina A

Nadalje je potrebno prikazati pogrešan unos negativne vrijednosti u polje za koje se očekuje unos samo pozitivnog broja. Na slici 6.3 je prikaz pogrešan unos podatka u listu I_y za podatak I_y, mm^4 broj vrijednosti -1.

Element broj	A, mm ²	Iy, mm ⁴	Element Cvor A	Element Cvor B	q, o
1	204	10132	0	1	1
2	204	10132	1	2	2
3	204	-1	2	3	3
4	204	10132	2	4	4
5	204	10132	4	5	5
6	204	10132	5	6	6

Slika 6.3 Prikaz negativnog unosa

Na sljedećoj slici 6.4 prikazan je ispis konzole za unos negativnog broja kod momenta tromosti I_y , mm^4 .

```
In [24]: runfile('C:/Users/Toma/Desktop/okvrni nosač_finalna verzija 2.py', wdir='C:/Users/Toma/Desktop')
U listi Iy, mm^4 broj mora biti veći od 0.
```

Slika 6.4 Ispis konzole za negativni broj

6.2. Ispis i usporedba rezultata

U ovom poglavlju prikazuje se dobivanje podataka iz Python kôda na tri različita načina: spremanje dijela podataka u Excel, ispisivanje proračunatih vrijednosti i prikazivanje dijagrama u konzoli. Te podatka ćemo usporediti sa izračunatim vrijednostima i dijagramima iz javnih dostupnih aplikacija *STRIAN* i *SW FEA 2D FRAME*. Na slici 6.5 je prikazan ispisi koji se nalazi u *python* konzoli [5][6].

```
reakcije_u_osloncima:
R1: 109.42503416519946
R2: 898.5293591076429
R3: -57.32438977567702
R11: 1718.5614800090905
R19: -109.42503416527376
R20: 382.90916088326674
R21: 88.56478622457801

Pomaci_po_cvorovima:
u1: 0
u2: 0
u3: 0
u4: 0.014687677891003142
u5: -4.1948149351430575e-05
u6: -0.04897321488684829
u7: 0.01468256934599076
u8: -8.02316283851116e-05
u9: 0.017141700189309304
u10: 0.048965969724609384
u11: 0
u12: 0.01714170018930932
u13: 0.014680015073484567
u14: -0.020248425120385306
u15: -0.009140677558265326
u16: 0.01467746080097836
u17: -1.7876244672421415e-05
u18: 0.01960807619489007
u19: 0
u20: 0
u21: 0
```

Slika 6.5 Prikaz ispisa podataka na konzoli

Iz Excel datoteke pod nazivom *Rezultati_okvirnog_nosaca* dobivene su tablice reakcije u osloncima, pomaci po cvorovima i sile u cvorovima. Ti rezultati bit će prikazani u nastavku.

U Tablici 6.1 prikazani su dobiveni podaci *Pomaci_po_cvorovima* koji se uspoređuju s podacima dobivenim iz aplikacije *SW FEA 2D FRAME*, a koji se nalaze u Tablici 6.2. Treba napomenuti da je dodan fiktivni čvor 4 koji je omogućio lakše računanje sile, no taj čvor treba zanemariti prilikom analize rezultata.

Tablica 6.1 Pomaci po čvorovima

	Pomaci_po_cvorovima (m)	Br.Čvora
u1	0	0
u2	0	
u3	0	
u4	0,014688	1
u5	-4,2E-05	
u6	-0,04897	
u7	0,014683	2
u8	-8E-05	
u9	0,017142	
u10	0,048966	3
u11	0	
u12	0,017142	
u13	0,01468	4
u14	-0,02025	
u15	-0,00914	
u16	0,014677	5
u17	-1,8E-05	
u18	0,019608	
u19	0	6
u20	0	
u21	0	

Tablica 6.2 Rezultati pomaka čvorova iz aplikacije *SW FEA 2D FRAME*

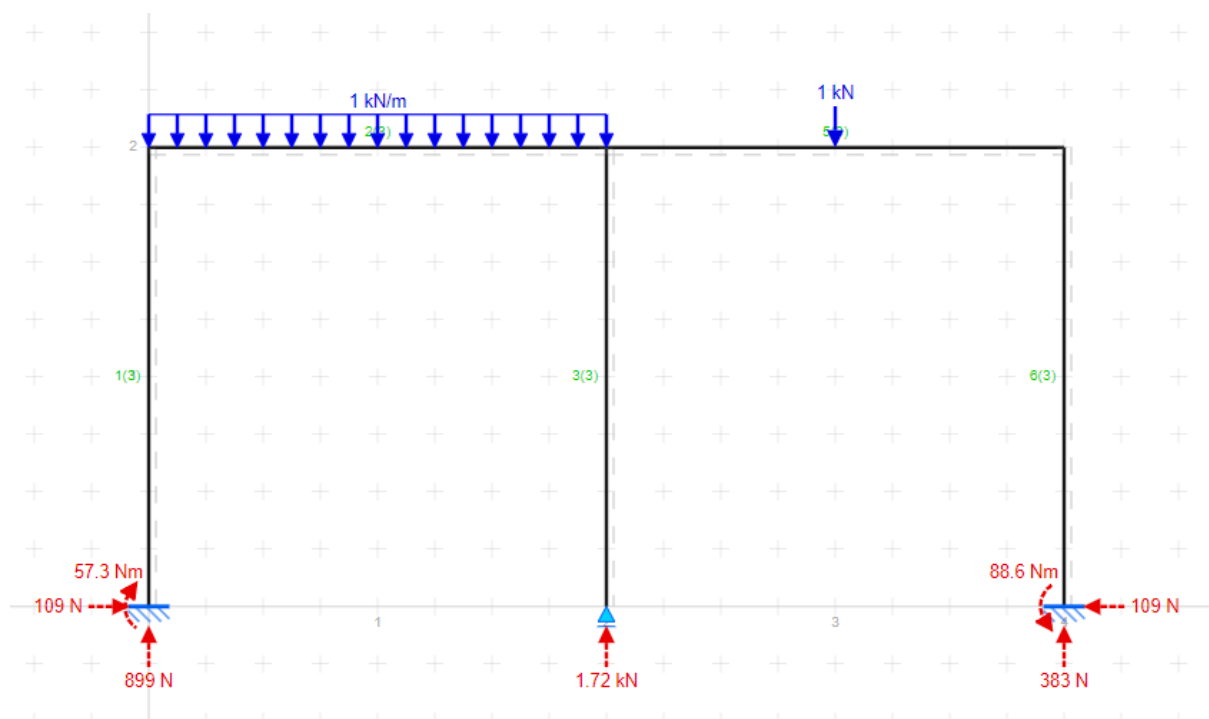
Node ID(BR.Čvora)	X (m)	Y (m)	dX (mm)	dY (mm)	Rot (rad)
0	0.000	0.000	0.000	0.000	0.000
1	0.000	2.000	14.688	-0.042	-0.049
2	2.000	2.000	14.683	-0.080	0.017
3	2.000	0.000	48.966	0.000	0.017
5	4.000	2.000	14.677	-0.018	0.020
6	4.000	0.000	0.000	0.000	0.000

U nastavku su prikazani dobiveni rezultati za reakcije u osloncima iz *Excel* dokumenta. Njih možemo usporediti s dobivenim podacima koji se nalaze na dijagramu na desnoj strani slike 6.6. Navedeni rezultati reakcija u osloncima prikazani su u Tablici 6.3.

Tablica 6.3 Rezultati reakcije u osloncima

	Reakcije_u_osloncima
R1	109,425
R2	898,5294
R3	-57,3244
R11	1718,561
R19	-109,425
R20	382,9092
R21	88,56479

Na slici 6.6 je prikaz okvirnog nosača sa reakcijama u osloncima nacrtan u aplikaciju *STRIAN*.



Slika 6.6 Prikaz okvirnog nosača sa reakcijama u osloncima

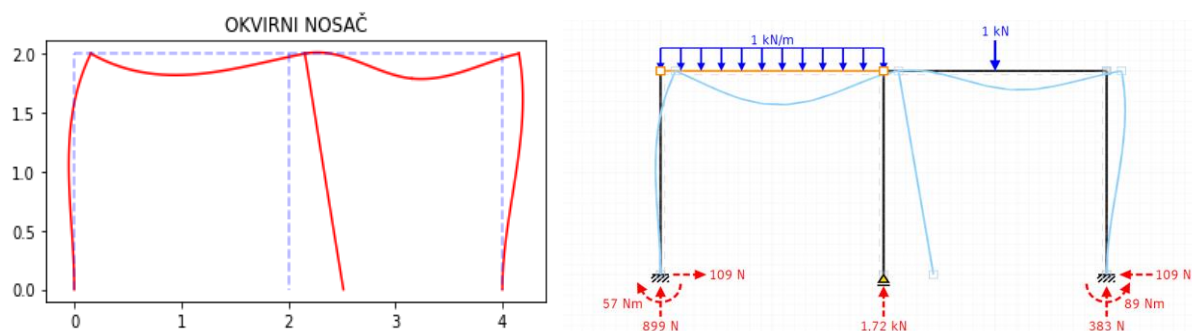
Dobiveni rezultati za sile u čvorovima uzeti su u *Excel* dokumentu. Navedeni rezultati sila u čvorovima prikazani su u Tablici 6.4.

Tablica 6.4 Prikaz rezultata sile u čvorovima

	N	Q	M	N	Q	M
0	898,5294	-109,425	-57,3244	-898,529	109,425	-161,526
1	109,425	898,5294	161,5257	-109,425	1101,471	-364,467
2	1718,561	7,11E-15	-1,4E-14	-1718,56	-7,1E-15	2,84E-14
3	109,425	617,0908	364,467	-109,425	-617,091	252,6239
4	109,425	-382,909	-252,624	-109,425	382,9092	-130,285
5	382,9092	109,425	130,2853	-382,909	-109,425	88,56479

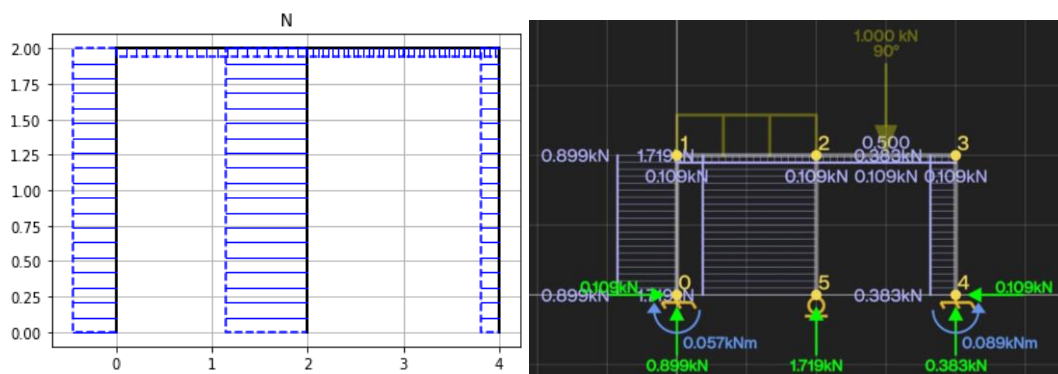
Dijagrami su generirani u Python konzoli i bit će uspoređeni s dijagramima dobivenim iz aplikacija *STRAN* i *SW FEA 2D FRAME*, koje koriste istu metodu. Dijagrami koji će bit prikazani deformiranog i nedeformiranog okvirnog nosača, momenta uzdužnih i poprečnih sila

Na slici 6.7 prikazana je usporedba dijagrama deformiranog i nedeformiranog okvirnog nosača. Lijevi dijagram je dobiven korištenjem *Python* programa, dok je desni dijagram dobiven korištenjem aplikacije *STRAN*.



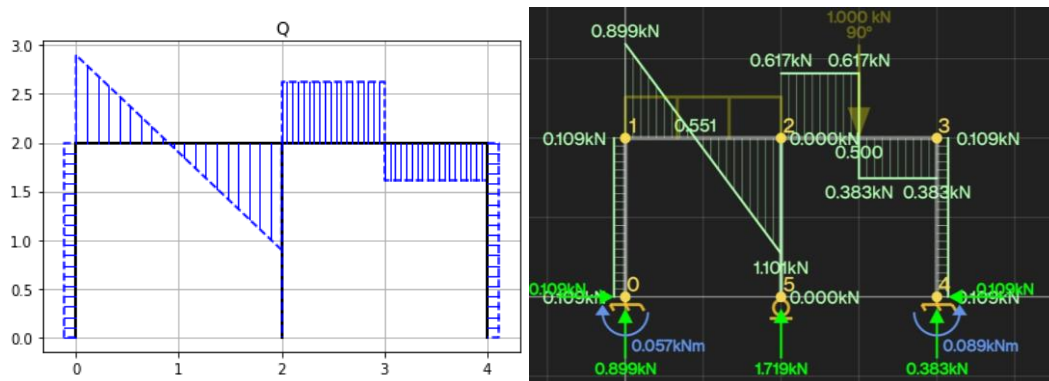
Slika 6.7 Usporedba dijagrama deformiranog i nedeformiranog okvirnog nosača

Na slici 6.8 nalazi se usporedba dijagrama uzdužnih sila gdje je lijevi dobiven u *Python* programu, a desni u aplikaciji *STRIAN* i *SW FEA 2D FRAME*.



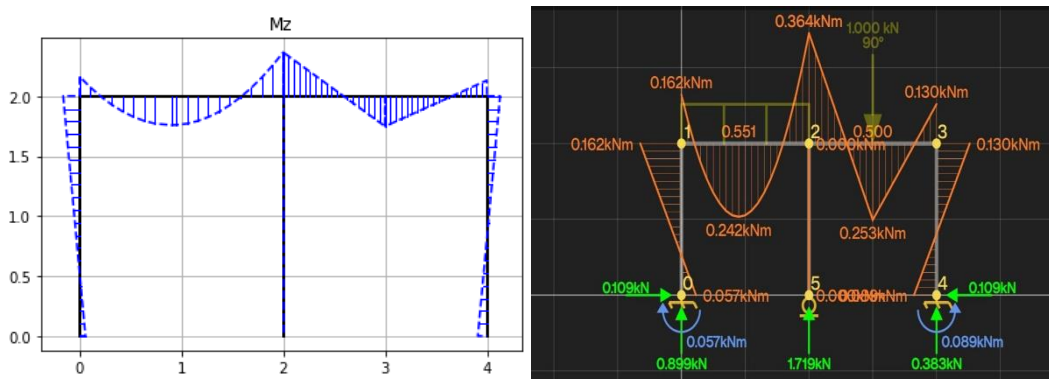
Slika 6.8 Usporedba dijagrama uzdužnih sila

Na slici 6.9 nalazi se usporedba dijagrama poprečnih sila gdje je lijevi dobiven u *Python* programu, a desni u aplikaciji *SW FEA 2D FRAM*.



Slika 6.9 Usporedba dijagrama poprečnih sila

Na slici 6.10 nalazi se usporedba dijagrama momenata gdje je lijevi dobiven u *Python* programu, a desni u aplikaciji *SW FEA 2D FRAME*.



Slika 6.10 Usporedba dijagrama momenata

7. Zaključak

U ovom radu je prikazano programsko rješenje primjene metode konačnih elemenata za rješavanje općeg problema ravninskih okvirnih nosača. Prvo je izvršena matematička analiza postupka rješavanja problema ravninskih okvirnih nosača primjenom metode konačnih elemenata, a kako bi se razumjela metodologija i koraci proračuna. Nakon matematičke analize je izvršena implementacija metode u programski kôd.

U rad je uključena i izrada dijagrama toka kako bi se jasno vizualizirali struktura, logika i tok izvršavanja programa, pružajući bolji uvid u njegovu funkcionalnost i organizaciju.

Programskom implementacijom omogućena je analiza i proračun okvirnih nosača primjenom metode konačnih elemenata, uz prikaz rezultata na konzoli te njihovo pohranjivanje u Excel datoteku za daljnju analizu i dokumentaciju.

Korištenjem *numpy* biblioteke za proračune, program značajno ubrzava rješavanje problema naročito u slučaju problema s velikim brojem konačnih elemenata. *Numpy* omogućava efikasnu manipulaciju velikih matrica i vektora, što je ključno za brzinu i točnost numeričke analize.

Nakon izrade programa isti je testiran te su rezultati uspoređeni s rezultatima dobivenim primjenom sličnog programa [5][6] koji su dostupni na internetu. Usporedbom dobivenih rezultata potvrđena je ispravnost rada izrađenog programa. Kod nekih javno dostupnih aplikacija može se uočiti da postoje neke razlike u dijagramima momenta, uzdužnih i poprečnih sila što se može pripisati različitim metodama implementacije proračuna unutrašnjih sila i momenata po segmentima elemenata.

Također, program nije ograničen na broj konačnih elemenata, brojem oslonaca, sila, momenata i kontinuiranih opterećenja što je čest slučaj kod javno dostupnih "besplatnih" programa.

Za daljnje poboljšanje ovog programskog rješenja moglo bi se razmotriti dodavanje grafičkog sučelja (GUI) kako bi se olakšao unos podataka i vizualizacija rezultata. Grafičko sučelje omogućilo bi korisnicima intuitivnije korištenje programa bez potrebe za poznavanjem programskog jezika ili složenih naredbi.

8. Reference

1. Kim, D. H., & Sankar, B. V.: „Introduction to Finite Element Analysis and Design“, Wiley (2009).
2. Hughes, T. J. R.: „The Finite Element Method: Linear Static and Dynamic Finite Element Analysis“, Dover Publications (2000).
3. David V.H.: „FUNDAMENTALS OF FINITE ELEMENT ANALYSIS“, McGraw-Hill (2004)
4. Meštrović, M.: „Metoda Konačnih Elemenata“, Građevinski fakultet Zagreb (2020).
5. STRIAN [Online] . Dostupno na : <https://structural-analyser.com/> dana 10.06.2024
6. SW FEA 2D FRAME [Online]. Dostupno na:
<https://play.google.com/store/apps/details?id=np.com.softwel.swframe2d&hl=en>.
dostupno dana 10.06.2024.