



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A
BIOMECHANIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND
BIOMECHANICS

**DEFORMAČNĚ-NAPĚŤOVÁ ANALÝZA PRUTOVÝCH SOUSTAV
METODOU KONEČNÝCH PRVKŮ**

STRESS-STRAIN ANALYSIS OF A BAR SYSTEMS USING FINITE ELEMENT METHOD

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

JÁN FODOR

VEDOUcí PRÁCE
SUPERVISOR

Ing. TOMÁŠ NÁVRAT, Ph.D

BRNO 2015

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav mechaniky těles, mechatroniky a biomechaniky
Akademický rok: 2014/2015

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

student(ka): Ján Fodor

který/která studuje v **bakalářském studijním programu**

obor: **Základy strojního inženýrství (2341R006)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Deformačně-napěťová analýza prutových soustav metodou konečných prvků

v anglickém jazyce:

Stress-strain analysis of a bar systems using finite element method

Stručná charakteristika problematiky úkolu:

Cílem práce je naprogramovat algoritmus metody konečných prvků pro řešení prutových soustav. Pro řešení primárně využít volně dostupné prostředky (Python, knihovny NumPy, SciPy, překladač Fortranu, apod.). Ověření funkčnosti realizovat výpočtem v programu ANSYS.

Cíle bakalářské práce:

1. Přehled používaných MKP programů se stručným popisem jejich možností.
2. Možnost využití volně dostupných prostředků pro vědecké výpočty.
3. Naprogramovat algoritmus MKP pro prutovou soustavu.
4. Verifikovat vypočtené výsledky s výsledky získanými v programu ANSYS.

Seznam odborné literatury:

1. Zienkiewicz, O., C., Taylor, R. L.: The finite element method, 5th ed., Arnold Publishers, London, 2000
2. Kolář, V., Kratochvíl, J., Leitner, F., Ženíšek, A.: Výpočet plošných a prostorových konstrukcí metodou konečných prvků, SNTL Praha, 1979
3. Ferreira, A.J.M.: MATLAB Codes for Finite Element Analysis: Solids and Structures, 2009

Vedoucí bakalářské práce: Ing. Tomáš Návrat, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2014/2015.

V Brně, dne 17.11.2014

L.S.

prof. Ing. Jindřich Petruška, CSc.
Ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
Děkan fakulty

Abstrakt

Cieľom tejto bakalárskej práce je primárne naprogramovať algoritmus pre riešenie prútových sústav pomocou metódy konečných prvkov. K naprogramovaniu sú použité voľne dostupné softvérové prostriedky – jazyk Python a jeho knižnice NumPy, SciPy a Matplotlib. Výsledky takto získané sú overené v programe ANSYS. Práca ďalej podáva stručný prehľad komerčne a voľne dostupného vedeckého softvéru.. Metóda konečných prvkov je numerická metóda využívaná pre riešenie problémov technickej praxe v strojárstve, stavebníctve, elektrotechnike a príbuzných odboroch.

Abstract

The aim of this bachelors thesis is primarily programming of algorithm for solving bar systems (i.e. trusses) using finite element method. Programming language Python and his modules NumPy, SciPy and Matplotlib are used . The results thus obtained are verified in the program ANSYS . This work also gives a brief overview of commercial and freely accessible scientific software. Finite Element Method is a numerical method used for solving technical problems in mechanical engineering, structural engineering, electrical engineering and related fields.

Kľúčové slová:

Prútové sústavy, Metóda konečných prvkov, Python, ANSYS

Key words:

Bar systems, Finite element method, Python, ANSYS

Bibliografická citácia:

FODOR, J. *Deformačně-napěťová analýza prutových soustav metodou konečných prvků*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2015. 66 s. Vedoucí bakalářské práce Ing. Tomáš Návrat, Ph.D..

Prehlásenie:

Prehlasujem, že som bakalársku prácu vypracoval samostatne, pod vedením Ing. Tomáša Návrata, Ph.D a s použitím zdrojov uvedených na konci práce.

V Brne dňa:.....

.....
Ján Fodor

Pod'akovanie:

Chcel by som pod'akovať Ing. Tomášovi Návratovi, Ph.D za rady a pripomienky pri písaní práce a mojím rodičom za podporu pri štúdiu.

OBSAH

1. ÚVOD, CIEĽ PRÁCE	9
2. MOŽNOSŤ VYUŽITIA VOLNE DOSTUPNÉHO SOFTVÉRU PRE VEDECKÉ A INŽINIERSKE VÝPOČTY	10
2.1 ZÁKLADNÉ POJMY SÚVISIACE S VOLNE DOSTUPNÝM SOFTVÉROM	10
2.2 PREHLAD VOLNE DOSTUPNÉHO SOFTVÉRU, KTORÝ V SEBE OBSAHUJE ALGORITMUS MKP	12
2.3 STRUČNÁ CHARAKTERISTIKA MKP SOFTVÉRU A NIEKTORÝCH ĎALŠÍCH POUŽÍVANÝCH PRE VEDECKÉ A INŽINIERSKE VÝPOČTY	12
2.3.1 FEBio.....	12
2.3.2 Calculix.....	13
2.3.3 Code_Aster	14
2.3.4 OOFEM	14
2.3.5 Z88Aurora	15
2.3.6 Python.....	16
2.3.7 Scipy	16
2.3.8 NumPy	16
2.3.9 Knižnica Scipy.....	16
2.3.10 Matplotlib	16
2.3.11 Anaconda	17
2.3.12 GNU Octave	17
2.3.13 Scilab	18
3. PREHLAD KOMERČNE DOSTUPNÝCH PROGRAMOV OBSAHUJÚCICH ALGORITMUS MKP	19
3.1 ZOZNAM NAJZNÁMEJŠÍCH MKP PROGRAMOV.....	19
3.2 ANSYS	19
3.2_NASTRAN.....	21
3.3 RFEM.....	22
3. TEORETICKÝ ÚVOD DO PRÚTOVÝCH SÚSTAV A MKP	23
3.1 PRÚT – VYMEDZENIE POJMU	23
3.2 PRÚTOVÉ SÚSTAVY	23

3.3 VZPERNÁ STABILITA PRÚTU	24
3.3 MOŽNOSTI RIEŠENIA PRÚTOVÝCH SÚSTAV	25
3.4 TEORETICKÝ ÚVOD DO MKP	25
3.4.1 Základné rovnice obecnej pružnosti	25
3.4.2 Metóda konečných prvkov z hľadiska riešenia priamej úlohy pružnosti	27
3.4.3 Lagrangeov variačný princíp	27
3.4.4 Diskretizácia v MKP	28
3.4.5 Základná rovnica MKP	28
3.4.6 Prvok prenášajúci osovú zaťaženie v lokálnom súradnicovom systéme	28
3.4.7 Globálny a lokálny súradnicový systém	29
3.4.8 Skladanie matic K,U,F pre celkovú sústavu	32
3.4.9 Ilustračný príklad:	32
4. PROGRAM V PYTHONĚ A KONTROLA VÝSLEDKOV V SYSTÉME ANSYS	40
4.1 POPIS PROGRAMU	40
4.2 2D ÚLOHA A JEJ OVERENIE	41
4.3 3D ÚLOHA	47
5. GRAFICKÉ UŽÍVATEĽSKÉ ROZHRAŇIE	56
6. ZÁVER	62
ZDROJE	63
ZOZNAM PRÍLOH	66

1. Úvod, cieľ práce

Metóda konečných prvkov (ďalej aj MKP) je numerická metóda pre riešenie parciálnych diferenciálnych rovníc. Od analytického prístupu sa líši tým, že miesto elementu nekonečne malých rozmerov zavádza prvok konečnej veľkosti. Týmto spôsobom sa dajú riešiť mnohé problémy, ktoré sa klasickým analytickým prístupom riešiť nedajú. Sú to napríklad analýzy tvarovo zložitých súčastí. Jediným obmedzením MKP je dostupnosť výpočtovej techniky, ktorej výkon a množstvo dostupnej pamäti ovplyvňuje rýchlosť riešenia rozsiahlych sústav algebraických rovníc, ktoré sa vyskytujú pri riešení úloh MKP. Na počiatku svojho vzniku slúžila metóda pre numerické riešenie problémov pružnosti v stavebníctve a leteckom priemysle. V dnešnej dobe nachádza široké uplatnenie v oblastiach strojárstva, stavebníctva, elektrotechniky a príbuzných odboroch. Metódou konečných prvkov sa dajú riešiť hlavne problémy pružnosti, plasticity, dynamiky, mechaniky tekutín, elektromagnetizmu a prenosu tepla. Pre tieto výpočty existuje veľké množstvo komerčných a voľne dostupných programov, ktoré majú v sebe často okrem algoritmu MKP implementované aj iné numerické metódy, napríklad metódu konečných objemov pre prípad riešenia problémov mechaniky tekutín.

V tejto práci sú prúťové sústavy poňaté ako zjednodušený model priehradových konštrukcií, ktoré zanedbávajú namáhanie prúťov ohybom a krutom. Toto je spôsobené modelovaním styčníc ako klbových miest reálnych, ktoré môžu byť tuhé a zanedbaním ohybu spôsobeného od zaťaženia prúťov v smere kolmom na strednicu. Jediný nenulový výsledný vnútorný účinok je normálová sila, a teda prúťy sú namáhané len na tlak a ťah. Medzi najdôležitejšie medzné stavy takto staticky namáhaných prúťov patrí medzný stav klzu a medzný stav vzpernej stability v prípade tlakového namáhania.

Priehradové konštrukcie sa využívajú v strojárnských a stavebných aplikáciách pre svoju schopnosť prenášať zaťaženia pri relatívne nízkej hmotnosti konštrukcie na veľké vzdialenosti (napríklad kovové mosty).

Cieľ práce

Cieľom tejto práce je primárne naprogramovať algoritmus MKP pre deformačne – napäťovú analýzu rovinných a priestorových prúťových sústav. Výstupom z programu budú deformácie a napätia v jednotlivých prúťoch sústavy, sily vo väzbách s okolím vrátane grafického zobrazenia deformovaného tvaru sústavy. K tomuto cieľu majú byť dosiahnuté voľne dostupné prostriedky. Algoritmus bude naprogramovaný v jazyku Python za použitia knižníc a funkcií pre prácu maticami, riešenie sústav lineárnych algebraických rovníc a pre vykresľovanie funkcií. Výsledky takto získané budú overené v programe ANSYS. Ďalší cieľ je stručný prehľad voľne dostupných programov pre inžinierske a vedecké výpočty všeobecne a prehľad komerčných programov, ktoré majú v sebe implementovaný algoritmus MKP. V úvode budú zhrnuté základné teoretické poznatky o MKP a jej aplikácii na riešenie rovinnéj a priestorovej prúťovej sústavy. Pre názornú predstavu bude jedna úloha riešená “ručne“ za pomoci kalkulačky.

2. Možnosť využitia voľne dostupného softvéru pre vedecké a inžinierske výpočty

2.1 Základné pojmy súvisiace s voľne dostupným softvérom

Na začiatok je vhodné charakterizovať niektoré pojmy, ktoré súvisia s voľne dostupným softvérom. Táto problematika je však veľmi široká, a tak budú stručne charakterizované len niektoré hlavné pojmy.

Softvérová licencia

Softvérová licencia je právny nástroj, ktorý reguluje podmienky používania a redistribúcie softvéru.

Copyleft

Pod označením copyleft, respektíve copyleftová licencia si je možno predstaviť druh licencií, ktoré znamenajú, že diela odvodené od softvéru s licenciou copyleft musia byť tiež dostupné pod touto licenciou. Tento druh licencií sa používa v takzvanom slobodnom softvéri. Príkladmi copyleftových licencií sú GPL, LGPL. Tento druh licencie teda v podstate zaručuje, aby slobodný softvér a ďalšie z neho odvodené ostali aj naďalej slobodné.[6]

Slobodný softvér

Slobodný softvér (free software) je softvér, ktorého nadobudnutie nie je podmienené poskytnutím finančnej alebo inej odmeny jeho autorovi. Je k nemu k dispozícii aj zdrojový kód. Softvér je možno ďalej používať, modifikovať a distribuovať. Distribuovať ho je možné zdarma a aj predávať jeho kópie a pozmenené verzie. Slovo slobodný (po anglicky free) sa teda nedá preložiť ako automaticky zadarmo a neznamená, že softvér sa nesmie komerčne využívať. Jedná sa hlavne o filozofiu dostupnosti zdrojového kódu, používania a ďalšej distribúcie. Kréda slobodného softvéru sa dajú zhrnúť takto:

- Sloboda používať program za akýmkoľvek účelom.
- Sloboda študovať, ako program pracuje a možnosť prispôsobiť ho svojim potrebám.
- Sloboda redistribuovať kópie programu.
- Sloboda vylepšovať program a zverejňovať zlepšenia, aby z nich mohla mať prospech celá komunita.

Najčastejšia licencia slobodného softvéru je GPL. Za myšlienku slobodného softvéru stojí hnutie Free Software Foundation. [7], [8]

Otvorený softvér

Po anglicky Open-source software je softvér s otvoreným zdrojovým kódom. Otvorenosť znamená možnosť za určitých podmienok zdrojový kód ďalej prehliadať a upravovať. Nie je to jediná charakteristika otvoreného softvéru, je však najzákladnejšia. Open source softvér môže byť poskytnutý zdarma. Základný rozdiel medzi otvoreným softvérom a slobodným softvérom by sa dal charakterizovať tak, že nie každý otvorený software je slobodný, ale každý slobodný je zároveň aj otvorený.

Z hľadiska nadobúdateľa softvéru je rozdiel medzi slobodným a otvoreným minimálny, z hľadiska ďalšej distribúcie sú však rozdielny. V prípade otvoreného softvéru

môže napríklad vzniknúť povinnosť nadobúdateľa informovať o zmenách, ktoré urobil v zdrojovom kóde, pokiaľ ho chce ďalej distribuovať, prípadne informovať poskytovateľa o ďalšej distribúcii. Ideovým autorom otvoreného software je Open Source Initiative. Medzi Free Software Foundation (FSF) a Open Source Initiative (OSI) sa vedú spory kvôli rôznym pohľadom na otvorený slobodný softvér, to však presahuje rozsah tejto práce. [9]

Proprietárny softvér

Tento pojem býva často používaný ako protiklad ku slobodnému softvéru. (Nadáciou Free Software Foundation) je tak označovaný každý program, do ktorého zdrojového kódu nemá užívateľ väčšinou právo zasahovať a ani si ho prezerat'. Pod tento pojem tak spadajú okrem "bežne predávaného" napríklad aj softvéry poskytované k užívaniu zdarma (freeware). [10]

Komerčný softvér

Komerčný software je software vyvíjaný za účelom zisku, teda za získanie jeho licencie je potrebné zaplatiť. Do tejto kategórie môže spadať otvorené, slobodné a aj proprietárny software.

Freeware

Softvér dostupný k užívaniu zdarma, spravidla bez možnosti prístupu k zdrojovému kódu. Autor často ponúka jeho ďalšie vylepšenia ako komerčné programy.

GNU General Public License

GNU General Public License, GNU GPL je druh licencie pre slobodný softvér, ktorá vyžaduje, aby ďalšie diela od neho odvodené boli distribuované pod licenciou GNU GPL. V priebehu vývoja bola táto licencia upravená do troch verzií, ktoré nahradzovali staršie verzie. [11]

BSD Licencia

Je to licencia umožňujúca voľné šírenie diela vyžadujúca uvedenie autora, informácie o licencií a upozornenie o zrieknutí sa zodpovednosti za dielo.

GNU Lesser General Public License

GNU LGPL je druh licencie, ktorá je považovaná za kompromis medzi GNU GPL a BSD licenciou.

2.2 Prehľad voľne dostupného softvéru, ktorý v sebe obsahuje algoritmus MKP

Tab. 2.1 voľne dostupný softvér s implementovaným algoritmom MKP [12]

názov	licencia	Operačný systém
Agros2D	GPL	Windows, Linux
CalxuliX	GPL	Windows, Linux
Code Aster	GPL	Linux, FreeBSD
Deal.II	LGPL	Linux, Unix, Windows, Max OS X
DUNE	GPL ver.2	Linux, Unix, Max OS X
Elmer	GPL	Linux, Windows, Max OS X
FEBio	Zvláštny typ	Linux, Windows, Max OS X
FEniCS Project	LGPL	Linux, Unix, Windows, Max OS X
FreeFem++	GPL	Linux, Unix, Windows, Max OS X, Solaris
GetFEM++	LGPL	Unix, Windows, Max OS X
Hermes Project	LGPL	Linux, Unix, Windows, Max OS X
jFem	GPL	Linux, Unix, Windows, Max OS X
MoFEM JosePH	LGPL	Linux, Unix, Max OS X
MOOSE	LGPL	Linux, Unix, Max OS X
OOFEM	LGPL	Windows, Unix
OpenFOAM	GPL	Linux, Unix
OpenSees	Zvláštny typ	Linux, Unix, Windows
Z88/Z88Aurora		Linux, Windows, Mac OS X

2.3 Stručná charakteristika MKP softvéru a niektorých ďalších používaných pre vedecké a inžierske výpočty

2.3.1 FEBio

FEBio (Finite elements for biomechanics) je využívaný v biomechanike a biofyzike. Je špeciálne určený pre nelineárne problémy veľkých deformácií. Je to voľne dostupný softvér pre nekomerčné použitie spustiteľný na operačných systémoch Windows, Linux a Mac OS X. Bol vyvinutý na University of Utah. Ponúka analýzy z oblastí:

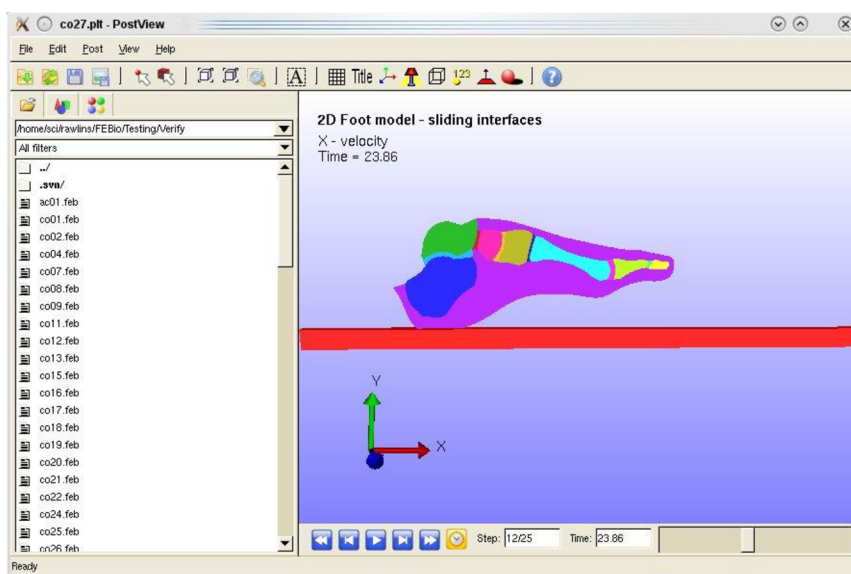
- Nelineárna elasticita a viskoelasticita.
- Mechanika tuhých telies.
- Analýzy z oblasti chemických reakcií, prúdenie v poréznom prostredí, osmóza, vedenie tepla, mechanika rastu tkaniva.

Podporuje okrajové podmienky odpovedajúce problémom biomechaniky. FEBio sa skladá z troch súčastí:

- preprocesoru PreView - vytvorenie geometrie, siete, okrajových podmienok, zadávanie materiálov a zadanie druhu analýzy. Výstupom z PreView sú

dáta, s ktorými ďalej pracuje riešič. Do preprocesoru možno importovať geometriu formátu DXF, ďalej geometriu zo softwaru ako napríklad ABAQUS, LSDYNA, HyperMesh a HyperSurface.

- vlastného riešiča FEBio
- postprocesoru PostView – rozličné možnosti vizualizácie výsledkov. Umožňuje aj uloženie grafického výstupu vo formáte .bmp a .tiff. [13]



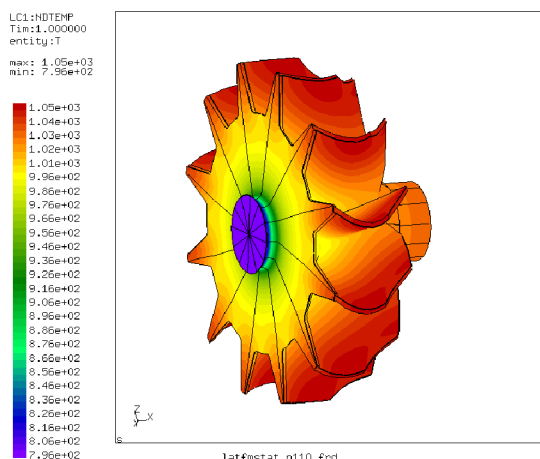
Obr. 2.1 Ukážka z prostredia PostView [14]

2.3.2 Calculix

Calculix je voľne dostupný softvér s implicitným a explicitným riešičom. Pôvodne bol dostupný iba pre Linux, dnes je dostupný aj pre OS Windows. Riešič je schopný lineárnych aj nelineárnych výpočtov. Dokáže riešiť problémy:

- statiky
- dynamiky
- teplotné úlohy.

Je možné používať vlastný preprocesor Calculixu, alebo je možno model vytvoriť v komerčných programoch ako Nastran, Ansys a Abaqus. Je v ňom tiež možný opačný postup, teda vytvárať geometriu pre NASTRAN, ABAQUS, Ansys, Code-aster pre voľne dostupné programy ako Dolfyn, Duns, ISAAC a OpenFOAM. [15],[16]



Obr. 2.2 Calculix [17]

2.3.3 Code_Aster

Softvér Code_Aster ponúka výpočty v oblasti lineárnej a nelineárnej statiky a dynamiky, únavy, pórovitých materiálov, termálnej analýzy a multifyzikálnych úloh. Bol písaný hlavne v jazykoch Python a Fortran. Využíva sa aj v problémoch súvisiacich s nukleárnymi zariadeniami. Je dostupný pre GNU/Linux a FreeBSD. Je spustiteľný aj na OS Windows, ale platí to len pre staršie verzie. [18], [19]

2.3.4 OOFEM

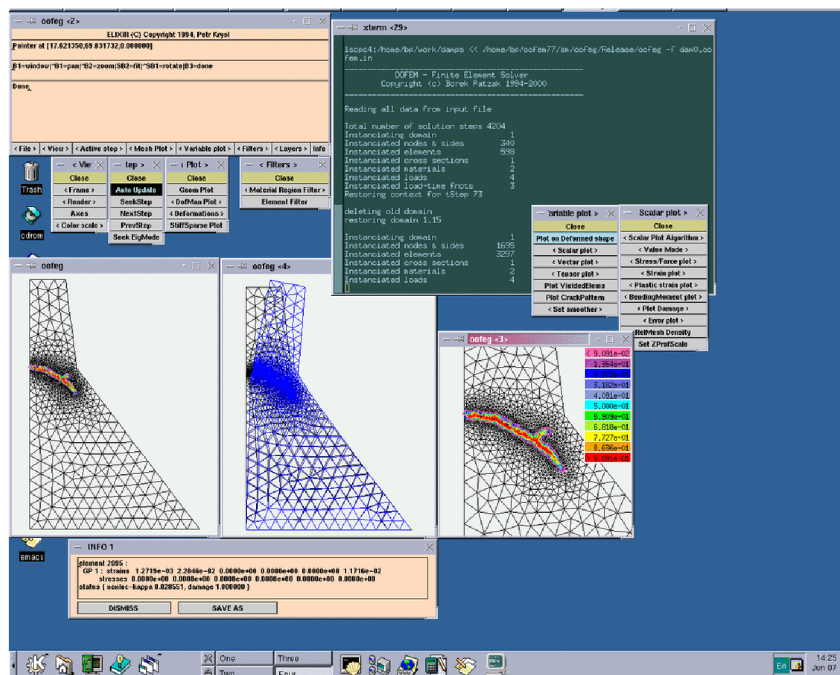
OOFEM je objektovo orientovaný program pre riešenie multifyzikálnych úloh pomocou MKP. Je stále vyvíjaný na katedre mechaniky Stavebnej Fakulty ČVUT pod licenciou GNU Lesser General Public license.

Medzi časti programu patria:

- Obecné modulárne výpočtové jadro (OOFEMlib)
- Modul pre mechaniku. Umožňuje riešiť napríklad problémy lineárnej a nelineárnej statiky a dynamiky, podpora paralelných výpočtov.
- Transportné problémy: vedenie tepla a vlhkosti, prepojenie transportných problémov a mechanickej analýzy.
- Mechanika tekutín
- Postprocesor

OOFEM umožňuje využitie externých programov:

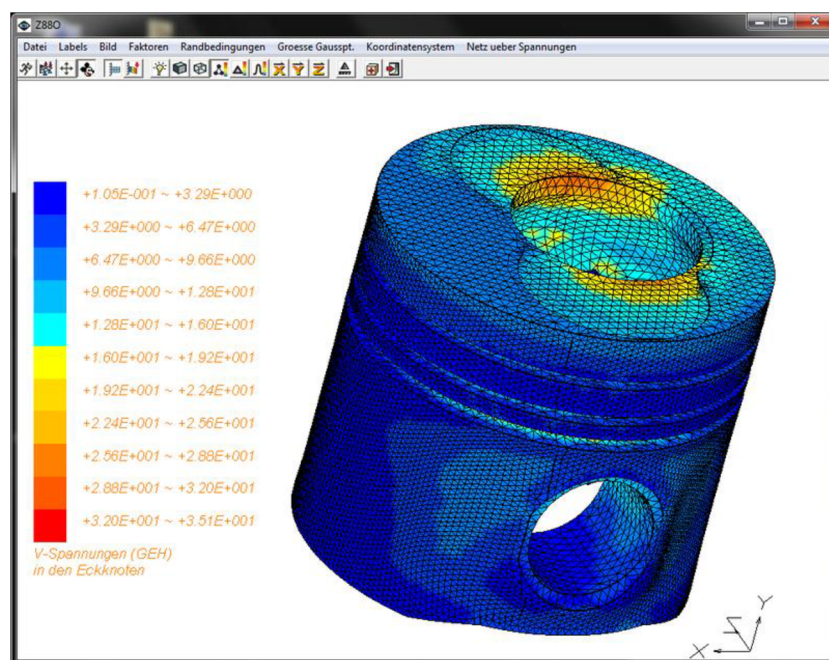
- T3d – generátor siete
- IML++ - knižnica iteračných metód
- SPOOLES – riedky priamy riešič
- MayaVi, ParaView – vizualizačné nástroje [20]



Obr. 2.3 analýza kolapsu priehrady Konya Dam [21]

2.3.5 Z88Aurora

Pod názvom Z88Aurora je myslený grafické užívateľské rozhranie pre program na Z88. Je dostupné ako freeware. Preprocesor umožňuje import geometrie vo formáte .STP, .STL a .DXF. Ďalej umožňuje import geometrie vytvorenej v komerčných programoch ako NASTRAN, ABAQUS, ANSYS a COSMOS. Z88Aurora je schopná riešiť lineárnu a nelineárnu mechaniku, termálnu analýzu, spojenie termálnej a mechanickej analýzy a výpočet vlastných frekvencií. [22]



Obr. 2.4 Z88 Aurora [23]

2.3.6 Python

Python je (najmä) objektovo orientovaný programovací jazyk, jeho prednosťou je hlavne jeho jednoduchosť a prehľadnosť, čím ho robí vhodným aj pre začiatočníkov. Pre inžinierske výpočty je sú významné jeho distribúcie a knižnice obsahujúce nástroje pre riešenie numerických úloh, napríklad nástroje pre riešenie rozsiahlych sústav rovníc, dátové typy, nástroje pre vykresľovanie dát podobné matlabu a mnohé ďalšie. Jazyk Python vytvoril v roku 1991 Guido van Rossum.

2.3.7 Scipy

Pojmom SciPy sa označuje zbierka nástrojov Pythonu pre vedecké a inžinierske výpočty. Medzi hlavné balíky, ktoré sú súčasťou SciPy patrí NumPy, knižnica SciPy, Matplotlib, Ipython, Sympy, Pandas.

2.3.8 NumPy

– Balík dátových typov a funkcií pre prácu s rozsiahlymi dátami. Jeho najdôležitejšou súčasťou je "ndarray", n-rozmerné pole (array). NumPy obsahuje rôzne funkcie umožňujúce prácu s týmto dátovým typom. V tejto práci sa ndarray používa napríklad ako matica tuhostí. Je tiež použitých niekoľko funkcií pre prácu s poľami, napríklad *numpy.reshape* (premena matice na maticu o iných rozmeroch s rovnakým počtom prvkov), alebo *numpy.zeros* (vytvorenie matice s prvkami o hodnote 0).

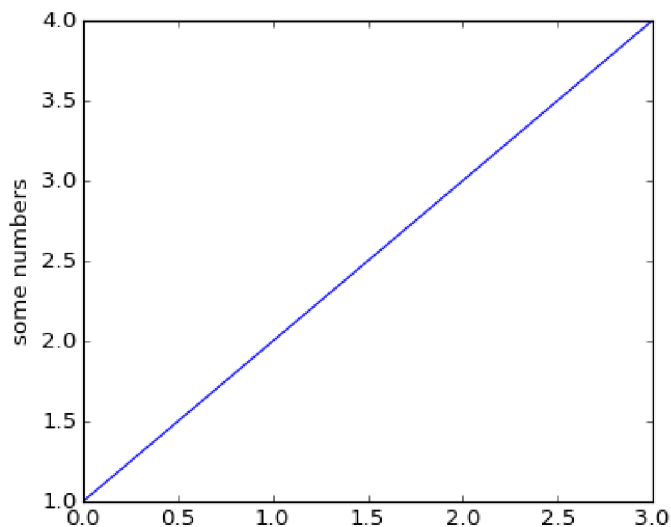
2.3.9 Knižnica Scipy

Je to knižnica obsahujúca nástroje pre riešenie optimalizácie, lineárnej algebry, integrácie, spracovania signály, riešenie obyčajných diferenciálnych rovníc a mnohé iné. Knižnica je aktuálne pod takzvanou BSD licenciou. (poznámka). V tejto práci je použitá funkcia *linalg.solve*, ktorá rieši sústavu lineárnych algebraických rovníc

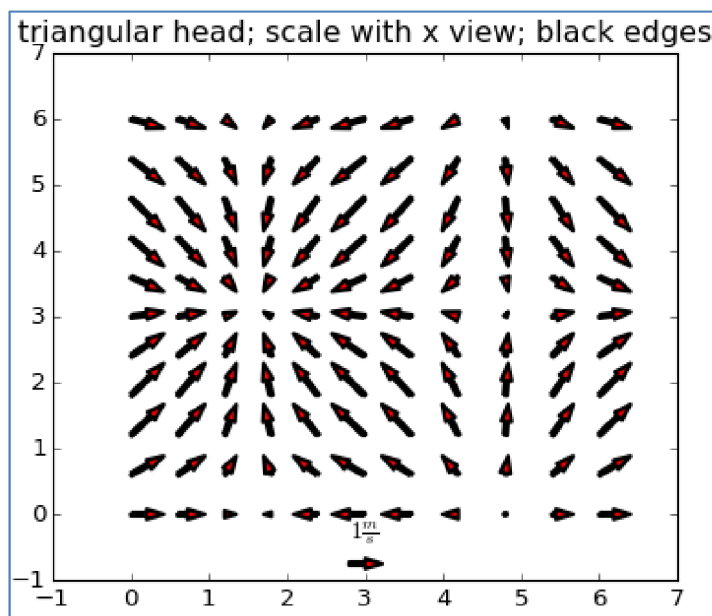
2.3.10 Matplotlib

Nástroj pre vizualizáciu 2D a 3D dát. Umožňuje vykresľovanie vzhľadom podobné Matlabu. Obsahuje veľké množstvo typov pre vykresľovanie, ako napríklad "klasický" plot, histogram, scatter, quiver a mnoho ďalších. V tejto práci je použitý na vykreslenie prúťových sústav, ich deformovaných tvarov, a pôsobiacich síl.


```
import matplotlib.pyplot as plt
plt.plot([1,2,3,4])
plt.ylabel('some numbers')
plt.show()
```



Obr. 2.5 príklad vykreslenia funkcie pomocou Matplotlib [24]



Obr 2.5 príklad vykreslenia funkcie pomocou Matplotlib [25]

2.3.11 Anaconda

Proprietárny softvér (inak softvér s uzavretým kódom). Je to distribúcia jazyka Python od firmy Continuum Analytics, dostupná k užívaniu zdarma. Obsahuje najpoužívanejšie balíky pre vedecké a inžinierske výpočty, pre túto prácu dôležité NumPy, Scipy, Matplotlib. Obsahuje tiež niekoľko užívateľsky pohodlných vývojových prostredí (po anglicky IDE – integrated development environment), v tejto práci je použité prostredie Spyder. [26]

2.3.12 GNU Octave

GNU Octave je vysokoúrovňový programovací jazyk, kompatibilný s MATLABom. Je to takzvaný slobodný software pod licenciou GNU General Public License. Je určený na

numericé výpočty. Príklady jeho použitia sú riešenie lineárnych a nelineárnych rovníc , numerické derivovanie, integrovanie, vizualizácia dát a manipulácia s dátami. Na vzniku GNU Octave sa podieľal okrem iných hlavne John W. Eaton. Je dostupný pre operačné systémy GNU/Linux Windows, UNIX, Cygwin a OS X. [27]

2.3.13 Scilab

Scilab je voľne dostupný inžiniersky a vedecký software pod licenciou CeCill(). Je dostupný pre Windows 7,8, Vista, GNU/Linux a Mac OS X. Medzi jeho takzvané funkcionality patria:

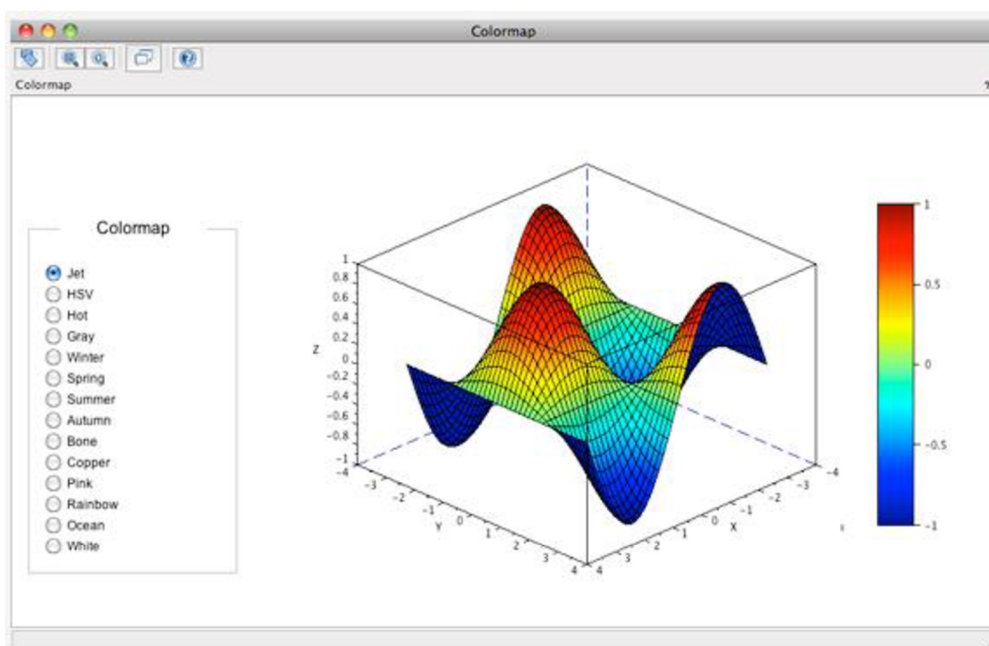
Maths & Simulation

Elementárne funkcie, lineárna algebra, riedke matice, polynómy, simulácia (diferenciálne rovnice)

2D a 3D Visualization

“Klasické čiary“, histogramy, animácie, plochy, koláčový graf.

Medzi ďalšie možnosti patrí optimalizácia, štatistika, riadenie, spracovanie signálu, vývoj aplikácií, modelovanie mechanických systémov, hydraulických obvodov. [28]



Obr. 2.6 Vizualizácia v Scilabe [29]

3. Prehľad komerčne dostupných programov obsahujúcich algoritmus MKP

3.1 Zoznam najznámejších MKP programov

V dnešnej dobe umožňuje väčšina programov riešenie nielen úloh klasickej deformačne-napät'ovej analýzy, ale sú to skôr multifyzikálne produkty pre riešenie úloh štruktúrálnej analýzy, prúdenia, prenosu tepla, únav, dynamiky a elektromagnetizmu nakoľko pri vývoji produktu väčšinou samotná deformačne-napät'ová analýza nestačí a výsledky z rôznych analýz sú často použité pre tvorbu inej analýzy. Nie všetky tieto súčasti sú postavené na metóde konečných prvkov, ale napríklad aj na metóde konečných objemov.

Tab. 3.1 prehľad komerčného softéru s implementovaným algoritmom MKP [30]

Abaqus	ADINA	Advance Design
ALGOR	Altair HyperWorks	ANSA
ANSYS	AutoForm	CFD-FASTRAN
Computers and Structures	COMSOL Multiphysics	CosmosWorks
CST Studio Suite	DIANA_FEA	Diffpack
Extreme Loading for Structures	FEFLOW	FlexPDE
HydroGeoSphere	HyperSizer	JMAG
LS-DYNA	LUSAS	MADYNO
Nastran	NEi Fusion	NEi Nastran
NEi Works	Pam-Crash	Quickfield
RadioSS	RAM Elements	Range Software
RFEM	S-FRAMW	SAMCEF
SAP2000	SESAM	Sherlock Autom. Design Analysis
STRAND7	StressCheck	Vflo
VisualFEA		

3.2 ANSYS

Pod mojmom ANSYS si dnes možno predstaviť rozsiahly inžiniersky software pre modelovanie veľkého množstva fyzikálnych dejov. Vďaka rozsiahlemu záberu možností simulácie a analýzy jeho použitie siaha do mnohých sfér vedy a priemyslu od leteckého, strojárenského, stavebného, elektrotechnického až po energetiku, procesné inžinierstvo a zdravotníctvo.

ANSYS sú vlastne desiatky produktov, ktoré by sa podľa oblasti záujmu dali rozdeliť do niekoľkých základných skupín:

- Electronics
- Fluid dynamics
- Structural analysis Multiphysics

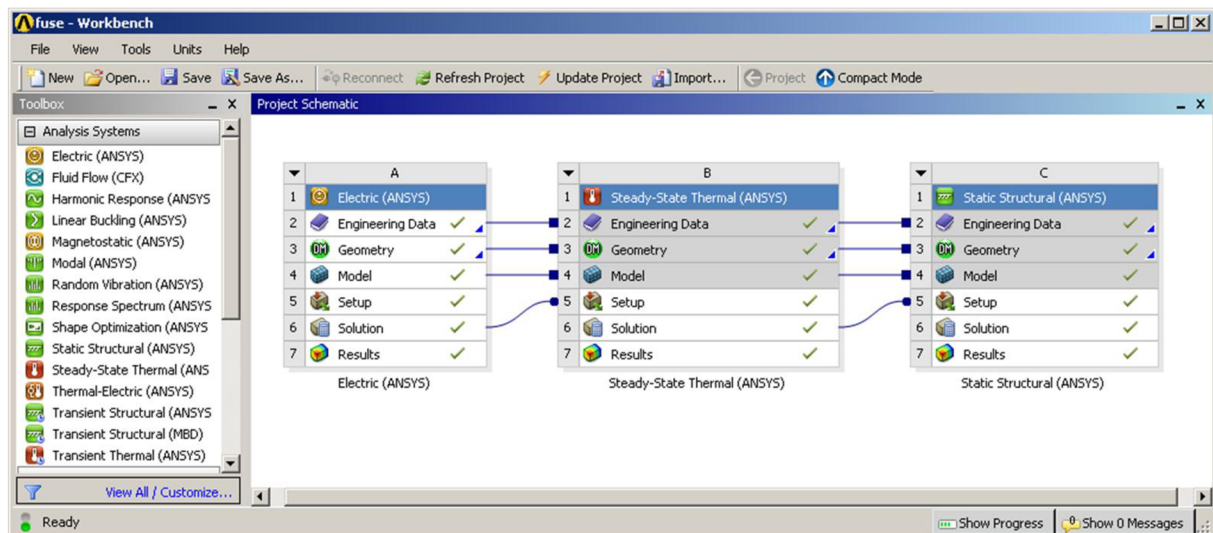
Structural analysis

Structural analysis je z hľadiska pevnostných výpočtov strojných súčastí najdôležitejšia skupina, ponúka riešenie úloh lineárnej a nelineárnej statiky a úlohy dynamiky. Medzi súčasti tejto skupiny patria:

- (ANSYS) Mechanical
- Structural
- Professional
- DesignSpace
- ACT
- Rigid Body Dynamics
- Composite PrepPost
- nCode DesighLife
- Explicit Dynamics

ANSYS Workbench

Platforma ANSYS Workbench umožňuje simuláciu a analýzu multifyzikálnych úloh vďaka systému prepojenia jednotlivých analýz, ktorých výsledky navzájom súvisia. Je to užívateľsky príjemné prostredie ovládané “klikaním” narozdiel napríklad od prostredia APDL, ktoré sa ovláda predovšetkým pomocou skriptov, čo je pre neskúseného užívateľa náročné, hlavne, ak je potrebné zadať zložitú úlohu. Workbench obsahuje “Toolbox“, v ktorom sa vyberajú rôzne druhy analýz ako štruktúrna, analýza prúdenia, elektromagnetického poľa, teplotná úloha a podobne. [31]



Obr. 3.1 prostredie Workbench [32]

3.2_NASTRAN

NA(SA)STR(ucture)AN(alysis) bol vyvinutý pre NASA v šedťdesiatych rokoch 20. storočia. Jeho vývoj financovala vláda USA. Firma Computer Sciences Corporation (CSC) bola vybratá Pre vývoj softvéru. V dnešnej dobe sa v širokej miere používa v leteckom, lodnom a automobilovom priemysle. Medzi programy, ktoré sú postavené na pôvodnom zdrojovom kóde NASTRANu, patria:

- MSC Nastran
- NASTRAN-xMG
- NEi Nastran
- NX Nastran [33]

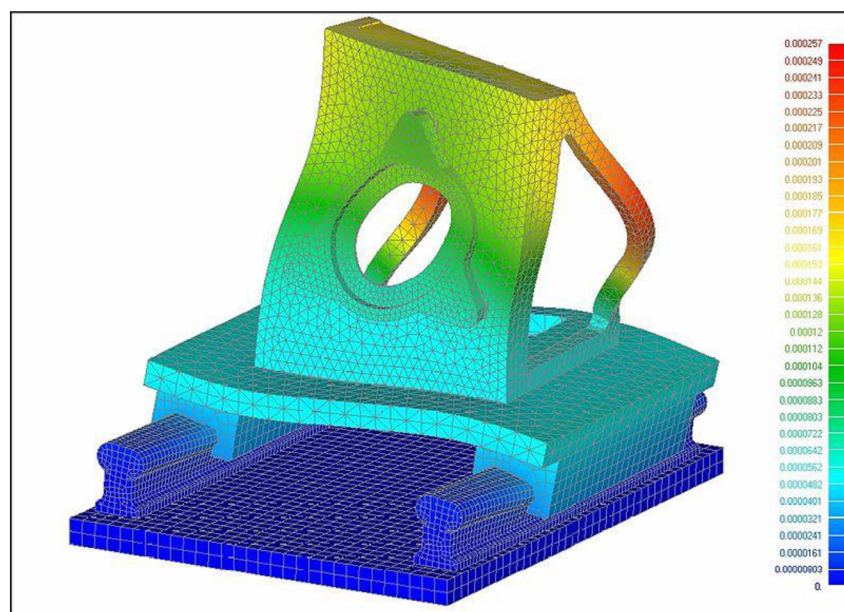
MSC Nastran

MSC Nastran je originálny NASTRAN pre komerčné použitie, dostupný od roku 1963. V dnešnej dobe ho poskytuje sponočnosť MSC Software spolu s veľkým množstvom iných produktov, schopných vzájomne spolupracovať. Samotný MSC Nastran je multidisciplinárny software pre statické, dynamické a termálne výpočty lineárne a nelineárne, dynamiku rotorov a výpočty na posúdenie únavy. [34]

NEi Nastran

NEi Nastran je software od firmy NEi software. Ponúka možnosti pre riešenie:

- Lineárnej a nelineárnej statiky
- Dynamiky
- Stabilitných výpočtov
- Výpočtov prenosu tepla [35]

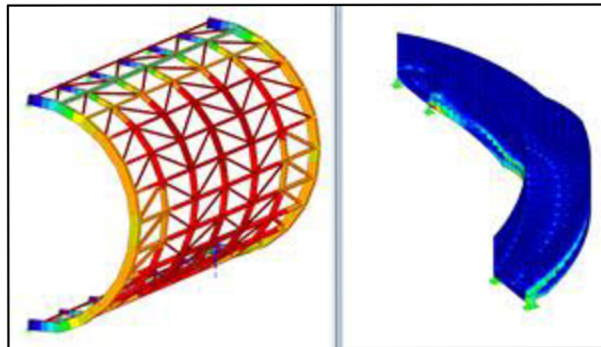


Obr. 3.2 NEi Nastran [36]

3.3 RFEM

RFEM je program používaný v stavebníctve pre statické výpočty konštrukcií. Pochádza od firmy Dlubal Software. K hlavnému programu RFEM sú pridávané moduly pre riešenie konkrétnych stavebných konštrukcií, delené do kategórií:

- Ocel
- Betón
- Drevo
- Sklo
- Prípoje
- Dynamika
- Ostatné [37]



Obr.3.3 RFEM [38]

3. Teoretický úvod do prútových sústav a MKP

3.1 Prút – vymedzenie pojmu

Prút predstavuje z hľadiska deformácie a napätosti najjednoduchší model telesa. Pre vymedzenie pojmu prút sa zavádzajú takzvané prútové predpoklady, ktoré prút charakterizujú:

- Geometrické predpoklady
 - Prút je určený krivkou γ , ktorá sa nazýva strednica. V každom bode strednice je prút definovaný priečnym prierezom ψ , ktorý je kolmý na strednicu. Priesečník strednice a priečného prierezu sa nazýva geometrické ťažisko T prierezu.
 - Strednica je hladká a spojitá krivka konečnej dĺžky.
 - Priečny prierez je spojitá jedno alebo viacnásobne súvislá oblasť.
 - Dĺžka strednice je rádovo minimálna taká veľká ako najväčší rozmer priečného prierezu.
- Väzbové a zaťažovacie predpoklady
 - Väzby obmedzujú len posuvy a uhly natočenia strednice.
 - Zaťaženie prútu je sústredené na strednici.
- Deformačné predpoklady
 - Deformovaná strednica zostáva hladkou a spojitou krivkou.
 - Priečne prierezy si pri deformácii zachovávajú rovinnosť a kolmosť voči deformovanej strednici.
- Napätostné predpoklady
 - napätosť prútu je určená normálovým a šmykovým napätím v priečnom priereze, táto napätosť sa nazýva prútová. [1]

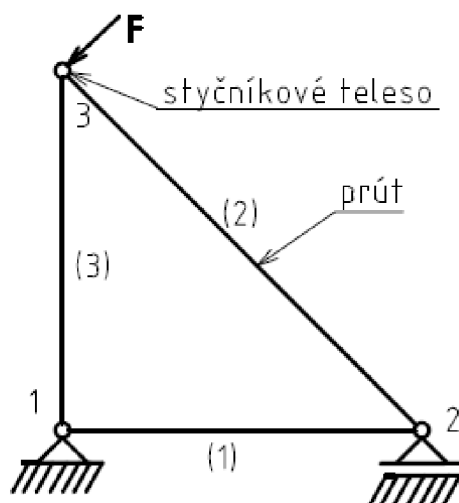
V tejto práci sa budú vyskytovať len prúty priame, teda so strednicou popísateľnou ako úsečka v nedeformovanom a deformovanom stave. Prúty budú namáhané len na ťah a tlak, a teda napätie vznikajúce prúte bude len normálové.

3.2 Prútové sústavy

Prútové sústavy riešené v tejto práci sú najjednoduchší model priehradových a prútových konštrukcií. Sú vymedzené nasledujúcimi predpokladmi:

- Väzby medzi prútmi sú tvorené sférickými a rotačnými kinematickými dvojicami typu NNTN.
- Prútová sústava je tvorená prútmi a styčnickovými telesami. Prúty sú tvorené styčnickovými telesami, pričom stredy kinematických dvojíc splývajú a tento bod sa označuje ako styčník.
- Vonkajšie sily pôsobiace na prútovú sústavu sú sústredené v styčníkoch.
- K základnému telesu je sústava uložená prostredníctvom styčnickov.

- Každý prút je uložený nepohyblivo a je prostredníctvom styčnickových telies viazaný aspoň ku dvom ďalším prútom. Sústava tvorí prútové teleso. [2]



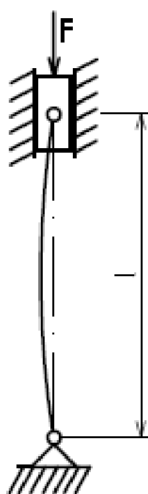
Obr 3.1 Prútová sústava

3.3 Vzperná stabilita prútu

Pri namáhaní dlhého priameho prútu tlakovou silou totožnou so strednicou prútu dojde pri dosiahnutí určitej kritickej sily k strate vzpernej stability. Pre túto silu je možné odvodiť vzťah:

$$F_{krit} = \frac{\alpha \cdot \pi^2 \cdot E \cdot J}{l^2}$$

kde F_{krit} je Eulerova kritická sila a α je súčiniteľ závislý na uložení prútu. Pre prút s rotačnými väzbami platí $\alpha = 1$. [3]



Obr. 3.2 Prút s rotačnými väzbami

3.3 Možnosti riešenia prúťových sústav

V priebehu histórie bolo odvodených a používaných množstvo spôsobov riešení prúťových sústav. Medzi tieto metódy patria:

- Metóda styčnicková. Spočíva v zostavovaní rovníc statickej rovnováhy pre styčníky prúťovej sústavy. Z týchto rovníc sa získajú sily, ktoré sú zároveň aj normálové výsledné vnútorné účinky prúťov sústavy. Z nich sa spočíta deformácia a napätosť sústavy. Samotné použitie styčnikovej metódy je obmedzené statickou určitosťou prúťovej sústavy. V prípade statickej neurčitosti treba riešenie kombinovať napríklad s použitím castiglianovej vety.
- Cremonova metóda. Je to vlastne varianta styčnikovej metódy vo forme grafického riešenia.
- Priesečná metóda (Ritterova). Zmysel riešenia spočíva v myslenom reze prúťovej sústavy takým spôsobom, že pre "odrezanú" časť sa zostavia rovnice silovej a momentovej rovnováhy. Z toho vyplýva, že rez by mal byť vedený tak, aby bola sústava rovníc pre túto časť prúťovej sústavy riešiteľná.
- Medzi ďalšie metódy patria: metóda momentových kĺbov, metóda myslených kĺbov, metóda neurčitej mierky, metóda náhradných prúťov, metóda silová a deformačná, metóda konečných prvkov [40]

3.4 Teoretický úvod do MKP

3.4.1 Základné rovnice obecnej pružnosti

Základné rovnice obecnej pružnosti sa používajú na riešenie takzvanej priamej úlohy pružnosti, ktorú je možno formulovať nasledovne:

"Pre dané teleso so známou geometriou, materiálom, zatažením a väzbami s okolím určite jeho deformáciu a napätosť." [5]

Rovnice obecnej pružnosti sú parciálne diferenciálne rovnice, ktoré navzájom uvádzajú do vzťahu napätosť a deformáciu, konkrétne:

- Tri posuvy u, v, w
- Šesť pretvorení $\epsilon_x, \epsilon_y, \epsilon_z, \gamma_{xy}, \gamma_{yz}, \gamma_{zx}$
- Šesť napätí $\sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{yz}, \tau_{zx}$

Základné rovnice obecnej pružnosti sa delia na rovnice rovnováhy, geometrické rovnice a konštitutívne vzťahy.

Rovnice rovnováhy

Sú to rovnice rovnováhy (vnútorného) elementárneho prvku, na ktorý pôsobia zložky napätia a merné objemové sily, konkrétne rovnice rovnováhy pre smery x, y a z.

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} + o_x = 0 \quad (3.1)$$

$$\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{yz}}{\partial z} + o_y = 0 \quad (3.2)$$

$$\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma_z}{\partial z} + o_z = 0 \quad (3.3)$$

Rovnice geometrické

Sú to vzťahy medzi zložkami posuvov a pretvorení.

$$\begin{aligned} \varepsilon_x &= \frac{\partial u}{\partial x} & \varepsilon_y &= \frac{\partial v}{\partial y} & \varepsilon_z &= \frac{\partial w}{\partial z} \\ \gamma_{xy} &= \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & \gamma_{yz} &= \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} & \gamma_{zx} &= \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \end{aligned} \quad (3.4)$$

Konštitutívne vzťahy

Určujú vzťah medzi napätosťou a deformáciou. Inak sa nazývajú zobecnený hookeov zákon.

$$\begin{aligned} \varepsilon_x &= \frac{1}{E} [\sigma_x - \mu(\sigma_y + \sigma_z)] \\ \varepsilon_y &= \frac{1}{E} [\sigma_y - \mu(\sigma_x + \sigma_z)] \\ \varepsilon_z &= \frac{1}{E} [\sigma_z - \mu(\sigma_x + \sigma_y)] \\ \gamma_{xy} &= \frac{1}{G} \tau_{xy} & \gamma_{yz} &= \frac{1}{G} \tau_{yz} & \gamma_{zx} &= \frac{1}{G} \tau_{zx} \end{aligned} \quad (3.5)$$

kde G je modul pružnosti v šmyku a je daná vzťahom

$$G = \frac{E}{2(1 + \mu)} \quad (3.6)$$

Základné rovnice obecnej pružnosti musia byť doplnené okrajovými podmienkami geometrickými a silovými [5].

3.4.2 Metóda konečných prvkov z hľadiska riešenia priamej úlohy pružnosti

V prístupe ku riešeniu priamej úlohy pružnosti možno MKP definovať:

Z hľadiska matematickej formulácie problému ako úlohu variačnú. To znamená nájsť hodnotu energie telesa takú, že dosahuje určitej extrémnej hodnoty. V prípade MKP sa využíva Lagrangeov variačný princíp. Z hľadiska výberu nezávislých funkcií pružnosti ide o metódu deformačnú, teda ako neznáme funkcie vyberáme zložky posuvov. Z hľadiska realizácie riešenia je MKP metóda numerická. [5]

3.4.3 Lagrangeov variačný princíp

Základom pre rovnice deformačnej varianty MKP je Lagrangeov variačný princíp. Dá sa formulovať takto: “Medzi všetkými funkciami posuvov, ktoré zachovávajú spojitosť telesa a spĺňajú geometrické okrajové podmienky, sa realizujú tie, ktoré udelia celkovej potenciálnej energii Π stacionárnu hodnotu.” [5]

Celková potenciálna energia telesa sa dá vyjadriť ako

$$\Pi = W - P \quad (3.7)$$

Kde W je energia napätosti telesa

$$W = \frac{1}{2} \int_{\Omega} \boldsymbol{\sigma}^T \boldsymbol{\varepsilon} dV \quad (3.8)$$

a P je potenciál vonkajšieho zaťaženia.

$$P = \int_{\Omega} \mathbf{u}^T \cdot \mathbf{o} \cdot dV + \int_{\Gamma_p} \mathbf{u}^T \cdot \mathbf{p} \cdot dS \quad (3.9)$$

Jednotlivé členy a symboly majú význam:

- (vektor) posuvov $\mathbf{u}^T = (u, v, w)$
- pretvorení $\boldsymbol{\varepsilon}^T = (\varepsilon_x, \varepsilon_y, \varepsilon_z, \gamma_{xy}, \gamma_{yz}, \gamma_{zx})$
- napätí $\boldsymbol{\sigma}^T = (\sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{yz}, \tau_{zx})$
- objemového zaťaženia $\mathbf{o}^T = (o_x, o_y, o_z)$
- plošného zaťaženia $\mathbf{p}^T = (p_x, p_y, p_z)$
- riešené teleso Ω
- plocha riešeného telesa Γ_p

3.4.4 Diskretizácia v MKP

V prípade MKP sa riešená oblasť rozdelí na konečný počet prvkov a hľadajú sa posuvy konečného počtu uzlov týchto prvkov. Posuvy celej oblasti sa potom aproximujú pomocou bázových funkcií. V prípade použitia MKP na riešenie prútových sústav nie je potrebné problematiku bázových funkcií ďalej rozoberať.

3.4.5 Základná rovnica MKP.

Pre celkovú potenciálnu energiu je možné pre diskretizovanú oblasť odvodiť vzťah

$$\Pi = \frac{1}{2} \mathbf{U}^T \mathbf{K} \mathbf{U} - \mathbf{U}^T \mathbf{F} \quad (3.10)$$

kde \mathbf{U} predstavuje stĺpcovú maticu neznámych deformačných parametrov konečného počtu uzlov, \mathbf{K} je matica tuhosti sústavy a \mathbf{F} je stĺpcová matica zaťaženia sústavy. Podrobnejší výklad k týmto maticiam sa nachádza v ďalšej časti textu. Z lagrangeovho variačného princípu vyplýva, že pre získanie vektoru posuvov treba nájsť pre celkovú potenciálnu energiu stacionárnu hodnotu, v tomto prípade minimum. Pre nájdenie minima je potrebné potenciál zderivovať podľa neznámeho vektoru posuvov

$$\frac{\partial \Pi}{\partial \mathbf{U}} = 0$$

Získaný výraz má tvar

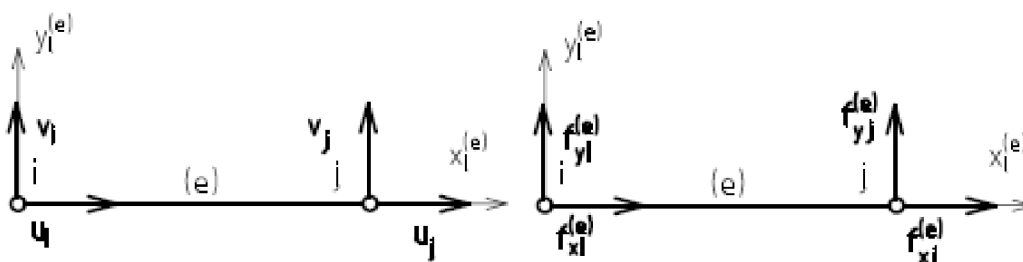
$$\mathbf{K} \cdot \mathbf{U} = \mathbf{F} \quad (3.11)$$

a nazýva sa základná rovnica MKP [5].

Matica \mathbf{K} je singulárna, a teda sústava nemá riešenie. Preto je treba predpísať okrajové podmienky tak, aby sa zabránilo pohybu sústavy v priestore. Tým sa z matice tuhosti odstráni riadky a stĺpce s indexmi predpísaných nulových posuvov v matici deformačných parametrov (matica bude regulárna).

3.4.6 Prvok prenášajúci osovú zaťaženie v lokálnom súradnicovom systéme

Matica tuhosti prvku



Obr 3.3 Prvok prenášajúci osovú zaťaženie

Na obrázku 3.3 je prvok (e) s lokálnym súradnicovým systémom so znázorneným zaťažením a štyrmi deformačnými parametrami. Jeho maticu tuhosti je možno si predstaviť ako tuhosť prvku v rôznych smeroch, v tomto prípade v smeroch lokálnych súradnicových osí.

Prvková matica tuhosti v 2D lokálnom súradnicovom systéme má podobu [5]

$$\mathbf{k}_l = \frac{E \cdot S}{l} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.12)$$

kde E je youngov modul pružnosti, S je plocha priečného prierezu a l je dĺžka prvku. Prvý a tretí riadok tejto matice má význam tuhosti v smere lokálnej osi x . Druhý a štvrtý riadok obsahuje len nuly, čo naznačuje, že v smere lokálnej osi y má prvok nulovú tuhosť, čo zodpovedá predstave prútového prvku prenášajúceho len osovú silu.

Matica zaťaženia prvku

Prvková matica zaťaženia v 2D lokálnom súradnicovom systéme má tvar stĺpcovej matice, ktorej prvky predstavujú zložky zaťaženia prvku v hraničných uzloch.

$$\mathbf{f}_l^T = (f_{ixl}, f_{iy l}, f_{jxl}, f_{jyl})$$

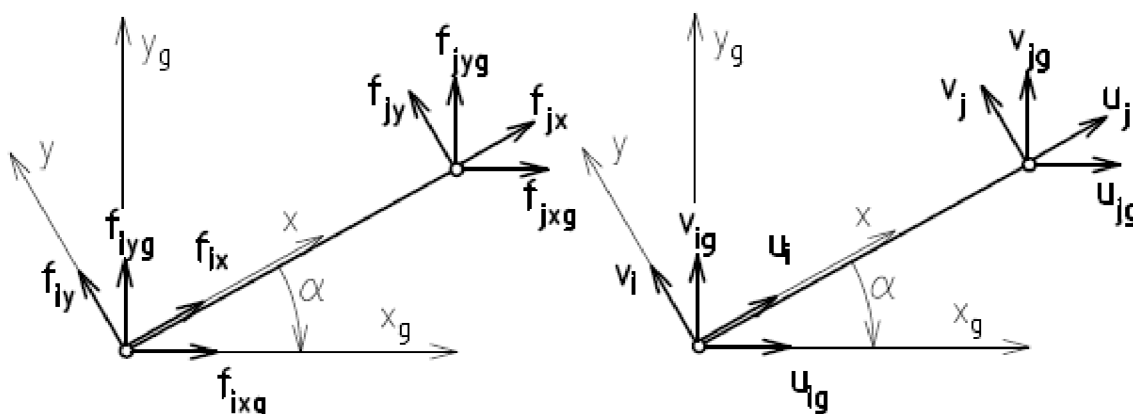
Matica deformačných parametrov

Matica deformačných parametrov v 2D lokálnom súradnicovom systéme má tvar stĺpcovej matice, ktorá vyjadruje možné posuvy prvku v rovine v smeroch osí lokálneho súradnicového systému

$$\mathbf{\delta}_l^T = (u_{il}, v_{il}, u_{jl}, v_{jl})$$

3.4.7 Globálny a lokálny súradnicový systém

Pre riešenie úloh v 2D a 3D je treba matice tuhosti, zaťaženia a deformačných parametrov jednotlivých prvkov transformovať do takzvaného globálneho súradnicového systému pomocou transformačných vzťahov uvedených nižšie.

Prútový prvok v 2D

Obr.3.4 Vzťahy medzi globálnym a lokálnym súr. systémom

Pre transformáciu do globálneho súradnicového systému platia vzťahy

$$\delta_l = T \cdot \delta \quad (3.13)$$

$$f_l = T \cdot f \quad (3.14)$$

Kde T je transformačná matica

$$T = \begin{pmatrix} \cos \alpha & \sin \alpha & 0 & 0 \\ -\sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & \cos \alpha & \sin \alpha \\ 0 & 0 & -\sin \alpha & \cos \alpha \end{pmatrix} \quad (3.15)$$

Uhol α je uhol medzi lokálnou a globálnou osou x (v najbližšom smere od lokálnej ku globálnej).

Po úpravách možno pre prvkovú maticu tuhosti písať [5]

$$\mathbf{k} = T^T \cdot \mathbf{k}_l \cdot T \quad (3.16)$$

Podoba prvkovej matice tuhosti v globálnom súradnicovom systéme je teda [39]

$$\mathbf{k} = \frac{E \cdot S}{l} \cdot \begin{pmatrix} c^2 & sc & -c^2 & -sc \\ sc & s^2 & -sc & -s^2 \\ -c^2 & -sc & c^2 & sc \\ -sc & -s^2 & sc & s^2 \end{pmatrix} \quad (3.17)$$

kde

$$c = \cos \alpha$$

$$s = \sin \alpha$$

V prípade prútových sústav bude potrebné pre zistenie normálového napätia v prúte spočítať deformáciu prvku v smere lokálnej osi x. Na to bude slúžiť transformačná matica \mathbf{T}_u . Platí

$$d = u_{jl} - u_{il} = (-c, -s, c, s) \cdot \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \end{pmatrix} \quad (3.18)$$

a teda

$$\mathbf{T}_u = (-c, -s, c, s) \quad (3.19)$$

Osovo zaťažovaný prvok v 3D

Matica tuhosti v lokálnom súradnicovom systéme je podobná predchádzajúcemu prípadu, akurát je rozšírená o 2 riadky a stĺpce, odpovedajúce pridanému deformačnému parametru na ose z [5]

$$\mathbf{k}_l = \frac{E \cdot S}{l} \begin{pmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.20)$$

Matica tuhosti v globálnych súradniciach má podobu

$$\mathbf{k} = \frac{ES}{l} \cdot \begin{pmatrix} c_x^2 & c_{xy} & c_{xz} & -c_x^2 & -c_{xy} & -c_{xz} \\ c_{xy} & c_y^2 & c_{yz} & -c_{xy} & -c_y^2 & -c_{yz} \\ c_{xz} & c_{yz} & c_z^2 & -c_{xz} & -c_{yz} & -c_z^2 \\ -c_x^2 & -c_{xy} & -c_{xz} & c_x^2 & c_{xy} & c_{xz} \\ -c_{xy} & -c_y^2 & -c_{yz} & c_{xy} & c_y^2 & c_{yz} \\ -c_{xz} & -c_{yz} & -c_z^2 & c_{xz} & c_{yz} & c_z^2 \end{pmatrix} \quad (3.21)$$

kde

$$c_x = \frac{x_2 - x_1}{l}, \quad c_y = \frac{y_2 - y_1}{l}, \quad c_z = \frac{z_2 - z_1}{l}$$

$$c_{xy} = c_x \cdot c_y, \quad c_{yz} = c_y \cdot c_z, \quad c_{xz} = c_x \cdot c_z$$

l má význam dĺžky prvku (rozdiel súradníc uzlov prvku na lokálnej osi x) a x,y,z, sú globálne súradnice uzlov. [4]

Matica deformačných parametrov prvku je

$$\boldsymbol{\delta}^T = (u_i, v_i, w_i, u_j, v_j, w_j)$$

Prvková matica zaťaženia má tvar

$$\mathbf{f}^T = (f_{ix}, f_{iy}, f_{iz}, f_{jx}, f_{jy}, f_{jz})$$

Transformačná matica pre zistenie deformácie prvku v smere lokálnej osi x je

$$\mathbf{T}_{u3D} = (-c_x, -c_y, -c_z, c_x, c_y, c_z)$$

3.4.8 Skladanie matíc K,U,F pre celkovú sústavu

Celková matica deformačných parametrov v globálnych súradniciach je podobná maticiam deformačných parametrov prvkov, akurát má rozmer odpovedajúci počtu deformačných parametrov celej sústavy.

$$\mathbf{U} = \begin{pmatrix} u_1 \\ \vdots \\ \vdots \\ \vdots \\ v_n \end{pmatrix} \quad (3.22)$$

Celková matica vonkajšieho silového zaťaženia (bez bližšieho odvodenia) je stĺpcová matica rozmermi zodpovedajúca celkovej matici deformačných parametrov, vytvorená súčtom prvkových matíc zaťaženia. Jej prvky zodpovedajú zaťaženiu uzlov v smeroch os globálneho súradnicového systému.

$$\mathbf{F} = \sum_{i=1}^m \mathbf{f}_i = \begin{pmatrix} f_{1x} \\ \vdots \\ \vdots \\ \vdots \\ f_{ny} \end{pmatrix} \quad (3.23)$$

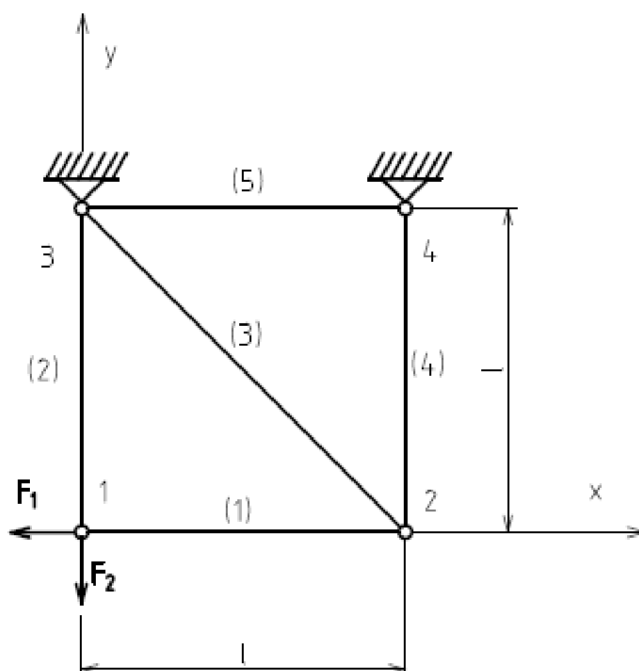
Celková matica tuhosti sústavy je súčtom matíc tuhosti prvkov. Spôsoby zostavovania matíc zaťaženia, deformačných parametrov a vonkajšieho zaťaženia budú rozobraté v ilustračnom príklade.

$$\mathbf{K} = \sum_{i=1}^m \mathbf{k}_i \quad (3.24)$$

Symboly n, m odpovedajú počtu uzlov, respektíve prvkov [39].

3.4.9 Ilustračný príklad:

Pre danú prútovú sústavu je treba zistiť posuvy uzlov, normálové sily napätia v prútoch a sily vo väzbách.



Obr. 3.5 zadanie ilustračnej úlohy

$$E = 210 \cdot 10^3 \text{ MPa}$$

$$S = 314 \text{ mm}^2$$

$$l = 1000 \text{ mm}$$

$$f_{1x} = -\text{abs}(F_1) = -50000 \text{ N}$$

$$f_{1y} = -\text{abs}(F_2) = -40000 \text{ N}$$

statický rozbor (z hľadiska MKP nie je podstatný, ale je uvedený pre kompletnosť)

Vnútorne hľadisko:

Jedá sa o prútovú sústavu zloženú zo 4 styčníc (uzly) a 5 prútov (prvky).

$$p = 5$$

$$k = 4$$

$$p = 2k - 3$$

$$5 = 2 \cdot 4 - 3$$

Úloha je vnútorne staticky určitá.

Vonkajšie hľadisko:

Sú k dispozícii 3 podmienky statickej rovnováhy, počet neznámych parametrov vonkajších stykových síl je 4

$$v = 3$$

$$\mu_A = 4 = \{f_{3x}, f_{3y}, f_{4x}, f_{4y}\}$$

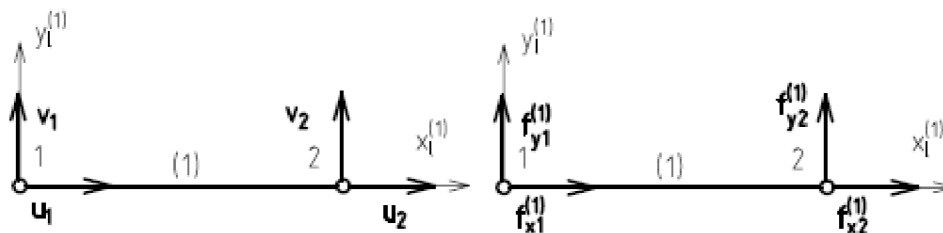
$$s_e = \mu_A - v = 4 - 3 = 1$$

Úloha je 1 krát externe staticky neurčitá.

Postup

Pre jednotlivé prvky sa zostavia prvkové matice tuhosti v lokálnych súradniciach, tie sa potom transformujú do globálnych súradnic. Tieto sa potom poskladajú do celkovej (globálnej) matice tuhosti. Spôsob poskladania riadia indexy i riadkov a stĺpcov prvku. Podobným spôsobom sa poskladá aj globálna matica vonkajšieho zaťaženia. Okrajové podmienky (deformačné) sú:

$$u_3 = v_3 = u_4 = v_4 = 0$$

Prvok (1)

Obr. 3.6 prvok 1

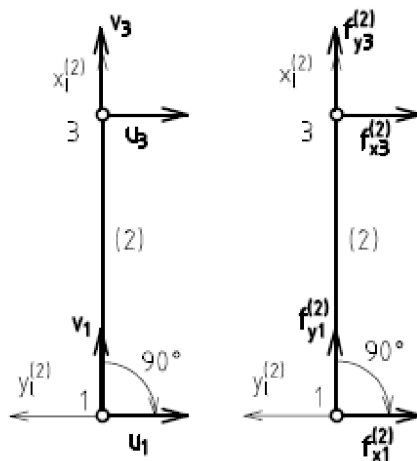
$$\alpha = 0^\circ, \quad i = (1, 2, 5, 6)$$

$$\mathbf{k}_1 = \frac{E \cdot S}{l} \begin{pmatrix} c^2 & c \cdot s & -c^2 & -c \cdot s \\ c \cdot s & s^2 & -c \cdot s & -s^2 \\ -c^2 & -c \cdot s & c^2 & c \cdot s \\ -c \cdot s & -s^2 & c \cdot s & s^2 \end{pmatrix} = \begin{pmatrix} 65940 & 0 & -65940 & 0 \\ 0 & 0 & 0 & 0 \\ -65940 & 0 & 65940 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} N/mm$$

po formálnom rozšírení takom, aby matica tuhosti odpovedala rozmerom v globálnej matici zaťaženia a deformačných parametrov

$$\mathbf{k}_1 = \begin{pmatrix} 65940 & 0 & 0 & 0 & -65940 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -65940 & 0 & 0 & 0 & 65940 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} N/mm$$

Prvok (2)



Obr.3.7 prvok 2

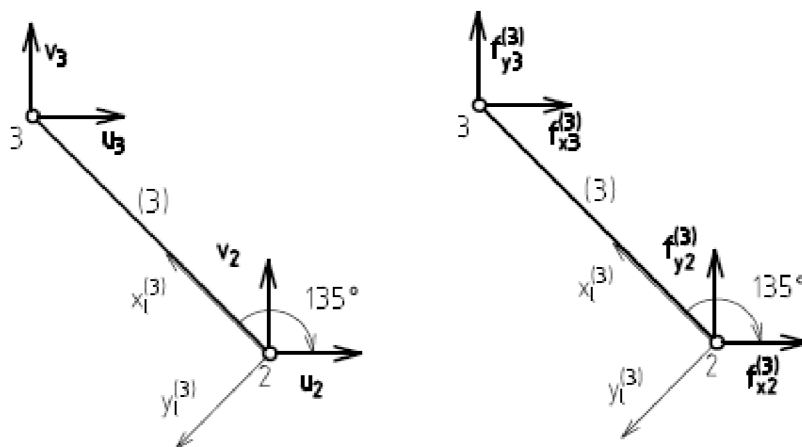
$\alpha = 90^\circ, i = (1, 2, 3, 4)$

$$\mathbf{k}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 65940 & 0 & -65940 \\ 0 & 0 & 0 & 0 \\ 0 & -65940 & 0 & 65940 \end{pmatrix} N/mm$$

po formálnom rozšírení

$$\mathbf{k}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 65940 & 0 & -65940 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -65940 & 0 & 65940 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} N/mm$$

Prvok (3)



Obr. 3.8 prvok 3

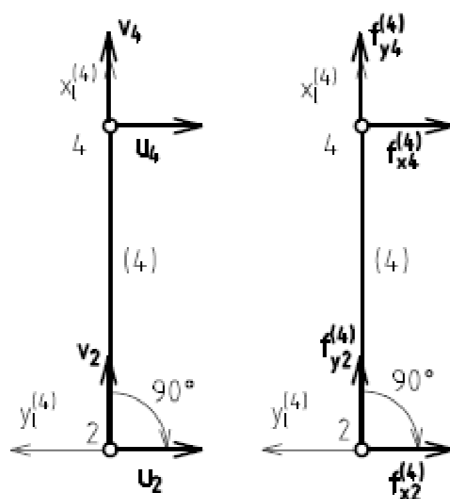
$\alpha = 135^\circ, i = (3, 4, 5, 6)$

$$\mathbf{k}_3 = \begin{pmatrix} 23313,311 & -23313,311 & -23313,311 & 23313,311 \\ -23313,311 & 23313,311 & 23313,311 & -23313,311 \\ -23313,311 & 23313,311 & 23313,311 & -23313,311 \\ 23313,311 & -23313,311 & -23313,311 & 23313,311 \end{pmatrix} N/mm$$

po formálnom rozšírení (a=23313,311)

$$\mathbf{k}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a & -a & -a & a & 0 & 0 \\ 0 & 0 & -a & a & a & -a & 0 & 0 \\ 0 & 0 & -a & a & a & -a & 0 & 0 \\ 0 & 0 & a & -a & -a & a & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} N/mm$$

Prvok (4)



Obr. 3.9 prvok 4

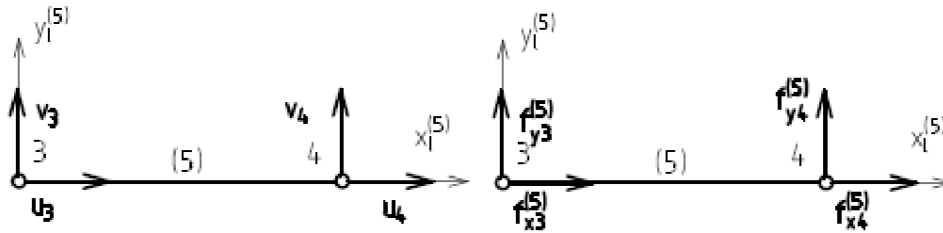
$\alpha = 90^\circ$, $i = (3, 4, 7, 8)$

$$\mathbf{k}_4 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 65940 & 0 & -65940 \\ 0 & 0 & 0 & 0 \\ 0 & -65940 & 0 & 65940 \end{pmatrix} N/mm$$

po formálnom rozšírení

$$\mathbf{k}_4 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 65940 & 0 & 0 & 0 & -65940 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -65940 & 0 & 0 & 0 & 65940 \end{pmatrix} N/mm$$

Prvok (5):



Obr. 3.10 prvok 5

$\alpha = 0^\circ$, $i = (5, 6, 7, 8)$

$$\mathbf{k}_5 = \begin{pmatrix} 65940 & 0 & -65940 & 0 \\ 0 & 0 & 0 & 0 \\ -65940 & 0 & 65940 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} N/mm$$

po formálnom rozšírení

$$\mathbf{k}_5 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 65940 & 0 & -65940 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -65940 & 0 & 65940 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} N/mm$$

Celková matica tuhosti sústavy je teda:

$$\mathbf{K} = \mathbf{k}_1 + \mathbf{k}_2 + \mathbf{k}_3 + \mathbf{k}_4 + \mathbf{k}_5$$

Kde a=65940, b=23313,31, c=89253,311

$$\mathbf{K} = \begin{pmatrix} a & 0 & -a & 0 & 0 & 0 & 0 & 0 \\ 0 & a & 0 & 0 & 0 & -a & 0 & 0 \\ -a & 0 & c & -b & -b & b & 0 & 0 \\ 0 & 0 & -b & c & b & -b & 0 & -a \\ 0 & 0 & -b & b & c & -b & -a & 0 \\ 0 & -a & b & -b & -b & c & 0 & 0 \\ 0 & 0 & 0 & 0 & -a & 0 & a & 0 \\ 0 & 0 & 0 & -a & 0 & 0 & 0 & a \end{pmatrix} N/mm$$

Sústava $\mathbf{K} \cdot \mathbf{U} = \mathbf{F}$ pred aplikáciou okrajových podmienok

$$\begin{pmatrix} a & \dots & 0 \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & \ddots & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ 0 & \dots & a \end{pmatrix} N/mm \cdot \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \\ u_4 \\ v_4 \end{pmatrix} mm = \begin{pmatrix} -50000 \\ -40000 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} N$$

Teraz je potrebné aplikovať okrajové podmienky, sústava dostane tvar

$$\begin{pmatrix} 65940 & 0 & -65940 & 0 \\ 0 & 65940 & 0 & 0 \\ -65940 & 0 & 89253,311 & -23313,311 \\ 0 & 0 & -23313,311 & 89253,311 \end{pmatrix} N/mm \cdot \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \end{pmatrix} mm = \begin{pmatrix} -50000 \\ -40000 \\ 0 \\ 0 \end{pmatrix} N$$

Vektor získaných posuvov uzlov v globálnych súradniciach (po doplnení predpísaných nulových posuvov je:

$$\mathbf{U} = \begin{pmatrix} -3,661 \\ -0,607 \\ -2,903 \\ -0,758 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} mm$$

Keď týmto vektorom prenásobíme globálnu maticu tuhosti, dostaneme vektor vonkajších síl. Okrem známych síl sú v ňom obsiahnuté aj sily vo väzbách (vyznačené hrubo).

$$\mathbf{K} \cdot \mathbf{U} = \mathbf{F} = \begin{pmatrix} -50000 \\ -40000 \\ 0 \\ 0 \\ \mathbf{50000} \\ -\mathbf{10000} \\ 0 \\ \mathbf{50000} \end{pmatrix} N$$

V ďalšej časti je potrebné posuvy z globálnych súradníc transformovať späť do lokálnych, aby sa pre jednotlivé prvky získali hodnoty predĺženia/skrátenia. Z nich sa získajú hodnoty normálových síl a normálových napätí v prvkoch.

Prvok 1:

$$\boldsymbol{\delta}_{(1)} = \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \end{pmatrix} = \begin{pmatrix} -3,661 \\ -0,607 \\ -2,903 \\ -0,758 \end{pmatrix} mm$$

$$d_{(1)} = \mathbf{T}_u \cdot \boldsymbol{\delta}_{(1)} = 0,758 mm$$

$$\sigma_{(1)} = \frac{E \cdot d_{(1)}}{l} = 159,236 MPa$$

$$N_{(1)} = \frac{\sigma_{(1)}}{S} = 50000 N$$

Obdobným spôsobom sa získajú deformácie, sily a napätia pre ostatné prvky.

Tab. 3.1 Hodnoty získané pre ostatné prvky

prvok	d [mm]	N [kN]	σ [MPa]
(2)	0,607	40	127,389
(3)	-1,517	-70,711	-225,193
(4)	0,758	50	159,236
(5)	0	0	0

4. Program v Pythone a kontrola výsledkov v systéme ANSYS

4.1 Popis Programu

Pre vytvorenie programu je použitý jazyk Python 2.7, konkrétne jeho distribúcia Anaconda. V Anaconde samotnej je zvolené prostredie Spyder. Anaconda bola zvolená hlavne preto, že má v sebe zabudované všetky potrebné knižnice a moduly.

Do samotného programu sa zadajú vstupné parametre: súradnice styčníc, prútoch, silové zaťaženie, deformačné podmienky. Program na základe vstupných údajov o prútoch a styčníkoch vytvorí maticu tuhosti sústavy. Ďalej dopočíta neznáme posuvy styčníc a z nich ostatné parametre – predovšetkým napätosť v prútoch, a väzbové sily. Následne sa celá sústava vykreslí v deformovanom a nedeformovanom tvare. Niektoré časti kódu:

Vytvorenie matice tuhosti

```
for i in range(0, len(elements)):
    #x-ova dlzka prvku
    xLength=nodeCoordinates[elements[i,3]-1,0]-nodeCoordinates[elements[i,2]-1,0]
    #y-ova dlzka prvku
    yLength=nodeCoordinates[elements[i,3]-1,1]-nodeCoordinates[elements[i,2]-1,1]
    #celkova dlzka prvku
    elementLength=np.sqrt((xLength)**2+(yLength)**2)
    #kosinus
    c=xLength/elementLength
    #sinus
    s=yLength/elementLength
    #prvkova matica tuhosti
    kElement=np.array([[c*c,c*s,-c*c,-c*s],
                       [c*s,s*s,-c*s,-s*s],
                       [-c*c,-c*s,c*c,c*s],
                       [-c*s,-s*s,c*s,s*s]])*elements[i,0]*elements[i,1]/elementLength
    #riadky a stlpce v globalnej matici tuhosti
    colNrow=np.array([[elements[i,2]*2-2,elements[i,2]*2-1,
                       elements[i,3]*2-2,elements[i,3]*2-1]])
    #indexy poli musia byt integery
    colNrow=colNrow.astype(int)
    #skladanie do globalnej matice tuhosti
    globStifMat[colNrow,colNrow.T]+=kElement
```

Získanie normálových síl a napätí

```
for i in range(0,len(elements)):
    xLength=nodeCoordinates[elements[i,3]-1,0]-nodeCoordinates[elements[i,2]-1,0]
    yLength=nodeCoordinates[elements[i,3]-1,1]-nodeCoordinates[elements[i,2]-1,1]
    elementLength=np.sqrt((xLength)**2+(yLength)**2)
    c=xLength/elementLength
    s=yLength/elementLength
    #transformacna matica
    transfMat=np.array([-c,-s,c,s])
    #indexy prvkovych posuvov z celkoveho vektora posuvov
    dispRow=np.array([[elements[i,2]*2-2,elements[i,2]*2-1,
```



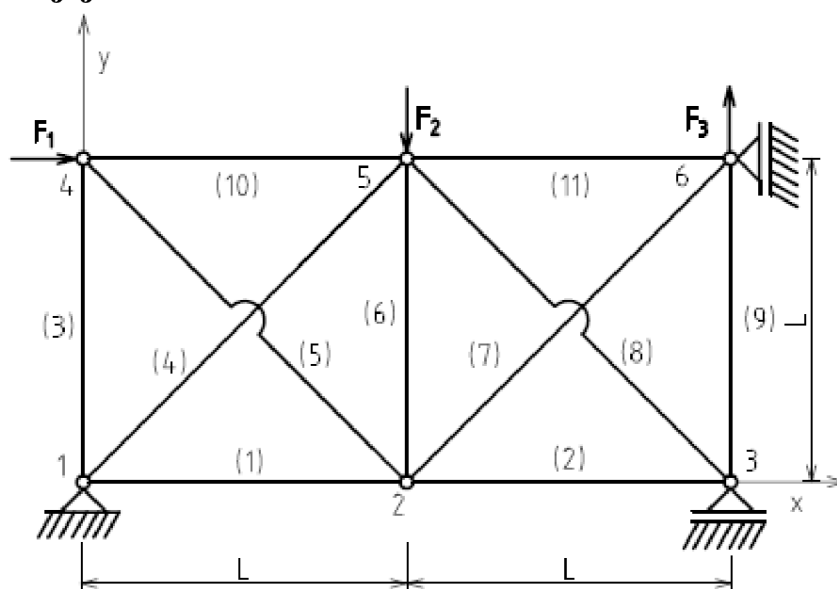
```

elements[i,3]*2-2,elements[i,3]*2-1]])
dispRow=dispRow.astype(int)
#uzlove posuvy prvku v globalnych suradniciach
elementNodesDispGlob=np.array([nodeDisplacements[dispRow]])
#deformacia prvku (rozdiel xovych posuvov v LS) [mm]
elementDispLoc[i]=np.dot(transfMat,np.reshape(elementNodesDispGlob, (4)))
#normalova sila pre prvok
#F=E*S*d/l
#N=Mpa*mm2*mm/mm
#-->F [N]
normalForces[i]=elements[i,0]*elements[i,1]*(elementDispLoc[i])/elementLength
#vpisanie normalovych napati do vektoru napati pre vsetky prvky
#sigma [MPa]
normalStress[i]=normalForces[i]/elements[i,1]

```

Spôsob zostavovania matice tuhosti a získania napätí je podobný ako v [4].

4.2 2D úloha a jej overenie



Obr. 4.2 zadanie 2D úlohy

$$F_1 = 50 \text{ kN}$$

$$F_2 = 50 \text{ kN}$$

$$F_3 = 50 \text{ kN}$$

$$L = 1 \text{ m}$$

$$E = 210 \text{ GPa}$$

$$S = \pi * \frac{30^2}{4} \text{ [mm}^2\text{]}$$

Fyzikálne rozmery jednotlivých veličín budú pre 2D aj 3D úlohu nasledovné:

- normálová sila [N]

- normálové napätie [MPa]
- uzlový posun [mm]

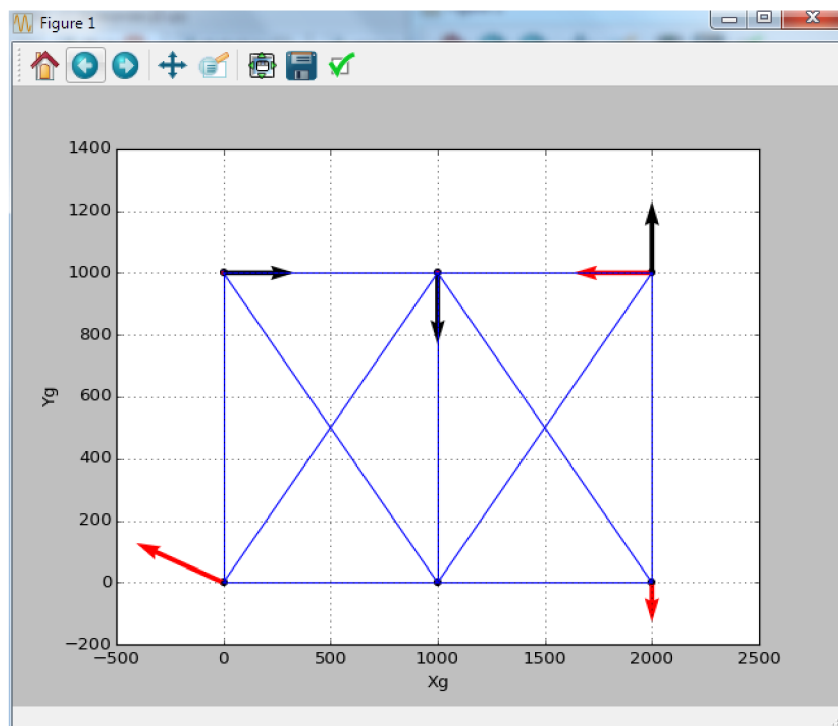
Výstup z prostredia Sypyder:

nodeDisplacementsRes			totalExtForceRes		
	0	1		0	1
0	0.000	0.000	0	-26237.583	11881.209
1	0.254	-0.376	1	0.000	0.000
2	0.391	0.000	2	0.000	-11881.209
3	0.619	-0.003	3	50000.000	0.000
4	0.280	-0.499	4	0.000	-50000.000
5	0.000	0.217	5	-23762.417	50000.000

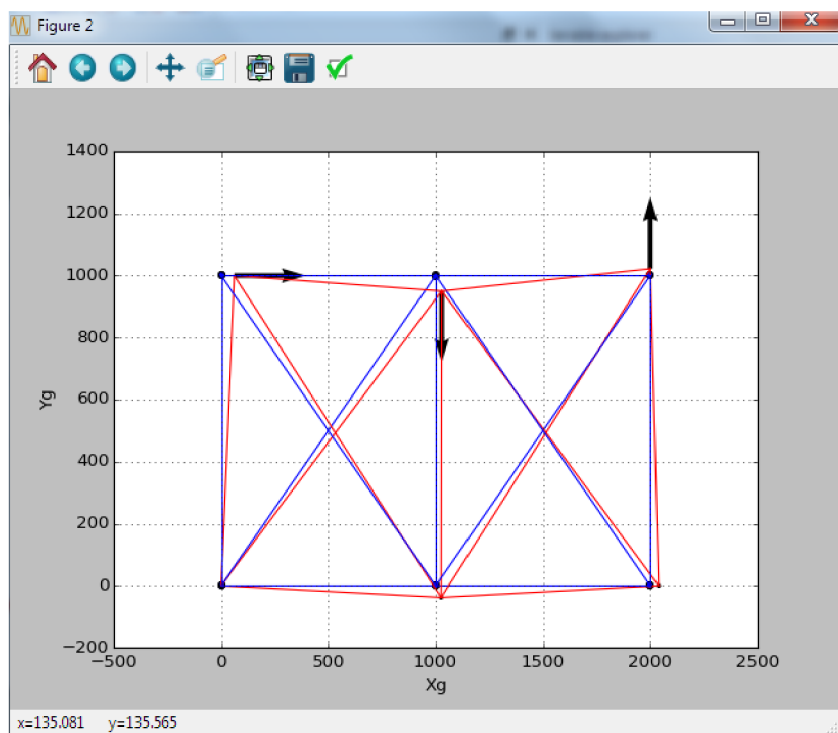
Obr.4.3 uzlové posuvy a vonkajšie zaťaženie uzlov

normalForces		normalStress	
	0		0
0	37711.093	0	53.350
1	20331.954	1	28.764
2	-407.699	2	-0.577
3	-16225.993	3	-22.955
4	576.573	4	0.816
5	-18194.536	5	-25.740
6	25154.387	6	35.586
7	-28753.725	7	-40.678
8	32213.162	8	45.572
9	-50407.699	9	-71.312
10	-41549.255	10	-58.780

Obr. 4.4 normálové sily a napätia v prvkoch



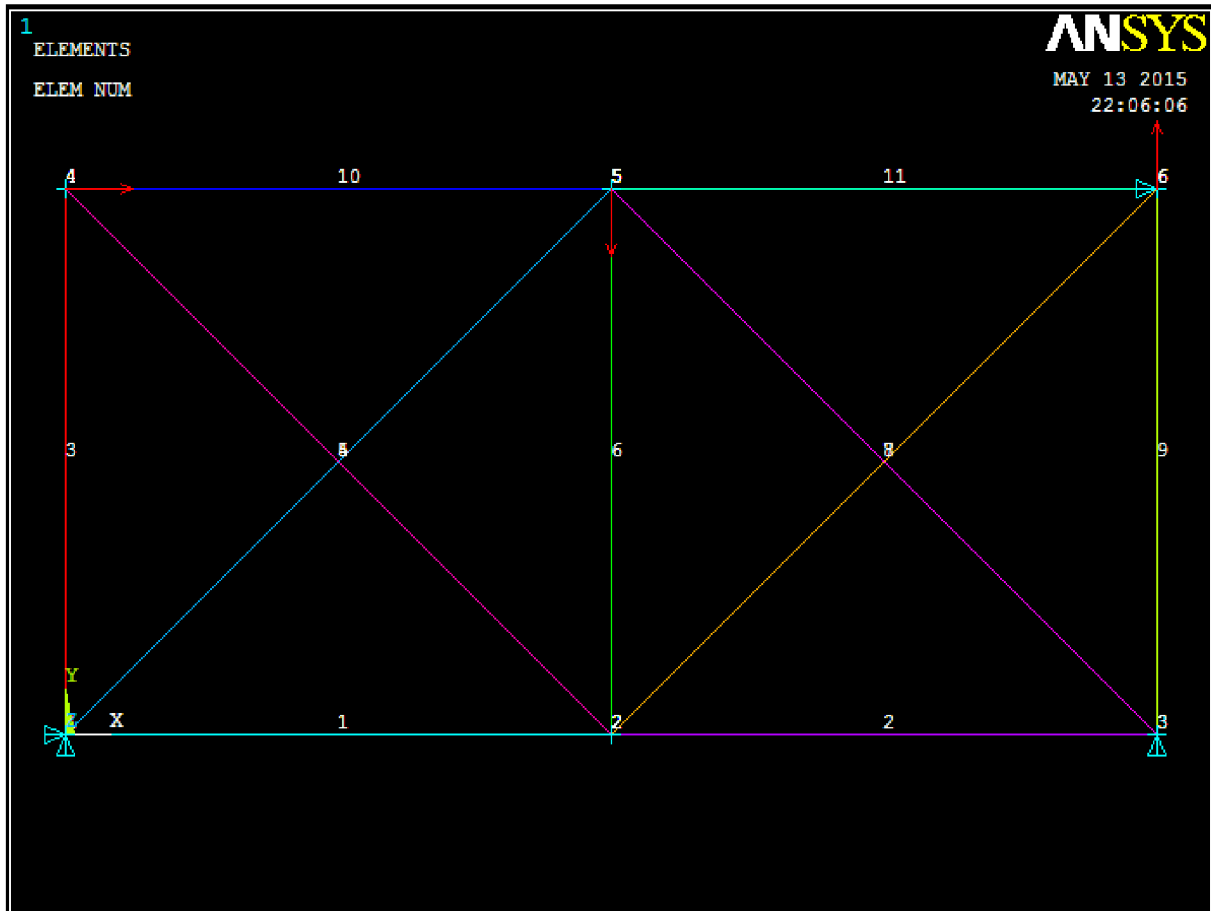
Obr. 4.5 nedeformovaná sústava, červená-reakcie, čierna-vonkajšie sily



Obr. 4.6 porovnanie deformovaného a nedeformovaného tvaru v mierke 100:1

Kontrola v ANSYSe:

Bol použitý ANSYS Mechanical APDL. Pre kontrolu bol použitý prvok LINK180, materiálový model linear elastic isotropic. Sú porovnávané hodnoty zaokrúhlené na 3 desatinné miesta.



Obr. 4.7 zadaná sústava v ANSYSe

PRNSOL Command

File

PRINT U NODAL SOLUTION PER NODE

**** POST1 NODAL DEGREE OF FREEDOM LISTING ****

LOAD STEP= 1 SUBSTEP= 1
TIME= 1.0000 LOAD CASE= 0

THE FOLLOWING DEGREE OF FREEDOM RESULTS ARE IN THE GLOBAL COORDINATE SYSTEM

NODE	UX	UY	UZ	USUM
1	0.0000	0.0000	0.0000	0.0000
2	0.25405	-0.37595	0.0000	0.45374
3	0.39102	0.0000	0.0000	0.39102
4	0.61949	-0.27466E-02	0.0000	0.61949
5	0.27991	-0.49853	0.0000	0.57173
6	0.0000	0.21701	0.0000	0.21701

MAXIMUM ABSOLUTE VALUES

NODE	4	5	0	4
VALUE	0.61949	-0.49853	0.0000	0.61949

Obr. 4.8 uzlové posuny

PRESOL Command

File

```

PRINT ELEM ELEMENT SOLUTION PER ELEMENT
**** POST1 ELEMENT SOLUTION LISTING ****
LOAD STEP= 1 SUBSTEP= 1
TIME= 1.0000 LOAD CASE= 0
EL= 1 NODES= 1 2 MAT= 1 XC, YC, ZC= 500.0 0.000 0.000 AREA= 706.86 LINK180
FORCE= 37711. STRESS= 53.350 EPEL= 0.25405E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 2 NODES= 2 3 MAT= 1 XC, YC, ZC= 1500. 0.000 0.000 AREA= 706.86 LINK180
FORCE= 20332. STRESS= 28.764 EPEL= 0.13697E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 3 NODES= 1 4 MAT= 1 XC, YC, ZC= 0.000 500.0 0.000 AREA= 706.86 LINK180
FORCE= -407.70 STRESS= -0.57678 EPEL= -0.27466E-05
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 4 NODES= 1 5 MAT= 1 XC, YC, ZC= 500.0 500.0 0.000 AREA= 706.86 LINK180
FORCE= -16226. STRESS= -22.955 EPEL= -0.10931E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 5 NODES= 2 4 MAT= 1 XC, YC, ZC= 500.0 500.0 0.000 AREA= 706.86 LINK180
FORCE= 576.57 STRESS= 0.81568 EPEL= 0.38842E-05
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 6 NODES= 2 5 MAT= 1 XC, YC, ZC= 1000. 500.0 0.000 AREA= 706.86 LINK180
FORCE= -18195. STRESS= -25.740 EPEL= -0.12257E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 7 NODES= 2 6 MAT= 1 XC, YC, ZC= 1500. 500.0 0.000 AREA= 706.86 LINK180
FORCE= 25154. STRESS= 35.586 EPEL= 0.16946E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 8 NODES= 3 5 MAT= 1 XC, YC, ZC= 1500. 500.0 0.000 AREA= 706.86 LINK180
FORCE= -28754. STRESS= -40.678 EPEL= -0.19371E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 9 NODES= 3 6 MAT= 1 XC, YC, ZC= 2000. 500.0 0.000 AREA= 706.86 LINK180
FORCE= 32213. STRESS= 45.572 EPEL= 0.21701E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 10 NODES= 4 5 MAT= 1 XC, YC, ZC= 500.0 1000. 0.000 AREA= 706.86 LINK180
FORCE= -50408. STRESS= -71.312 EPEL= -0.33958E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 11 NODES= 5 6 MAT= 1 XC, YC, ZC= 1500. 1000. 0.000 AREA= 706.86 LINK180
FORCE= -41549. STRESS= -58.780 EPEL= -0.27991E-03
TEMP= 0.00 0.00 EPTH= 0.0000

```

Obr. 4.9 výsledky pre jednotlivé prvky

PRRSOL Command

File

```

PRINT REACTION SOLUTIONS PER NODE
**** POST1 TOTAL REACTION SOLUTION LISTING ****
LOAD STEP= 1 SUBSTEP= 1
TIME= 1.0000 LOAD CASE= 0
THE FOLLOWING X,Y,Z SOLUTIONS ARE IN THE GLOBAL COORDINATE SYSTEM

```

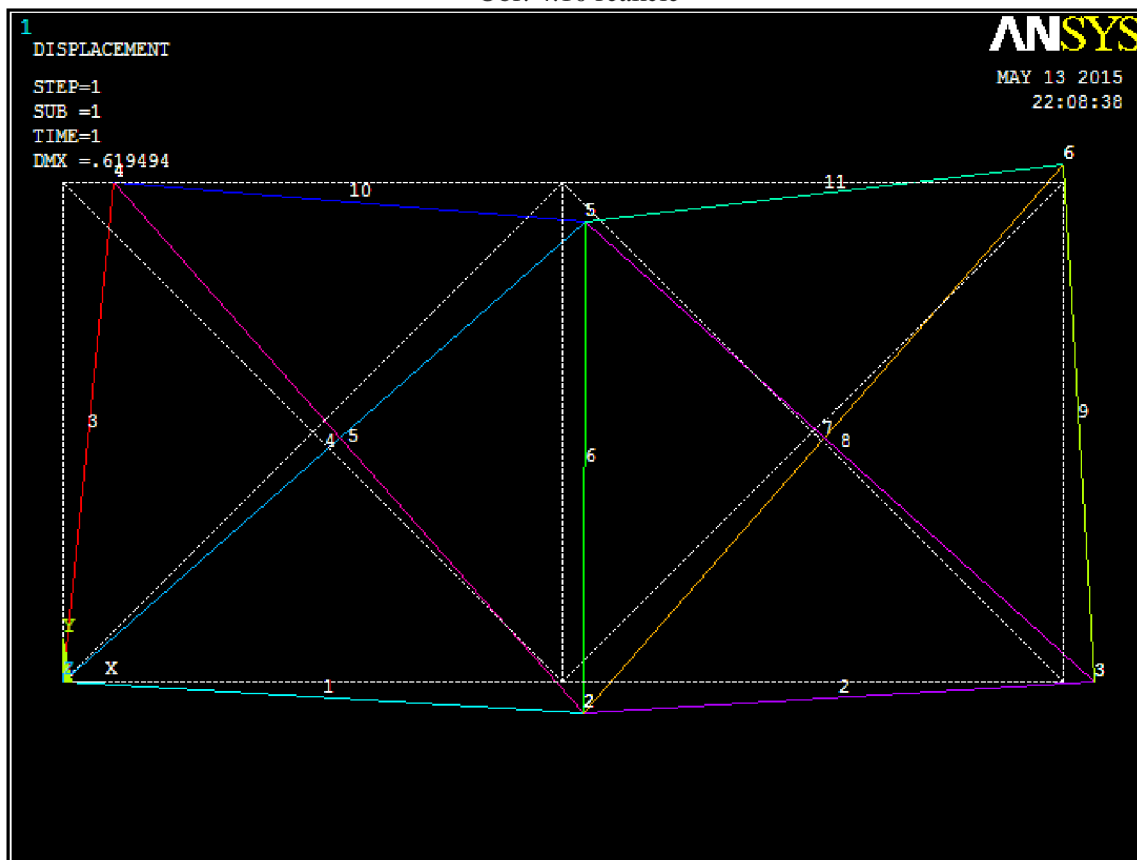
NODE	FX	FY	FZ
1	-26238.	11881.	0.0000
2			0.0000
3		-11881.	0.0000
4			0.0000
5			0.0000
6	-23762.		0.0000

```

TOTAL VALUES
VALUE -50000. -0.25466E-10 0.0000

```

Obr. 4.10 reakcie



Obr. 4.11 deformovaná sústava a tvar nedeformovanej v mierke

Porovnanie výsledkov

Tab. 4.1 porovnanie normálových napätí

prvok	Python ANSYS	
	σ [MPa]	
1	53,350	53,350
2	28,764	28,764
3	-0,577	-0,577
4	-22,955	-22,955
5	0,816	0,816
6	-25,740	-25,740
7	35,586	35,586
8	-40,678	-40,678
9	45,572	45,572
10	-71,312	-71,312
11	-58,780	-58,780

Tab. 4.2 porovnanie uzlových posuvov

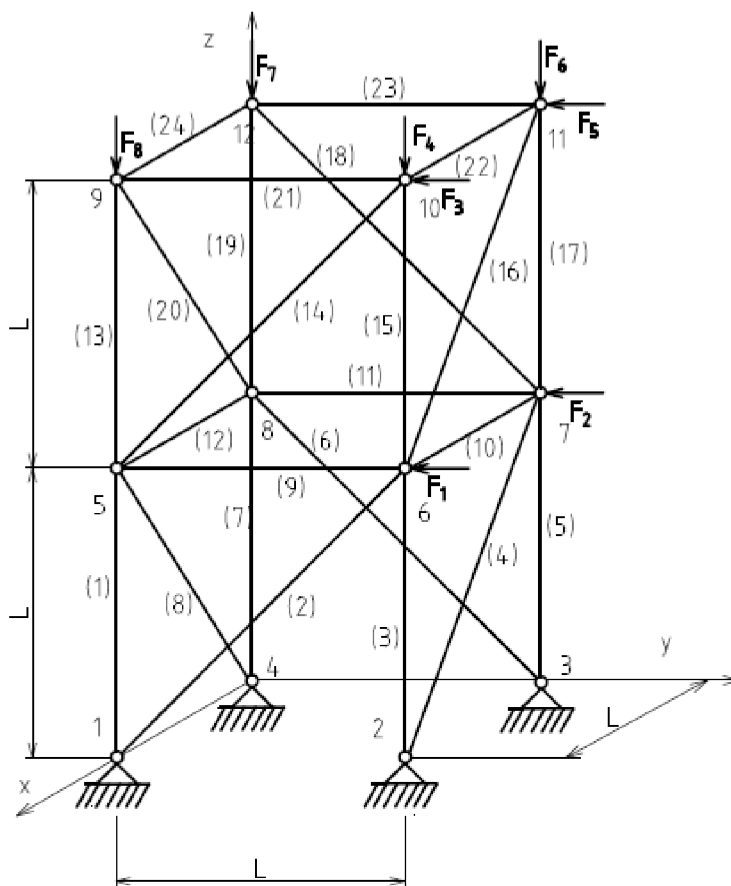
uzol	Python		ANSYS	
	u[mm]	v[mm]	u[mm]	v[mm]
1	0	0	0	0
2	0,254	0,376	0,254	0,376
3	0,391	0	0,391	0
4	0,619	-0,003	0,619	-0,003
5	0,280	-4,499	0,280	-4,499
6	0	0,217	0	0,217

4.3 3D úloha

Skript sa v prípade 3D úlohy len rozšíri o príslušnú dimenziu, pre maticu tuhosti platia odlišné vzťahy. Zostavovanie je opäť podobné [4].

Matica tuhosti pre 3D

```
#prvkove matice tuhosti a ich doplnenie do globalnej
for i in range(0,len(elements)):
#x-ova dlzka prvku
xLength=(nodeCoordinates[elements[i,3]-1,0]-nodeCoordinates[elements[i,2]-1,0])
#y-ova dlzka prvku
yLength=(nodeCoordinates[elements[i,3]-1,1]-nodeCoordinates[elements[i,2]-1,1])
#z-ova dlzka prvku
zLength=(nodeCoordinates[elements[i,3]-1,2]-nodeCoordinates[elements[i,2]-1,2])
#celkova dlzka prvku
elementLength=np.sqrt((xLength)**2+(yLength)**2+(zLength)**2)
#uhol medzi xlok a Xglob
cX=xLength/elementLength
#uhol medzi xlok a Yglob
cY=yLength/elementLength
#uhol medzi xlok a Zglob
cZ=zLength/elementLength
#prvkova matica tuhosti v glob suradniciach
kElement=np.array([[cX*cX,cX*cY,cX*cZ,-cX*cX,-cX*cY,-cX*cZ],
[cX*cY,cY*cY,cY*cZ,-cX*cY,-cY*cY,-cY*cZ],
[cX*cZ,cY*cZ,cZ*cZ,-cX*cZ,-cY*cZ,-cZ*cZ],
[-cX*cX,-cX*cY,-cX*cZ,cX*cX,cX*cY,cX*cZ],
[-cX*cY,-cY*cY,-cY*cZ,cX*cY,cY*cY,cY*cZ],
[-cX*cZ,-cY*cZ,-
cZ*cZ,cX*cZ,cY*cZ,cZ*cZ]])*elements[i,0]*elements[i,1]/(elementLength)
#riadky a stlpce v globalnej matici tuhosti
colNrow=np.array([[elements[i,2]*3-3,elements[i,2]*3-2,elements[i,2]*3-1,
elements[i,3]*3-3,elements[i,3]*3-2,elements[i,3]*3-1]])
#indexy poli musia byt integery
colNrow=colNrow.astype(int)
#skladanie do globalnej matice tuhosti
globStifMat[colNrow,colNrow.T]+=kElement
```



Obr 4.13 zadanie 3D úlohy

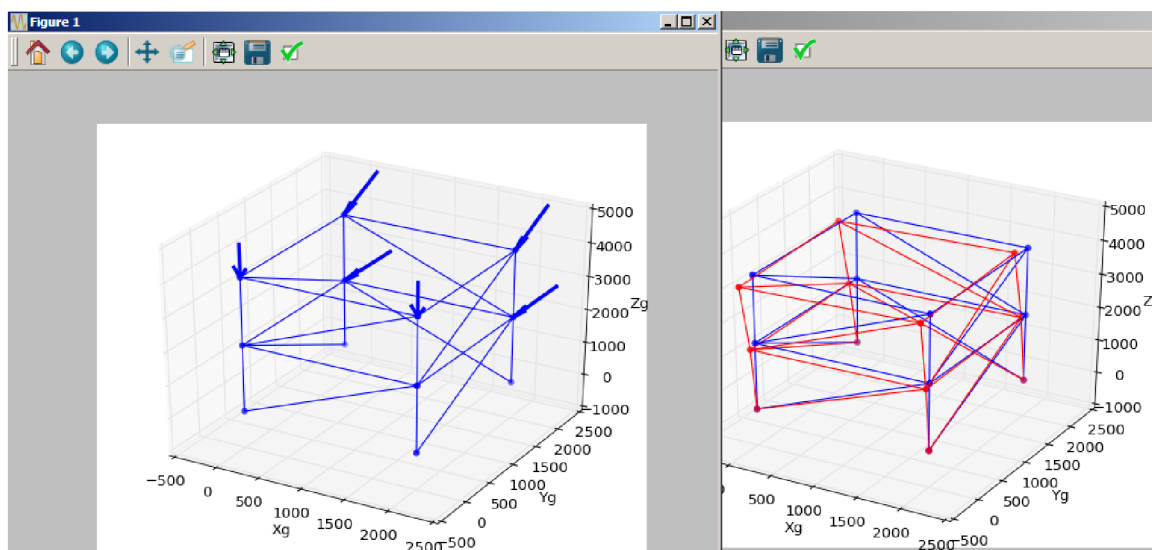
$$F_i = F = -10000N$$

$$L = 1m$$

$$E = 210GPa$$

$$S = \pi * \frac{30^2}{4} [mm^2]$$

Riešenie v Pythone



Obr. 4.14 grafický výstup z programu

nodeDisplacementsRes			
	0	1	2
0	0.000	0.000	0.000
1	0.000	0.000	0.000
2	0.000	0.000	0.000
3	0.000	0.000	0.000
4	0.269	-1.166	-0.269
5	0.000	-1.032	0.269
6	0.000	-1.571	0.000
7	0.269	-1.301	-0.539
8	0.135	-2.086	-0.404
9	-0.404	-2.086	0.269
10	-0.404	-2.895	-0.135
11	0.135	-2.760	-0.808

Obr. 4.15 uzlové posuvy

normalForces		normalStress	
	0		0
0	-20000.000	0	-28.294
1	-28284.271	1	-40.014
2	20000.000	2	28.294
3	-0.000	3	-0.000
4	0.000	4	0.000
5	28284.271	5	40.014
6	-40000.000	6	-56.588
7	-0.000	7	-0.000
8	10000.000	8	14.147
9	0.000	9	0.000
10	-20000.000	10	-28.294
11	0.000	11	0.000
12	-10000.000	12	-14.147
13	-14142.136	13	-20.007
14	-0.000	14	-0.000
15	-0.000	15	-0.000
16	-10000.000	16	-14.147
17	14142.136	17	20.007
18	-20000.000	18	-28.294
19	0.000	19	0.000
20	0.000	20	0.000
21	0.000	21	0.000
22	-10000.000	22	-14.147
23	-0.000	23	-0.000

Obr. 4.16 normálové sily a napätia

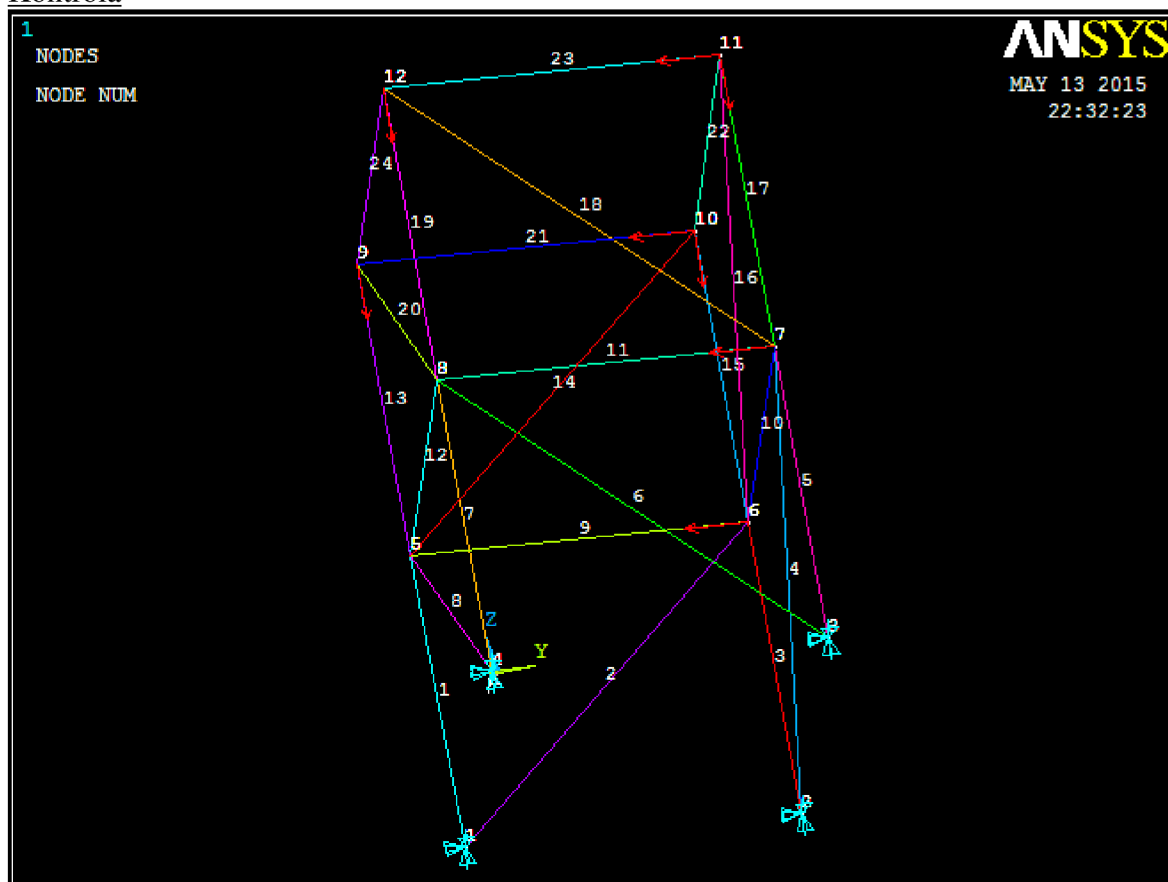
	0	1	2
0	0.000	20000.000	40000.000
1	-0.000	0.000	-20000.000
2	0.000	20000.000	-20000.000
3	0.000	0.000	40000.000
4	0.000	0.000	-0.000
5	0.000	-10000.000	0.000
6	0.000	-10000.000	0.000
7	-0.000	0.000	-0.000
8	0.000	0.000	-10000.000
9	0.000	-10000.000	-10000.000
10	0.000	-10000.000	-10000.000
11	0.000	0.000	-10000.000

Format Resize Background color

OK Cancel

Obr. 4.17 vonkajšie zaťaženie uzlov

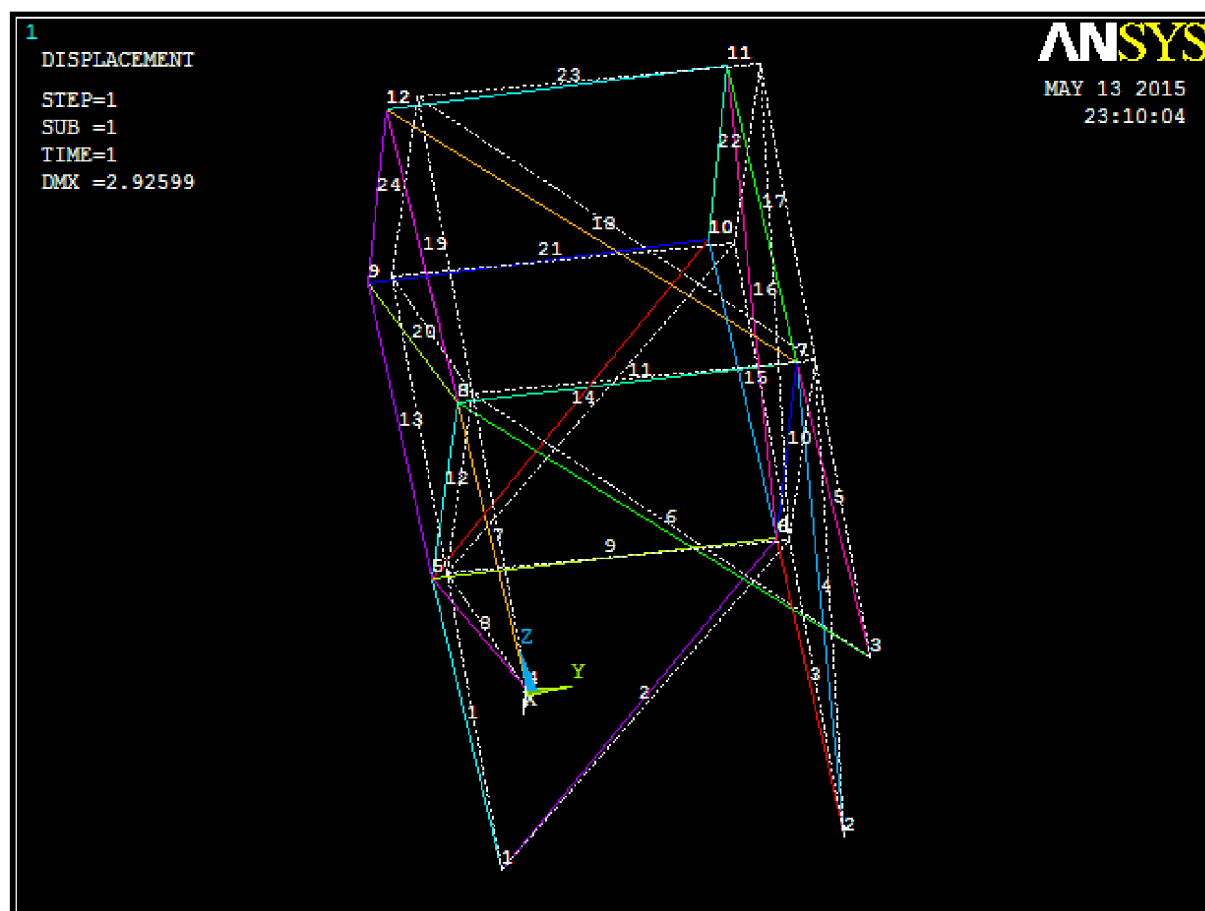
Kontrola



Obr. 4.18 zadanie sústavy

PRNSOL Command				
File				
PRINT U MODAL SOLUTION PER NODE				
**** POST1 NODAL DEGREE OF FREEDOM LISTING ****				
LOAD STEP= 1 SUBSTEP= 1				
TIME= 1.0000 LOAD CASE= 0				
THE FOLLOWING DEGREE OF FREEDOM RESULTS ARE IN THE GLOBAL COORDINATE SYSTEM				
NODE	UX	UY	UZ	USUM
1	0.0000	0.0000	0.0000	0.0000
2	0.0000	0.0000	0.0000	0.0000
3	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000	0.0000
5	0.26947	-1.1664	-0.26947	1.2271
6	-0.18041E-15	-1.0316	0.26947	1.0663
7	-0.16609E-15	-1.5706	-0.10049E-15	1.5706
8	0.26947	-1.3011	-0.53894	1.4339
9	0.13473	-2.0864	-0.40420	2.1295
10	-0.40420	-2.0864	0.26947	2.1422
11	-0.40420	-2.8948	-0.13473	2.9260
12	0.13473	-2.7601	-0.80841	2.8792
MAXIMUM ABSOLUTE VALUES				

Obr. 4.19 uzlové posuvy



Obr. 4.20 deformovaný tvar

```

PRRSOL Command
File
PRINT REACTION SOLUTIONS PER NODE
**** POST1 TOTAL REACTION SOLUTION LISTING ****
LOAD STEP= 1 SUBSTEP= 1
TIME= 1.0000 LOAD CASE= 0
THE FOLLOWING X,Y,Z SOLUTIONS ARE IN THE GLOBAL COORDINATE SYSTEM
NODE    FX      FY      FZ
  1    0.0000  20000.  40000.
  2    0.17215E-11  0.0000 -20000.
  3    0.0000  20000. -20000.
  4   -0.28450E-11  0.0000  40000.
TOTAL VALUES
VALUE -0.11235E-11  40000.  40000.

```

Obr. 4.21 reakcie v uzloch

```

PRESOL Command
File
PRINT ELEM ELEMENT SOLUTION PER ELEMENT
**** POST1 ELEMENT SOLUTION LISTING ****
LOAD STEP 1 SUBSTEP= 1
TIME= 1.0000 LOAD CASE= 0
EL= 1 NODES= 1 5 MAT= 1 XC,YC,ZC= 2000. 0.000 1000. AREA= 706.86 LINK180
FORCE= -28284. STRESS= -28.294 EPEL=-0.13473E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 2 NODES= 1 6 MAT= 1 XC,YC,ZC= 2000. 1000. 1000. AREA= 706.86 LINK180
FORCE= -28284. STRESS= -40.014 EPEL=-0.19054E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 3 NODES= 2 6 MAT= 1 XC,YC,ZC= 2000. 2000. 1000. AREA= 706.86 LINK180
FORCE= 20000. STRESS= 28.294 EPEL= 0.13473E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 4 NODES= 2 7 MAT= 1 XC,YC,ZC= 1000. 2000. 1000. AREA= 706.86 LINK180
FORCE= 0.24346E-11 STRESS= 0.34442E-14 EPEL= 0.16401E-19
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 5 NODES= 3 7 MAT= 1 XC,YC,ZC= 0.000 2000. 1000. AREA= 706.86 LINK180
FORCE=-0.74582E-11 STRESS=-0.10551E-13 EPEL=-0.50244E-19
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 6 NODES= 3 8 MAT= 1 XC,YC,ZC= 0.000 1000. 1000. AREA= 706.86 LINK180
FORCE= 28284. STRESS= 40.014 EPEL= 0.19054E-03
TEMP= 0.00 0.00 EPTH= 0.0000

```

Obr. 4.22 výsledky pre prvky časť 1

PRESOL Command												
File												
EL=	9	NODES=	5	6	MAT=	1	XC, YC, ZC=	2000.	1000.	2000.	AREA= 706.86	LINK180
FORCE=	10000.				STRESS=	14.147		EPEL=	0.67367E-04			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	10	NODES=	6	7	MAT=	1	XC, YC, ZC=	1000.	2000.	2000.	AREA= 706.86	LINK180
FORCE=	-0.10628E-11				STRESS=	-0.15036E-14		EPEL=	-0.71600E-20			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	11	NODES=	7	8	MAT=	1	XC, YC, ZC=	0.000	1000.	2000.	AREA= 706.86	LINK180
FORCE=	-20000.				STRESS=	-28.294		EPEL=	-0.13473E-03			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	12	NODES=	5	8	MAT=	1	XC, YC, ZC=	1000.	0.000	2000.	AREA= 706.86	LINK180
FORCE=	-0.40235E-11				STRESS=	-0.56921E-14		EPEL=	-0.27105E-19			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	13	NODES=	5	9	MAT=	1	XC, YC, ZC=	2000.	0.000	3000.	AREA= 706.86	LINK180
FORCE=	-10000.				STRESS=	-14.147		EPEL=	-0.67367E-04			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	14	NODES=	5	10	MAT=	1	XC, YC, ZC=	2000.	1000.	3000.	AREA= 706.86	LINK180
FORCE=	-14142.				STRESS=	-20.007		EPEL=	-0.95272E-04			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	15	NODES=	6	10	MAT=	1	XC, YC, ZC=	2000.	2000.	3000.	AREA= 706.86	LINK180
FORCE=	0.16094E-10				STRESS=	0.22768E-13		EPEL=	0.10842E-18			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	16	NODES=	6	11	MAT=	1	XC, YC, ZC=	1000.	2000.	3000.	AREA= 706.86	LINK180
FORCE=	0.30176E-11				STRESS=	0.42690E-14		EPEL=	0.20329E-19			
TEMP=	0.00				0.00			EPTH=	0.0000			

Obr. 4.23 výsledky pre prvky časť 2

PRESOL Command												
File												
EL=	17	NODES=	7	11	MAT=	1	XC, YC, ZC=	0.000	2000.	3000.	AREA= 706.86	LINK180
FORCE=	-10000.				STRESS=	-14.147		EPEL=	-0.67367E-04			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	18	NODES=	7	12	MAT=	1	XC, YC, ZC=	0.000	1000.	3000.	AREA= 706.86	LINK180
FORCE=	14142.				STRESS=	20.007		EPEL=	0.95272E-04			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	19	NODES=	8	12	MAT=	1	XC, YC, ZC=	0.000	0.000	3000.	AREA= 706.86	LINK180
FORCE=	-20000.				STRESS=	-28.294		EPEL=	-0.13473E-03			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	20	NODES=	8	9	MAT=	1	XC, YC, ZC=	1000.	0.000	3000.	AREA= 706.86	LINK180
FORCE=	0.40235E-11				STRESS=	0.56921E-14		EPEL=	0.27105E-19			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	21	NODES=	9	10	MAT=	1	XC, YC, ZC=	2000.	1000.	4000.	AREA= 706.86	LINK180
FORCE=	0.0000				STRESS=	0.0000		EPEL=	0.0000			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	22	NODES=	10	11	MAT=	1	XC, YC, ZC=	1000.	2000.	4000.	AREA= 706.86	LINK180
FORCE=	0.0000				STRESS=	0.0000		EPEL=	0.0000			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	23	NODES=	11	12	MAT=	1	XC, YC, ZC=	0.000	1000.	4000.	AREA= 706.86	LINK180
FORCE=	-10000.				STRESS=	-14.147		EPEL=	-0.67367E-04			
TEMP=	0.00				0.00			EPTH=	0.0000			
EL=	24	NODES=	9	12	MAT=	1	XC, YC, ZC=	1000.	0.000	4000.	AREA= 706.86	LINK180
FORCE=	0.0000				STRESS=	0.0000		EPEL=	0.0000			
TEMP=	0.00				0.00			EPTH=	0.0000			

Obr. 4.24 výsledky pre prvky časť 3

Tab. 4.3 porovnanie uzlových posuvov

uzol	Python			ANSYS		
	u (ux) [mm]	v (uy) [mm]	w (uz) [mm]	u (ux) [mm]	v (uy) [mm]	w (uz) [mm]
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0,269	-1,166	-0,269	0,269	-1,666	-0,269
6	0	-1,032	0,269	0	-1,032	0,269
7	0	-1,571	0	0	-1,571	0
8	0,269	-1,301	-0,539	0,269	-1,301	-0,539
9	0,135	-2,086	-0,404	0,135	-2,086	-0,404
10	-0,404	-2,086	0,269	-0,404	-2,086	0,269
11	-0,404	-2,895	-0,135	-0,404	-2,895	-0,135
12	0,135	-2,760	-0,808	0,135	-2,760	-0,808

Tab. 4.4 porovnanie normálových síl a napätí

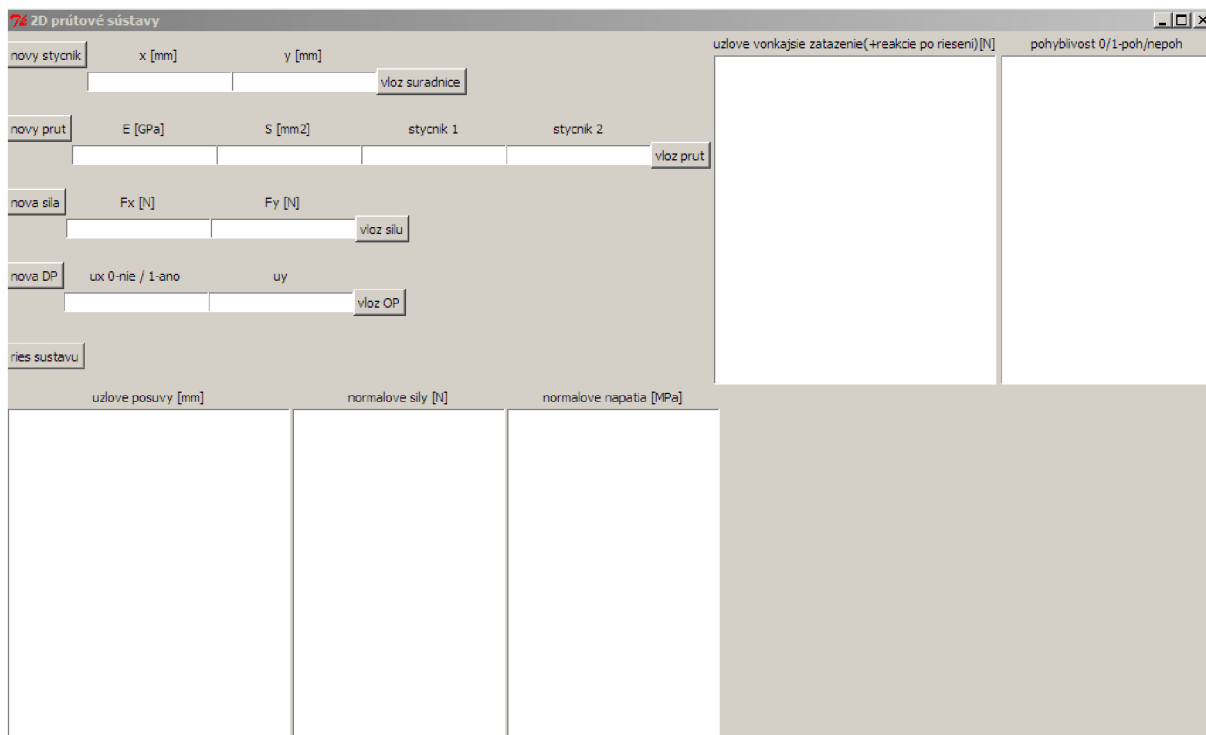
prvok	Python		ANSYS	
	N [N]	σ [MPa]	N [N]	σ [MPa]
1	-20000	-28,294	-20000	-28,294
2	-28284,271	-40,014	-28284	-40,014
3	20000	28,294	20000	28,294
4	0	0	0	0
5	0	0	0	0
6	28284,271	40,014	28284	40,014
7	-40000	-56,588	-40000	-56,588
8	0	0	0	0
9	10000	14,147	10000	14,147
10	0	0	0	0
11	-20000	-28,294	-20000	-28,294
12	0	0	0	0
13	-10000	-14,147	-10000	-14,147
14	-14142,136	-20,007	-14142	-20,007
15	0	0	0	0
16	0	0	0	0
17	-10000	-14,147	-10000	-14,147
18	14142,136	20,007	14142	20,007
19	-20000	-28,294	-20000	-28,294
20	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	-10000	-14,147	-10000	-14,147
24	0	0	0	0

Tab. 4.5 porovnanie reakcií

uzol	Python			ANSYS		
	F_x [N]	F_y [N]	F_z [N]	F_x [N]	F_y [N]	F_z [N]
1	0	20000	40000	0	20000	40000
2	0	0	-20000	0	0	-20000
3	0	20000	-20000	0	20000	-20000
4	0	0	40000	0	0	40000

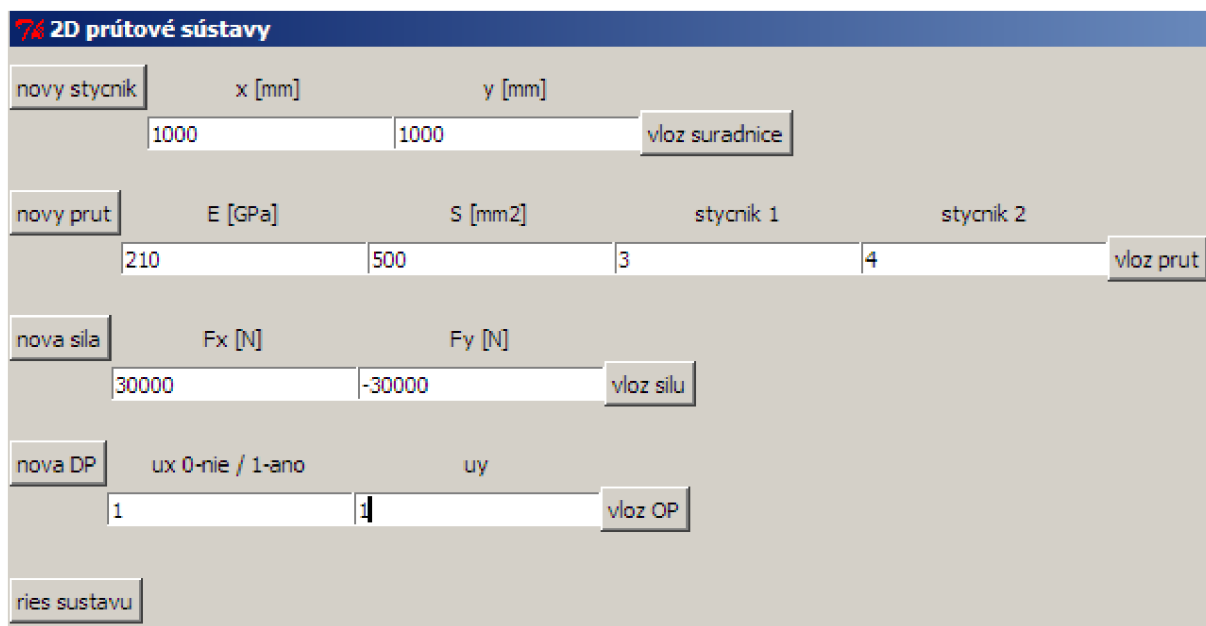
5. Grafické užívateľské rozhranie

Pomocou knižnice Tkinter bolo v Pythone pre 2D úlohy vytvorené jednoduché grafické užívateľské rozhranie (GUI - Graphic User Interface). Zadávajú sa do neho všetky potrebné vstupné údaje ako súradnice styčníc, prúty, sily a deformačné podmienky. Výstupom je vykreslená sústava, hodnoty napätí, síl v prútoch a reakcie vo väzbách.



Obr. 5.1 grafické užívateľské rozhranie

Jednoduchá úloha riešená pomocou GUI:



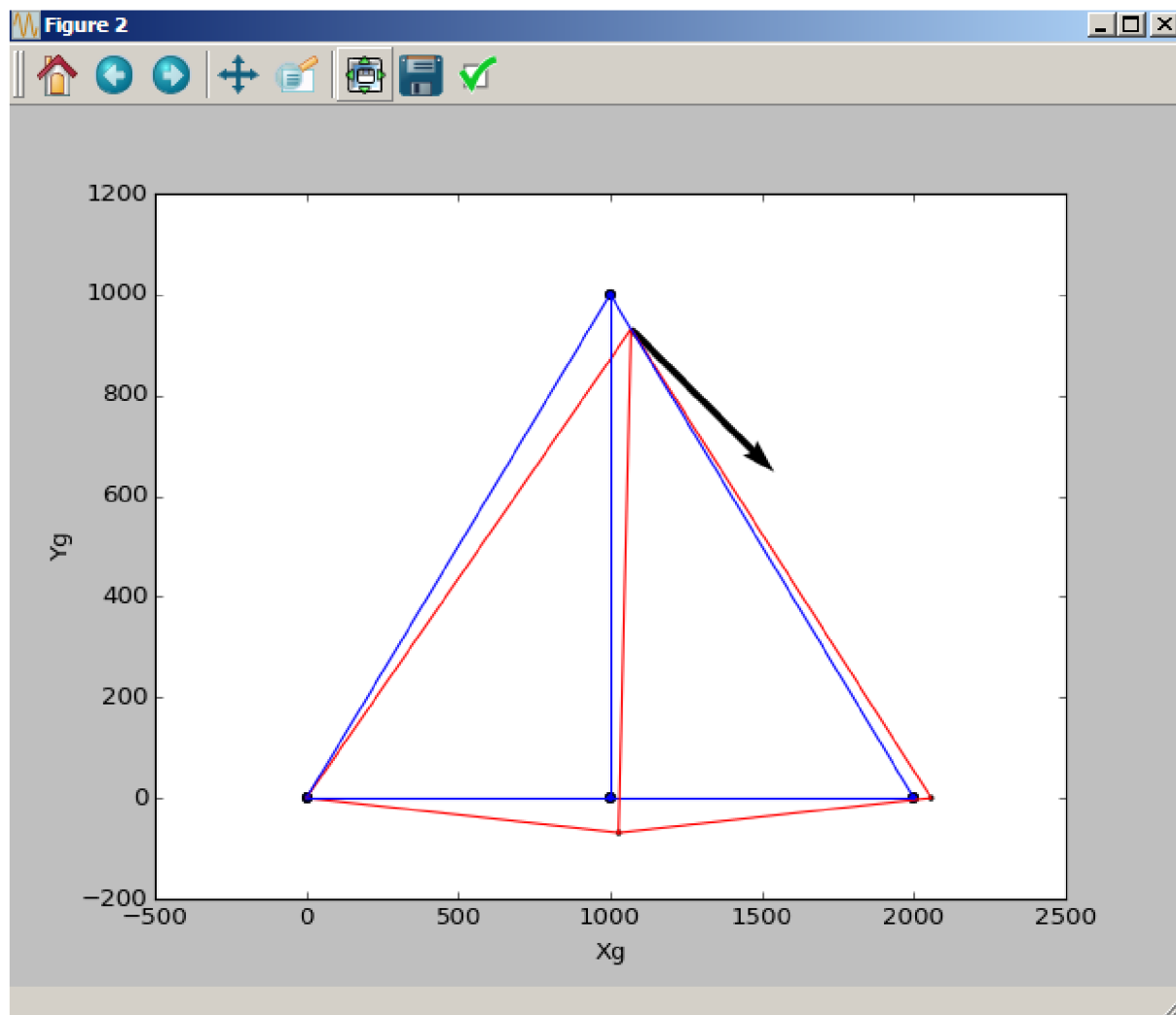
Obr. 5.2 ukážka zadávania vstupných hodnôt

uzlove posuvy [mm]	normalove sily [N]	normalove napatia [MPa]
[0.0, 0.0]	30000.00000000001	60.00000000000002
[0.2857142857142858, -0.6897753035351704]	30000.00000000001	60.00000000000002
[0.5714285714285716, 0.0]	-8.242985408089818e-12	-1.6485970816179636e-14
[0.6897753035351702, -0.6897753035351704]	0.0	0.0
	-42426.40687119286	-84.85281374238572

Obr. 5.4 získané výstupné hodnoty

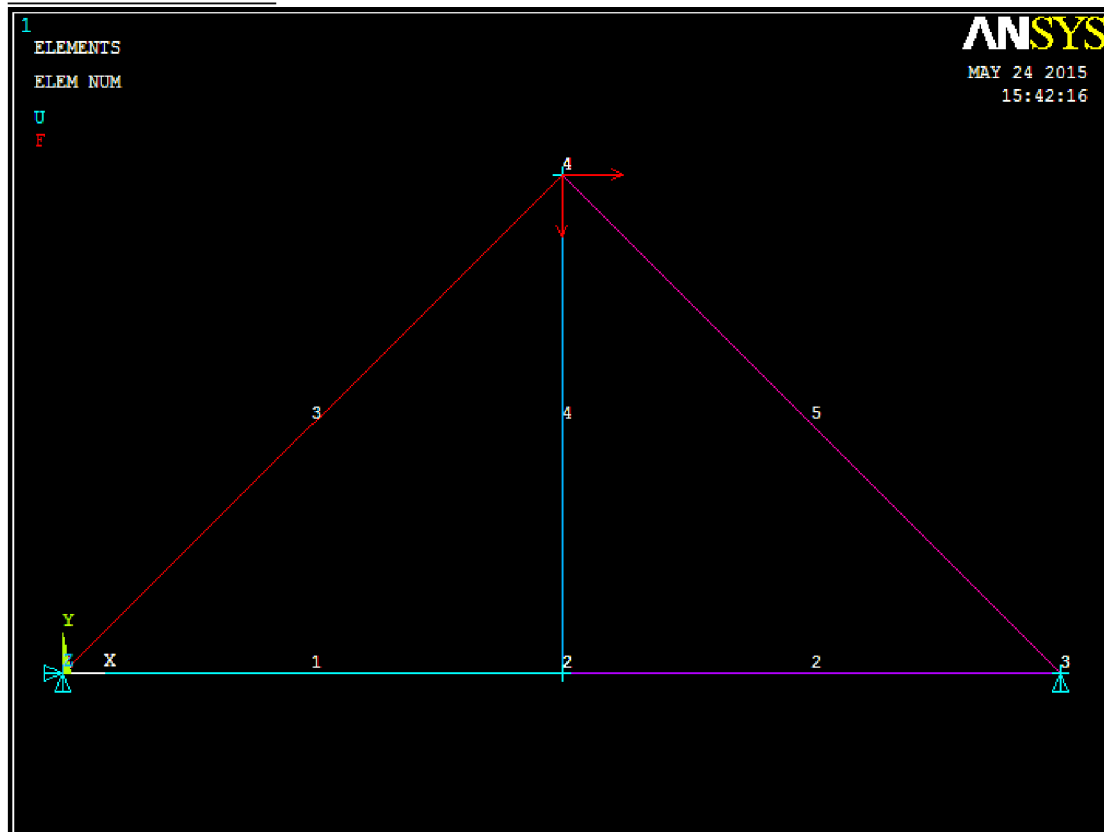
uzlove vonkajšie zaťaženie(+reakcie po riešení)[N]	pohyblivosť 0/1-poh/nepoh
[-3.00000000e+04 3.63797881e-12]	[0, 0]
[0. 0.]	[1, 1]
[-3.63797881e-12 3.00000000e+04]	[1, 0]
[30000. -30000.]	[1, 1]

Obr. 5.5 vonkajšie zaťaženie uzlov a ich pohyblivosť

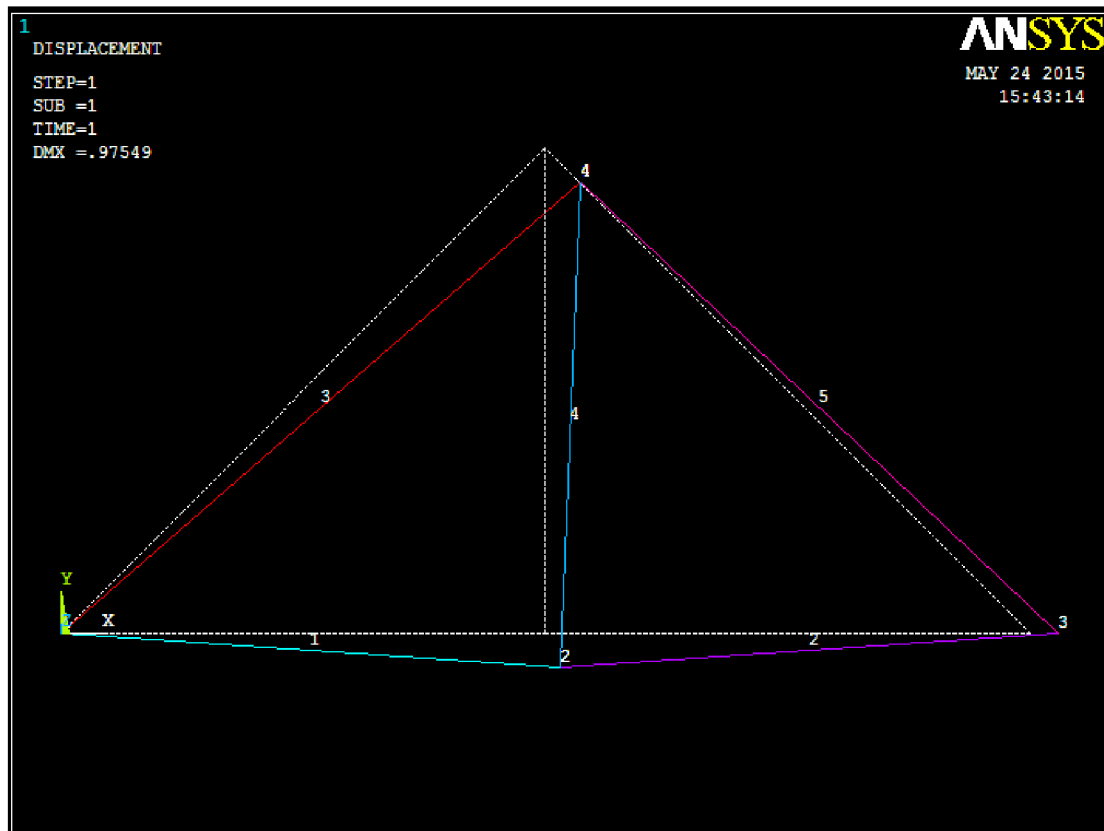


Obr. 5.6 grafický výstup z programu

Overenie v ANSYSe



Obr. 5.7 nastavenie sústavy v ANSYSe



Obr. 5.8 deformovaná sústava v ANSYSe

```

PRNSOL Command
File
PRINT U   NODAL SOLUTION PER NODE

**** POST1 NODAL DEGREE OF FREEDOM LISTING ****

LOAD STEP=   1  SUBSTEP=   1
TIME=   1.0000  LOAD CASE=   0

THE FOLLOWING DEGREE OF FREEDOM RESULTS ARE IN THE GLOBAL COORDINATE SYSTEM

  NODE      UX          UY          UZ          USUM
    1      0.0000      0.0000      0.0000      0.0000
    2      0.28571     -0.68978      0.0000      0.74661
    3      0.57143      0.0000      0.0000      0.57143
    4      0.68978     -0.68978      0.0000      0.97549

MAXIMUM ABSOLUTE VALUES
NODE      4          4          0          4
VALUE     0.68978   -0.68978   0.0000   0.97549

```

Obr. 5.9 uzlové posuvy získané v ANSYSe

```

PRNSOL Command
File
PRINT REACTION SOLUTIONS PER NODE

**** POST1 TOTAL REACTION SOLUTION LISTING ****

LOAD STEP=   1  SUBSTEP=   1
TIME=   1.0000  LOAD CASE=   0

THE FOLLOWING X,Y,Z SOLUTIONS ARE IN THE GLOBAL COORDINATE SYSTEM

  NODE      FX          FY          FZ
    1     -30000.     -0.80498E-11   0.0000
    2
    3           30000.   0.0000
    4
TOTAL VALUES
VALUE     -30000.    30000.    0.0000

```

Obr. 5.10 reakcie

```

PRESOL Command
File
PRINT ELEM ELEMENT SOLUTION PER ELEMENT
**** POST1 ELEMENT SOLUTION LISTING ****
LOAD STEP 1 SUBSTEP= 1
TIME= 1.0000 LOAD CASE= 0
EL= 1 NODES= 1 2 MAT= 1 XC,YC,ZC= 500.0 0.000 0.000 AREA= 500.00 LINK180
FORCE= 30000. STRESS= 60.000 EPEL= 0.28571E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 2 NODES= 2 3 MAT= 1 XC,YC,ZC= 1500. 0.000 0.000 AREA= 500.00 LINK180
FORCE= 30000. STRESS= 60.000 EPEL= 0.28571E-03
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 3 NODES= 1 4 MAT= 1 XC,YC,ZC= 500.0 500.0 0.000 AREA= 500.00 LINK180
FORCE= 0.11384E-10 STRESS= 0.22768E-13 EPEL= 0.10842E-18
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 4 NODES= 2 4 MAT= 1 XC,YC,ZC= 1000. 500.0 0.000 AREA= 500.00 LINK180
FORCE=-0.11384E-10 STRESS=-0.22768E-13 EPEL=-0.10842E-18
TEMP= 0.00 0.00 EPTH= 0.0000
EL= 5 NODES= 3 4 MAT= 1 XC,YC,ZC= 1500. 500.0 0.700 AREA= 500.00 LINK180
FORCE= -42426. STRESS= -84.853 EPEL=-0.40406E-03
TEMP= 0.00 0.00 EPTH= 0.0000

```

Obr. 5.11 výsledky pre prúty v ANSYSe

Tab. 5.1 porovnanie uzlových posuvov

uzol	GUI		ANSYS	
	u [mm]	v [mm]	u [mm]	v [mm]
1	0	0	0	0
2	0,286	-0,690	0,286	-0,690
3	0,571	0	0,571	0
4	0,690	-0,690	0,690	-0,690

Tab 5.2 porovnanie normálových síl a napätí

prút	N [N]		σ [MPa]	
	GUI	ANSYS	GUI	ANSYS
1	30000	30000	60	60
2	30000	30000	60	60
3	0	0	0	0
4	0	0	0	0
5	-42426,407	-42426	-84,853	-84,853

6. Záver

Cieľom tejto bakalárskej práce bolo primárne naprogramovať algoritmus pre riešenie rovinných a priestorových prúťových sústav. Výsledky mali byť overené v programe ANSYS. Hlavným cieľom bolo dosiahnuť zhodu vo výsledkoch, v tomto prípade deformácii a napätosti prúťovej sústavy. Táto zhoda bola dosiahnutá pre prípad rovinný aj priestorový, ako dokladajú výstupy z oboch programov. Vzhľadom na to, že oba spôsoby riešenia v sebe obsahujú v zásade rovnaký algoritmus pre riešenie a prvok siete je v oboch prípadoch vlastne prút, a teda diskretizácia je pre program v Pythone a ANSYS-e rovnaká, sa táto zhoda v prípade vylúčenia hrubej chyby pri zostavovaní algoritmu v jazyku Python dala očakávať. Zhoda je obmedzená len počtom platných desatinných miest. Ďalej bol v úvode práce uvedený základný prehľad voľne, prípadne komerčne dostupného dostupného software pre vedecké a inžinierske výpočty. MKP sa ukazuje ako prostriedok pre rýchle riešenie deformácie a napätosti prúťových sústav, vhodný k naprogramovaniu na počítači. Prúťové sústavy sú pritom veľmi jednoduchý výpočtový model a skutočná sila MKP spočíva hlavne v riešení zložitých úloh analyticky len veľmi obmedzene riešiteľných alebo vôbec neriešiteľných. Bohužiaľ sa mi však nepodarilo stihnúť naprogramovať algoritmus pre riešenie sústav prúťových telies s tuhými styčnými, kde by sa bral do úvahy aj ohybový moment a ani vytvoriť plne funkčný .exe súbor s grafickým užívateľským rozhraním.

Zdroje

- [1] JANÍČEK, Přemysl. Mechanika těles: pružnost a pevnost I. 3., přeprac. vyd., V Akademickém nakladatelství CERM 1. Brno: Akademické nakladatelství CERM, 2004, 287 s. ISBN 80-214-2592-x.
- [2] FLORIAN, Zdeněk, Emanuel ONDRÁČEK a Karel PŘIKRYL. Mechanika těles: statika. Vyd. 7., V Akademickém nakladatelství CERM 2. Brno: Akademické nakladatelství CERM, 2007, 182 s. ISBN 978-80-214-3440-0.
- [3] SHIGLEY, Joseph Edward, Charles R MISCHKE a Richard G BUDYNAS. Konstruování strojních součástí. 1. vyd. Editor Miloš Vlk. Překlad Martin Hartl. V Brně: VUTIUM, 2010, xxv, 1159 s. Překlady vysokoškolských učebnic. ISBN 978-80-214-2629-0.
- [4] FERREIRA, A. MATLAB codes for finite element analysis: solids and structures. Dordrecht: Springer Science & Business Media, c2009, ix, 235 p. Solid mechanics and its applications, 157. ISBN 1402092008-.
- [5] PETRUŠKA, Jindřich. MKP v inženýrských výpočtech [elektronická skripta]. [cit. 2015-05-24]. Plný text dostupný z :
<http://www.umt.fme.vutbr.cz/img/fckeditor/file/opory/RIV/MKP2011.pdf>
- [6] Copyleft. [online]. 24.5.2015 [cit. 2015-05-24]. Dostupné z:
<http://cs.wikipedia.org/wiki/Copyleft>
- [7] Co je to svobodný software?. [online]. 21.12.2014 [cit. 2015-05-24]. Dostupné z:
<http://www.gnu.org/philosophy/free-sw.cs.html>
- [8] Svobodný software. [online]. 25. 4. 2015 [cit. 2015-05-24]. Dostupné z:
http://cs.wikipedia.org/wiki/Svobodn%C3%BD_software
- [9] Open Source software. Root.cz. [online]. [cit. 2015-05-24]. Dostupné z:
<http://www.root.cz/specially/licence/open-source-software>
- [10] Tzv. proprietární software. Root.cz. [online]. [cit. 2015-05-24]. Dostupné z:
<http://www.root.cz/specially/licence/tzv-proprietarni-software>
- [11] GNU General Public License. Wikipedia, the free encyclopedia. [online]. 13. 4. 2015 [cit. 2015-05-24]. Dostupné z:
http://cs.wikipedia.org/wiki/GNU_General_Public_License#Dr.C5.BEitel.C3.A9_autorsk.C3.BDch_pr.C3.A1v
- [12] List of finite element software packages. Wikipedia, the free encyclopedia [online]. 7.5.2015 [cit. 2015-05-24]. Dostupné z:
http://en.wikipedia.org/wiki/List_of_finite_element_software_packages
- [13] FEBio Software Suite. febio.org [online]. [cit. 2015-05-24]. Dostupné z: <http://febio.org>

- [14] AUTOR NEUVEDEN. febio.org [online]. [cit. 24.5.2015]. Dostupný na WWW: <http://febio.org/preview/preview-gallery/#prettyPhoto/1/>
- [15] Calculix Review. iMechanica. [online]. [cit. 2015-05-24]. Dostupné z: <http://imechanica.org/node/1628>
- [16] CALCULIX. calculix. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.calculix.de/>
- [17] AUTOR NEUVEDEN. [online]. [cit. 24.5.2015]. Dostupný na WWW: <http://www.calculix.de/fw05lat.html>
- [18] Presentation of Code_Aster. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.code-aster.org/V2/spip.php?rubrique18>
- [19] Code_Aster. Wikipedia, the free encyclopedia. [online]. [cit. 2015-05-24]. Dostupné z: http://en.wikipedia.org/wiki/Code_Aster
- [20] OOFEM – program pro výpočty metodou konečných prvků. volně šiřitelný program pro řešení multifyzikálních problémů metodou konečných prvků. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.oofem.org/cz/oofem.html>
- [21] <http://www.oofem.org/cz/gallery/gallery.html>. volně šiřitelný program pro řešení multifyzikálních problémů metodou konečných prvků. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.oofem.org/gallery/screenshots/koyna.dam.gif>
- [22] Z88AURORA. [online]. [cit. 2015-05-24]. Dostupné z: <http://en.z88.de/z88aurora/#>
- [23] Postprocessor Z88 V14. Wikipedia, the free encyclopedia. [online]. [cit. 2015-05-24]. Dostupné z: http://en.wikipedia.org/wiki/Z88_FEM_software#/media/File:Programsnapshot_Postprocessor_Z88_V14.png
- [24] Pyplot tutorial. [online]. [cit. 2015-05-24]. Dostupné z: http://matplotlib.org/users/pyplot_tutorial.html
- [25] pylab_examples example code: quiver_demo.py. [online]. [cit. 2015-05-24]. Dostupné z: http://matplotlib.org/examples/pylab_examples/quiver_demo.html
- [26] Anaconda. Continuum documentation. [online]. [cit. 2015-05-24]. Dostupné z: <http://docs.continuum.io/anaconda/>
- [27] GNU Octave. GNU Octave. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.gnu.org/software/octave/>
- [28] About Scilab. Scilab. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.scilab.org/scilab/about>
- [29] Graphics. Scilab. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.scilab.org/scilab/gallery/graphics>

- [30] List of finite element software packages. Wikipedia, the free encyclopedia. [online]. [cit. 2015-05-24]. Dostupné z: http://en.wikipedia.org/wiki/List_of_finite_element_software_packages
- [31] ANSYS - Simulation Driven Product Development. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.ansys.com/>
- [32] Wilde Analysis. [online]. [cit. 2015-05-24]. Dostupné z: http://wildeanalysis.co.uk/system/photos/178/original/ANSYS_Multiphysics_User_Environment.jpg?1271162613
- [33] Nastran. Wikipedia, the free encyclopedia. [online]. 24.5.2015 [cit. 2015-05-24]. Dostupné z: <http://en.wikipedia.org/wiki/Nastran>
- [34] MSC Nastran Multidisciplinary Structural Analysis. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.mssoftware.com/product/msc-nastran>
- [35] NEi Nastran Finite Element Analysis and Simulation Software. NEi Nastran. [online]. [cit. 2015-05-24]. Dostupné z: <http://www.nenastran.com/nei-nastran.php>
- [36] Example of NEi Nastran analysis result. Wikipedia, the free encyclopedia. [online]. [cit. 2015-05-24]. Dostupné z: http://en.wikipedia.org/wiki/NEi_Nastran#/media/File:Noraneng.JPG
- [37] Finite Element Analysis Software. Structural Analysis and Design Software. [online]. [cit. 2015-05-24]. Dostupné z: <https://www.dlupal.com/cz/rfem-5xx.aspx>
- [38] Finite Element Analysis Software. Structural Analysis and Design Software. [online]. [cit. 2015-05-24]. Dostupné z: <https://www.dlupal.com/netgenium/files/141517.jpg>
- [39] Carlos Felippa. . Introduction to Finite Element Methods. [online]. [cit. 2015-05-25]. Dostupné z: <http://www.colorado.edu/engineering/cas/courses.d/IFEM.d/>
- [40] SEDLÁČEK, R. Staticky neurčité prutové soustavy. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2012. 66 s. Vedoucí bakalářské práce Ing. Zdeněk Florian, CSc..

Zoznam príloh:

Na CD:

- skript 2DprutoveSustavy.py
- skript 3DprutoveSustavy.py
- skript 2DprutoveSustavyGUI.py