Университет по архитектура, строителство и геодезия
Строителен факултет
Катедра „Строителна механика“

Инж. Неделчо Живков Ганчовски

АНАЛИЗ НА СТРОИТЕЛНИ КОНСТРУКЦИИ ЧРЕЗ ФУНКЦИОНАЛНО МОДЕЛИРАНЕ В ОБЛАЧНА ИЗЧИСЛИТЕЛНА СРЕДА

Автореферат

към дисертационен труд за присъждане на научна и образователна степен „Доктор“ в професионално направление 5.7. „Архитектура, строителство и геодезия“, научна специалност „Строителна механика, съпротивление на материалите и теоретична механика (Строителна механика)“

Научен ръководител:

доц. д-р инж. Александър Трайков

Научно жури:

проф. д-р инж. Михаела Кутева-Генчева – катедра АИТ, УАСГ
проф. д-р инж. Светлана Лилкова-Маркова – кат. „Техн. механика“, УАСГ
проф. д-р инж. Николай Жечев – МГУ „Св. Иван Рилски“
доц. д-р инж. Анита Хандрулева – ВСУ „Л. Каравелов“
доц. д-р инж. Дария Михалева – ВСУ „Черноризец Храбър“

гр. София

2025 г.

Заглавие: Анализ на строителни конструкции чрез функционално моделиране в облачна изчислителна среда

Професионално направление: 5.7 „Архитектура, строителство и геодезия“

Научна специалност: „Строителна механика, съпротивление на материалите и теоретична механика (Строителна механика)“

Катедра: „Строителна механика“

Автор: Инж. Неделчо Живков Ганчовски

Научен ръководител: доц. д-р инж. Александър Трайков

Научно жури:

Вътрешни за УАСГ:

проф. д-р инж. Михаела Кутева-Генчева – катедра „Автоматизация на инженерния труд“

проф. д-р инж. Светлана Лилкова-Маркова – катедра „Техническа механика“

Резервен член: проф. д-р инж. Соня Първанова-Йончева – кат. „Строителна механика“

Външни за УАСГ:

проф. д-р инж. Николай Жечев – МГУ „Св. Иван Рилски“

доц. д-р инж. Анита Хандрулева – ВСУ „Л. Каравелов“

доц. д-р инж. Дария Михалева – ВСУ „Черноризец Храбър“

Резервен член: доц. д-р инж. Илиана Стойнова – ВСУ „Л. Каравелов“

Председател на първото заседание: проф. д-р инж. Светлана Лилкова-Маркова

Кандидатът е зачислен за докторант на свободна форма на обучение със Заповед 354/15.05.2023 г. на Ректора на УАСГ, на основание на решение на Факултетен съвет на Строителен факултет, протокол № 32 от 09.05.2023 г. на основание чл. 46 от ПУДПЗСДС.

Отчислен е с право на защита със Заповед 323/29.04.2025 г. на Ректора на УАСГ, на основание на решение на Факултетен съвет на Строителен факултет, протокол № 55/15.04.2025 г., по доклад на ръководителя на катедра „Строителна механика“, на основание чл. 59 от *Правилника за учебната дейност, правата и задълженията на студентите, докторантите и специализантите* и чл. 54 (1) от *Стандарта за обучение на докторантите в УАСГ*.

Данни за дисертационния труд:

Дисертационният труд е разработен в обем от 257 стр. като съдържанието е структурирано в 12 глави и е онагледено с 64 фигури и 14 таблици. Към тях са представени 5 бр. приложения с общ обем от 95 стр. Библиографията обхваща 83 източника, от които 12 на кирилица и 71 на латиница. По темата на дисертационния труд са направени 7 публикации – 3 самостоятелни и 4 в съавторство. Четири от тях са в международни издания. До момента са известни 2 цитирания на разработените публикации, както следва:

[1] Galvão H., Silva V. Classes de Métodos Numéricos não Convencionais para Determinação de Raízes de Funções. 02.2024. Thesis. Universidade Federal do Pará. Faculdade de Matemática. DOI: 10.13140/RG.2.2.24088.57606

[2] Roots Fortran. Numerical Library for Modern Fortran. NASA Johnson Space Center, Houston, Texas
https://jacobwilliams.github.io/roots-fortran/

СЪДЪРЖАНИЕ

I. Увод....4
I.1 Исторически преглед на изчислителните устройства....4
I.2 Развитие на програмните средства....5
I.3 Съвременни тенденции в развитието на компютърните изчисления....6
I.4 Актуално състояние на проблема....8
I.5 Цели, задачи и обхват....9
I.5.1 Цели....9
I.5.2 Задачи....9
I.5.3 Обхват....9
I.6 Етапи на дисертационния труд....10
I.7 Методологически особености....11
II. Теоретични основи на компютърните изчисления....11
II.1 Особености при работа в машинна изчислителна среда....12
II.1.1 Машинно представяне на числата с плаваща запетая....12
II.1.2 Машинна аритметика....13
II.1.3 Видове грешки при компютърните изчисления....13
II.1.4 Относителна работна точност....15
II.1.5 Сравняване на числа с плаваща запетая....15
II.2 Алгоритми....16
II.2.1 Кратко въведение в теорията на алгоритмите....16
II.2.2 Сложност на алгоритмите....17
II.2.3 Обусловеност и числена устойчивост....17
II.2.4 Сходимост на итерационни алгоритми....17
II.2.5 Оценка на грешката....18
III. Числени методи за анализ на строителни конструкции....19
III.1 Методи за решаване на нелинейни уравнения....19
III.1.1 Постановка на задачата....19
III.1.2 Критерии за избор на подходящ метод....20
III.1.3 Обзорен преглед на основните методи за решаване на нелинейни уравнения 21
III.1.4 Обобщение на свойствата на различните методи....32
III.1.5 Предложение за нов хибриден метод....32
III.1.6 Числени експерименти....38
III.1.7 Изводи....40
III.2 Екстремум на функция (оптимизационна задача)....41
III.2.1 Постановка на задачата....41
III.2.2 Метод на златното сечение....41
III.2.3 Сходимост и оценка на грешката....43
III.2.4 Програмна реализация....43
III.3 Числено диференциране....43
III.3.1 Производна на функция....43
III.3.2 Дефиниция и видове формули....44

III.3.3 Изчислителни проблеми и оценка на грешката....44
III.3.4 Избор на оптимална стъпка....46
III.3.5 Методи за подобряване на точността....48
III.3.6 Екстраполация на Ричардсън....49
III.3.7 Програмна реализация....52
III.3.8 Допълнителни особености....52
III.3.9 Примери....53
III.3.10 Изводи....59
III.4 Числено интегриране....60
III.4.1 Дефиниция....60
III.4.2 Видове квадратурни формули....60
III.4.3 Проблеми при числеността интегриране....70
III.4.4 Методи за подобряване на точността....70
III.4.5 Адаптивна квадратура на Симпсън....72
III.4.6 Адаптивна квадратура на Гаус-Кронрод....73
III.4.7 Адаптивна квадратура на Лобато....74
III.4.8 Сравнителен анализ и избор на подходящ алгоритъм....75
III.5 Адаптивна интерполация....77
IV. Възможности на изчислителната среда....79
IV.1 Програмен език....80
IV.1.1 Класификация и характеристики на програмния език....80
IV.2 Комплексни числа....81
IV.2.1 Видове запис и разпознаване....81
IV.2.2 Предефиниране на основните аритметични операции....82
IV.2.3 Комплексни функции....82
IV.3 Програмиране с мерни единици....83
IV.3.1 Видове мерни единици....84
IV.3.2 Математически модел и машинно представяне....85
IV.3.3 Библиотека мерни единици....86
IV.4 Визуализация на резултатите....86
IV.5 Вектори и матрици....89
IV.5.1 Вектори....90
IV.5.2 Матрици....93
IV.6 Функционално моделиране....108
IV.6.1 Основни принципи....108
IV.6.2 Пример за функционален подход....109
IV.6.3 Предимства и недостатъци....110
IV.6.4 Приложение....110
IV.6.5 Програмна реализация....111
V. Програмируема платформа за изчисляване на конструкции в облачна среда....114
V.1 Принципна схема и архитектура на приложението....114
V.1.1 Облачна компонента на платформата (back-end)....115
V.1.2 Клиентска част (front-end)....115

V.1.3 Принципна схема на работа....115
V.2 Структура и основни възможности....116
V.2.1 Среда за разработка (online IDE)....116
V.2.2 Онлайн калкулатор (online calculator)....117
V.2.3 Библиотека от подпрограми (online library)....117
V.2.4 Документация (online help)....118
V.2.5 Потребителски модул....119
V.2.6 Административен модул....119
V.3 Десктоп версия....120
V.4 Ресурси и достъп до платформата....121
VI. Приложение в изчисляването на строителни конструкции....121
VI.1 Изследване на колона с променливо напречно сечение с произволна форма с отчитане на геометрична и физична нелинейност....121
VI.1.1 Постановка на задачата....121
VI.1.2 Теоретични основи и методи за анализ....121
VI.1.3 Пример – стоманена колона с елипсовидно сечение....133
VI.2 Гранична носимоспособност на Ст.Б. колона за условен центричен натиск....144
VI.3 Някои класически решения от теорията на еластичността....148
VI.4 Анализ на гредостена по МКЕ....158
VI.5 Анализ на плоча по метода на крайните елементи....172
VI.6 Програма за статически анализ на равнинни рамки....182
VI.7 Функционален модел на пръстов краен елемент с произволно напречно сечение и променливи еластични характеристики по дължина....192
VI.8 Анализ на мигновено изменяема диада с геометрична нелинейност с големи премествания (теория от 3-ти ред)....212
VI.9 Анализ на несиметрична мигновено изменяема диада с предварително напрягане, физична и геометрична нелинейност с големи премествания (теория от 3-ти ред)....215
VI.10 Собствени форми на трептене на стоманен комин с променливо напречно сечение с произволна форма....222
VI.11 Анализ за удар от падащо тяло (метална сфера)....229
VI.12 Оптимизация на размерите на единичен правоъгълен фундамент, натоварен с кос нецентричен натиск....239
VII. Възможности за внедряване в учебния процес и проектантската практика....242
VII.1 Внедряване в процеса на обучение....242
VII.2 Внедряване в проектантската практика....242
VIII. Приноси....248
IX. Насоки за бъдеща работа....248
X. Публикации по темата на дисертационния труд....249
XI. Приложения....250
XII. Литература....250

I. Увод

Развитието на методите за анализ на строителни конструкции се намира в пряка зависимост от развитието на изчислителните средства. И това е съвсем естествено - без наличието на подходящи средства за изчисление, много от съвременните методи щяха да бъдат прекалено трудни, дори невъзможни за прилагане в практиката. Например, без компютри, които да могат да обработват матрици с големи размери, методът на крайните елементи нямаше да получи толкова широко разпространение.

В настоящата дисертация ще бъде разгледан един по-различен подход за разработване на математически модели за анализ на строителните конструкции, който да позволи използването на съвременни програмни средства с различна архитектура спрямо традиционните.

I.1 Актуално състояние на проблема

Към днешна дата проблемите, които стоят пред технологичното развитие на софтуера за строително проектиране, са най-общо следните:

- Сравнително малко приложения използват ефективно предимствата на новите технологии. Повечето програмни продукти за строително проектиране все още работят като десктоп приложения.
- Масово използваните програмни езици и платформи за разработка стават все по-абстрактни и сложни, което изисква наемането на високо квалифицирани софтуерни специалисти. С оглед на това, процесът на разработване е необходимо да се автоматизира и оптимизира, така че да може да се извършва от специалисти по строително инженерство или строителна механика.
- Наличните приложения за строително проектиране са трудни за тестване и проверка. Изчислителните процедури са скрити в изходния код, който не е достъпен за крайния потребител. За него софтуерът представлява „черна кутия“, но отговорността за верността на резултатите е изцяло на проектанта.

За решаването на описаните по-горе проблеми, е необходимо да се потърсят нови и алтернативни методи за моделиране и анализ на строителни конструкции, които да послужат за основа при създаването на съответните софтуерни продукти.

I.2 Цели, задачи и обхват

I.2.1 Цели

Цел на настоящата дисертация е разработването на обща методика за решаване на задачи от строителната механика чрез функционално моделиране, както и нейната софтуерна реализация в облачна среда.

I.2.2 Задачи

1. Да се разработят числени методи и алгоритми за функционалната реализация на основните изчислителни процедури, свързани със задачите на строителната механика.
2. Да се проектира прост и интуитивен „програмен“ език, с възможност за функционално програмиране, който да включва и необходимия математически апарат. Той трябва да може да се използва от практикуващи инженери и студенти без специални умения по програмиране.
3. Да се разработи изчислително ядро, което да разпознава, оптимизира и изчислява описания чрез програмен код функционален изчислителен модел.
4. Да се разработи облачна изчислителна платформа с уеб интерфейс, която да осъществява връзката между потребителите и изчислителното ядро.
5. Да се състави библиотека от готови модули за решаване на различни задачи от областта на строителната механика и проектирането, които да се внедрят за ползване в обучението и практиката.

I.2.3 Обхват

Разработката обхваща различни задачи от математиката, информатиката и строителната механика, които имат отношение към автоматизирането изчисляване на строителните конструкции. Разгледани са следните въпроси:

- изчисления в машинна среда, точност, грешки, изчислителни проблеми и начини за тяхното преодоляване;
- числени методи и алгоритми с приложение в строителната механика;
- формални езици и граматики; лексикален и синтактичен анализ; теория на автоматите и методи за транслация;
- изчисляване на строителни конструкции с начални несъвършенства, геометрична и физична нелинейност с помощта на функционално моделиране.

При разглеждането на съответните теми, включени в обхвата, освен литературен обзор е направен и сравнителен анализ на използваните средства и методи от гледна точка на разглежданите в дисертацията проблеми, като някои от параграфите съдържат нови разработки, свързани с поставените за решаване задачи.

I.3 Етапи на дисертационния труд

Повечето научни разработки се изграждат на следния принцип – първо се формулира проблема, а след това се търсят методите за неговото решаване. Тъй като предметът на настоящата дисертация е изграждането на „уни-

версално“ изчислително средство, с възможност за решаване на широк кръг задачи от строителната механика, тук се подхожда в обратен ред: Първо ще се дефинират необходимите теоретични основи в областта на математиката, информатиката и строителната механика. След това ще се реализират програмно съответните алгоритми и числени методи и накрая ще се приложат за решаване на различни проблеми в проектирането на строителни конструкции. В тази връзка, дисертационният труд обхваща следните етапи:

1. Проучване
   - Анализ на наличните програмни средства за разработване на инженерни изчисления, техните предимства и недостатъци;
   - Формулиране на концепция за нова изчислителна среда, която да направи по ефективно разработването на програмни средства в областта на строителната механика;
   - Изясняване на теоретичните основи от математиката и информатиката. Проучване на свойствата на машинните изчислителни среди;
   - Изясняване на теоретичните основи и предпоставки от строителната механика.
2. Разработка:
   - Разработване и внедряване на ефективни числени алгоритми;
   - Разработване на програмен език (наречен Calyx) за управление на системата;
   - Програмна реализация на изчислителното ядро;
   - Програмиране на софтуерна платформа, която ще осигурява взаимодействието на потребителите с изчислителното ядро.
3. Приложение и внедряване:
   - Разработване на програмни процедури за изчисляване на строителни конструкции;
   - Внедряване на процедурите в проектантската практика;
   - Внедряване на платформата като средство за обучение по строителна механика, с което студентите да могат да експериментират.

I.4 Методологически особености

Дисертацията е с научно-приложен характер. Тя обединява съществуващи знания в областта на математиката, информатиката и строителната механика за разработване на нови методи и програмни средства за статическо изчисляване и оразмеряване на строителните конструкции. По същество, разработката е интердисциплинарна, но има насоченост и конкретно приложение в областта на строителната механика.

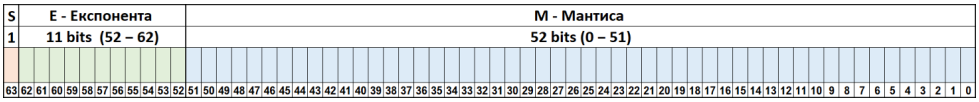
Съвременните методи на строителната механика са ориентирани предимно към прилагането на компютърни системи в проектирането. При това надеждността на изчисленията е изключително важна, с оглед на отго-

ворността, която носят строителните конструктори. Тези методи трябва да се прилагат внимателно, с отчитане на характеристиките на използваната машинна изчислителна среда. Иначе, може да се окаже, че компютърни програми, които привидно са разработени коректно от гледна точка на механиката и работят правилно в общия случай, могат да станат неустойчиви и да дадат грешни резултати в отделни частни случаи.

Това изисква изчисленията да се поставят на солидни теоретични основи в областта на математиката, информатиката и теорията на алгоритмите, които ще бъде направено в следващите няколко глави. Вгаждането пък на универсални числени методи в системата ще ни позволи да дефинираме зависимостите на строителната механика в системата, използвайки директно техния интегрален или диференциален вид, след което да оставим на нейното ядро да извърши необходимите изчисления.

II. Теоретични основи на компютърните изчисления

В математическите и инженерни изчисления се използват предимно реални и комплексни числа в десетична бройна система. Теоретично, те могат да бъдат с произволно избрана точност. Компютрите обаче работят с машинни числа – цели (integer) или с плаваща запетая (floating point). Те се съхраняват в двоична бройна система чрез краен брой битове. За да извършваме точни и надеждни изчисления с компютър, трябва да имаме в предвид начините, по които компютърът работи с числени стойности.



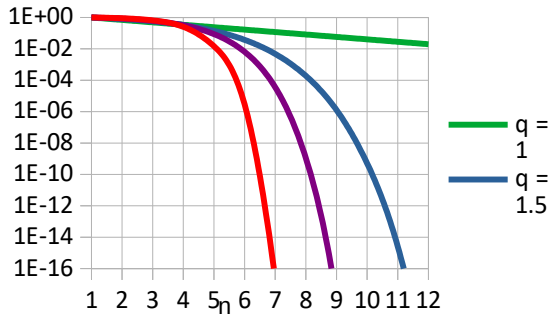
Фигура 1: Машинно представяне на стойности от тип double

Когато съставяме алгоритми за автоматизирано изчисляване на строителни конструкции е важно да знаем, че пресмятанията с компютър са съпроводени от грешки от закръгление. Влиянието им върху крайния резултат може да е съществено и зависи от обусловеността на задачата и числената устойчивост на използвания алгоритъм.

II.1 Алгоритми

Целите и задачите на настоящата дисертация изискват разработването и прилагането на изчислителни алгоритми за решаване на различни задачи от областта на строителната механика. Те трябва да бъдат изградени в съответствие с теорията на алгоритмите и да бъде направен анализ и оценка на тяхното действие – сложност, обусловеност, устойчивост сходимост, точност и пр. За оценка на сходимостта на интерационните алгоритми се използва следният критерий (т.н. “q-сходимост”):

$$\lim_{n\to\infty} \frac{\left|x_{n+1} - x_0\right|}{\left|x_n - x_0\right|^q} = C, \text{ където } 0 < C \leq 1. \qquad (1)$$



Фигура 2: Графики на сходимостта при различни стойности на q

В зависимост от стойността на q се определя типа на сходимост:

$q = 1$ – сходимостта е линейна;

$q \in (1, 2)$ – сходимостта е суперлинейна;

$q = 2$ – сходимостта е квадратична;

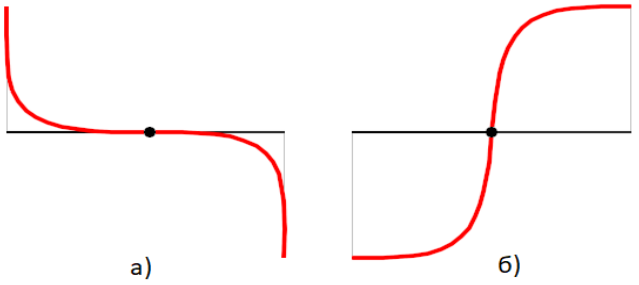
$q = 3$ – сходимостта е кубична.

II.1.1 Оценка на грешката

Обикновено тя се избира от порядъка на работната точност на изчислителната среда. За числа с плаваща запетая и двойна точност, тя е близка до $\varepsilon = 10^{-15}$. Алтернативно, може да се използва следният вариант:

$$\left|x_n - x_{n-1}\right| \le \varepsilon \frac{\left|x_2\right| + \left|x_1\right|}{2} \qquad (2)$$

Проверката за сходимост по стойността на аргумента може обаче да не е достатъчна. Ако функцията има вида, показан на Фигура 3 б), малка промяна на аргумента ще доведе до голяма промяна в стойността на функцията.



Фигура 3: Примерни функции за намиране на корен

Поради това често се налага и проверка по стойността на функцията. Тя обикновено има следния вид:

$$\left|f(x_n) - f(x_{n-1})\right| \le \varepsilon \frac{\left|f(x_2)\right| + \left|f(x_1)\right|}{2} \qquad (3)$$

Използването на средната стойност на функцията намалява риска случайно да „уцелим“ неподходяща стойност, например близка до нула.

III. Числени методи за анализ на строителни конструкции

III.1 Методи за решаване на нелинейни уравнения

III.1.1 Постановка на задачата

Реалните строителни конструкции почти винаги работят с физична и/или геометрична нелинейност. Изследването им в нелинейна постановка обаче често води до уравнения, които са прекалено сложни за да се представи решението в затворен вид. Този проблем се решава чрез прилагането на числени методи. При тях се извършва определен брой итерации, в резултат на които се получава приблизителна стойност на търсеното решение. Съществуват различни методи за намиране на приближение на корените на нелинейни уравнения по итерационен път. Те могат да се групират най-общо в следните типове:

- Методи с вложени интервали (bracketing);
- Методи с последователни приближения;
- Хибридни методи – Методи на Декер, Брент и др.

III.1.2 Критерии за избор на подходящ метод

Всеки от горните методи има своите предимства и недостатъци и е подходящ за различен вид задачи. Нашата цел е да изберем такъв метод, който да е подходящ за разработването на универсален алгоритъм за различни видове уравнения. Това изисква добро познаване на свойствата на отделните методи. Основните критерии за избор са:

- устойчивост;
- универсалност;
- сходимост и бързодействие;
- обем и сложност на изчисленията.

Критериите в горния списък са подредени по важност. За нашата цел ще търсим алгоритъм, който на първо място да е устойчив и универсален. Скоростта е на трето място, поради високата производителност на съвременния хардуер.

III.1.3 Обзорен преглед на основните методи за решаване на нелинейни уравнения

В дисертацията са разгледани подробно, описани и анализирани следните числени методи за решаване на нелинейни уравнения:

- метод на разполовяването (на свиващите изображения)
- метод на хордите и производни на него
- „Илиноис“ метод
- метод на Андерсън-Бьорк
- метод на Ридърс

- метод на Нютън-Рафсон
- метод на Стефенсен
- метод на Халей
- метод на Мюлер
- метод на Островски
- метод на Брент.
- ITP Метод.

Свойствата на отделните методи за решаване на нелинейни уравнения са обобщени в Таблица 1.

Таблица 1: Методи за решаване на нелинейни уравнения

Метод	Тип	Базово уравнение от	Сходимост	q	E	Устойчивост
Разполовяване	Вложени интервали	I-ва степен	линейна	1	1	устойчив
Хорди		I-ва степен	линейна	1	1	непостоянен
Илиноис		I-ва степен	суперлинейна	1.44	1.44	полуустойчив
Андерсън-Бьорк		I-ва степен	суперлинейна	1.7	1.7	полуустойчив
Ридърс		II-ра степен	суперлинейна	2	$\sqrt{2} = 1.41$	полуустойчив
Нютън-Рафсон	Итерационни	I-ва степен	квадратична	2	$\sqrt{2} = 1.41$	неустойчив
Секущи		I-ва степен	суперлинейна	1.62	1.62	неустойчив
ODF [7]		II-ра степен	от IV-ти ред	4	$4^{1/4} = 1.41$	неустойчив
Мюлер		II-ра степен	суперлинейна	1.84	1.84	неустойчив
Халей		II-ра степен	кубична	3	$3^{1/3} = 1.44$	неустойчив
Островски		II-ра степен	от IV-ти ред	4	$4^{1/3} = 1.59$	неустойчив
Брент	Хибридни	I-ва и II-ра	суперлинейна	1.84	1.84	полуустойчив
ITP [20]		I-ва	суперлинейна	1.62	1.62	устойчив

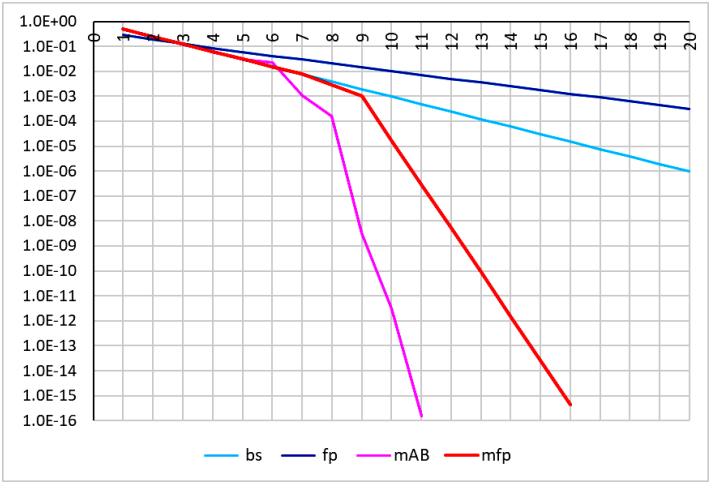
В последната колона е дадена и оценка на устойчивостта, с оглед на използването на дадения алгоритъм като общ метод за решаване на нелинейни уравнения с различни функции и начални условия както и внедряването му в системата.

III.1.4 Предложение за нов хибриден метод

Тъй като характеристиките на съществуващите методи не са достатъчно подходящи за поставените цели е предложен подход, базиран на съчетанието на два алгоритъма – на разполовяването (bs) и хордите (fp). В зависимост от свойствата на функцията в разглеждания интервал ще се превк-

лючва към единия или другия метод като за критерий ще се използва кривината на функцията, изразена като относителна разлика в ординатите.

Методът на хордите обаче има един недостатък и той е, че ако знакът на кривината е постоянен в границите на интервала, единият край остава неподвижен, а прибирането става единствено от другия край. Решение на този проблем дава методът на Андерсън-Бьорк. Така можем да направим още едно подобрение като заменим метода на хордите с този на Андерсън-Бьорк (AB). Резултатът е даден на Фигура 4.



Фигура 4: Сходимост на решението при различните методи

bs = bisection; fp = false position; mfp = bs + fp; mAB = bs + AB

1. Започваме с метода на разполовяването. Намираме средата на интервала;
2. Определяме стойността на функцията в средната точка;
3. Изчисляваме ординатата на хордата в същата точка;
4. Проверяваме дали методът на Андерсън-Бьорк ще даде по-добро приближение, като сравняваме стойността на функцията с тази на хордата;
5. Ако проверката е изпълнена, продължаваме с метода на Андерсън-Бьорк;
6. Аналогично, ако сходимостта на метода на Андерсън-Бьорк е бавна, превключваме обратно на разполовяването.

За да проверим как работи алгоритъмът, ще проведем числени експерименти с голям брой тестови функции и ще сравним резултатите с някои от най-широко използваните методи.

III.1.5 Числени експерименти

Изследвани са седем от най-популярните числени методи за решаване на нелинейни уравнения като получените резултати са сравнени с предложения метод. Изходният код на използваните алгоритми е даден в Приложение 1 (Алгоритми 2 - 9). За числения експеримент са избрани 50 функции, като са използвани литературни данни от подобни сравнения в

Грешка: не е намерен източникът на обръщение, [16], [17], [19], [23] и др. Уравненията и графиките им са дадени Приложение 2. За всяка функция е дадено изменението на грешката в зависимост от броя итерации за всеки от разглежданите алгоритми (Приложение 3). Необходимият брой итерации за достигане на сходимост е даден в табличен вид, както следва:

Обобщение на резултатите

Функ-ция	Уравнение	Брой итерации при метод [1]							
		bs	fp	ill	ab	ITP	mAB	rid	br
Sum	Общ брой итерации	2364	5201	1940	1967	1377	952	1262	1624
Ave	Среден брой	47	104	39	39	28	19	25	32
Max	Максимален брой	50	200	200	200	55	50	84	143
Ф-ция	Метод	bs	fp	ill	ab	ITP	mAB	rid	br

[1] Легенда

bs – Метод на разполовяването (Bisection)
fp – Метод на хордите (False Position)
ill – Илиноис метод (Illinois)
ab – Андерсън - Бьорк (Anderson-Björck)
ITP – Interpolate, truncate, project метод
mAB – Мод. метод на Андерсън-Бьорк (Modified Anderson-Björck)
rid – Метод на Ридърс (Ridders)
bre – Метод на Брент (Brent)

При метода на Ридърс функцията се извиква повече от веднъж на итерация. Поради това стойностите в таблицата показват броя изчислявания на функцията, вместо броя итерации. Това дава по-добра представа за реалната скорост на работа на алгоритъма.

III.1.6 Изводи

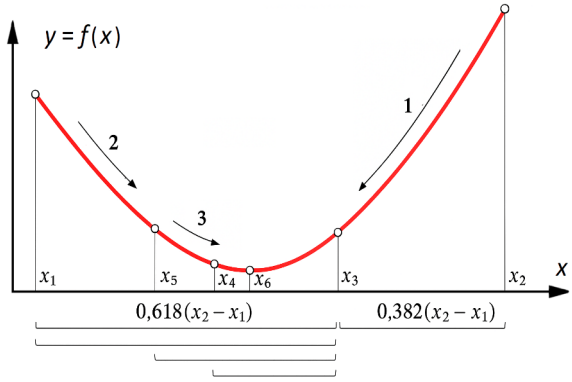
Сходимостта на повечето методи надвишава тази на разполовяването, но при функции с неблагоприятно поведение, може да падне значително под нея. За разглеждания набор от функции предложеният алгоритъм показва най-добър среден и максимален брой итерации и следователно ще бъде вграден в изчислителното ядро на платформата.

III.2 Екстремум на функция (оптимизационна задача)

Задачата за екстремум (минимум и максимум) на функция има широко приложение в проектирането на строителни конструкции, като например – оптимизация на формата на конструктивните елементи (виж примера в т. VI.5), както и при енергетичните методи. Поради това тя представлява важно допълнение към разработваната изчислителна платформа.

За целта ще използваме метод на златното сечение. Той е разработен от Джак Кийфър [27] през 1953 г. и се състои от следните стъпки (Фигура 5):

1. В интервала $[x_1, x_2]$ вмъкваме т. x_3 така, че да го дели в отн. $1 : \varphi$.
2. В подинтервала$[x_1, x_3]$ (по-големия) вмъкваме нова т. x_4 така, че да го разделя също в отношение $1 : \varphi$ и точката да бъде по-близо до x_3.
3. Ако $x_4 < x_3$ свиваме интервала откъм x_3, тоест задаваме $x_2 = x_3$.
4. Ако $x_3 < x_4$ свиваме интервала откъм x_4, тоест задаваме $x_1 = x_4$.
5. Проверяваме за сходимост и ако е изпълнено, връщаме резултата.
6. Иначе, се връщаме на стъпка 2 и повтаряме итерацията.



Фигура 5: Последователност на работа на метода на златното сечение

Сходимостта на алгоритъма е линейна с коефициент на свиване на интервала, равен на $q = \varphi = 0.618$. Като критерий за оценка на грешката и прекратяване на итерациите се използва следният израз:

$$\left|x_2 - x_1\right| < \left(\left|x_3\right| + \left|x_4\right|\right)\sqrt{\epsilon} \tag{4}$$

Реализацията на алгоритъма на програмния език C# 12.0 е дадена в Приложение 1 (Алгоритъм 10).

III.3 Числено диференциране

Диференцирането е една от най-често срещаните математически операции в строителната механика: Деформациите се изразяват като производни на преместванията. Огъващият момент в греда е втора производна на уравнението на еластичната линия, а напречната сила – първа производна на огъващия момент. При метода на крайните елементи е необходимо да диференцираме функциите на формата за да получим напреженията и разрезните усилия в елементите. Много производни обаче не могат да бъдат решени аналитично, което изисква използването на числени методи.

По дефиниция първа производна на функция представлява границата:

$$f'(x) = \lim_{\Delta x \to 0}\left(\frac{f(x+\Delta x) - f(x)}{\Delta x}\right) \tag{5}$$

Ако функцията е диференцируема, от горната формула може да се изведе аналитичен израз за производната във функция от x. В много случаи това не е възможно и тогава се налага използването на числени методи. Най-често се използват формули с крайни разлики (КР):

- дясна несиметрична КР от два възела:

$$f'(x) \approx \frac{f(x+\Delta x) - f(x)}{\Delta x} \tag{6}$$

- лява несиметрична КР от два възела:

$$f'(x) \approx \frac{f(x) - f(x-\Delta x)}{\Delta x} \tag{7}$$

- симетрична КР от два възела:

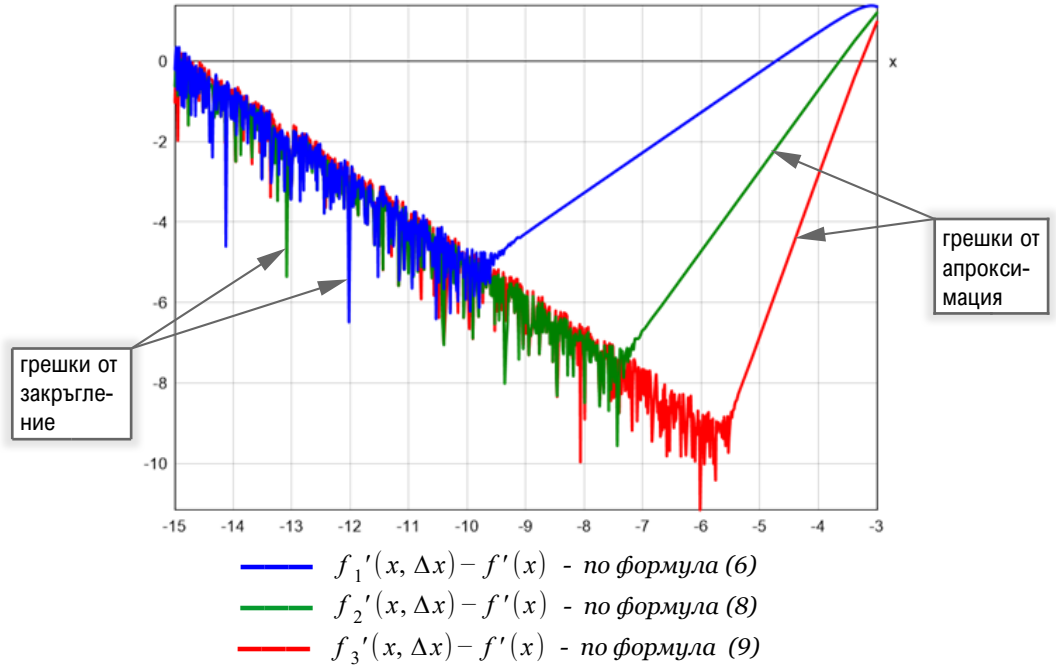
$$f'(x) \approx \frac{f(x+\Delta x) - f(x-\Delta x)}{2 \cdot \Delta x} \tag{8}$$

- симетрична КР от пет възела:

$$f'(x) \approx \frac{-f(x+2\cdot\Delta x) + 8\cdot f(x+\Delta x) - 8\cdot f(x-\Delta x) + f(x-2\cdot\Delta x)}{12\cdot\Delta x} \tag{9}$$

III.3.1 Изчислителни проблеми и оценка на грешката

Един от основните проблеми на числено диференциране е каква стойност да изберем за стъпката Δx. Получените резултати са съпроводени както от грешки от апроксимация, така и грешки от закръгление, които нарастват с намаляване на Δx, както е дадено на Фигура 6 по-долу.



Фигура 6: *Изменение* на грешката при числено *диференциране* в зависимост от Δx

Видно е, че с намаляване на Δx, влиянието на грешките от апроксимация намалява, а това на грешките от закръгление нараства. Абсцисата при минимума на сумарната грешка е оптималната стъпка за съответната крайна разлика. Виждаме, че дори и тогава точността на решението е ограничена. Това показва, че използването само на крайни разлики не е достатъчно и е необходим по-прецизен, по възможност адаптивен метод за изчисление.

III.3.2 Екстраполация на Ричардсън

Това е общ метод за подобряване на сходимостта на апроксимации с числови редове. Той може да се прилага ефективно както при числено диференциране, така и при интегриране и други видове задачи.

Методът комбинира две апроксимации с двойна разлика в гъстотата на мрежата за да оцени и отстрани остатъчната грешка. При това се получава следната рекурентна зависимост:

$$R_n(\Delta x) = R_{n-1}\left(\frac{\Delta x}{2}\right) + \frac{R_{n-1}\left(\frac{\Delta x}{2}\right) - R_{n-1}(\Delta x)}{4^n - 1} \qquad (10)$$

$$f'(x) \approx R_n(\Delta x) + O\left(\Delta x^{2n}\right)$$

Грешката при n-тата итерация е равна на съответния остатък от реда на Тейлър, който от своя страна е равен на разликата между текущата и предишната итерация:

$$O_n = R_n - R_{n-1}$$

Решението продължава докато грешката не стане по-малка от необходимата точност или докато не достигнем заложения максимален брой итерации. Програмна реализация на алгоритъма е дадена в Приложение 1 (Алгоритъм 11). Направено е и допълнително подобрение като рекурсивната процедура е заместена с итеративна с едномерен помощен масив (вместо двумерен) за по-гиляма ефективност В долната таблица направено сравнение на грешките от различни методи за избрани стойности на x:

Таблица 2: Грешки при отделните методи за числено диференциране

$f(x)$	x	$f'(x)$	Calcpad	Mathcad	Mathematica	Scilab	ф-ла (8)	ф-ла (9)
$\cos(x^2)$	1	-1.68	$1.18\cdot10^{-13}$	$6.11\cdot10^{-15}$	$1.28\cdot10^{-14}$	$8.21\cdot10^{-12}$	$8.21\cdot10^{-12}$	$4.53\cdot10^{-13}$
	10	10.13	$5.36\cdot10^{-13}$	$1.17\cdot10^{-14}$	$8.77\cdot10^{-14}$	$1.73\cdot10^{-10}$	$1.73\cdot10^{-10}$	$2.47\cdot10^{-12}$
	100	61.12	$1.37\cdot10^{-11}$	$2.46\cdot10^{-12}$	$1.94\cdot10^{-13}$	$5.26\cdot10^{-10}$	$5.26\cdot10^{-10}$	$1.16\cdot10^{-12}$
	1000	699.99	$6.67\cdot10^{-11}$	Няма решение	$2.90\cdot10^{-11}$	$1.03\cdot10^{-11}$	$1.03\cdot10^{-11}$	$5.85\cdot10^{-11}$
e^x	3	20.09	$1.42\cdot10^{-14}$	$3.89\cdot10^{-15}$	$8.71\cdot10^{-17}$	$2.25\cdot10^{-11}$	$2.25\cdot10^{-11}$	$3.15\cdot10^{-13}$
	30	$1.069\cdot10^{13}$	$2.39\cdot10^{-14}$	0	$3.68\cdot10^{-16}$	$1.79\cdot10^{-10}$	$1.79\cdot10^{-10}$	$6.53\cdot10^{-13}$
	300	$1.94\cdot10^{130}$	$7.65\cdot10^{-12}$	1	$7.23\cdot10^{-15}$	$1.68\cdot10^{-10}$	$1.68\cdot10^{-10}$	$2.96\cdot10^{-10}$

III.3.3 Изводи

Резултатите потвърждават, че разработването на проста, универсална и надеждна процедура за числено диференциране съвсем не е лесна задача. С така избраните универсални параметри, предложеният алгоритъм осигурява сравнително висока точност, съпоставима с другите методи при използване на оптимални параметри. Също така той се характеризира и с висока устойчивост и скорост на изпълнение. Прилагането му е лесно и не изисква задълбочени познания в областта на числените методи и програмирането.

III.4 Числено интегриране

В строителната механика численото интегриране намира широко приложение за решаване на различни видове задачи. Като пример може да посочим определянето на равнодействащата сила на напреженията в дадено сечение със сложна форма и нелинейна зависимост между напрежения и деформации.

III.4.1 Дефиниция

Предмет на численото интегриране е определянето на приблизителна стойност на определения интеграл $\int_a^b f(x)\,\mathrm{d}x$. То може да се прилага дори и когато функцията няма аналитично представяне (но са известни стойностите ѝ в отделни точки). Съществуват и функции, чийто неопределен интеграл не може да се представи чрез аналитичен израз.

За приблизителното определяне на стойностите на интегралите се използват т. нар. квадратурни формули. При тях, интегралът се представя като линейна комбинация от стойностите на функцията в отделни точки.

$$\int_a^b f(x)\,\mathrm{d}x \approx \sum_{i=0}^{n} w_i\, f(x_i) \tag{13}$$

Точките x_i се наричат възли, а числата w_i – тегловни коефициенти. Обикновено те се получават чрез апроксимиране на функцията $f(x)$ с избран интерполационен полином.

III.4.2 Видове квадратурни формули

Квадратурните формули се разделят най-общо на два основни класа:

- Квадратурни формули на Нютън-Коутс;
- Квадратурни формули на Гаус.

Съществуват и други методи за числено интегриране, които в определени случаи могат да бъдат и по-ефективни като:

- Двойно-експоненциална квадратура (Tanh-Sinh);
- Квадратурна формула на функцията на грешката;
- Квадратурна формула на Кленшоу-Къртис;
- Монте-Карло методи.

III.4.3 Проблеми при численото интегриране

Както при останалите числени методи, основният проблем е постигането на висока точност с минимален брой изчисления. За съжаление не съществува „най-добър“ метод, който да обхваща всички възможни случаи-.Нашата цел обаче е да изберем сравнително прост и универсален метод, който да е подходящ за прилагане при различни задачи от строителното проектиране. Той трябва да осигурява добър баланс между устойчивост, прецизност и бързодействие. Едновременно с това трябва да и бъде лесен за използване и от непрофесионалисти. Поради това, ще се насочим към адаптивните методи за интегриране, които осигуряват висока точност и надеждна автоматична оценка на грешката.

III.4.4 Адаптивна квадратура на Симпсън

Алгоритъмът е предложен от G.F. Kuncir през 1962 г [41]. Той използва добре познатата квадратурна формула на Симпсън. Решението се извършва в следната последователност: Първо изчисляваме стойността на интеграла в текущия интервал по формулата на Симпсън. След това го разделяме на две равни части и прилагаме формулата спрямо двата подинтервала. Процесът продължава рекурсивно до достигане на търсената точност. Критерият за сходимост има вида [42]:

$$\left|S(a,c)+S(c,b)-S(a,b)\right|<15\,\epsilon, \tag{14}$$

където ε е желаната точност, а функциите „S” представляват формулите на Симпсън, приложени спрямо отделните подинтервали, както следва:

$$S(a,b)=\frac{x_b-x_a}{6}(f_a+4\,f_c+f_b)$$

$$S(a,c)=\frac{x_c-x_a}{6}(f_a+4\,f_d+f_c)$$

$$S(c,b)=\frac{x_b-x_c}{6}(f_c+4\,f_e+f_b)$$

Програмна реализация на алгоритъма на езика C# 12.0 е дадена в Приложение 1 (Алгоритъм 13).

III.4.5 Адаптивна квадратура на Лобато

Квадратурната формула на Лобато също е от Гаусов тип, но за разлика от останалите, използва възлите и в краищата на интервала. Това я прави подходяща за адаптивна реализация тъй като позволява преизползването на стойностите на функцията в крайните възли.

На база на квадратурата на Лобато, през 2000 г., Гандер и Гаучи разработват адаптивен алгоритъм за числено интегриране [44], който е внедрен в програмните системи Matlab и GNU Octave.

Алгоритъмът използва квадратура на Гаус-Лобато (GL) от 4 възела, която има следния вид:

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \approx \frac{1}{6}\left(f(1)+f(-1)\right)+\frac{5}{6}\left(f\left(-\sqrt{\frac{1}{5}}\right)+f\left(\sqrt{\frac{1}{5}}\right)\right) \quad (15)$$

За оценка на грешката е добавено и второ решение, което използва разширение на Кронрод (GLK) с 3 допълнителни възела:

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \approx \frac{16}{35}f(0)+\frac{11}{210}\left(f(1)+f(-1)\right)+\frac{72}{245}\left(f\left(-\sqrt{\frac{2}{3}}\right)+f\left(\sqrt{\frac{2}{3}}\right)\right)+\ldots$$
$$\ldots+\frac{125}{294}\left(f\left(-\sqrt{\frac{1}{5}}\right)+f\left(\sqrt{\frac{1}{5}}\right)\right)$$

Разликата между двата резултата представлява грешката от апроксимация. Ако проверката за сходимост не е удовлетворена, процедурата се извиква рекурсивно за всеки от шестте подинтервала, затворени между възлите на GLK квадратурата. В Приложение 1 е дадена програмна реализация на алгоритъма, на езика C# 12.0 (Алгоритъм 14).

III.4.6 Адаптивна двойно-експоненциална квадратура

Алгоритъмът е предложен през 1974 година от Такахаши и Мори [36]. Разработен е на базата на сума на Ойлер-Маклорън и използва интегриране чрез трансформация (смяна на променливите). В зависимост от избраната трансформация се различават няколко версии на алгоритъма:

Tanh-Sinh: $x=\tanh\left(\frac{\pi}{2}\sinh t\right)$ – за интеграли от вида $\int_{-1}^{1} f(x)\,\mathrm{d}x$

Exp-Sinh: $x=\exp\left(\frac{\pi}{2}\sinh t\right)$ – за интеграли от вида $\int_{0}^{+\infty} f(x)\,\mathrm{d}x$

Exp-Exp: $x=\exp\left(t-\exp\left(-t\right)\right)$ – за интеграли от вида $\int_{0}^{+\infty} f_1(x)\cdot\exp(-x)\,\mathrm{d}x$

Sinh-Sinh: $x=\sinh\left(\frac{\pi}{2}\sinh t\right)$ – за интеграли от вида $\int_{-\infty}^{+\infty} f(x)\,\mathrm{d}x$

Подобно на квадратурните формули от Гаусов тип, двойно-експоненциалната квадратура се използва успешно в експерименталната математика и изчисленията с висока точност. Тя е изключително ефективна, включително и когато функцията има сингулярности в краищата на разглеждания интервал. Двойно-експоненциалната трансформация проектира интервала [-1, 1] в (−∞, +∞), при което сингулярностите в краищата „изчезват“ в безкрайността.

Проведените сравнителни анализи на Tanh-Sinh квадратурата с други методи за числено интегриране [37] демонстрират значително предимство на този метод, спрямо конкурентите му варианти (Гаус, Errf). Към настоящия момент той се счита за най-ефективния известен метод за числено интегриране.

Както при Гаусовите квадратурни формули, възлите са разположени неравномерно. Абсцисите и теглата се определят сравнително лесно, с помощта на формулите:

$$x_k = \tanh\left(\frac{\pi}{2}\sinh kh\right) \tag{17}$$

$$w_k = \frac{\pi h \cosh kh}{2\cosh^2\left(\frac{\pi}{2}\sinh kh\right)} \tag{18}$$

III.4.7 Сравнителен анализ и избор на подходящ метод

Както споменахме по-горе, за целите на нашата изчислителна платформа е необходим ефективен и универсален алгоритъм, който да осигурява добър баланс между точност, устойчивост и скорост, без да поставя допълнителни изисквания за гладкост и непрекъснатост на функцията. Адаптивните алгоритми са добър вариант тъй като дават висока точност с минимален брой извиквания на разглежданата функция.

За да изберем по-добрият вариант ще проведем числени експерименти и ще сравним получените резултати. Като критерий ще използваме броят извиквания на подинтегралната функция, необходими за постигането на избрана точност на решение. Резултатите са обобщени в следната таблица:

Таблица 3: Сравнителен анализ на адаптивни квадратурни формули

Функция	AdaptiveSimpson		AdaptiveLobatto		Tanh-Sinh	
	n	***ε***	***n***	***ε***	***n***	***ε***
$f_1(x)$	3085	0	517	0	59	1.89×10^{-16}
$f_2(x)$	2349	1.32×10^{-16}	577	1.32×10^{-16}	59	2.64×10^{-16}
$f_3(x)$	14449	1.76×10^{-16}	2077	3.53×10^{-16}	117	7.05×10^{-16}
$f_4(x)$	25177	0	3157	0	55	2.83×10^{-16}
$f_5(x)$	209	3.02×10^{-16}	607	1.36×10^{-15}	15	0.869 [3]
$f_6(x)$	12597	0	1597	1.24×10^{-16}	59	2.49×10^{-16}
$f_7(x)$	–	Грешка [1]	–	Грешка [1]	467	4.46×10^{-9}
$f_8(x)$	∞ [2]	–	∞ [2]	–	467	0.00679 [3]

Легенда:

1) Грешка! Няма решение.

2) Прекалено дълго време за изпълнение (зацикляне).

3) Достигнатата точност е недостатъчна.

В таблицата са използвани следните означения:

n – брой итерации;

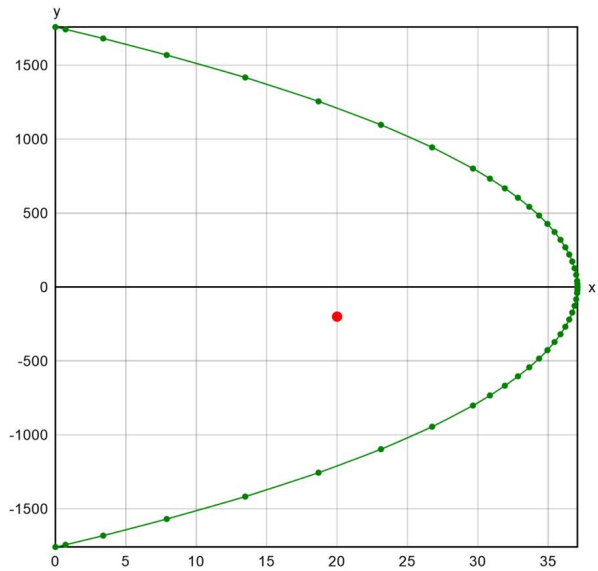
ε – относителна грешка.

Въз основа на горните резултати, адаптивната квадратура на Лобато е избрана като най-подходяща за внедряване в изчислителната система. Тя е най-универсална като област приложение, с висока устойчивост, за сметка на известен компромис със скоростта. Поради изключителната си ефективност, Tanh-Sinh квадратурата също е включена в системата като алтернативна възможност, с изричното указание да се използва само за гладки и непрекъснати функции.

III.5 Адаптивна интерполация

Това е метод, при който дадена функция се апроксимира с дискретно множество от точки в даден интервал, а определянето на междинните стойности се извършва чрез адаптивна интерполация – линейна, spline и др. При това, възлите са разпределени неравномерно, така че грешката от апроксимация да бъде минимална в целия интервал.

Когато имаме тежка за изчисляване функция, този подход може да ускори многократно работата на алгоритъма. Той е много подходящ за изчертаване на графики като показаната на Фигура 7.



Фигура 7: Построяване на графика чрез адаптивна интерполация

Методът е тривиален и лесен за програмиране. Той може да се реализира рекурсивно или итеративно. Ние ще използваме итеративната версия. В Приложение 1 е дадена програмната реализация на алгоритъма , разработена на езика C# 12.0 (Алгоритъм 15).

Адаптивната интерполация е важен елемент от концепцията на функционалното моделиране. Когато веригата от последователно свързани функции стане прекалено дълга и най-вече, когато част от функциите съдържат числени методи, скоростта на изчисленията може да се забави значително. В тези случай е целесъобразно някои от функциите да се апроксимират, което може да стане чрез адаптивна spline интерполация. Така, ще трябва да изчислим стойностите им само за ограничен брой точки, което ще ускори работата на алгоритъма до десетки и дори стотици пъти.

IV. Възможности на изчислителната среда

За да може разглежданата изчислителната среда да се прилага за изчисляване на строителни конструкции в практиката тя трябва да разполага с програмен език, чрез който да се осъществява връзката между потребителя и изчислителното ядро, както и следните основни възможности:

- пълен набор от аритметични, релационни и логически оператори и стандартни функции за работа с „реални“ и комплексни числа;
- дефиниране на променливи и функции от потребителя с един или множество параметри;
- поддръжка на мерни единици в изчисленията с автоматично изравняване на дименсиите при отделните операции;
- готова библиотека с единици за измерване от различни системи: SI (метрична), USCS (US) и Imperial (UK);
- изчисления с вектори и матрици, решения на системи линейни уравнения, търсене на собствени числа и вектори и др.
- блокове за цикли и условни разклонения на програмния поток;
- функции за сумиране на редове, итерации и числени методи, както са описани по-горе в настоящата дисертация;
- изчертаване на графики на функции като диаграми и повърхнини;
- изход на резултатите от изчисленията към отчети в HTML, PDF файл или DOCX формат за MS Word;
- вграждане на динамично и интерактивно съдържание чрез HTML, CSS и JS както и SVG за създаване на параметрични векторни схеми и чертежи;
- генериране на формуляри за попълване на входните данни;

Без тези възможности, изчислителната платформа би останала единствено „на хартия“ – на концептуално ниво, без да може да намери своето приложение в проектантската практика. Поради това по-нататък ще разгледаме накратко най-важните особености при реализацията на гореописа-

ните възможности на новата платформа. При това ще е необходимо да ползваме знания и от други дисциплини – математика и информатика като основа върху която ще изградим работеща среда за решаване на задачи по строителна механика.

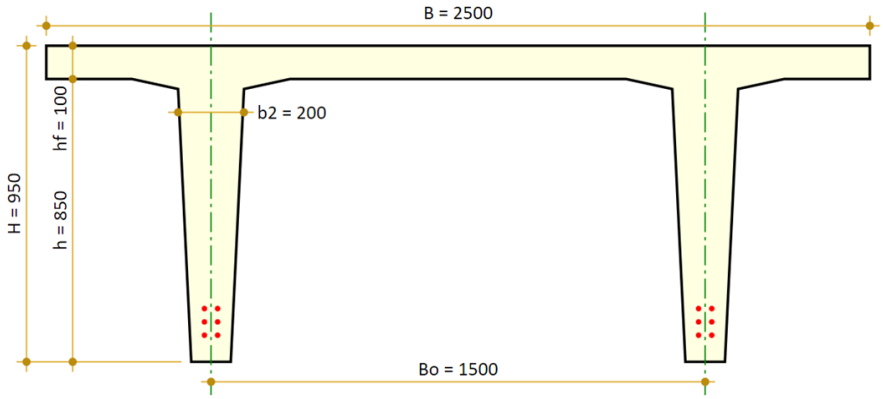
IV.1 Програмен език

Програмният език на системата се нарича **Calyx**. Той има опростен синтаксис, така че да може да се използва от строителни инженери без специални познания по програмиране. Едновременно с това той трябва да осигурява и достъп до всички възможности на изчислителната среда.

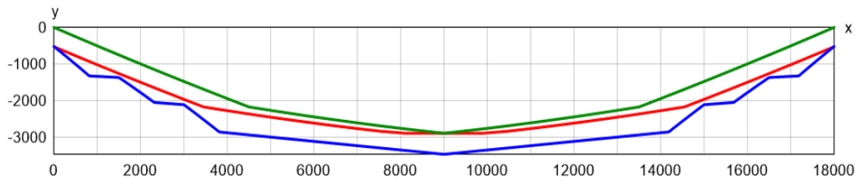
IV.1.1 Класификация и характеристики

Calyx е опростен, тясно специализиран, хибриден език за процедурно/функционално програмиране и моделиране на инженерни задачи, с възможности за автоматизирано документиране на изчисленията, директно от изходния код. Описание на възможностите на програмния език и процесите при неговата транслация е дадено в Приложение 4.

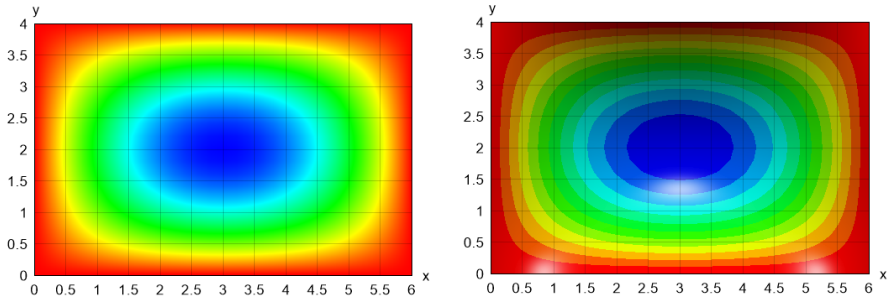
В дисертацията е дадено подробно описание на методите за изчисляване с комплексни числа, мерни единици и визуализация на резултатите. Може да изчертавате графики на функции на една или две променливи като използвате предвидените за целта команди. За разлика от други подобни системи, може да създавате и параметрични векторни чертежи чрез вграждането на SVG графики и дори анимации.



Фигура 8: Параметрична схема на ТТ панел по зададени размери със SVG



Фигура 9: Паралелни диаграми на покритието на М-диаграма на Ст.Б. напрегната греда.



а) Преливащо оцветяване *б) Изохромни ивици*

Фигура 10: Диаграма на преместванията в правоъгълна плоча

IV.2 Вектори и матрици

Векторите и матриците в Calcpad са реализирани по ефективен начин чрез динамични структури и съвременни алгоритми, осигуряващи удобство на работа, оптимално използване на паметта и висока скорост на изчисленията. Това е осъществено при максимално използване на възможностите на езика за програмиране C# и принципите на обектно-ориентираното програмиране – капсулиране, наследяване, полиморфизъм, предефиниране на функции и оператори и др.

Програмната реализация е съобразена със специфичните изисквания за прилагането им в областта на строителната механика и инженерните изчисления. При тях е характерно използването на вектори и матрици с големи размери, при които малък брой елементи са запълнени с ненулеви стойности. Това дава възможност за оптимизация в две насоки:

1. Нулевите стойности не се съхраняват, което води до икономия на памет.

2. При много от операциите нулевите стойности не се обработват, което пък води до икономия на време.

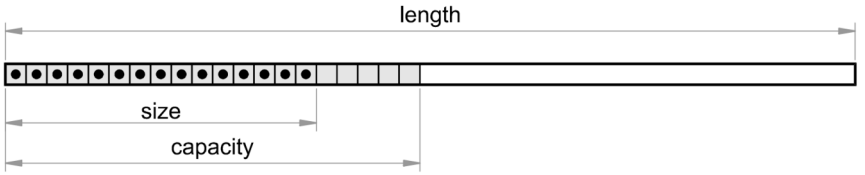
Трябва да се има в предвид и, че съвременните процесори разполагат с по няколко изчислителни ядра, които могат да работят едновременно и независимо едно от друго. Поради това за някои по-тежки изчисления се използват паралелни алгоритми, което допълнително увеличава тяхната производителност.

IV.2.1 Вектори

С оглед на поставените по-горе цели директното използване на блок от последователна памет тип статичен масив (Array) не е подходящо, тъй като още при създаването се заделя цялата памет с посочената дължина. Използването на динамични структури от данни като „свързан списък“, решава този проблем, но поставя друг. Достъпът до елемент по даден индекс изисква последователно обхождане на всички елементи до търсения, което е операция със сложност по време $O(n)$.

Вътрешна структура и организация на паметта

Като оптимален вариант между разход на памет и време, векторът в Calcpad е реализиран като „динамичен“ масив, който нараства според нуждите на потребителя. Той има структурата, показана на Фигура 11.



Фигура 11: Машинно представяне на вектор

Означенията на фигурата по-горе са както следва:

length – математическата дължина на вектора;

capacity – размерът на заделената статична памет като масив;

size – размерът на действително използваната област от вектора, извън която всички елементи гарантирано са равни на нула.

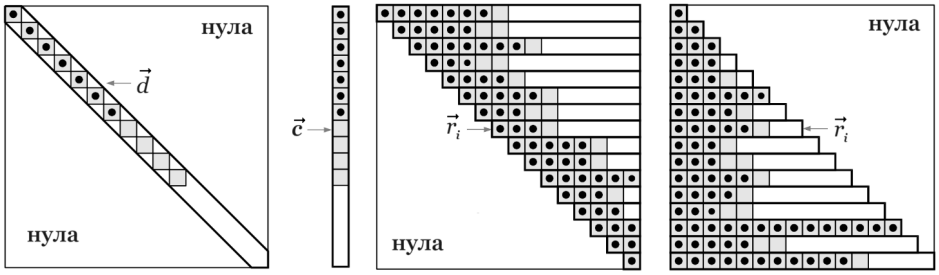
Операции с вектори

Резултатът от аритметичните, релационни и логически оператори върху вектори е вектор, който съдържа резултатите от поелементното изпълнение на оператора. Ако векторите са с различна дължина, по-късият се допълва с нули до размера на по-дългия. Скоростта на изчисленията е оптимизирана допълнително чрез използване на SIMD инструкциите на процесора.

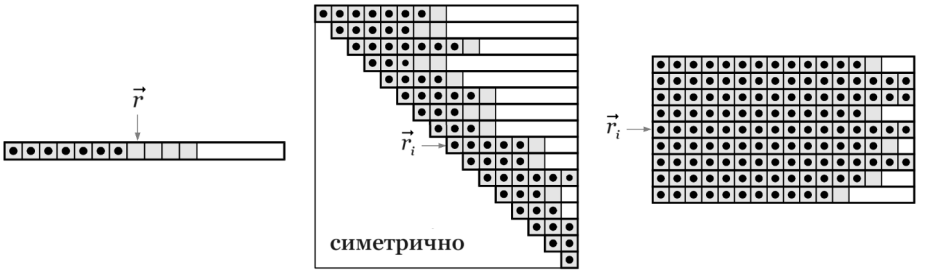
IV.2.2 Матрици

Аналогично на едномерните масиви при векторите, статичните двумерни масиви не са подходящи за програмна реализация на матрици за инженерни изчисления. Трябва да се има в предвид и, че много често се използват специални видове матрици, като единични, диагонални, триъгълни (горна и долна), матрица-стълб/ред и най-вече - симетрични, частично запълнени матрици. Съхраняването и обработването на такива матрици в двумерни масиви би представлявало огромен преразход на памет и време.

Тяхната структура е представена на Фигура 12:



а) б) в) г)



д) е) ж)

а) диагонална матрица - $M[i,j] = d[i]$, ако $i = j$ и 0, ако $i \neq j$;

б) матрица-стълб - $M[i,j] = c[i]$, ако $j = 1$, иначе – грешка;

в) горна триъгълна матрица - $M[i,j] = r_i[j - i + 1]$, ако $j \geq i$, иначе – 0;

г) долна триъгълна матрица - $M[i,j] = r_i[j]$, ако $j \leq i$, иначе - 0;

д) матрица-ред - $M[i,j] = r[j]$, ако $i = 1$, иначе – грешка;

е) симетрична матрица - $M[i,j] = r_i[j - i + 1]$, ако $i \geq j$, иначе = $r_i[i - j + 1]$;

ж) правоъгълна матрица - $M[i,j] = r_i[j]$;

Фигура 12: Машинно представяне на матрици

Операции с матрици

С изключение на умножението, което ще разгледаме отделно, всички аритметични, релационни и логически оператори се изпълняват поелементно. Като резултат се получава матрица с резултатите от действието на оператора върху отделните елементи на матриците – операнди. При бинарните операции, матриците трябва да имат еднакви размери.

Умножение на матрици

Матричното умножение се поддава лесно на паралелизация, което води до няколкократно ускоряване на алгоритъма. Освен това, изчисленията могат да бъдат паралелизирани и в самото ядро чрез векторизация, посредством SIMD инструкции (Single Instruction Multiple Data). Това става като данните се подават на блокове с размерите на разширения регистър, а всеки блок се обработва на един такт. Съвременните процесори могат да работят с поне четири числа тип double едновременно. И двата метода за паралелизация се използват интензивно в разработваната платформа.

Някои специални функции с матрици

С оглед на приложението на настоящата разработка в областта на инженерните изчисления, са разгледани и някои специални матрични задачи, които съответно са програмирани и внедрени в платформата:

- Норма и число на обусловеност;
- Декомпозиция (факторизация);
  - Декомпозиция на Холецки;
  - LDLT декомпозиция;

    - LU декомпозиция;
    - QR декомпозиция;
    - Декомпозиция по сингулярни стойности (SVD);
- Ранг и детерминанта;
- Системи линейни уравнения;
    - Чрез LU или LDLT декомпозиция;
    - Чрез декомпозиция на Холецки;
    - Чрез метода на преобусловения спрегнат градиент;
- Системи нелинейни уравнения;
    - Метод на последователните приближения;
    - Метод на последователните натоварвания;
    - Метод на Нютон-Рафсон;
    - Метод на Бройден;
    - Метод на дъгите (arc-length);
- Собствени числа и собствени вектори;
    - Симетричен QL алгоритъм с неявно отместване;
    - Метод на Ланцош.

IV.3 Функционално моделиране

IV.3.1 Основни принципи

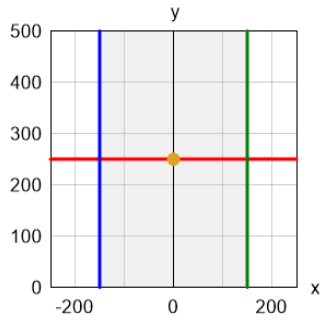
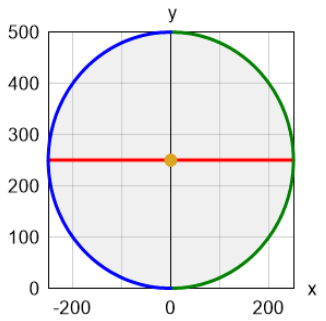
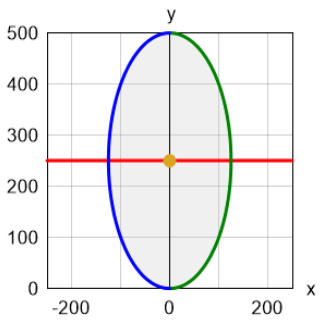
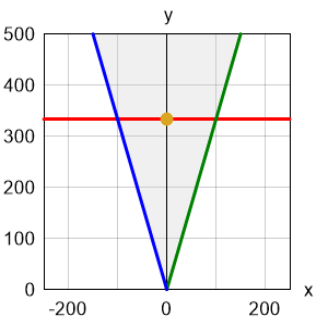
Един изчислителен алгоритъм може да се опише императивно – като последователност от изчислителни изрази или функционално – като зависимостите между отделните параметри на задачата се дефинират като функции. По този начин се изгражда **функционален математически модел** на дадената задача, който дава само общия метод за нейното решаване. В последствие, може да използваме този модел за изчисляване на различни частни случаи, с конкретни входни данни. Освен функции, моделът може да включва и числени методи, които дава възможност да се дефинират директно интегрални и диференциални зависимости. Calcpad дава възможност за прилагане и на двата подхода – императивен и функционален.

IV.3.2 Пример за функционален подход

Като пример може да се състави алгоритъм за определяне на инерционните характеристики на напречно сечение с произволна форма. За тази цел ще представим контура на сечението като функция. Кодът на алгоритъма е даден в Приложение 1 (Алгоритъм 16).

Чрез него последствие ще можем лесно да разширяваме възможностите на програмата с нови сечения. За всяко едно от тях ще трябва да добавим само един ред в кода, който да съдържа функцията на новия контур. Чрез промяна на променливата Shape може да изберем кое от сеченията да изчисляваме. Това разбира се може да се направи в удобен за потребителя

вид, като например списък за избор. Резултатите от работата на програмата за различните случаи са обобщени в следната таблица:

Форма	Схема	Инерционни характеристики
Правоъгълник $Shape = 0$ $b(z;\ h) = b$		Площ - $A = 1500\ cm^2$ Център на тежестта - $z_c = 250\ mm$ Инерционен момент - $I = 312500\ cm^4$ Съпрот. момент - $W = 12500\ cm^3$
Кръг $Shape = 1$ $b(z;\ h) = \ldots$ $2\sqrt{\left(\frac{h}{2}\right)^2 - \left(z - \frac{h}{2}\right)^2}$		Площ - $A = 1963.5\ cm^2$ Център на тежестта - $z_c = 250\ mm$ Инерционен момент - $I = 306796\ cm^4$ Съпрот. момент - $W = 12271.8\ cm^3$
Елипса $Shape = 2,\ k = 0.5$ $b(z;\ h) = \ldots$ $2 \cdot k \cdot \sqrt{\left(\frac{h}{2}\right)^2 - \left(z - \frac{h}{2}\right)^2}$		Площ - $A = 981.75\ cm^2$ Център на тежестта - $z_c = 250\ mm$ Инерционен момент - $I = 153398\ cm^4$ Съпрот. момент - $W = 6135.92\ cm^3$
Триъгълник $Shape = 3$ $b(z;\ h) = \frac{z \cdot b}{h}$		Площ - $A = 750\ cm^2$ Център на тежестта - $z_c = 333.3\ mm$ Инерционен момент - $I = 104167\ cm^4$ Съпрот. момент - $W = 3125\ cm^3$

IV.3.3 Предимства и недостатъци

Предимство на функционалното програмиране е, че спестява доста писане на код. Ако имаме 20 различни сечения и решим да реализираме горния пример императивно, ще трябва да съставим 20 отделни подпрограми. Освен че са по-кратки, функционалните процедури са и по-универсални и ни позволяват да преизползваме многократно един и същ код.

Основният недостатък е, че резултатите са трудно проверяеми. При императивния поход всяка стъпка от изчисленията се документира автоматично в записката със заместени стойности на променливите. При функционалния подход това не е възможно.

Друг проблем е, че скоростта на изпълнение на програмата се забавя, особено когато ползваме числени методи. Освен това, числените методи са натоварени и с грешки от апроксимация, които, макар и малки, могат да се отразят неблагоприятно върху точността на крайния резултат. При компетентно и внимателно програмиране обаче проблемите със скоростта и точността на решението могат да бъдат преодолени.

IV.3.4 Приложение

Може да комбинираме няколко функционални модела в един по-сложен. По този начин, може да изградим йерархия от модели, които да организираме в библиотека. За целите на строителната механика и проектирането, те могат да бъдат систематизирани в следните групи:

- модели на материала – например като обобщени зависимости „напрежения-деформации“;
- модели на напречното сечение – чрез обобщени зависимости за формата $b(z;\ h)$. Може да имаме няколко различни материала в едно сечение, като например при стоманобетонните и комбинираните конструкции;
- модели на геометрията – чрез зависимости от вида $y(x)$, а ако имаме променливо сечение – $b(z;\ h(x))$ и пр.;
- модели на натоварването – например чрез интензитета му по дължина на елемента $q(x)$;
- модели на механиката – зависимости между напрежения, разрезни усилия, премествания, завъртания, кривини и пр.

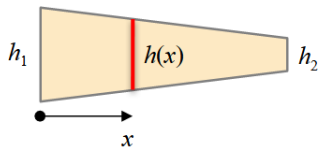
На базата на шбазова библиотека, могат лесно да бъдат програмирани различни видове задачи чрез вмъкване на съответните функции.

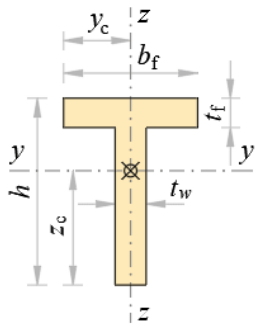
IV.3.5 Програмна реализация

При функционалното програмиране, точната последователност на изчисляване не е строго дефинирана и остава скрита за потребителя. Тук е възможно да се приложат различни подходи. За илюстрация ще разгледаме следния пример:

Пример

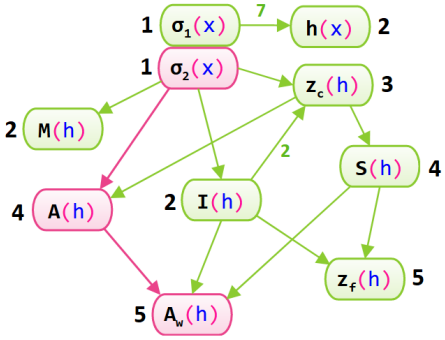
```
'Напрежения в греда с Т сечение, постоянна осова сила и променлива височина h1 ÷ h2
b1 = bf - b
Af = b1*hf
h(x) = h1 + i*x
zf(h) = h - hf/2
Aw(h) = b*h
A(h) = Aw(h) + Af
S(h) = Aw(h)*h/2 + Af*zf(h)
zc(h) = S(h)/A(h)
I(h) = b*h^3/12 + b1*hf^3/12 +
       Aw(h)*(h/2 - zc(h))^2 +
       Af*(zf(h) - zc(h))^2
σ1(x) = N/A(h(x)) + M(x)/I(h(x))*zc(h(x))
σ2(x) = N/A(h(x)) - M(x)/I(h(x))*(h(x) - zc(h(x)))
```





Граф на програмния поток

Видно от горния пример е, че някои функции зависят от други функции, които на свой ред зависят от трети такива и т.н. Тези зависимости са еднопосочни – от извикващата функция, към извикваната. Ако изобразим тези връзки графично, ще получим математически обект, който се нарича „**ориентиран граф**“, а когато става въпрос за структура на програма – „**граф на програмния поток**“ (PFG). За дадения по-горен пример, той изглежда, както е показано на Фигура 13 по-долу:



Фигура 13: Граф на програмния поток за разглеждания пример

Тези структури се изучават от теорията на графите, която е част от дискретната математика. Нейните методи могат да бъдат приложени успешно за изследване, оптимизация и изчисляване на програмния код.

Преди да преминем към самите изчисления, трябва да направим някои важни проверки. Една от тях е за наличие на неявна рекурсия. В Calcpad тя е забранена, тъй като функциите са едноредови и няма как да добавим условие за прекъсване. В теорията на графите, това се свежда до **проверка за цикличност в ориентиран граф**.

Методи за изчисляване

Изчисляването на резултатите се извършва чрез последователно обхождане на възлите от графа в определен ред. Това може да стане по различни начини, като основните от тях са два:

1. Предварително извършваме топологично сортиране на графа. Това е наредба, при която всеки връх се намира след предшествениците си. На Фигура 13 възлите са номерирани в тяхната топологична последователност. След това, извършваме изчисленията в обратен на номерирането ред. Ефектът от този метод е все едно да заместим функционалната процедура с еквивалентна на нея императивна.
2. Рекурсивно спускане с връщане. Това е по-интуитивният метод за изчисляване на функции. При него сканираме последователно тялото на функцията и когато стигнем до друга функция, на свой ред я отваряме за изчисляване. Продължаваме процеса, докато не достигнем функция, която не зависи от други. Изчисляваме я и връщаме резултата по обратния път. В теорията на графите, това е еквивалентно на **обхождане на граф в дълбочина**. На Фигура 13, с червено е показан пътят при първото обхождане, което ще направим при изчисляването на напреженията.

Предимството на първия метод е, че не използва рекурсия и работи доста по-бързо. Недостатъкът му е, че при компилирането се генерира допълнителен междинен код.

Оптимизация на програмния поток

От схемата на Фигура 13 е видно, че много от пътищата в графа се пресичат. Това означава, че при рекурсивния вариант на изчисление, ще трябва да преминем по няколко пъти през едни и същи възли. Иначе казано, ще трябва да изчисляваме една и съща функция многократно за един и същ аргумент. Това е излишно, тъй като всеки път ще получаваме еднакъв резултат. Ако проследим последователността на изчисленията, ще видим, че имаме общо 55 извиквания на 10 различни функции. От това следва, че вторият алгоритъм ще бъде около 5.5 пъти по-бавен от първия. Бихме могли обаче да подобрим този резултат значително, ако използваме техника, наречена „динамично оптимиране“.

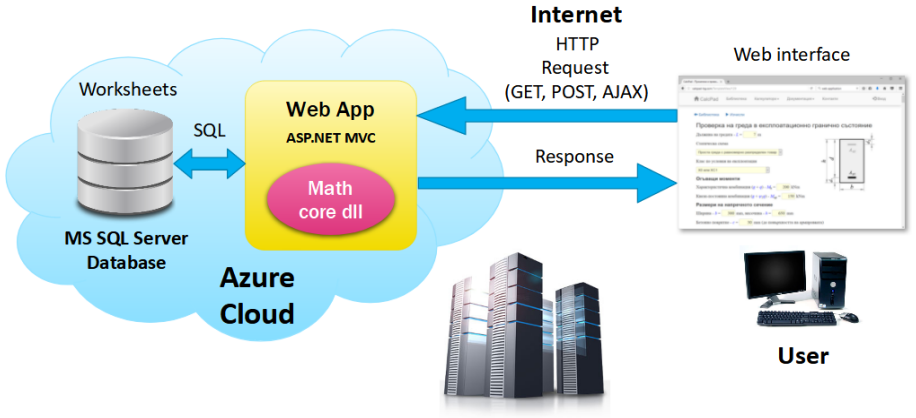
Идеята е да използваме допълнителен буфер, в който да записваме вече изчислените стойности и когато извикаме дадена функция повторно със същия аргумент, тя да върне готовия резултат, вместо да го изчислява отново.

За целта е най-удобно да използваме хеш таблица. По този начин спестяваме не само повторното изчисляване на самата функция, но и спускането в дълбочина по останалите възли на графа и изчисляването на още функции. Така, с малко допълнителна памет, може да спестим доста изчислителна работа и да подобрим значително скоростта на изпълнение.

В допълнение, всяка една функция се компилира за по-бързо изпълнение, при което се прилагат допълнителни оптимизации като сгъване и разпространение на константи. Когато даден израз съдържа само константни операнди, той може да бъде изчислен в процеса на компилиране да бъде заместен с получената стойност.

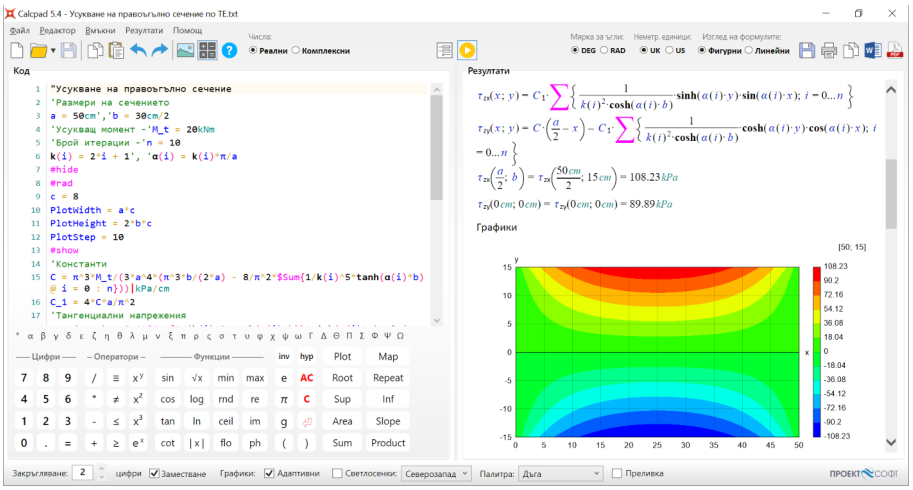
V. Програмируема платформа за изчисляване на конструкции в облачна среда

Въз основа на предложения в настоящата дисертация програмен език, числени методи и алгоритми е разработена облачна изчислителна платформа, наречена Calcpad. С нейна помощ строителни инженери могат да създават, качват и използват програми за изчисляване и оразмеряване на строителни конструкции, в облачна среда. Платформата е разработена на базата на Microsoft ASP.NET Core, с обектно ориентиран програмен език C# и Microsoft Visual Studio. Принципна схема на работа на платформата е показана на Фигура 14.



Фигура 14: Принципна схема на облачното приложение

Платформата има и десктоп версия, разработена на .NET 8.0 и WPF, отново с помощта на езика C#. Използва същото изчислително ядро, но вече с десктоп интерфейс, дефиниран на езика XAML. Приложението може да бъде изтеглено и инсталирано на компютъра на потребителя, където да се използва без необходимост от постоянна връзка с Интернет.



Фигура 15: Десктоп приложение

Инсталационните файлове, документацията и изходния код на софтуера са достъпни на следните линкове:

Уеб версия - https://calcpad.eu

Десктоп версия - https://calcpad.eu/download/calcpad-setup-en-x64.zip

Изходен код - https://github.com/Proektsoftbg/Calcpad

https://sourceforge.net/p/calyx/code/ref/master

Софтуерът е с отворен код и се разпространява безплатно с MIT лиценз.

VI. Приложение в изчисляването на строителни конструкции

VI.1 Изследване на колона с променливо напречно сечение с произволна форма с отчитане на геометрична и физична нелинейност

VI.1.1 Постановка на задачата

Задачата е да се разработи процедура на Calcpad за изчисляване и оразмеряване на колони с произволна форма на напречното сечение с променлива височина по дължина на елемента. Изчисленията да се извършат с отчитането на нелинейните свойства на материала, началните несъвършенства и ефектите от II ред.

VI.1.2 Теоретични основи и методи за анализ

Проблемът за изчисляването на пръстови елементи с пластични деформации е разгледан подробно в [62]. Относно устойчивостта на елементи с променливо напречно сечение информация може да се намери в [63], където е направен обширен преглед на наличните методи за решение.

Ние ще предложим един по-различен подход, който използва концепцията за функционално моделиране в Calcpad. Първо ще дефинираме необходимите геометрични и физични зависимости като функции. С помощта на вградените в Calcpad числени методи, ще ги представим в обобщен вид. Кратко описание на последователността на изчисляване и използваните формули е дадено по-долу, както следва:

Модел на материала

Свойствата на материала, ще бъдат представени чрез зависимостта „напрежения-деформации“ в общ вид. Зависимостта ще бъде зададена чрез функцията $\sigma(\varepsilon)$. Като пример ще използваме стомана с линейно-пластичен модел на работа:

$\sigma(\varepsilon)=max\left(-f_{yd};\ min\left(\varepsilon\cdot E;\ f_{yd}\right)\right)$, където

E е модула на еластичност на стоманата.

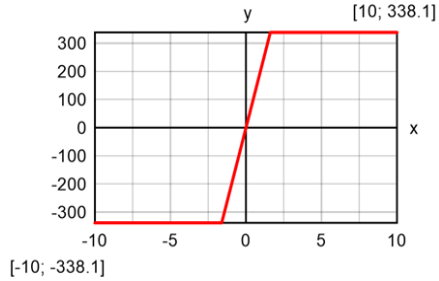
Обикновено се приема: $E=210\,GPa$

$f_{yd}=\frac{f_{yk}}{\gamma_s}$ е изчислителната граница на провлачване;

f_{yk} е характеристичната (нормативна) граница на провлачване;

γ_{ud} е коефициент на сигурност на материала.

Деформацията ε е ограничена в интервала: $\varepsilon\in[-\varepsilon_{ud},\varepsilon_{ud}]$. Приемаме, че разрушението настъпва при достигане на граничната деформация ε_{ud}.



Фигура 16: Работна диаграма на материала

По подобен начин могат да се дефинират и други модели на материала. Те могат да имат различно поведение при опън и натиск със сложен нелинеен характер. За съществуващи конструкции може да се използват и опитно определени работни диаграми.

Модел на сечението

Напречното сечение на елемента ще представим чрез зависимостта $b(z)$, където b е ширината на сечението, $z\in[0,\ h]$ е разстоянието от долния ръб, а h – височината на сечението.

Функцията $b(z)$ за различните видове сечения има вида:

Правоъгълно

$b(h;\ z)=b$, където b е въведената от потребителя ширина.

Кръгло

$$b(h;\ z)=2\cdot\sqrt{\left(\frac{h}{2}\right)^2-\left(z-\frac{h}{2}\right)^2}$$

Елипсовидно

$$b(h;\ z)=\frac{2\cdot b}{h}\cdot\sqrt{\left(\frac{h}{2}\right)^2-\left(z-\frac{h}{2}\right)^2}$$

Трапецовидно

$b(h;\ z)=b+\frac{z\cdot(b_2-b)}{h}$, където b_2 е ширината в горния край.

Т-образен профил

$b(h;\ z)=if\left(z>h-h_2;\ b_2;\ b\right)$, където b_2 и h_2 са размерите на горния пояс.

Двойно-Т профил

$b(h;\ z)=if\left(z>h-h_2;\ b_2;\ if\left(z<h_1;\ b_1;\ b\right)\right)$,

където b_1 и h_1 са размерите на долния пояс.

Тук като параметър сме включили и височината на сечението, тъй като тя ще бъде променлива по дължина на елемента. По аналогичен начин ще може да добавяме и други видове сечения само с по един допълнителен ред в кода. За тази цел обаче геометричните характеристики на сечението трябва да се дефинират чрез общи интегрални зависимости, както следва:

Площ - $A(h)=\int_0^h b(h;\ z)\,\mathrm{d}z$

Статичен момент- $S(h)=\int_0^h b(h;\ z)\cdot z\,\mathrm{d}z$

Център на тежестта - $z_c(h)=\frac{S(h)}{A(h)}$

Инерционен момент - $I(h)=\int_0^h b(h;\ z)\cdot\left(z-z_c(h)\right)^2\mathrm{d}z$

Съпротивителни моменти

За долен ръб - $W_{el,1}(h)=\frac{I(h)}{z_c(h)}$

За горен ръб - $W_{el,2}(h)=\frac{I(h)}{h-z_c(h)}$

Геометрия на елемента

Тя ще бъде представена чрез дължина на елемента – L, функция $e(x)$, която описва отклонението на оста (когато е криволинейна) и функция на изменение на височината на сечението, по дължина $h(x)$. Когато височината се изменя линейно, функцията има следния вид:

$h(x)=h_b+\frac{(h_t-h_b)\cdot x}{L}$, където h_b и h_t са височините на сечението, съответно в началото и края на елемента.

Дори и ако оста на елемента е праволинейна, пак ще ни е необходима функция за да дефинираме производствените и монтажните отклонения, дължащи се на началните несъвършенства. По отношение на граничните условия засега ще приемем статическа схема – конзола.

Натоварване

По аналогичен начин бихме могли да въведем произволни хоризонтални и вертикални натоварвания като функции, а разрезните усилия да определим чрез числено им интегриране по дължина на елемента.

За опростяване обаче ще изчислим елемента само за хоризонтална сила H и вертикална сила P на върха на конзолата. При това определянето на разрезните усилия се извършва, както следва:

Огъващ момент - $M_{Ed}(x)=H\cdot(L-x)$

Напречна сила - $V_{Ed}(x)=H$

Осова сила - $N_{Ed}(x)=-P$

Носимоспособност на напречното сечение

Тъй като формата на сечението не е фиксирана предварително, ще определим носимоспособността на сечението чрез интегрални зависимости, базирани на функциите на материала и сечението като входни данни.

Разпределението на деформациите по височина на сечението ще дефинираме също като функция. Ще приемем линейно разпределение, в съответствие с хипотезата на Бернули:

$\varepsilon_z(\xi;\ \varepsilon_1;\ \varepsilon_2)=\varepsilon_1+(\varepsilon_2-\varepsilon_1)\cdot\xi$, където

$\xi = z/h$ е относителната височина на нулевата линия, а

ε_1 и ε_2 са деформациите в долния и горния ръб на сечението.

Носимоспособност за чисто огъване:

$$M_{pl,Rd}=\int_0^h \sigma\left(\varepsilon_z\left(\frac{z}{h};\ \varepsilon_{ud};\ -\varepsilon_{ud}\right)\right)\cdot b(h;\ z)\cdot(z_c(h)-z)\,\mathrm{d}z$$

Гранична осова сила:

Центричен опън - $N_{pl,Rd,t}=\int_0^h \sigma\left(\varepsilon_z\left(\frac{z}{h};\ \varepsilon_{ud};\ \varepsilon_{ud}\right)\right)\cdot b(h;\ z)\,\mathrm{d}z$

Центричен натиск - $N_{pl,Rd,c}=\int_0^h \sigma\left(\varepsilon_z\left(\frac{z}{h};\ -\varepsilon_{ud};\ -\varepsilon_{ud}\right)\right)\cdot b(h;\ z)\,\mathrm{d}z$

За нецентричен опън/натиск трябва да се отчете влиянието на осовата сила върху носимоспособността на огъване. За стоманени елементи в БДС

EN 1993-1-1 [71] са дадени приблизителни формули за някои видове сечения. Например, за правоъгълно сечение формулата има следния вид:

$$M_{N,Rd} = M_{pl,Rd} \cdot \left[1 - \left(\frac{N}{N_{pl,Rd,c}} \right)^2 \right]$$

Ние обаче ще използваме по-общ подход, който обикновено се прилага при стоманобетонните и комбинираните конструкции. Предварително ще построим интеракционната диаграма за носимоспособност при огъване с осова сила. Това става чрез вариране на диаграмата на деформациите и интегриране на съответната диаграма на напреженията в рамките на сечението. За да обхванем всички възможни гранични състояния ще процедираме в следната последователност:

1. Започваме процедурата от състояние на чист опън: $\varepsilon_1 = \varepsilon_{ud}$, $\varepsilon_2 = \varepsilon_{ud}$;
2. Изменяме деформацията в долния ръб от опън до натиск:
   $\varepsilon_1 = \varepsilon_{ud} \rightarrow -\varepsilon_{ud}$, $\varepsilon_2 = \varepsilon_{ud}$;
3. Изменяме деформацията в горния ръб от опън до натиск:
   $\varepsilon_1 = -\varepsilon_{ud}$, $\varepsilon_2 = \varepsilon_{ud} \rightarrow -\varepsilon_{ud}$;
4. Завършваме със състояние на чист натиск: $\varepsilon_1 = -\varepsilon_{ud}$, $\varepsilon_2 = -\varepsilon_{ud}$;

За удобство при по-нататъшните изчисления ще представим деформираното състояние на сечението чрез обобщен параметър на деформациите ε, както следва:

- за долния ръб на сечението: $\varepsilon_1(\varepsilon) = \min(\varepsilon + \varepsilon_{ud};\ \varepsilon_{ud})$
- за горния ръб на сечението: $\varepsilon_2(\varepsilon) = \max(-\varepsilon_{ud};\ \varepsilon - \varepsilon_{ud})$
- по височина на сечението: $\varepsilon_{zz}(z;\ \varepsilon) = \varepsilon_z\left(\frac{z}{h};\ \varepsilon_1(\varepsilon);\ \varepsilon_2(\varepsilon)\right)$

Така ще получим всички описани по-горе етапи единствено като изменяме обобщения параметър на деформациите в границите:

$\varepsilon = [\varepsilon_{max}; -\varepsilon_{max}]$, където

Граничните разрезни усилия ще представим като функция на обобщения параметър:

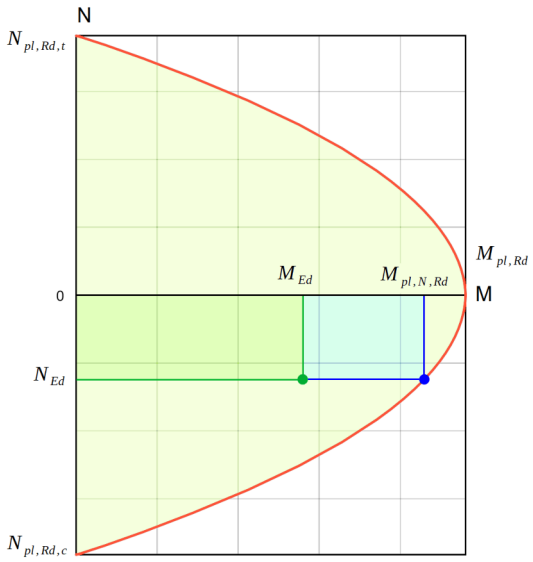
Огъващ момент: $M_{pl,Rd}(\varepsilon) = \int_0^h \sigma\left(\varepsilon_{zz}(z;\ \varepsilon)\right) \cdot b(h;\ z) \cdot \left(z_c(h) - z\right) \mathrm{d}z$

Осова сила: $N_{pl,Rd}(\varepsilon) = \int_0^h \sigma\left(\varepsilon_{zz}(z;\ \varepsilon)\right) \cdot b(h;\ z) \mathrm{d}z$

Накрая ще построим интеракционната диаграма с командата „`$Plot`“:

$$\$\mathrm{Plot}\left\{ M_{pl,Rd}(\varepsilon) \,|\, N_{pl,Rd}(\varepsilon) @ \varepsilon = -\varepsilon_{max} : \varepsilon_{max} \right\}$$

За стоманени сечения, тя има следния общ вид:



Оттук-нататък проверката за носимоспособност за конкретно външно натоварване M_{Ed} и N_{Ed} се свежда до чисто геометричната задача да се провери дали точката с координати [M_{Ed}, N_{Ed}] е вътрешна за контура на интеракционната диаграма.

За тази цел през точката на външното натоварваме ще построим права линия, която ще пресечем с интеракционната диаграма. Абсцисата на пресечната точка представлява търсения граничен огъващ момент, който ще сравним с този от външни въздействия. Тук отново ще използваме числен метод, както следва:

Обобщена деформация, съответстваща на осовата сила:

$$\varepsilon_N = \$\mathrm{Root}\left\{N_{pl,Rd}(\varepsilon) = -N_{Ed};\ \varepsilon \in \left[0;\ \varepsilon_{max}\right]\right\}$$

Граничен огъващ момент:

$$M_{pl,N,Rd} = M_{pl,Rd}\left(\varepsilon_N\right)$$

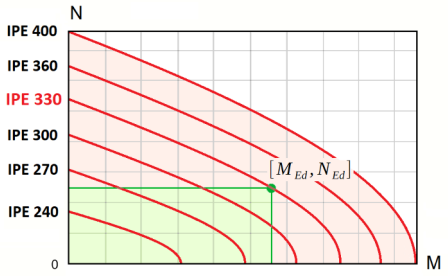
Проверка за носимоспособност:

$$M_{Ed} \le M_{pl,N,Rd}$$

Така, с помощта на Calcpad и принципите на строителната механика, съставихме обща процедура за проверка на стоманени сечения с произволна форма. Тя може да се прилага и за профили, които не попадат в обхвата на т. 6.2.9 от БДС EN 1993-1-1 [71].

Интеракционните диаграми дават по-богата и нагледна представа за носимоспособността на сечението, отколкото оразмерителните формули. За горещо-валцуваните профили, чиито размери са стандартизирани, задачата може да се реши еднократно като предварително се построят номограми за отделните видове стомани.

С тяхна помощ практикуващите инженери ще могат бързо и удобно да оразмеряват стоманени профили без да извършват каквито и да било изчисления. Чрез нанасяне на външното натоварване като точка върху номограмата, ще може директно да се отчете подходящия профил, както е показано на схемата:



Диаграма момент-кривина

Това е важна характеристика на сечението, която ще ни позволи да отчетем физичната нелинейност в по-нататъшните изчисления. Кривината в дадено сечение представлява наклона на линията на деформациите. Нейното определянето е тривиално. Огъващият момент от своя страна ще получим чрез интегриране на съответната диаграма на напреженията по височина на сечението. Като свободен параметър ще използваме деформацията в долния ръб на сечението ε_b. Деформацията в центъра на тежестта ще бъде фиксирана така, че да съответства на средната деформация от осова сила. Така получаваме следните изрази за огъващия момент и кривината:

Средна деформация в сечението - $\varepsilon_N(h)=\frac{-N_{Ed}}{A(h)\cdot E}$

Деформация в горния ръб на сечението - $\varepsilon_t(h;\ \varepsilon_b)=\frac{\varepsilon_b\cdot(z_c(h)-h)+\varepsilon_N(h)\cdot h}{z_c(h)}$

Огъващ момент:

$$M_{pl}(h;\ \varepsilon_b)=\int_0^h \sigma\left(\varepsilon_z\left(\frac{z}{h};\ \varepsilon_b;\ \varepsilon_t(h;\ \varepsilon_b)\right)\right)\cdot b(h;\ z)\cdot(z_c(h)-z)\,\mathrm{d}z$$

Кривина - $\theta(h;\ \varepsilon_b)=\frac{\varepsilon_b-\varepsilon_N(h)}{z_c(h)}$

За да може да прилагаме формулите и за променлива височина на сечението тя също е включена като параметър. Може да изчертаем диаграмата като използваме вградената команда в Calcpad:

$$\$\mathrm{Plot}\left\{\theta(h;\ \varepsilon_b)\,|\,M_{pl}(h;\ \varepsilon_b)@\varepsilon_b=\varepsilon_N(h)\ :\ \varepsilon_{ud}\right\}$$

Еластопластична коравина

Коравината на огъване по дефиниция представлява отношението на момента към кривината и се определя по формулата:

$$K(h;\ \varepsilon_b)=\frac{M_{pl}(h;\ \varepsilon_b)}{\theta(h;\ \varepsilon_b)}$$

След като имаме зависимостите между коравина, момент и кривина, може да преминем към следващия етап на изчисляване на елемента.

Начални несъвършенства

Загубата на устойчивост в Ойлерова постановка изисква идеални условия, които в практиката не съществуват. Дори и при пълната липса на хоризонтални натоварвания, елементите на строителните конструкции имат производствени неточности в размерите и отклонения на техните оси от правата линия. Към това в последствие се добавят и монтажните неточности спрямо проектното им положение в конструкцията. И едните и другите обаче подлежат на контрол, а стойностите им са нормативно ограничени. Те са посочени в съответните производствени стандарти или правила за изпълнение и приемане на СМР. Ние може да заложим тези стойности като начални и да изследваме елемента, с отчитане на геометрична нелинейност, съгласно теорията от II ред.

Съгласно БДС EN 1993-1-1 [71] началните несъвършенства се дефинират като:

- глобално начално наклоняване, по формула (5.5):

$\varphi=\varphi_0\cdot\alpha_h\cdot\alpha_m$, където:

$\varphi_0=\frac{1}{200}$ - базова стойност на наклоняването;

$\alpha_h=\max\left(\frac{2}{3};\ \min\left(\sqrt{\frac{4}{L}};\ 1\right)\right)$ - коеф. за височина на колоната;

$\alpha_m=\sqrt{0.5\left(1+\frac{1}{m}\right)}$ - коефициент за брой колони в един ред m;

- локално начално огъване на елемента по ф-ла (5.6) и табл. 5.1 от [71]:

$e_0=\frac{L}{150}$ - за крива на изкълчване "C"

Въз основа на горното може лесно да съставим функция, която отчита отклонената геометрия на оста вследствие на началните несъвършенства:

$$e_0(x)=\varphi\cdot x-\frac{4\cdot x\cdot(L-x)}{L^2}\cdot e_0$$

Геометрична нелинейност

При линейно-еластичния анализ усилията в елементите се определят за недеформираното състояние на конструкцията (теория от I-ри ред). Ако преместванията са малки, а натисковите сили са далеч от критичните, това може да се приеме като достатъчно добро приближение. В противен случай усилията трябва да се определят за деформираното състояние на конструкцията.

За разглежданата колона преместването на върха ще предизвика допълнителен ексцентрицитет на вертикалната сила спрямо оста на елемента. Това ще доведе и до допълнителни огъващи моменти по височина на колоната (моменти от II-ри ред). Проблемът е, че преместванията на свой ред зависят от огъващите моменти. Зависимостта е нелинейна и за да определим стойностите на преместванията и усилията, трябва да решим съответното нелинейно уравнение. За целта отново ще използваме числен метод – този на последователните приближения.

Ще започнем с усилията от недеформирана схема, с приложени начални несъвършенства. Първоначално ще изчислим преместванията с отчитането само на физичната нелинейност. За целта ще използваме интеграла на Максуел-Мор като в знаменателя поставим дефинираната по-горе нееластична коравина. След това ще определим огъващите моменти за деформираното състояние на елемента. От тях на свой ред ще получим нови премествания и пр. Ще продължим итерациите или докато не достигнем сходимост, или докато не установим, че решението е разходящо. Това ще стане когато натисковата сила е много близо или по-голяма от критичната. Като критерий за сходимост ще използваме относителната разлика в преместването на върха на колоната между две последователни итерации.

Изчислителна процедура за анализ по деформирана схема:

Критерий за сходимост - $\Delta_{max}=0.01$ (1%)

Диаграма от един. сила, приложена на височина "x" от основата:

$$\overline{M}(\xi;\ x)=x-\xi$$

Деформация в долния ръб на сечението, за даден огъващ момент:

$$\varepsilon_{b,M}(h;\ M)=\$\mathrm{Find}\left\{M_{pl}(h;\ \varepsilon_b)-M;\ \varepsilon_b\in\left[0;\ 2\cdot\varepsilon_{ud}\right]\right\}$$

Начална итерация

Огъващи моменти от външни сили и начални несъвършенства:

$$M_{e0}(x)=H\cdot(L-x)+P\cdot(e_0(L)-e_0(x))$$

Подинтегрална функция на Максуел-Мор:

$$f_0(\xi;\ x)=\frac{M_{e0}(\xi)\cdot\overline{M}(\xi;\ x)}{K(h(\xi);\ \varepsilon_{b,M}(h(\xi);\ M_{e0}(\xi)))}$$

Начална диаграма на преместванията: $d_0(x)=\int_0^x f_0(\xi;x)\,\mathrm{d}\xi$

За проверка, на този етап може да приложим разработения модел върху примера, даден от Г. Тодоров в т. 1.2.4 от [62]. Там е разгледана конзолна греда с правоъгълно сечение с постоянна височина и без осова сила. След въвеждане на данните от примера в Calcpad и стартиране на решението получаваме следното нееластично преместване:

$$d_0(L)=d_0(5\ m)=63.35\ mm$$

Дадените в [62] стойности са както следва:

- от анализа по МКЕ - $w_{KE}(5)=0.06367\,m=63.67\,mm$ (ANSYS 11, BEAM 189)

- теоретична стойност:

$$w_p(5)=\varepsilon_Y\frac{l^2}{h}\left(\frac{M_p}{M_0}\right)^2\left[\frac{40}{27}-\frac{4}{\sqrt{27}}\left(\frac{M_0}{M_p}+2\right)\sqrt{1-\frac{M_0}{M_p}}\right]=0.06331\,m=63.31\,mm$$

Относителната грешка спрямо теоретичното решение в [62] е:

$$\delta=\frac{63.35-63.31}{63.31}\cdot 100=0.0632\,\%$$

С оглед на това, че резултатите са получени при зададена работна точност в Calcpad = 10^{-3}, съвпадението с теоретичното решение е доста добро. За сравнение, грешката от решението по МКЕ е 0.57%, при използвани 60000 крайни елемента. Разликата може да се дължи и на приноса на напречните сили, който в останалите решения не се отчита.

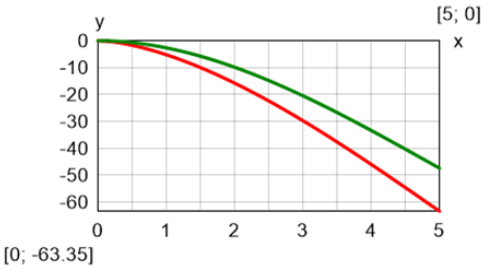
За информация, еластичното преместване е:

$$d_e(x)=\int_0^x\frac{M(\xi)\cdot\overline{M}(\xi;\,x)}{EI}d\xi,\qquad d_e(L)=d_e(5\,m)=47.37\,mm$$

Теоретичната му стойност е:

$$d_{e,L}=\frac{1}{3}\cdot\frac{H\cdot L^3}{EI}=\frac{1}{3}\cdot\frac{497.42\,kN\cdot(5\,m)^3}{437500\,kN\cdot m^2}=47.37\,mm$$

Диаграма на еластичното и нееластичното преместване:



Итерация - „*i*“

Огъващи моменти - $M_i(x)=H\cdot(L-x)+P\cdot(d_{i-1}(L)-d_{i-1}(x)+e_0(L)-e_0(x))$

От горния израз се вижда, че всяка следваща итерация използва функциите от предишната. Това ще създаде верига от рекурсивни извиквания между отделните функции с нарастваща дълбочина. Въпреки че системата кешира вътрешно резултатите, това се оказва недостатъчно когато използваме интензивно числени методи. При по-голям брой итерации има опасност от значително забавяне на алгоритъма или от препълване на кеша.

Поради това ще използваме друг по-ефективен подход. На всяка итерация ще апроксимираме моментовата диаграма с подходящ интерполационен полином. За целта ще определим стойностите на функцията в няколко

отделни точки, а за останалите ще интерполираме от вече изчислените стойности. Това ще ускори значително работата на алгоритъма.

В конкретния случай ще приложим интерполационни полиноми на Чебишов от първи тип [64]. Първите пет от тях са дефинирани чрез следните изрази:

$$T_0(x)=1;\ T_1(x)=x;\ T_2(x)=2\cdot x^2-1;\ T_3(x)=4\cdot x^3-3\cdot x;\ T_4(x)=8\cdot x^4-8\cdot x^2+1$$

С тяхна помощ ще апроксимираме диаграмата на огъващите моменти като използваме следната формула:

$$M_{i,Ch}(x)\approx\sum_{k=0}^{n-1} c(k)T_k(2\frac{x}{L}-1)-\frac{1}{2}c(0),\ \text{където} \qquad (19)$$

$T_k(x)$ е k-тия полином на Чебишов;

$c(k)=\frac{2}{n}\sum_{j=0}^{n-1} M_i(x(j))\cdot\cos\left(\frac{(j+0.5)k}{n}\pi\right)$ - интерполационен коефициент;

$x(k)=\frac{L}{2}\cdot\left(1+\cos\left(\frac{k+0.5}{n}\cdot\pi\right)\right)$ - коорд. на *k*-тия интерполационен възел.

Интерполационните възли са корени на полиномите на Чебишов. Изчислените в тях стойности ще съвпадат с изходната функция.

Директното прилагане формула (19) обаче изисква допълнителен брой аритметични операции спрямо оптималния вариант. Бихме могли да оптимизираме изчисленията, ако приложим схемата на Хорнер. За целта заместваме изразите на полиномите $T_{\mathrm{k}}(x)$ във формула (19). След известни преобразувания, получаваме следния полином:

$$P_{Ch}(x)=a_0+x\cdot\left(a_1+x\cdot\left(a_2+x\cdot\left(a_3+x\cdot a_4\right)\right)\right).$$

Коефициентите a_{k} се определят съгласно формулите:

$$a_0=\frac{c_0}{2}-c_2+c_4;\quad a_1=c_1-3\cdot c_3;\quad a_2=2\cdot c_2-8\cdot c_4;\quad a_3=4\cdot c_3;\quad a_4=8\cdot c_4.$$

Даденият по-горе запис е оптимален за компютърни изчисления защото спестява операциите по повдигане на степен. Тъй като горният израз ще се използва многократно, това ще ускори значително работата на алгоритъма. Оттук-нататък итерацията продължава в следната последователност:

Апроксимация на моментовата диаграма - $M_{i,Ch}(x)=P_{Ch}\left(\frac{2\cdot x}{L}-1\right)$

Подинтегрална функция - $f_i(\xi;\ x)=\frac{M_{i,Ch}(\xi)\cdot\overline{M}(\xi;\ x)}{K\left(h(\xi);\ \varepsilon(h(\xi);\ M_{i,Ch}(\xi))\right)}$

Премествания - $d_i(x)=\int_0^x f_i(\xi;\ x)\,\mathrm{d}\xi$

Максимално преместване - $d_{i,L}=d_i(L)$

Относителна грешка - $\Delta_i = \frac{d_{i,L} - d_{i-1,L}}{d_{i,L}}$

Ако $\Delta_i < \Delta_{max}$, необх. точност е достигната и итерациите се прекратяват.

Ако $\Delta_i > \Delta_{i-1}$, решението е разходящо и итерациите също се прекратяват.

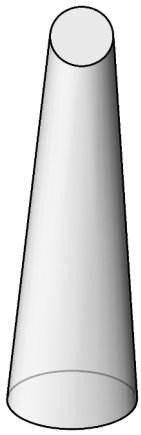
Иначе преминаваме към следващата итерация. За да избегнем „зацикляне“ на алгоритъма, ще ограничим максималния брой итерации. Когато той бъде достигнат, ще считаме че няма сходимост. Колкото по-близо е осовата сила до критичната, толкова повече се влошава сходимостта на алгоритъма. Когато е твърде близо или по-голяма от критичната, алгоритъмът става разходящ. Това свойство може да се използва за приблизително определяне на критичната сила.

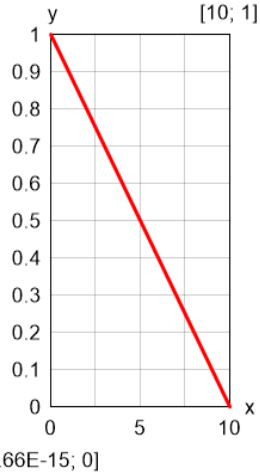
За илюстрация на предложения алгоритъм по-долу е даден пример за изчисляване и оразмеряване на стоманена колона с елипсовидно сечение.

VI.1.3 Пример – стоманена колона с елипсовидно сечение

Статическа схема

Конзола с дължина - $L = 1$ m





Натоварване

Вертикална сила - $P = 598$ kN

Хоризонтална сила - $H = 10$ kN

Огъващ момент по дължина на колоната

$M(x) = H \cdot (L - x)$

Диаграма на огъващите моменти

Материал

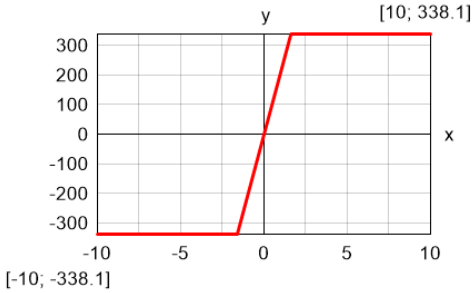
Стомана

Модул на еластичност - $E = 210$ GPa

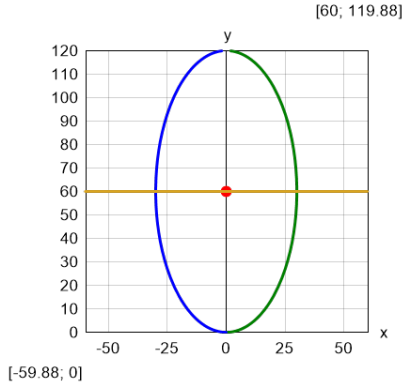
Характеристична граница на провлачване - $f_{yk} = 355$ MPa

Коефициент на сигурност на материала - $\gamma_s = 1.05$

Зависимост напрежения - деформации - $\sigma(\varepsilon) = \max\left(-f_{yd}; \min\left(\varepsilon \cdot E;\ f_{yd}\right)\right)$



Напречно сечение



Темпорална:

- в долния край - $h_b = 120$ mm
- в горния край - $h_t = 60$ mm

Ширина - $b = 60$ mm

Форма на сечението - **Елипсовидно**

Функция на контура на сечението

$$b(h;\ z) = \frac{2 \cdot b}{h} \cdot \sqrt{\left(\frac{h}{2}\right)^2 - \left(z - \frac{h}{2}\right)^2}$$

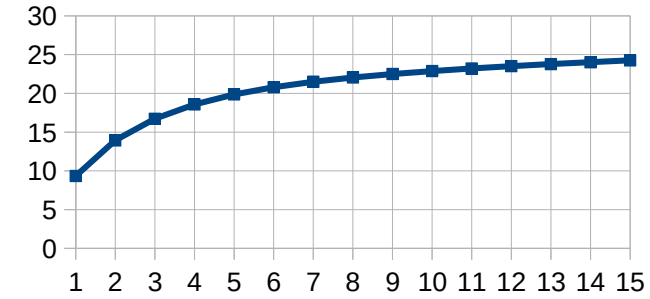
Резултати

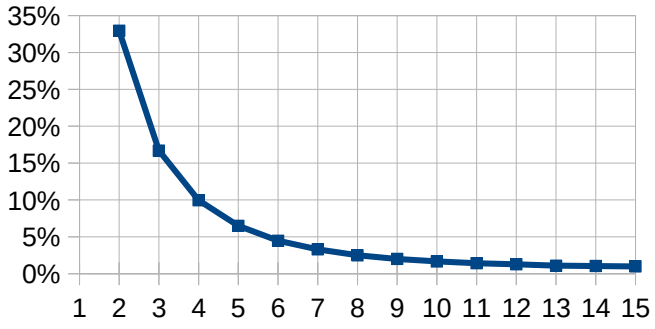
Преместване от линейно-еластичното решение:

$$d_{el}(x) = \int_{0\,m}^{x} \frac{M(\xi) \cdot \overline{M}(\xi;\ x)}{EI(h(\xi))}\, d\xi,\ d_{el,L} = d_{el}(L) = d_{el}(1\text{ m}) = 5.1\text{ mm}$$

Преместване от нелинейното решение - $d_{pl,L} = 24.26$ mm

i	d, mm	Δ, %
1	9,34	
2	13,93	32,95%
3	16,72	16,69%
4	18,57	9,96%
5	19,86	6,50%
6	20,79	4,47%
7	21,50	3,30%
8	22,05	2,49%
9	22,50	2,00%
10	22,88	1,66%
11	23,21	1,42%
12	23,51	1,28%
13	23,77	1,09%
14	24,02	1,04%
15	24,26	0,99%





Проверка на огъващия момент в основата в пластичен стадий

$$M_{Ed,II} = M_{Ed} + P \cdot (d_{pl,L} + e_{0L}) = 10\text{ kNm} + 598\text{ kN} \cdot (24.26\text{ mm} + 5\text{ mm}) = 27.5\text{ kNm}$$

$M_{Ed,II} = 27.5\ \text{kNm} \le M_{pl,N,Rd} = 43.31\ \text{kNm}$ Проверката е удовлетворена!

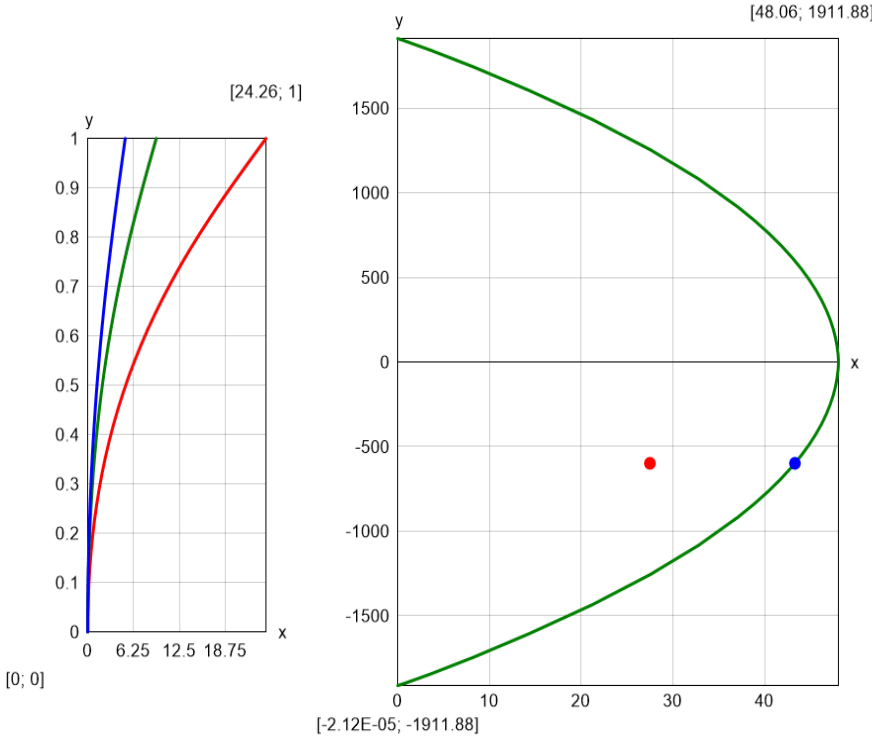
Коефициент на носимоспособност - $K = \dfrac{M_{Ed,II}}{M_{pl,N,Rd}} = \dfrac{27.5\ \text{kNm}}{43.31\ \text{kNm}} = 0.635$

Проверка в еластичен стадий - $\sigma = \left| \dfrac{N_{Ed}}{A(h_b)} + \dfrac{M_{Ed,II} \cdot (z_c(h_b) - h_b)}{I(h_b)} \right| =$

$\left| \dfrac{-598\ \text{kN}}{A(120\ \text{mm})} + \dfrac{27.5\ \text{kNm} \cdot (z_c(120\ \text{mm}) - 120\ \text{mm})}{I(120\ \text{mm})} \right| = 429.91\ \text{MPa}$

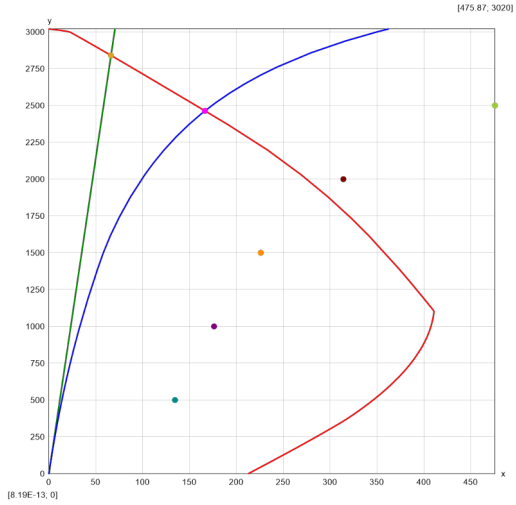
$\sigma = 429.91\ \text{MPa} > f_{yd} = 338.1\ \text{MPa}$ Проверката не е удовлетворена!

Коефициент на носимоспособност - $K = \dfrac{\sigma}{f_{yd}} = \dfrac{429.91\ \text{MPa}}{338.1\ \text{MPa}} = 1.27$



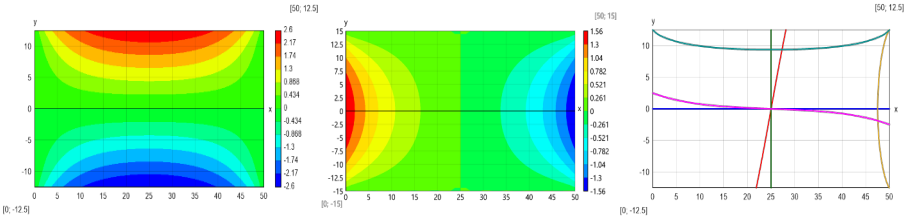
В дисертацията са разработени още 11 примера от областта на статичния и динамически линеен и нелинеен анализ на строителните конструкции с аналитични, полу-аналитични и числени методи в комбинация с предложената методика за функционално моделиране. Резултатите на проверени чрез алтернативни проверочни изчисления с програмни продукти SAP 2000, Stadyps и OpenSeesPy. Установено е добро съвпадение на резутатите в рамките на приетата точност на изчисление. Представени са решения на следните примери:

VI.2 Гранична носимоспособност на Ст.Б. колона за условен центричен натиск

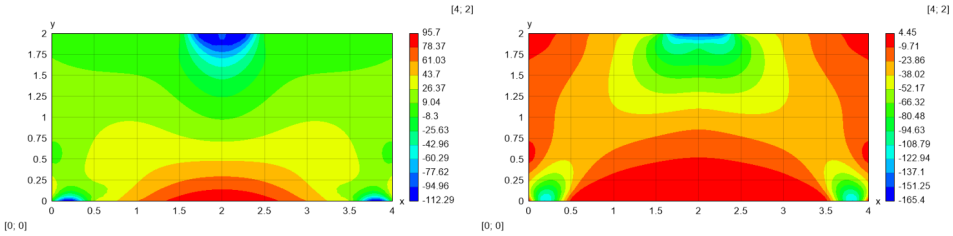


VI.3 Някои класически решения от теорията на еластичността

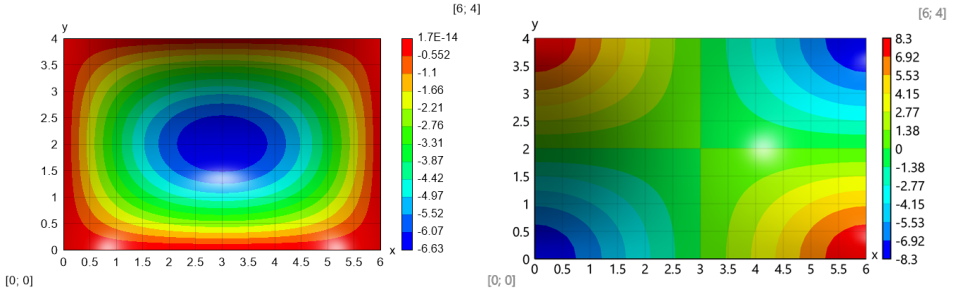
1. Усукване на правоъгълно сечение, съгласно теорията на Сен Венан



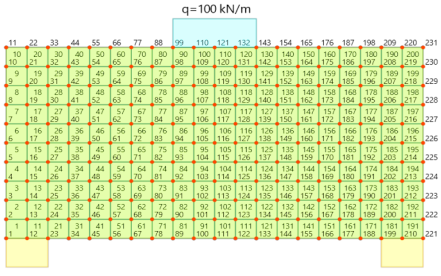
2. Решение на гредостена на две опори

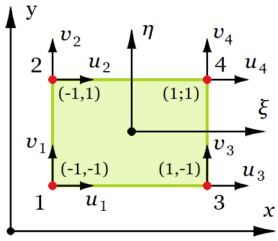


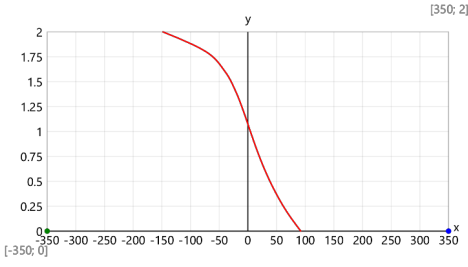
3. Изчисляване на правоъгълна плоча

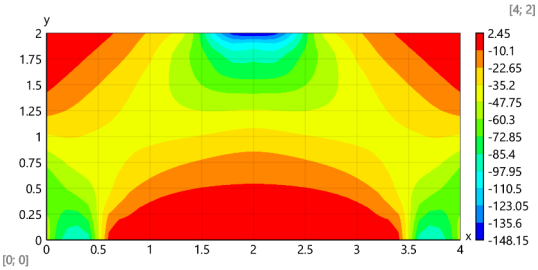


VI.4 Анализ на гредостена по МКЕ

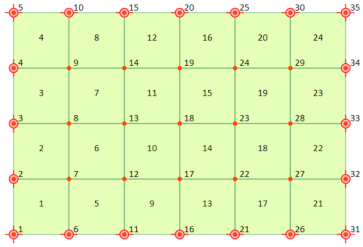


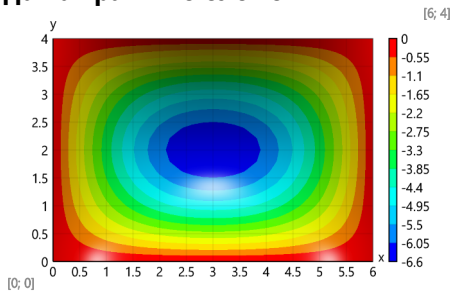




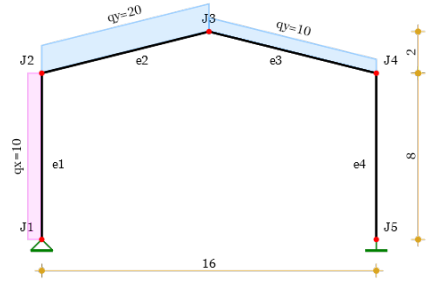


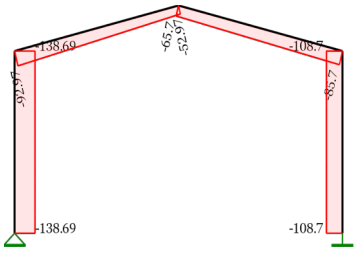
VI.5 Анализ на плоча по метода на крайните елементи

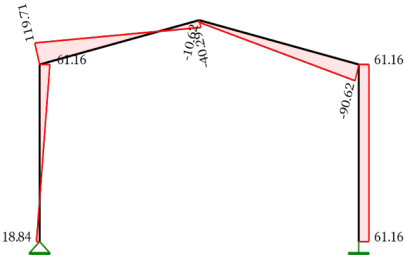


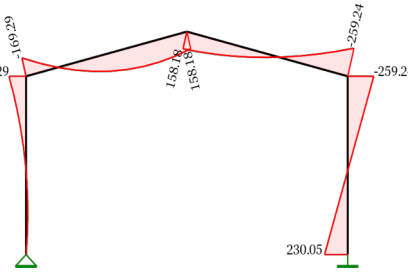


VI.6 Програма за статически анализ на равнинни рамки

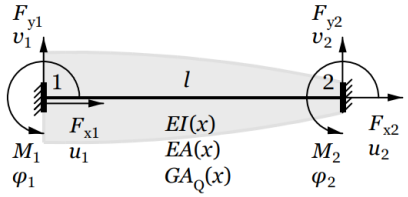








VI.7 Функционален модел на прътов краен елемент с произволно напречно сечение и променливи еластични характеристики по дължина

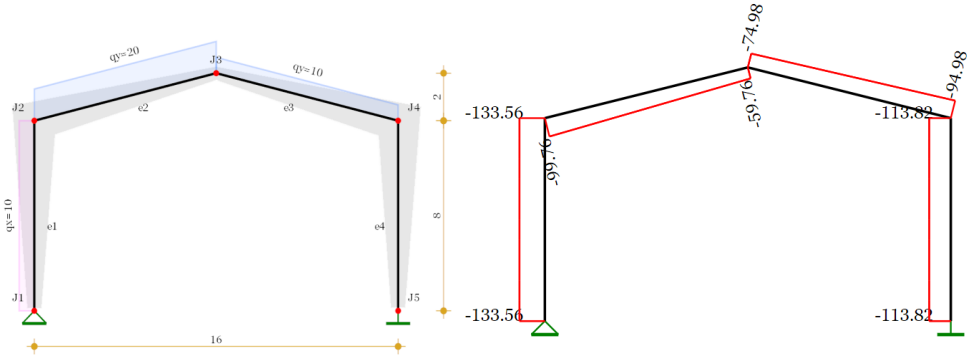


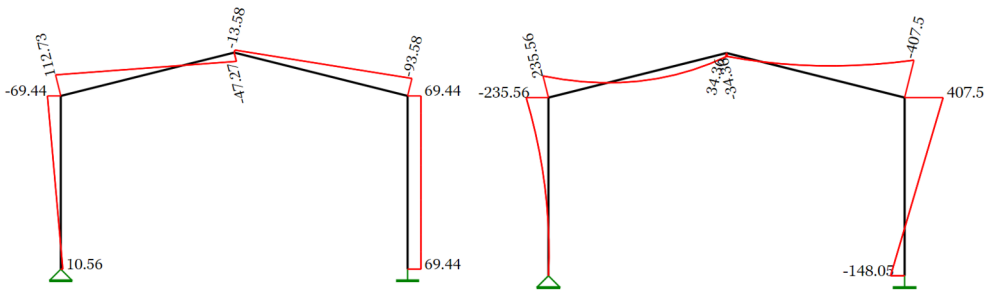
Матрица на коравина

$K_E =$

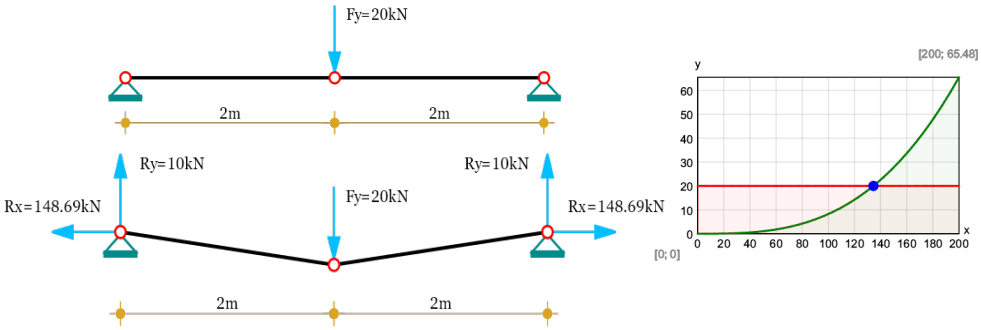
	u_1	v_1	φ_1	u_2	v_2	φ_2
F_{x1}	$\frac{1}{u_{F1}}$	0	0	$-\frac{1}{u_{F1}}$	0	0
F_{y1}	0	$\frac{\varphi_{M1}}{D_1}$	$-\frac{\varphi_{F1}}{D_1}$	0	$-\frac{\varphi_{M2}}{D_2}$	$\frac{\varphi_{F2}}{D_2}$
M_1	0	$-\frac{\varphi_{F1}}{D_1}$	$\frac{v_{F1}}{D_1}$	0	$\frac{\varphi_{F2}-\varphi_{M2}\,l}{D_2}$	$-\frac{v_{F2}-\varphi_{F2}\,l}{D_2}$
F_{x2}	$-\frac{1}{u_{F1}}$	0	0	$\frac{1}{u_{F1}}$	0	0
F_{y2}	0	$-\frac{\varphi_{M1}}{D_1}$	$\frac{\varphi_{F1}}{D_1}$	0	$\frac{\varphi_{M2}}{D_2}$	$-\frac{\varphi_{F2}}{D_2}$
M_2	0	$\frac{\varphi_{F1}+\varphi_{M1}\,l}{D_1}$	$-\frac{v_{F1}+\varphi_{F1}\,l}{D_1}$	0	$-\frac{\varphi_{F2}}{D_2}$	$\frac{v_{F2}}{D_2}$

Пример: Анализ на равнинна рамка с променливо сечение

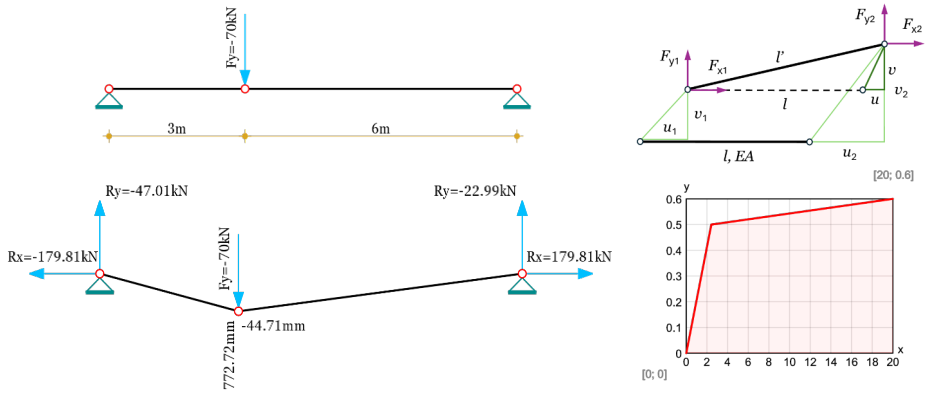




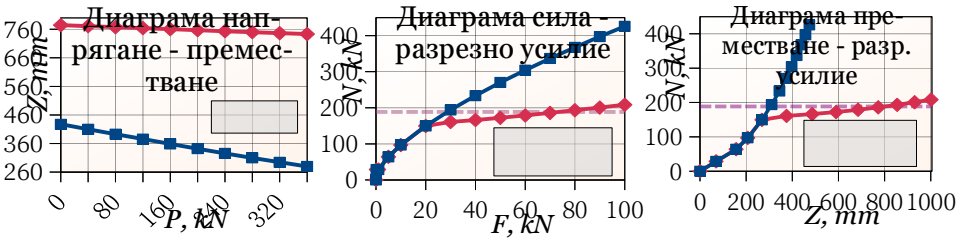
VI.8 Анализ на мигновено изменяема диада с геометрична нелинейност с големи премествания (теория от 3-ти ред)



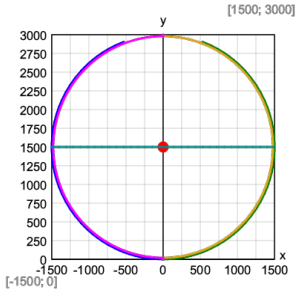
VI.9 Анализ на несиметрична мигновено изменяема диада с предварително напрягане, физична и геометрична нелинейност с големи премествания (теория от 3-ти ред)

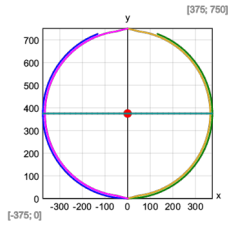


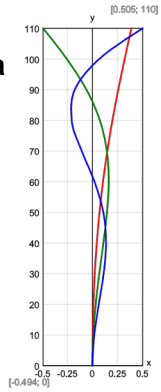
Графики на поведението на системата и сравнение с линейно-еластичен модел



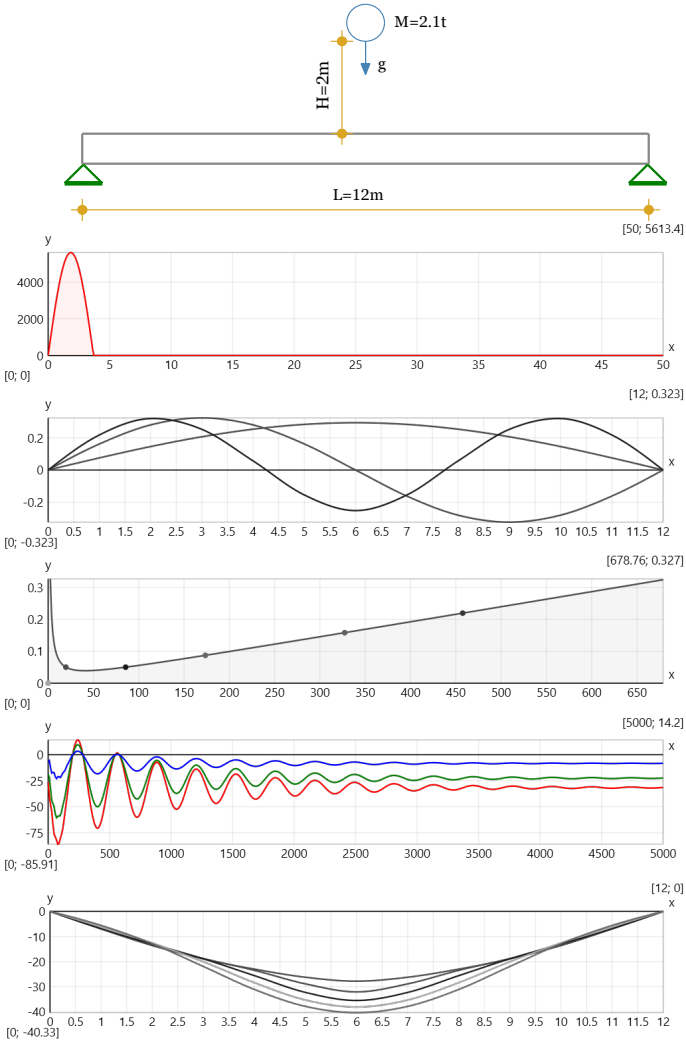
VI.10 Собствени форми на трептене на стоманен комин с променливо напречно сечение с произволна форма



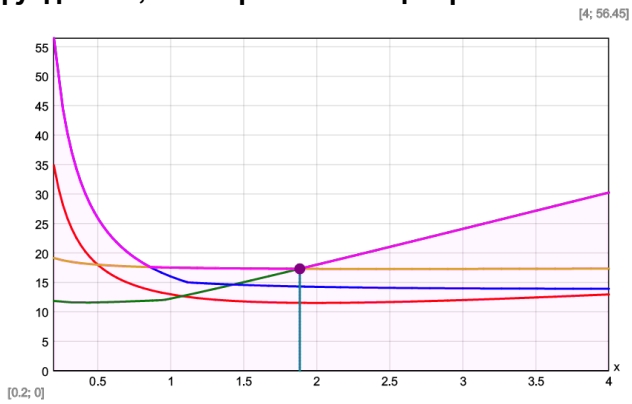




VI.11Анализ за удар от падащо тяло (метална сфера)



VI.12 Оптимизация на размерите на единичен правоъгълен фундамент, натоварен с кос нецентричен натиск



VII. Възможности за внедряване в учебния процес и проектантската практика

VII.1 Внедряване в процеса на обучение

В тази насока дисертацията може да намери приложение като:

- безплатно приложение за самостоятелно разработване на изчислителни записки, курсови и дипломни работи от студенти и докторанти;

- библиотека от интерактивни ръководства за изчисляване на строителни конструкции в Интернет. Освен че могат да видят начина за решаването на определена задача, потребителите могат да експериментират с различни входни данни и да видят как се изменят резултатите за отделните примери.

VII.2 Внедряване в проектантската практика

Към настоящия момент има разработени десетки подпрограми за решаването на различни видове задачи от строителното проектиране, които се прилагат успешно в практиката. По-долу е даден кратък списък на по-важните от тях:

Натоварвания

Земен натиск

Стоманобетонна сграда (вертикален товар)

Изчислителен спектър на реагиране

Натоварване от вятър за различни видове сгради и съоръжения

Натоварване от сняг

Снежна торба

Стоманобетонни конструкции

Греди

Правоъгълно сечение за огъване

Плочогредово сечение за огъване

Проверка на плочогредово сечение за огъване
Правоъгълно сечение за напречна сила
Греда за усукване, срязване и огъване

Колони

Интеракционна диаграма
Колона за нецентричен натиск
Колона за двоен нецентричен натиск

Пукнатини и провисвания

Пукнатини и провисване на греда
Пукнатини и провисване на греда с Т сечение

Полета

Свободно подпряно поле
Поле със запъната дълга страна
Поле със запъната къса страна
Поле със запънати дълги страни
Поле със запънати къси страни
Поле със запънати съседни страни
Запънато поле със свободно подпряна дълга страна
Запънато поле със свободно подпряна къса страна
Четиристранно запънато поле

Продънване

Продънване на правоъгълна вътрешна колона
Продънване на правоъгълна ръбова колона
Продънване на правоъгълна ъглова колона

Конструиране на армировката

Конструиране на греда за DCM
Конструиране на колона за DCM
Конструиране на шайба за DCM

Стоманени конструкции

Инерционни характеристики

C сечение
L сечение
T сечение
Z сечение
Двойно T сечение
Кръг
Кръгла тръба
Правоъгълна кутия
Правоъгълник

Напречни сечения

Двойно-Т сечение

Съединения и възли

Единично болтово съединение

Напрегнато болтово съединение

Болтова става

База на колона

Възел греда-колона

Възел греда-греда

Ъглов заваръчен шев

Сглобяеми и предварително напрегнати конструкции

Напрегната трапецовидна столица с изолиране

Напрегната Т столица с изолиране

Напрегната главна греда с изолиране

Сглобяема чашка

Къса конзола

Закладна част

Геотехника

Изчисляване и оразмеряване на ъглова подпорна стена

Математика

Площи на фигури

Обеми на тела

Решаване на квадратно уравнение

Решаване на кубично уравнение

Механика

Прости греди

Едностранно запънати греди

Двустранно запънати греди

Конзоли

Усукване на правоъгълно сечение по ТЕ

Анализ на гредостена по ТЕ

Анализ на правоъгълна плоча по ТЕ

Описаните по-горе подпрограми са внедрени в десетки проектантски бюра и фирми и се прилагат успешно в проектирането на различни видове строителни конструкции.

VIII. Приноси

1. Разработена е обща методика за функционално моделиране на строителни конструкции.
2. Разработена е универсална изчислителна платформа за създаване и използване на приложни програми за строително проектиране в облачна среда.
3. Направен е сравнителен анализ на различни числени методи по отношение на тяхната универсалност, устойчивост и сходимост като са избрани най-подходящите за внедряване в строителното проектиране. В допълнение на горното:
    3.1. Предложен е нов метод за решаване на нелинейни уравнения, който осигурява едновременно висока устойчивост и сходимост на решението, като комбинира предимствата метода на разполовяването и на Андерсън-Бьорк.
    3.2. Направено е подобрение на интегрирането с Tanh-Sinh квадратурна формула. Чрез предварителното кеширане на абсцисите и теглата е повишена скоростта на работа на алгоритъма.
4. Разработен е нов специализиран програмен език, предназначен за строителни инженери без специални умения по програмиране и съответен интерпретатор/компилатор.
5. Разработена е универсална процедура за изчисляване и оразмеряване на конструктивни елементи с променливи размери и произволна форма на напречното сечение с отчитане на начални несъвършенства, геометрична и физична нелинейност.

IX. Насоки за бъдеща работа

Като изчислителни възможности на платформата, с приложение в строителната механика, с времето ще бъдат добавени следните възможности:

- високопроизводителни изчисления с вектори и матрици с използването на ядрата на графичните ускорители;

- оптимизация в повече измерения – линейно и нелинейно програмиране, downhill simplex метод и др.

- някои специални функции: гама-функции, бета-функции, елиптични интеграли, интеграли на Фреснел, функции на Бесел и др.

- числени методи за решаване на диференциални уравнения – Рунге-Кута и др.

Като приложение в областта на строителната механика, могат да се разработят решения на различни задачи по метода на крайните елементи, отново с участието на начални несъвършенства, геометрична и физична нелинейност, но за по-сложни статически схеми.

X. Публикации по темата на дисертационния труд

1. Ганчовски Н. Софтуерна платформа за инженерни изчисления в облачна среда. Механика, Транспорт, Комуникации ISSN 1312-3823, том 17, брой 3, 2019 г. статия № 1482

https://mtc-aj.com/library/1842.pdf

2. Ganchovski N. Traykov A. Modified Anderson-Björck's method for solving nonlinear equations in structural mechanics 2023 IOP Conf. Ser.: Mater. Sci. Eng. 1276 012010. DOI 10.1088/1757-899X/1276/1/012010

https://iopscience.iop.org/article/10.1088/1757-899X/1276/1/012010

Известни цитирания:

Galvão H., Silva V. Classes de Métodos Numéricos não Convencionais para Determinação de Raízes de Funções. 02.2024. Thesis. Universidade Federal do Pará. Faculdade de Matemática. DOI: 10.13140/RG.2.2.24088.57606

Roots Fortran. Numerical Library for Modern Fortran. NASA Johnson Space Center, Houston, Texas, https://jacobwilliams.github.io/roots-fortran/

3. Traykov A., Ganchovski N. Analysis of a column with variable, arbitrary shaped cross section, including initial imperfections, geometric and material nonlinearity. 2023 IOP Conf. Ser.: Mater. Sci. Eng. 1276 012012. DOI 10.1088/1757-899X/1276/1/012012

https://iopscience.iop.org/article/10.1088/1757-899X/1276/1/012012

4. Traykova M. et al. Capacity of axially loaded RC columns with arbitrary cross sections to Eurocode 2. 2023 IOP Conf. Ser.: Mater. Sci. Eng. 1276 012011. DOI 10.1088/1757-899X/1276/1/012011

https://iopscience.iop.org/article/10.1088/1757-899X/1276/1/012011

5. Ганчовски Н. Адаптивни методи за числено интегриране в строителната механика. Сравнителен анализ. XXIII Международна научна конференция ВСУ`2023. 3-21, ISSN: 1314-071X.

https://conference.vsu.bg/_vsu2023/article1921.pdf

6. Ганчовски Н. Анализ на равнинни рамки по МКЕ с Calcpad. XXIV Международна научна конференция ВСУ`2024. 2-17, ISSN: 1314-071X.

https://www.researchgate.net/publication/384883768

7. Ganchovski, N., Traykov, A. (2025). Functional FE Formulation for Analysis of Plane Frames with Arbitrary Variable Sections and Material Properties. In: Nia, E.M., Awang, M., Aulady, M.F.N., Traykova, M., Yola, L. (eds) Selected Articles from the 8th International Conference on Architecture and Civil Engineering. ICACE 2024 2024. Lecture Notes in Civil Engineering, vol 635. Springer, Singapore.

https://doi.org/10.1007/978-981-96-5654-7_42

XI. Приложения

Приложение 1. Програмна реализация на разработените и използвани в дисертацията алгоритми.

Приложение 2. Формули и графики на тестови функции за решаване на нелинейни уравнения

Приложение 3. Графики на сходимостта на методи за решаване на нелинейни уравнения

Приложение 4. Описание на програмния език Calyx

Приложение 5. Входни данни и резултати от проведените анализи с програмни продукти SAP 2000 и OpenSeesPy

XII. Литература

[1] Goldberg, D. „What Every Computer Scientist Should Know About Floating Point Arithmetic“, ACM Computing Surveys. 23 (1), 1991 г. DOI: 10.1145/103162.103163 https://www.itu.dk/%7Esestoft/bachelor/IEEE754_article.pdf

[2] William H. P., Saul A. T., William T. V., and Brian P. F. Numerical Recipes 3rd Edition: The Art of Scientific Computing (3rd ed). Cambridge University Press, USA, 2007. http://numerical.recipes

[3] Константинов, М. Числени основи на мат. методи. Изд. УАСГ, София, 1997. https://uacg.bg/filebank/att_1958.pdf

[4] Schonfelder, J. L. “Very high accuracy Chebyshev expansions for the basic trigonometric functions”, MATHEMATICS OF COMPUTATION, VOLUME 34, NUMBER 149, JANUARY 1980, PAGES 237-244 https://doi.org/10.1090/S0025-5718-1980-0551302-5

[5] Volder, Jack E. (1959-03-03). "The CORDIC Computing Technique" (PDF). Proceedings of the Western Joint Computer Conference (WJCC) (presentation). San Francisco, California, USA: National Joint Computer Committee (NJCC): 257–261. Retrieved 2016-01-02.
https://www.jacques-laporte.org/Volder_CORDIC.pdf

[6] C. Ridders, "A new algorithm for computing a single root of a real continuous function", IEEE Transactions on Circuits and Systems, vol. 26, no. 11, pp. 979-980, November 1979 https://doi.org/10.1109/TCS.1979.1084580

[7] Alicia Cordero, José L. Hueso, Eulalia Martínez, Juan R. Torregrosa, Steffensen type methods for solving nonlinear equations, Journal of Computational and Applied Mathematics, Volume 236, Issue 12, 2012, Pages 3058-3064, ISSN 0377-0427 https://doi.org/10.1016/j.cam.2010.08.043

[8] Muller, D. E. A Method for Solving Algebraic Equations Using an Automatic Computer. Mathematical Tables and Other Aids to Computation, vol. 10, no. 56, 1956, pp. 208–215. www.jstor.org/stable/2001916

[9] Ostrowski, A. M. “Solution of Equations and Systems of Equations”, Acad. Press, II ed. (1966).

[10] Xia Wang and Liping Liu, "New eighth-order iterative methods for solving nonlinear equations", Journal of Computational and Applied Mathematics 234 (2010) 1611-1620. https://www.sciencedirect.com/science/article/pii/S0377042710001433

[11] Kung, H. T. and Traub, J. F. “Optimal order of one-point and multipoint iteration” Journal of the ACM, vol. 21, no. 4, pp. 643–651, 1974.
https://doi.org/10.1145/321850.321860

[12] Chein-Shan Liu, Tsung-Lin Lee, "A New Family of Fourth-Order Optimal Iterative Schemes and Remark on Kung and Traub’s Conjecture", Journal of Mathematics, vol. 2021, Article ID 5516694, 9 pages, 2021.
https://doi.org/10.1155/2021/5516694

[13] Tao Y, Madhu K. Optimal Fourth, Eighth and Sixteenth Order Methods by Using Divided Difference Techniques and Their Basins of Attraction and Its Application. Mathematics. 2019; 7(4): 322.
https://doi.org/10.3390/math7040322

[14] Brent, R. P. An algorithm with guaranteed convergence for finding a zero of a function, *The Computer Journal*, Volume 14, Issue 4, 1971, Pages 422–425
https://doi.org/10.1093/comjnl/14.4.422

[15] Galdino, S. A family of regula falsi root-finding methods. Proceedings of 2011 World Congress on Engineering and Technology. Shanghai, China. IEEE Press. ISBN 978-1-61284-365-0. p. 514-517
http://sergiogaldino.pbworks.com/w/file/fetch/66011429/0130-1943543

[16] Stage, S. A. Comments on An Improvement to the Brent’s Method. International Journal of Experimental Algorithms (IJEA), Volume (4) : Issue (1) : 2013

[17] Swift, A. and Lindfield, G. R. Comparison of a continuation method with Brent's method for the numerical solution of a single nonlinear equation, *The Computer Journal*, Volume 21, Issue 4, 1978, Pages 359–362
https://doi.org/10.1093/comjnl/21.4.359

[18] Sikorski, K. Bisection is optimal. Numer. Math. 40, 111–117 (1982).
https://doi.org/10.1007/BF01459080

[19] Argyros, I. K. Hernández-Verón M.A., Rubio M.J. (2019) On the Convergence of Secant-Like Methods. In: Dutta H., Kočinac L.D.R., Srivastava H.M. (eds) Current Trends in Mathematical Analysis and Its Interdisciplinary Applications. Birkhäuser, Cham.
https://doi.org/10.1007/978-3-030-15242-0_5

[20] Oliveira, I. F. D. and Takahashi, R. H. C. “An Enhancement of the Bisection Method Average Performance Preserving Minmax Optimality.” ACM Transactions on Mathematical Software (TOMS) 47 (2021): 1 – 24.
https://doi.org/10.1145/3423597

[21] Ostrowski, A. M. Solution of Equations and Systems of Equations: Pure and Applied Mathematics: A Series of Monographs and Textbooks, Vol. 9. Vol. 9. Elsevier, 2016.

[22] Suhadolnik, A. Combined bracketing methods for solving nonlinear equations, Applied Mathematics Letters, Volume 25, Issue 11, 2012, Pages 1755-1760, ISSN 0893-9659
https://doi.org/10.1016/j.aml.2012.02.006

[23] Kumar, R. and Vipan, “Comparative Analysis of Convergence of Various Numerical Methods”, Journal of Computer and Mathematical Sciences, Vol.6(6), 290-297, June 2015
http://compmath-journal.org/dnload/Robin-Kumar-and-Vipan-/CMJV06I06P0290.pdf

[24] Anderson, N. and Björck, Å. A new high order method of regula falsi type for computing a root of an equation. BIT 13, 253–264 (1973).
https://doi.org/10.1007/BF01951936

[25] Вълчанов Н., Константинов М. Съвременни математически методи за компютърни пресмятания. Част 1 - Основи на компютърните пресмятания. Числено диференциране и интегриране. Студии на БИАП по математически науки, т. 1, София, 1996. 231 с.

[26] Ganchovski N. Traykov A. Modified Anderson-Björck's method for solving non-linear equations in structural mechanics 2023 IOP Conf. Ser.: Mater. Sci. Eng. 1276 012010. DOI 10.1088/1757-899X/1276/1/012010
https://iopscience.iop.org/article/10.1088/1757-899X/1276/1/012010

[27] Kiefer, J. (1953) 'Sequential minimax search for a maximum', Proceedings of the American Mathematical Society, Vol. 4, No. 3, pp.502–506
https://www.ams.org/journals/proc/1953-004-03/S0002-9939-1953-0055639-3/S0002-9939-1953-0055639-3.pdf

[28] Knut Mørken. Numerical Algorithms and Digital Representation, Univ. I Oslo, 12 Nov 2010
https://www.uio.no/studier/emner/matnat/math/MAT-INF1100/h10/kompendiet/komp.pdf

[29] Reiterer, S. (2021). Numerical Differentiation using local Chebyshev Approximation.
https://www.researchgate.net/publication/349045916_Numerical_Differentiation_using_local_Chebyshev-Approximation

[30] Richardson, L. F. (1911). "The approximate arithmetical solution by finite differences of physical problems including differential equations, with an application to the stresses in a masonry dam". Philosophical Transactions of the Royal Society A. 210 (459–470): 307–357. doi:10.1098/rsta.1911.0009
https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.1911.0009

[31] Ridders, C. J. F. Accurate computation of F'(x) and F'(x) F"(x), Advances in Engineering Software 4(2), 75-76, 1978
https://doi.org/10.1016/S0141-1195(82)80057-0

[32] George B. Arfken, Hans J. Weber, Frank E. Harris. Mathematical Methods for Physicists (VII Ed.), Chapter 11 – Complex Variable Theory, Academic Press, 2013, ISBN 9780123846549
https://doi.org/10.1016/B978-0-12-384654-9.00011-6

[33] Runge, C. Übera empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten. Z. Math. Phys.,46 (1901), 224–243
https://archive.org/details/zeitschriftfrma12runggoog

[34] Марков, И. Метод на крайните елементи. УАСГ 2014 г. ISBN 978-954-724-072-8
https://uacg.bg/filebank/att_7612.pdf

[35] Romberg, W. (1955), "Vereinfachte numerische Integration", Det Kongelige Norske Videnskabers Selskab Forhandlinger, Trondheim, 28 (7): 30–36
https://doi.org/10.1007/BF02559689

[36] Takahasi, Hidetosi; Mori, Masatake (1974), Double Exponential Formulas for Numerical Integration, Publications of the Research Institute for Mathematical Sciences, 9(3): 721–741
https://www.ems-ph.org/journals/show_pdf.php?issn=0034-5318&vol=9&iss=3&rank=12

[37] David H. Bailey, Karthik Jeyabalan, and Xiaoye S. Li, A Comparison of Three High-Precision Quadrature Schemes, Experimental Mathematics, Vol. 14 (2005), No. 3, p. 317 – 329
https://www.davidhbailey.com/dhbpapers/quadrature-em.pdf

[38] Michalski, K. and Mosig, J. "Efficient computation of Sommerfeld integral tails – methods and algorithms" Journal of Electromagnetic Waves and Applications 2016
https://doi.org/10.1080/09205071.2015.1129915

[39] Robert van Engelen, Improving the Double Exponential Quadrature Tanh-Sinh, Sinh-Sinh and Exp-Sinh Formulas Genivia Labs, June 27, 2021
https://www.genivia.com/files/qthsh.pdf

[40] Clenshaw, C. W. and Curtis, A. R. "A method for numerical integration on an automatic computer, Numerische Mathematik 2, 197 (1960)
https://doi.org/10.1007/BF01386223

[41] Guy F. Kuncir. Algorithm 103: Simpson's rule integrator. Comm. ACM 5,6 (June 1962), 347.
https://doi.org/10.1145/367766.368179

[42] Lyness. J. N. Notes on the Adaptive Simpson Quadrature Routine. J. ACM 16, 3 (July 1969), 483–495.
https://doi.org/10.1145/321526.321537

[43] Laurie, Dirk (1997) "Calculation of Gauss-Kronrod quadrature rules.", Mathematics of Computation of the American Mathematical Society, 66 (219): 1133–1145
https://doi.org/10.1090%2Fs0025-5718-97-00861-2

[44] Gander, W. and Gautschi, W. Adaptive Quadrature—Revisited. BIT Numerical Mathematics 40, 84–101 (2000).
https://doi.org/10.1023/A:1022318402393

[45] Gonnet, P. Adaptive quadrature re-revisited, Doctoral Thesis, SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH, 2009, ISBN 978-0-557-08761-7
https://doi.org/10.3929/ethz-a-005861903

[46] Dijkstra, E. W. "Algol 60 translation: An Algol 60 translator for the x1", Research Report 35, Mathematisch Centrum, Amsterdam, 1961
http://www.cs.utexas.edu/users/EWD/MCReps/MR35.PDF

[47] НАРЕДБА ЗА ЕДИНИЦИТЕ ЗА ИЗМЕРВАНЕ, РАЗРЕШЕНИ ЗА ИЗПОЛЗВАНЕ В РЕПУБЛИКА БЪЛГАРИЯ. Приета с ПМС No 275 от 29.11.2002 г. Обн. ДВ. бр.115 от 10 Декември 2002г., изм. ДВ. бр.40 от 16 Май 2006г., изм. ДВ. бр.8 от 29 Януари 2010г., изм. и доп. ДВ. бр.46 от 19 Май 2020г.
https://www.damtn.government.bg/wp-content/uploads/naredbi/naredba_za_edinicite_za_izmervane-1.pdf

[48] Volker, V. (1969). "Gaussian Elimination is not Optimal". Numerische Mathematik 13 (4): 354–356. doi:10.1007/BF02165411.
https://link.springer.com/article/10.1007/BF02165411

[49] Williams, Virginia Vassilevska, Yinzhan Xu, Zixuan Xu and Renfei Zhou. (2023) “New Bounds for Matrix Multiplication: from Alpha to Omega.” ACM-SIAM Symposium on Discrete Algorithms.
https://arxiv.org/pdf/2307.07970

[50] Cuthill, E. and McKee, J. Reducing the bandwidth of sparse symmetric matrices, Proc. of ACM, (1969), 157–172.
https://dl.acm.org/doi/pdf/10.1145/800195.805928

[51] George, J. A. Computer implementation of the finite element method, Standford Computer Science Dept., Tech. Report STAn-CS-71-208, Standford, California, (1971)
https://apps.dtic.mil/sti/pdfs/AD0726171.pdf

[52] Gibbs N.E., Poole W.G. and Stockmeyer P.K., An algorithm for reducing the bandwidth and profle of a sparse matrix, SIAM Journal on Numerical Analysis, 13, (1976), 236–250
https://www.jstor.org/stable/2156090

[53] Broyden, C. G. “A Class of Methods for Solving Nonlinear Simultaneous Equations.” *Mathematics of Computation*, vol. 19, no. 92, 1965, pp. 577–93. *JSTOR*, https://doi.org/10.2307/2003941

[54] Vasios N., Nonlinear Analysis of Structures. The Arc Length Method: Formulation, Implementation and Applications, Harvart University, 2015 https://scholar.harvard.edu/sites/scholar.harvard.edu/files/vasios/files/ArcLength.pdf

[55] Riks, E. An incremental approach to the solution of snapping and buckling problems, International Journal of Solids and Structures, Vol. 15, Issue 7, 1979, Pages 529-551, ISSN 0020-7683 https://doi.org/10.1016/0020-7683(79)90081-7

[56] Crisfield, M. A. A fast incremental/iterative solution procedure that handles “snap-through”, Computers & Structures, Volume 13, Issues 1–3, 1981, Pages 55-62, ISSN 0045-7949 https://doi.org/10.1016/0045-7949(81)90108-5

[57] Jacobi, C.G.J. (1846). "Über ein leichtes Verfahren, die in der Theorie der Säkularstörungen vorkommenden Gleichungen numerisch aufzulösen". Crelle's Journal (in German). 1846 (30): 51–94. doi:10.1515/crll.1846.30.51. S2CID 199546177. http://gdz.sub.uni-goettingen.de/dms/load/img/?PID=GDZPPN002144522

[58] Francis, J. G. F. The QR Transformation A Unitary Analogue to the LR Transformation—Part 1, The Computer Journal, Volume 4, Issue 3, 1961, Pages 265–271 https://doi.org/10.1093/comjnl/4.3.265

[59] Francis, J. G. F. The QR Transformation—Part 2, The Computer Journal, Volume 4, Issue 4, 1962, Pages 332–345 https://doi.org/10.1093/comjnl/4.4.332

[60] Householder, A. S. (1958). Unitary Triangularization of a Nonsymmetric Matrix. J. ACM, 5, 339-342. https://hal.science/hal-01316095/file/p339householderb.pdf

[61] James F. Blinn. 1977. Models of light reflection for computer synthesized pictures. In Proceedings of the 4th annual conference on Computer graphics and interactive techniques (SIGGRAPH '77). Association for Computing Machinery, New York, NY, USA, 192–198. https://doi.org/10.1145/563858.563893

[62] Тодоров, Г. Изследване на прътови конструкции при пластични деформации. Първо издание. УАСГ София, 2013, ISBN 978-954-724-056-8. https://uacg.bg/filebank/att_5271.pdf

[63] Светлана Лилкова-Маркова. Устойчивост на конструктивни елементи с променливо напречно сечение. Първо издание. “АБС техника”, 2014. https://www.researchgate.net/profile/Svetlana-Lilkova-Markova/publication/318672176_USTOJCIVOST_NA_KONSTRUKTIVNI_ELEMENTI_S_PROMENLIVO_NAPRECNO_SECENIE/links/5976ec56458515e26d25c65d/USTOJCIVOST-NA-KONSTRUKTIVNI-ELEMENTI-S-PROMENLIVO-NAPRECNO-SECENIE.pdf

[64] Mason, J. C., Handscomb, D. C. (2002). Chebyshev Polynomials (1st ed.). Chapman and Hall/CRC. http://inis.jinr.ru/sl/M_Mathematics/MRef_References/Mason,%20Hanscomb.%20Chebyshev%20polynomials%20(2003)/C0355-FM.pdf

[65] Traykov A., Ganchovski N. Analysis of a column with variable, arbitrary shaped cross section, including initial imperfections, geometric and material nonlinearity. 2023 IOP Conf. Ser.: Mater. Sci. Eng. 1276 012012. DOI 10.1088/1757-899X/1276/1/012012 https://iopscience.iop.org/article/10.1088/1757-899X/1276/1/012012

[66] Банков Б., Павлова Ю. Метод на крайните елементи в строителната механика. Изд. от УАСГ, София 1999 г.

[67] Levy, R., Spillers, W.R. Analysis of Geometrically nonlinear structures: Second Edition (2003) Analysis of Geometrically nonlinear structures: Second Edition, p. 272. DOI: 10.1007/978-94-017-0243-0 https://link.springer.com/book/10.1007/978-94-017-0243-0

[68] McKenna, F., Fenves, G. L, and Scott, M. H. (2000) Open System for Earthquake Engineering Simulation. University of California, Berkeley http://opensees.berkeley.edu

[69] Mazzoni, S., McKenna, F., Scott, M. H., and Fenves, G. L. (2006) OpenSees Command Language Manual. University of California, Berkeley http://opensees.berkeley.edu/manuals/usermanual

[70] Minjie Zhu, Frank McKenna, Michael H. Scott, OpenSeesPy: Python library for the OpenSees finite element framework, SoftwareX, Vol. 7, 2018, Pages 6-11, ISSN 2352-7110 https://doi.org/10.1016/j.softx.2017.10.009.

[71] БДС EN 1993-1-1:2005/A1:2014. Еврокод 3: Проектиране на стоманени конструкции. Част 1-1: Общи правила и правила за сгради https://bds-bg.org/bg/project/show/bds:proj:88306

[72] Норми за проектиране на бетонни и стоманобетонни конструкции, изм. и доп. 2008 г., http://kab-sofia.bg/images/stories/normi/3-b-4_stbeton.pdf

[73] БДС EN 1992-1-1:2005. Еврокод 2: Проектиране на бетонни и стоманобетонни конструкции. Част 1-1: Общи правила и правила за сгради. https://bds-bg.org/bg/project/show/bds:proj:49824

[74] Traykova M. et al. Capacity of axially loaded RC columns with arbitrary cross sections to Eurocode 2. 2023 IOP Conf. Ser.: Mater. Sci. Eng. 1276 012011. DOI 10.1088/1757-899X/1276/1/012011 https://iopscience.iop.org/article/10.1088/1757-899X/1276/1/012011

[75] Христо Върбанов, Ангел Тепавичаров, Танко Ганев. Приложна теория на еластичността и пластичността. София, 1992, ISBN 954-03-0043-6

[76] Minimum Design Loads and Associated Criteria for Buildings and Other Structures (ASCE/SEI 7-22). American Society of Civil Engineers. 2022 https://doi.org/10.1061/9780784415788

[77] Harris C. M., Piersol A.G., HARRIS' SHOCK AND VIBRATION HANDBOOK, Fifth Edition, McGraw-Hill 2002, ISBN 0-07-137081-1

[78] Qing Peng, Xiaoming Liu, Yueguang Wei, Elastic impact of sphere on large plate, Journal of the Mechanics and Physics of Solids, Volume 156, 2021, 104604, ISSN 0022 - 5096, https://doi.org/10.1016/j.jmps.2021.104604

[79] Hong Hao and Thong M. Pham, Performance of RC Beams with or without FRP Strengthening Subjected to Impact Loading, Proceedings of the 2nd World Congress on Civil, Structural, and Environmental Engineering (CSEE'17) Barcelona, Spain – April 3–4, 2017 ISSN:2371-5294

https://doi.org/10.11159/icsenm17.1

[80] Gugan, D. “Inelastic collision and the Hertz theory of impact.” American Journal of Physics 68 (2000): 920-924
http://www.oxfordcroquet.com/tech/gugan/index.asp

[81] Върбанов, Х. Устойчивост и динамика на еластичните системи. Изд. Техника, София, 1989

[82] Бобев Т., Тодоров Т., Рангелов, Р., Павлова, Ю., Байчев. И. Устойчивост и динамика на строителните конструкции. Ръководство за решаване на задачи. изд Техника, София 1994

[83] SAP 2000. Reference manual. Computers and Structures, Inc. Berkeley, California, USA. 1998