

České vysoké učení technické v Praze
Fakulta stavební
Katedra betonových a zděných konstrukcí



Optimalizace betonových prvků pomocí genetických algoritmů

Bakalářská práce

Václav Wudi

Praha, Květen 2020

Vedoucí bakalářské práce:

Ing. Martin Petřík, Ph.D.
Katedra betonových a zděných konstrukcí
Fakulta Stavební
České vysoké učení technické v Praze
Technická 2
160 00 Praha 6

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta stavební

Thákurova 7, 166 29 Praha 6



ZADÁNÍ BAKALÁŘSKÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: <u>Wudi</u>	Jméno: <u>Václav</u>	Osobní číslo: <u>468654</u>
Zadávací katedra: <u>Katedra betonových a zděných konstrukcí</u>		
Studijní program: <u>Stavební inženýrství</u>		
Studijní obor: <u>Konstrukce pozemních staveb</u>		

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce: <u>Optimalizace betonových prvků prostřednictvím genetického algoritmu</u>	
Název bakalářské práce anglicky: <u>Optimization of Concrete Elements Using Genetic Algorithm</u>	
Pokyny pro vypracování: - Vysvětlení genetického algoritmu (GA) a principu použité metody konečných prvků (MKP) - Definice optimalizovaného prvku + rozbor jeho působení a využití - Vytvoření programu v jazyce Python využívající GA a MKP za účelem optimalizace betonového prvku vzhledem k zadaným kritériím	
Seznam doporučené literatury: * ZELINKA, Ivan, et al. Evoluční výpočetní techniky: principy a aplikace. BEN, 2008. * HYNEK, Josef. Genetické algoritmy a genetické programování. Grada Publishing as, 2008. * BEGHINI, Lauren L., et al. Connecting architecture and engineering through structural topology optimization. Engineering Structures, 2014, 59: 716-726. * KATTAN, Peter I. MATLAB guide to finite elements: an interactive approach. Springer Science & Business Media, 2010. * BUCHANAN, George R. Schaum's outline of theory and problems of finite element analysis. McGraw-Hill, 1995. * LOGAN, Daryl L. A first course in the finite element method. Cengage Learning, 2011.	
Jméno vedoucího bakalářské práce: <u>Ing. Martin Petřík, Ph.D.</u>	
Datum zadání bakalářské práce: <u>27.2.2020</u>	Termín odevzdání bakalářské práce: <u>18.5.2020</u> <small>Údaj uveďte v souladu s datem v časovém plánu příslušného ak. roku</small>
<u>[Signature]</u> Podpis vedoucího práce	<u>[Signature]</u> Podpis vedoucího katedry

III. PŘEVZETÍ ZADÁNÍ

Beru na vědomí, že jsem povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je nutné uvést v bakalářské práci a při citování postupovat v souladu s metodickou příručkou ČVUT „Jak psát vysokoškolské závěrečné práce“ a metodickým pokynem ČVUT „O dodržování etických principů při přípravě vysokoškolských závěrečných prací“.

<u>27.2.2020</u> Datum převzetí zadání	<u>[Signature]</u> Podpis studenta(ky)
---	---

Prohlášení

Prohlašuji, že jsem bakalářskou práci na téma "Optimalizace betonových prvků pomocí genetických algoritmů" zpracoval samostatně za použití uvedené literatury a pramenů. Dále prohlašuji, že nemám závažný důvod proti užití tohoto školního díla ve smyslu § 60 zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze (24.5.2020)

.....
Václav Wudi

Poděkování

Chtěl bych poděkovat Ing. Martinu Petříkovi, Ph.D. za jeho trpělivost, čas a připomínky. Dále bych chtěl poděkovat mojí rodině za podporu během studia.

Abstrakt

Tato práce se zabývá optimalizací betonových prvků. Jedná se o tvarovou a rozměrovou optimalizaci příhradové konstrukce. Optimum je vztaženo ke čtyřem kritériím. Těmi jsou tahové napětí, tlakové napětí, průhyb a hmotnost.

Pro výpočet vnitřních sil je použita metoda konečných prvků. Optimalizace je provedena genetickým algoritmem. Výpočet je implementován v programovacím jazyce Python. Výsledky z výpočtu jsou lepší ve všech parametrech optimalizace, než původní uvažovaná konstrukce.

Klíčová slova: Genetické algoritmy, metoda konečných prvků, optimalizace, betonová konstrukce.

Abstract

This thesis focus of a optimization concrete element. It is shape and dimension optimization of truss structure. Optimum is taken from four criterion. That is pressure, tensile, deflection and mass of the element.

For calculate force is used finite element method. Optimization performed using genetic algorithm. Calculation is implement in programming language Python. Final result are better in all parameter comparison with the original construction.

Keywords: Genetic algorithm, finite element method, optimization, concrete construction.

Seznam tabulek

4.1	Liniové zatížení nosníku	17
6.1	Hodnoty parametrů při různých vahách. Název je odvozen od zvolených vah.	25
6.2	Porovnání hodnoty fitness funkce průměru nejlepších jedinců v různých generacích při proměnné a konstantní mutaci.	27
6.3	Průměrné hodnoty fitness funkce nejlepšího jedince v závislosti na selekčním tlaku.	28
6.4	Porovnání hodnot konstrukcí.	29

Seznam obrázků

1.1	Příklad optimalizované budovy - Shogakukan Building	1
2.1	Znázornění procesu výpočtu metodou konečných prvků.	3
2.2	Příklad převedení příhradové konstrukce na výpočtový model.	4
2.3	Příklad jednotlivých posunů ve směru x a y a jejich očíslování	6
2.4	Transformace souřadnic.	7
3.1	Postup optimalizace pomocí genetických algoritmů.	9
3.2	Příklad ruletové selekce.	10
3.3	Příklad binárního jednobodového křížení.	11
3.4	Příklad vytvoření jedinců mezilehlém křížením	12
3.5	Příklad vytvoření jedinců přímkovým křížením	13
4.1	Pohled na interiér budovy.	15
4.2	Optimalizované parametry.	16
4.3	Schéma uložení příhradové konstrukce.	18
4.4	Schéma ukotvení konce zábradlí do konstrukce.	19
5.1	Příklad výpočtu přepony pravoúhlého trojúhelníka napsaném v Pythonu a c++. Zdroj z Pythonu je vlevo; c++ napravo.	20
5.2	Ověření funkčnosti, hledaným parametrem je výška trojúhelníku.	21
5.3	Hodnota napětí v benchmarku je dána vyšším z napětí.	21
5.4	Tvar a hodnoty fitness funkce pro průhyb konstrukce.	22
5.5	Tvar a hodnoty fitness funkce pro hmotnost konstrukce.	23
5.6	Tvar a hodnoty fitness funkce pro napětí konstrukce.	23
5.7	Funkce pravděpodobnosti mutace	24
6.1	Výsledek kdy uvažujeme, že všechny tři kritéria (použitelnost, únosnost a váha) mají stejnou váhu.	25
6.2	Průběh fitness funkce z šesti výpočtů. Fitness funkce nejlepšího jedince je znázorněna modře. Červeně je znázorněn průměr fitness funkcí jedinců. . .	26
6.3	Porovnání konstantní a proměnné mutace.	27
6.4	Porovnání tří různých selekčních tlaků.	28
6.5	Porovnávané konstrukce.	29
6.6	Vizualizace zvolené konstrukce.	30
A.1	Příklady konstrukcí z první generace výpočtu	33

Obsah

Abstract	vi
Seznam tabulek	vii
Seznam obrázků	viii
1 Úvod	1
2 Metoda konečných prvků	2
2.1 Rozdělení konstrukce na prvky	4
2.2 Lokální matice tuhosti	4
2.3 Globální matice tuhosti	5
2.4 Vektor zatížení	5
2.5 Vektor posunů	5
2.6 Řešení rovnice $K^*u=f$	6
2.7 Určení vnitřních sil	7
3 Genetické algoritmy	8
3.1 Fitness funkce	8
3.2 Selektce	9
3.2.1 Vážená ruleta	9
3.2.2 Pořadová selektce	10
3.2.3 Zbytkový stochastický výběr	10
3.2.4 Turnajová metoda	10
3.3 Křížení	11
3.3.1 Binární křížení	11
3.3.2 Křížení jedinců s genomem reálných čísel	11
3.4 Mutace	13
3.5 Nahrazovací strategie	13
3.6 Podmínky ukončení	14
4 Optimalizovaná konstrukce	15
4.1 Parametry	15
4.1.1 Známé parametry	15
4.1.2 Hledané parametry	16
4.2 Stavebně technické parametry	17
4.2.1 Zatížení	17
4.2.2 Uložení nosníku	18
4.2.3 Pochozí plocha	18

4.2.4	Zábradlí	19
5	Implementace v Pythonu	20
5.1	Python	20
5.1.1	Důvod volby Pythonu	20
5.2	Ověření funkčnosti	21
5.3	Zvolené parametry optimalizace	22
5.3.1	Fitness funkce	22
5.3.2	Mutace	24
5.3.3	Křížení a selekce	24
6	Výsledky	25
6.1	Průběh výpočtu	26
6.2	Volba mutace	27
6.3	Selekční tlak	28
6.4	Porovnání účinnosti	29
6.5	Zvolená konstrukce	30
7	Závěr	31
A	Příloha 1	33

Kapitola 1

Úvod

Při návrhu konstrukcí se můžeme setkat s požadavkem na řešení, které nejenom vyhovuje základním požadavkům, ale je zároveň z některého hlediska nejlepší. Řešení může být nejlepší z estetického, statického, ekonomického, provozního i jiného hlediska. Může jít také o kombinaci více hledisek. (viz Obr.1.1)



Obrázek 1.1: Příklad optimalizované budovy - Shogakukan Building

Cílem této práce je nalezení architektonicky optimální příhradové konstrukce. Výpočet bude proveden genetickými algoritmy a bude implementován v programovacím jazyce Python. Vnitřní síly budou vypočteny pomocí metody konečných prvků.

Kapitola 2

Metoda konečných prvků

Metoda konečných prvků je široce používaná metoda,[1], která řeší dané fyzikální, mechanické a matematické problémy numericky. Řešený celek se rozdělí na prvky s daným rozměrem a danými materiálovými charakteristikami. Převede se tak řešení diferenciálních rovnic (jednoduchých, nebo parciálních) na úlohu řešení soustavy lineárních rovnic.

Matematicky se jedná o zobecněnou Galerkinovu metodu.[2]

Do soustavy rovnic vstupuje tuhost konstrukce charakterizovaná globální maticí tuhosti, vektor přemístění, který charakterizuje předepsané přemístění a vektor zatížení, který charakterizuje zatížení konstrukce. (viz Obr. 2.1)

Soustavu rovnic můžeme zapsat jako:

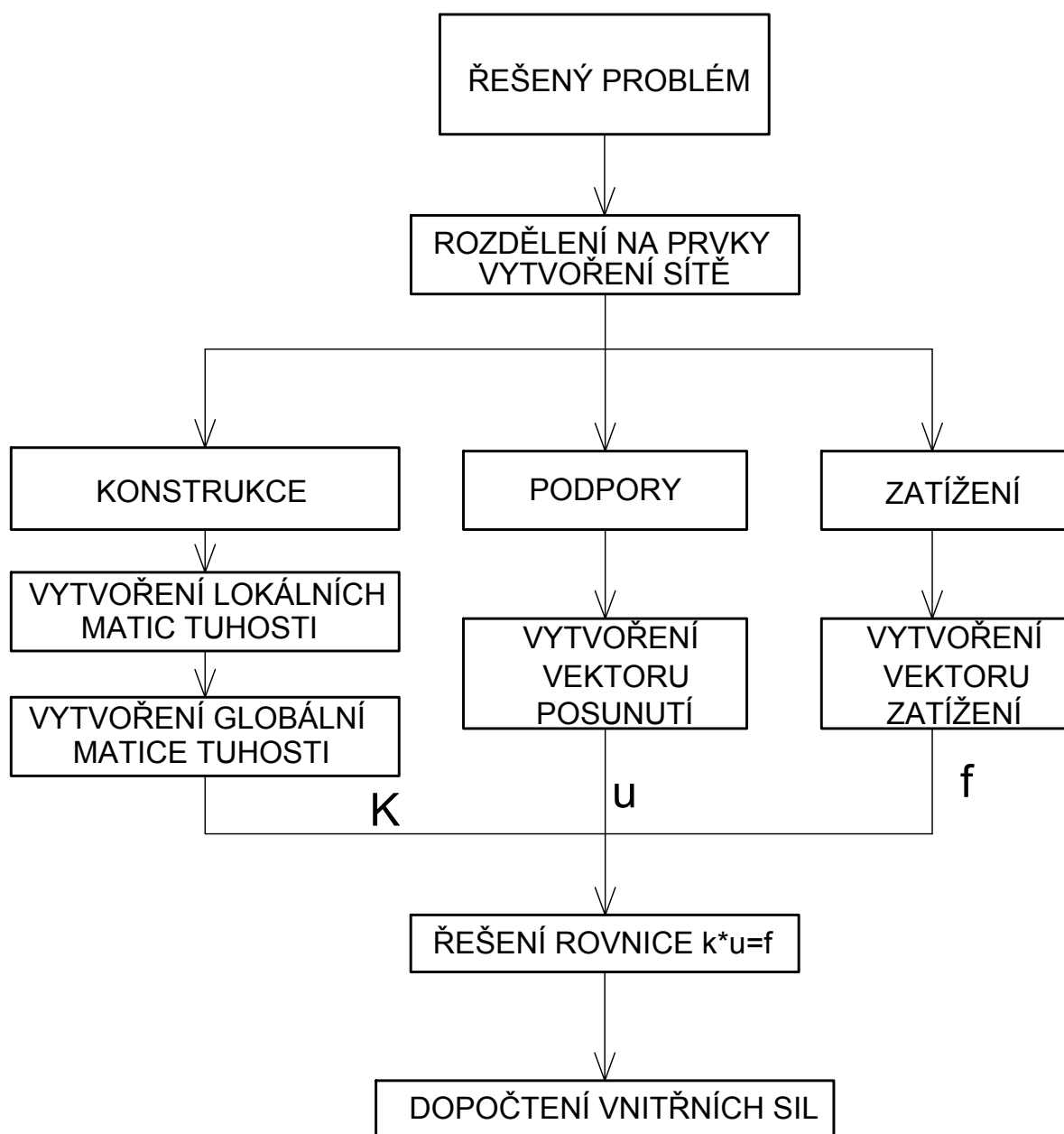
$$K * u = f \quad (2.1)$$

Kde K je matice tuhosti konstrukce, u je vektor přemístění a f vektor zatížení. Z dané rovnice poté můžeme dopočítat vnitřní síly.

V mém případě jde o metodu konečných prvků 2d prutových prvků. Pruty uvažuji jako tlačené, nebo tažené. Neuvažuji tedy s ohybovým momentem. Zároveň předpokládám, že pruty jsou výrazně větší než rozměry průřezu tedy že platí:

$$l \gg a \quad (2.2)$$

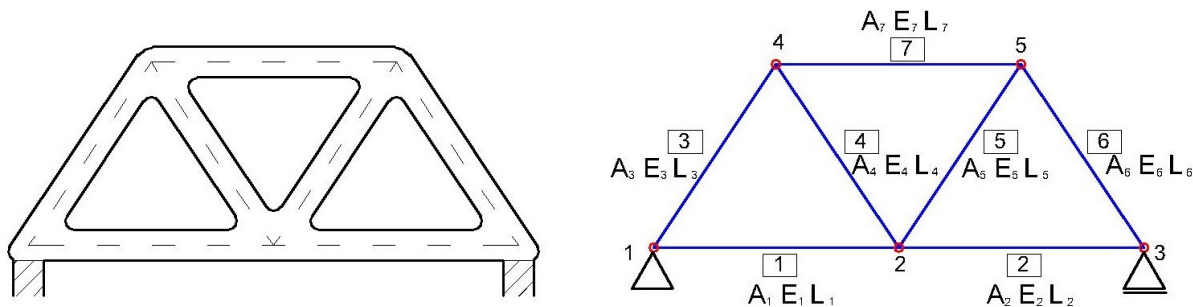
Kde l je délka prutu. Hodnota a je větší z rozměrů prutu. Pokud by rozdíl mezi délkou a rozměrem prutu byl příliš malý, už bychom nemohli uvažovat daný prvek jako prut, ale jako 2D, nebo 3D prvek.



Obrázek 2.1: Znázornění procesu výpočtu metodou konečných prvků.

2.1 Rozdělení konstrukce na prvky

Prvním krokem pro výpočet metodou konečných prvků je vytvoření sítě, která definuje prvky. U příhradové konstrukce se jedná o tažené/tlačené prvky spojené v uzlech. (viz Obr. 2.2) Při výpočtu tlakově /tahově namáhaných prutových konstrukcí je nejjednodušší možností zvolit jako prvek celé pruty.



Obrázek 2.2: Příklad převedení příhradové konstrukce na výpočtový model.

Každému prvku jsou přiřazeny jeho materiálové a geometrické vlastnosti, kterými jsou: průřezová plocha, délka, Youngův modul, počáteční a koncový uzel. U uzlu definujeme jeho polohu. Jednotlivé prvky jsou očíslovány.¹

2.2 Lokální matice tuhosti

Matice tuhosti vyjadřuje nakolik jednotková síla zdeformuje konstrukci.[3]

Matice tuhosti se dělí na lokální matice tuhosti, které definují tuhost prvku (prut) a globální matici, která definuje tuhost konstrukce.

Lokální matice prutu v 1D prostoru se vypočte z rovnice 2.3 [4]:

$$K_i = E_i * A_i * \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \quad (2.3)$$

Kde K je lokální matice tuhosti E je Youngův modul pružnosti, A je průřezová plocha prutu.

¹Číslování lze začít od 1, nebo od nuly. Při aplikaci v programovacím jazyku je lepší začít od 0, protože index matic a vektorů většiny programovacích jazyků začíná od 0. V tomto textu indexuji od 1, protože je to intuitivnější.

Při transformaci souřadnic prvku ve 2D prostoru matici tuhosti násobíme transformační maticí. (výpočet 2.4)

$$T = \begin{bmatrix} c^2 & c * s \\ c * s & s^2 \end{bmatrix} \quad (2.4)$$

Kde c značí cosinus úhlu a s sinus úhlu, který svírají lokální souřadnice prvku s globálními souřadnicemi.

Po roznásobení dostaneme matici lokální matici tuhosti. (výpočet 2.5,2.6)

$$K_i = T * K_1 \quad (2.5)$$

$$K_i = E_i * A_i * \begin{bmatrix} c^2 & c * S & -c^2 & -c * s \\ c * S & s^2 & -c * s & -s^2 \\ c^2 & c * s & c^2 & c * s \\ c * s & s^2 & c * s & S^2 \end{bmatrix} \quad (2.6)$$

Kde E je Youngův modul pružnosti, A je průřezová plocha prutu, c je cosinus úhlu α , s je sinus úhlu α . Úhel α je úhel který získáme natočením souřadnicové osy, tak abychom získali osu x rovnoběžnou s osou prutu.

2.3 Globální matice tuhosti

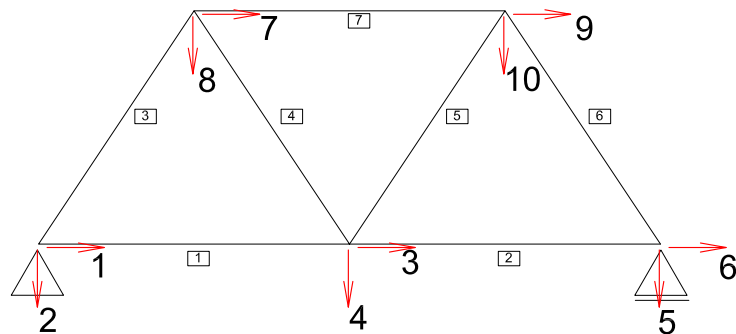
Globální matice tuhosti charakterizuje tuhost celé konstrukce. Globální matici tuhosti získáme dosazením lokálních matic tuhostí všech prvků do jedné matice. Dosazujeme pomocí kódových čísel.

2.4 Vektor zatížení

Vektor zatížení se čísluje obdobně jako vektor posunů. Na rozdíl od vektoru posunů, jsou hodnoty reakcí neznámé. U ostatních uzlů, se uvažuje hodnota rovné síle, která zde působí. Pokud do bodu nepůsobí žádná síla, počítáme se silou nulové velikosti.

2.5 Vektor posunů

Každý uzel má dva možné směry posunutí. Jedná se o posunutí ve směru x a y . Tyto posunutí značíme U_x a U_y . (viz Obr. 2.3) Ke každému prutu tak připadají čtyři možné posuny.



Obrázek 2.3: Příklad jednotlivých posunů ve směru x a y a jejich očíslování

Z přemístění uzlů lze sestavit vektor přemístění. V místech podpor je předepsán nulový posun. Nulové řádky můžeme z matice vyjmout. Tím získáme submatici tuhosti, se kterou dále postupujeme ve výpočtu. Ostatní posuny vstupují do výpočtu jako neznámé.

2.6 Řešení rovnice $K \cdot u = f$

Pokud máme globální matici tuhosti K a vektor zatížení f získáme rovnici $K \cdot u = f$. Neznámá u je vektor přemístění uzlů. Tato rovnice je maticovou rovnicí, ve které matice tuhosti K je pozitivně definitní. Existuje více možných způsobů jak tuto rovnici řešit. Volba způsobu řešení rovnice má vliv na časovou náročnost výpočtu. [5]

- Provedení výpočtu pomocí inverzní matice. Při provedení výpočtu skrze inverzní matici vynásobíme obě strany rovnice inverzní maticí matice tuhosti. Tím získáme rovnici:

$$u = f * K^{-1} \quad (2.7)$$

- LU faktorizace (někdy uváděno jako LU dekompozice) Rozdělení matice tuhosti na dvě trojúhelníkové matice (horní a dolní)

$$K = L * U \quad (2.8)$$

- Choleskyho dekompozice - zvláštní případ LU faktorizace s využitím pozitivní definitivnosti matice. Uvažujeme, že horní matice L a dolní matice U jsou stejné.
- Sdružené gradienty - jde o iterační metodu.

2.7 Určení vnitřních sil

Z vyřešení rovnice získáme hodnoty posunů uzlů ve směru X a Y. Pro prut vedený v 1D platí:

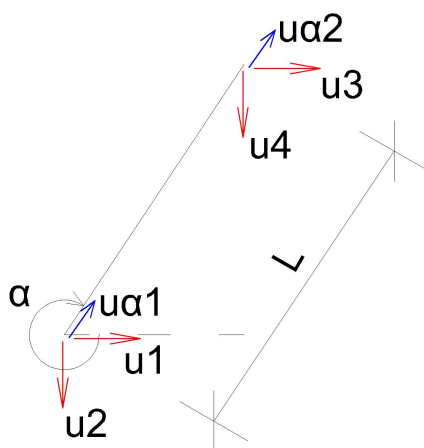
$$N = k_i * (u_2 - u_1) \quad (2.9)$$

Po dosazení transformace souřadnic do 2D prostoru platí (viz Obr. 2.4):

$$N = k_i * (u_{\alpha 2} - u_{\alpha 1}) \quad (2.10)$$

$$u_{\alpha 1} = \cos\alpha * u_1 + \sin\alpha * u_2 \quad (2.11)$$

$$u_{\alpha 2} = \cos\alpha * u_3 + \sin\alpha * u_4 \quad (2.12)$$



Obrázek 2.4: Transformace souřadnic.

Po dosazení získáme:

$$N = k_i * [\cos\alpha * (u_3 - u_1) + \sin\alpha * (u_4 - u_2)] \quad (2.13)$$

Z normálové síly v daném prutu vydělením průřezovou plochou dostaneme napětí.

Kapitola 3

Genetické algoritmy

Genetické algoritmy jsou analogií k přírodnímu výběru, kdy geny nejlépe vybaveného jedince se nejvíce předávají dále. Geny jedinců, kteří jsou málo uzpůsobeny pro přežití zanikají. Cíl výpočtu je získání nejlepšího jedince.

Základním prvkem je jedinec. Jedinec reprezentuje jedno řešení. Skupina řešení vzniklá ve stejném kroku optimalizace je generace.

Proces výpočtu pomocí genetických algoritmů se provádí tak, že nejdříve vytvoříme první náhodnou generaci s určeným počtem jedinců. Tyto jedince ohodnotíme pomocí fitness funkce. Následně vyselektujeme jedince a provedeme křížení. Tím vznikne nová generace. Poté se generace ohodnotí fitness funkcí a proces probíhá znovu dokud nezískáme optimální řešení. (viz Obr. 3.1)

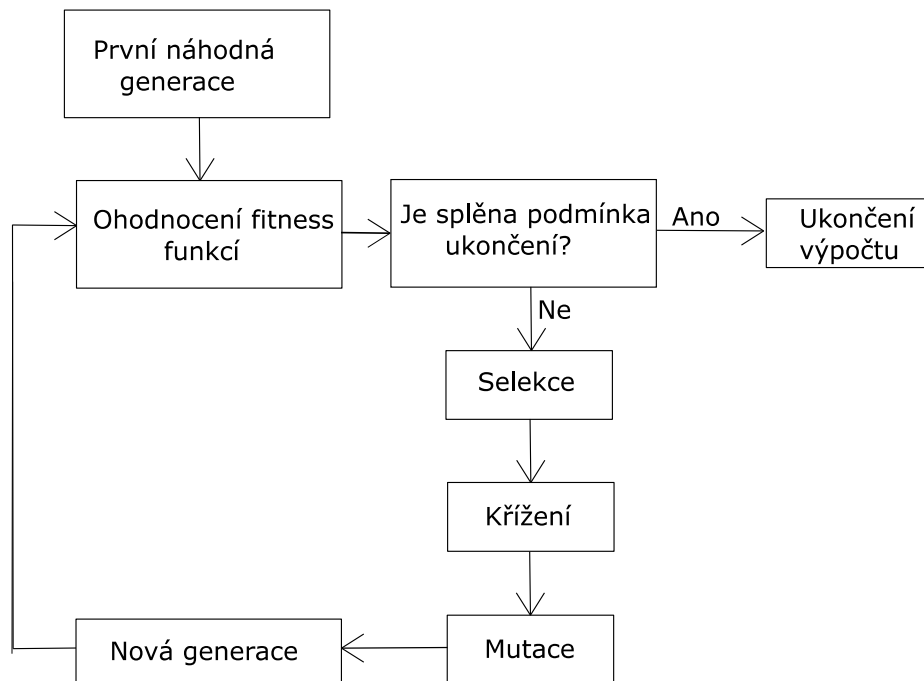
Optimální řešení je takové, ve kterém nabývá fitness funkce nejmenší hodnoty. Jedná se tedy o globální minimum funkce. V průběhu výpočtu se může stát, že výpočet skončí v lokálním minimu. Tomu se snažíme zabránit diverzitou populace a mutací.

3.1 Fitness funkce

Fitness funkce je ohodnocení daného jedince. Udává, nakolik jedinec vyhovuje podmínkám, které jsou na něj kladeny. Kritériem požadavků může být jedno nebo více. Při vícekritériální optimalizaci nedostáváme jedno ideální řešení, ale skupinu řešení.[6]

Volba matematického zápisu funkce by měla být funkcí, která má funkční hodnoty od 0 do 1. Kdy 0 jsou hodnoty nejhorší a 1 nejlepší. Případně je možné toto pořadí zvolit obráceně. Funkce by měla být matematicky vyjádřena tak, aby dostatečně vhodně popisovala užitečnost jedinců a její rozdíl.

Funkci není vhodné volit tak, aby v ní docházelo k ostrému zlomu, ani takovou, kde jsou hodnoty přesně nulové. Vhodnější je volit funkci tak, aby se hodnoty plynule blížily k nule.



Obrázek 3.1: Postup optimalizace pomocí genetických algoritmů.

3.2 Selektce

Selektce je proces výběru jedinců ke křížení. Při výběru chceme docílit toho, aby se vlastnosti jedinců, kteří mají větší fitness funkci, předávaly více.

3.2.1 Vážená ruleta

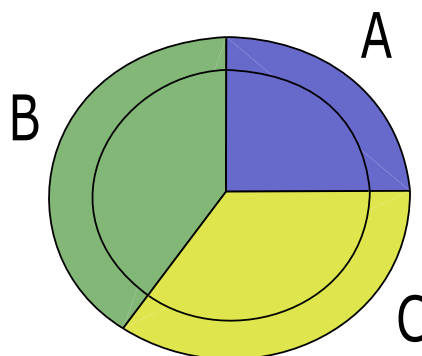
Vážená ruleta (*ruletová metoda, také označováno jako proporcionální výběr*) je odvozena od hraní rulety a ruletového kola. Velikost fitness funkce představuje počet polí na ruletě. (viz Obr 3.2)

Jde o případ selektce, kdy pravděpodobnost výběru jedince je přímo úměrnou poměru fitness funkce jedince k součtu všech fitness funkcí. Matematicky můžeme vyjádřit takto:

$$p = \frac{f_j}{\sum_{j=1}^n f_j} \quad (3.1)$$

kde p je pravděpodobnost výběru jedince, f značí fitness jedince, n je počet jedinců.

Jedinec	Fitness	prav. %
A	0,3	25
B	0,4	33,33
C	0,5	41,67



Obrázek 3.2: Příklad ruletové selekce.

3.2.2 Pořadová selekce

Pořadová selekce je obdobná vážené ruletě. Místo fitness funkce je použito pořadí jedince. Pravděpodobnost výběru nezávisí na velikosti fitness funkce, ale na pořadí jedince v populaci.

3.2.3 Zbytkový stochastický výběr

Mezi další metodu využívající principů ruletové metody můžeme zařadit zbytkový stochastický výběr. Při této metodě určíme na základě fitness funkce pravděpodobnost přežití jedince. Ta se vynásobí počtem jedinců a zaokrouhlí na celé číslo směrem dolů. Zbylá místa, která připadají na desetinná čísla, jsou obsazeny pomocí ruletové metody. [7]

3.2.4 Turnajová metoda

Další metodou, kterou můžeme zvolit je turnajová metoda. Při této metodě se náhodně zvolí n jedinců, mezi kterými se uspořádá turnaj. Zvítězí jedinec s nejlepší fitness funkcí. Počet turnajů se rovná počtu nově vzniklých jedinců.

Množstvím jedinců, kteří se účastní turnaje, je možno volit diverzitu populace. Při počtu n rovno jednomu jedinci v turnaji se jedná vlastně o náhodný výběr. Při volbě, kdy je n rovno velikosti populace se jedná o naprostý elitismus, kdy do následující generace přejde jen jeden nejlepší jedinec.

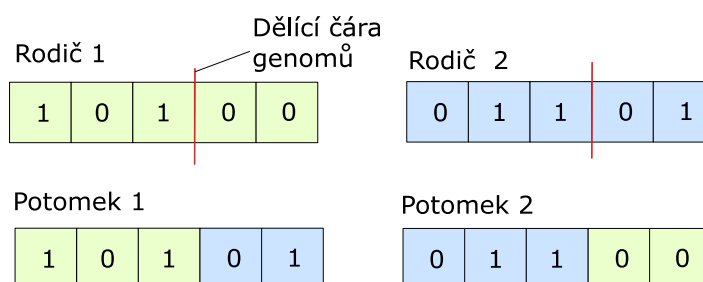
3.3 Křížení

Při křížení kombinujeme vlastnosti minimálně dvou jedinců. Z nich poté vznikne potomek nebo potomci. Typy křížení jsou rozdílné pro genom z binárních čísel a genom složený z reálných čísel.[8]

3.3.1 Binární křížení

Binární křížení se uplatňuje, pokud máme genom zapsaný pomocí binárního kódu.

Nejjednodušším řešením křížení je jednobodové křížení. V genomu rodičů se zvolí náhodně jedno místo. (dělicí čáru genomu) Potomek získá vstupní parametry do tohoto místa od jednoho rodiče a od zvoleného místa dál od druhého rodiče. Druhý potomek získá parametry opačné (viz Obr. 3.3)



Obrázek 3.3: Příklad binárního jednobodového křížení.

Při vícebodovém křížení volíme více bodů, kde dochází k přechodu přenosu parametrů z rodiče na potomka. Při náhodném křížení náhodně volíme, od kterého rodiče bude mít potomek genom.

3.3.2 Křížení jedinců s genomem reálných čísel

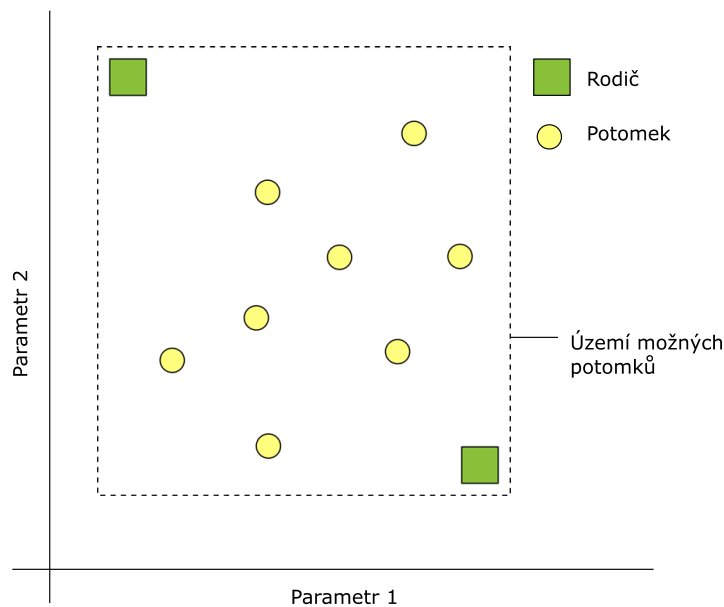
Metody křížení použité pro binární čísla jsou neúčinná pro reálná čísla. Způsoby křížení genomu reálných čísel, jsou založena na váženém průměru hodnot genu.

Mezilehlé křížení

Mezilehlé křížení (*Intermediate Recombination*) je druh křížení, při kterém genom potomků nabývá mezilehlých hodnot genomu rodičů. (viz Obr. 3.4). Hodnota genomu se získá pomocí vzorce:

$$Var_i^p = Var_i^r * \alpha_i + Var_i^r * (1 - \alpha_i) \quad (3.2)$$

Kde Var značí proměnnou v genomu, p potomka a r rodiče. Proměnná α nabývá hodnot od $-d$ do $1+d$. Hodnota α je dána náhodně pro každý genom.



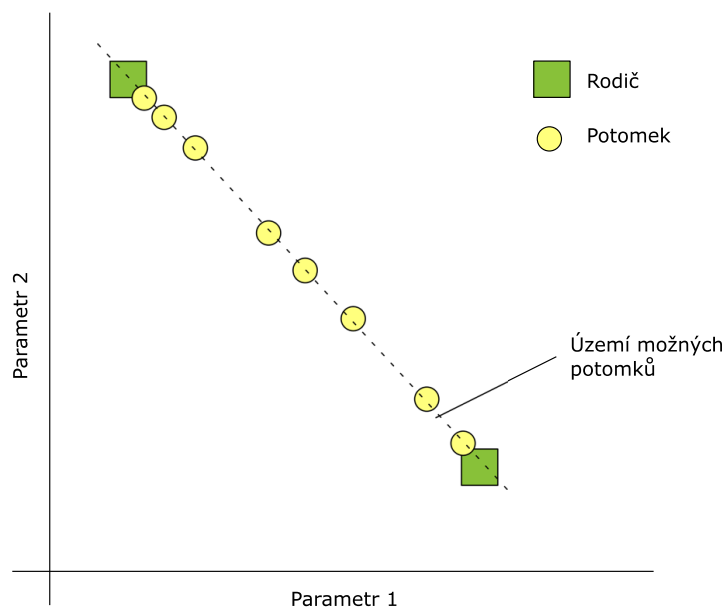
Obrázek 3.4: Příklad vytvoření jedinců mezilehlém křížením

Přímkové křížení

Přímkové křížení (*Line recombination*) je dalším druhem křížení jedinců. Při přímkovém křížení výsledná hodnota potomků leží na přímce. V tomto případě je hodnota genomu potomků na přímce, která se odvíjí od genomu rodičů. (viz Obr. 3.5)

$$Var_i^p = Var_i^r * \alpha + Var_i^r * (1 - \alpha) \quad (3.3)$$

Kde i značí jedince. Var značí proměnnou v genomu, p potomka a r rodiče. Proměnná α nabývá hodnot od $-d$ do $1+d$. Hodnota α je dána náhodně a je pro všechny geny stejná.



Obrázek 3.5: Příklad vytvoření jedinců přímkovým křížením

3.4 Mutace

Mutace je náhodná změna jednoho parametru, která probíhá s malou pravděpodobností. Mutací se snažíme zabránit, aby řešení neskončilo v lokálním minimu (maximu). Pravděpodobnost mutace může být konstantní, nebo se může zvyšovat s velikostí podobností daného parametru u jedinců.[9] ¹

3.5 Nahrazovací strategie

Do další generace mohou být zahrnuty jeden, nebo dva typy jedinců. Prvním jsou potomci vzniklí křížením a mutací. Potomky zařazujeme vždy. Dalším druhem jsou jedinci z předešlé generace. Jde buď o nejlepší jedince, nebo o jedince nejméně podobné potomkům.

Třetím možným typem, který je možno zahrnout do nové generace, jsou noví zcela náhodní jedinci.

¹To, jestli hledáme lokální maximum nebo minimum, záleží na zvolené fitness funkci. Pokud zvolíme fitness funkci tak, že nejlepší jedinci jsou s fitness funkcí rovni 0, tak hledáme globální minimum.

3.6 Podmínky ukončení

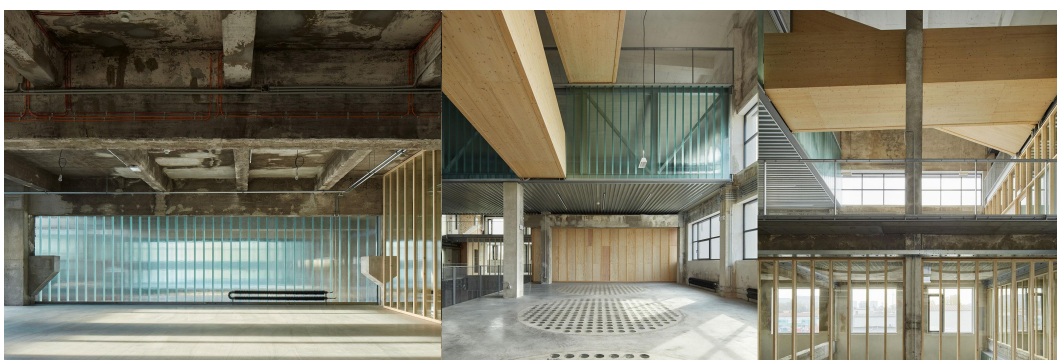
Podmínky k ukončení běhu výpočtu mohou být: [10] [11]

- Dosažení maximálního počtu generací.
- Bylo dosaženo dolní hodnoty rozsahu fitness funkce, případně je výpočet blízko ní.
- Pravděpodobnost, že se hodnoty změní, je příliš malá.

Kapitola 4

Optimalizovaná konstrukce

Předmětem optimalizace v této práci je nalezení tvaru příhradové lávky, která by spojila dvě části interiéru, mezi nimiž se nachází atrium. Jde o architektonický prvek, který bude osazen v objektu Bratislavské Mlýnice. V objektu se nachází několik lávek, které jsou ze dřeva nebo betonu. (viz Obr. 4.1)



Obrázek 4.1: Pohled na interiér budovy.

4.1 Parametry

Okolní prostor a účel lávky jsou pevně dané. Hledanými parametry jsou parametry popisující konstrukci. Předpokládám je, že se jedná o příhradovou konstrukci. Topologicky jiné tvary, nejsou ve výpočtu zohledněny.

4.1.1 Známé parametry

Světlá šířka prostoru, přes který bude provedena lávka, je 9 metrů. Šířka lávky bude činit 1,2 metru. Lávka je zatížena vlastní tíhou, stálým zatížením od skladby podlahy a zábradlí a užitným zatížením.

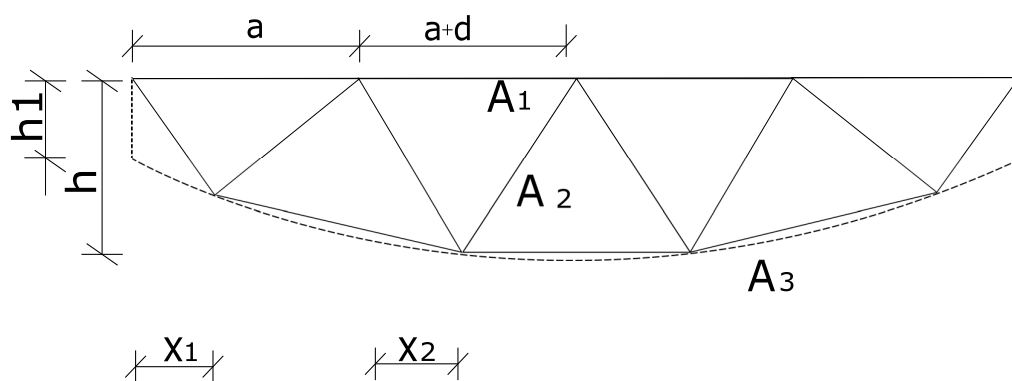
Beton, který bude použit na konstrukci, je pohledový. Bude mít tedy nejen funkci nosnou, ale zároveň funkci estetickou. Vzhledem k poměrně štíhlým prvkům lze do betonu přidávat jen jemné kamenivo. Jako výztuž bude použita běžná betonářská výztuž doplněná drátky ke snížení velikosti trhlin. V tažených částech bude výztuž dimenzovaná na příslušný tah. V tlačené prvky budou obsahovat konstrukční výztuž.

Uvažuji ocel B500B a beton C45/55 XC2 D-max 4 mm. Pohledová vzdálenost betonu je 15 metrů. Do betonu nebudou přidány žádné pigmenty ovlivňující barvu, bude mít tedy standardní šedou barvu.

4.1.2 Hledané parametry

Předmětem optimalizace je nalezení tvaru a rozměrů konstrukce. Konstrukci uvažujeme osově symetrickou. Parametry popisují tvar osy prvků příhradové konstrukce a velikost plochy průřezů.

Konstrukce je popsána dvanácti hledanými parametry. (viz obr. 4.2, příklady konstrukcí v první generaci jsou v příloze č 1.) Hledanými parametry je výška příhradové konstrukce, změna výšky, počet polí příhradové konstrukce a rozměr jednotlivých polí. Mezi rozměry polí uvažuji lineární závislost. Dalším parametrem je sklon diagonál. Ten je charakterizován dvěma parametry. Posledních šest parametrů popisuje velikost ploch dílčích prvků. Jde o plochy horní a dolní pásnice a diagonál. Plocha pásnic a jednotlivých diagonál není konstantní, ale je uvažována mezi jednotlivými pásnicemi kvadratická závislost.



Obrázek 4.2: Optimalizované parametry.

Kritéria optimalizace

Při optimalizaci konstrukce byla zvolena čtyři kritéria optimalizace.

- Tahové namáhání - nejvyšší tahové namáhání
- Tlakové namáhání - nejvyšší tlakové namáhání
- Průhyb - lineární průhyb uprostřed nosníku (největší průhyb)
- Hmotnost konstrukce

4.2 Stavebně technické parametry

Kromě vyřešení tvaru nosníku, jsem v rámci bakalářské práce vyřešil technické parametry, které jsou nutné pro realizaci. Jedná se o zatížení, uložení nosníku na průvlak, skladbu podlahy nad nosníkem a vyřešení kotvení zábradlí.

4.2.1 Zatížení

Nosník je zatížen proměnným, stálým zatížením od RHS profilů, zábradlí, dřevěné nášlapné vrstvy a vlastní tíhy (viz. Tabulka 4.1)[12]. Jde o prostý nosník, proto mohu uvažovat s jedním návrhovým stavem, který reprezentuje maximální možné zatížení.

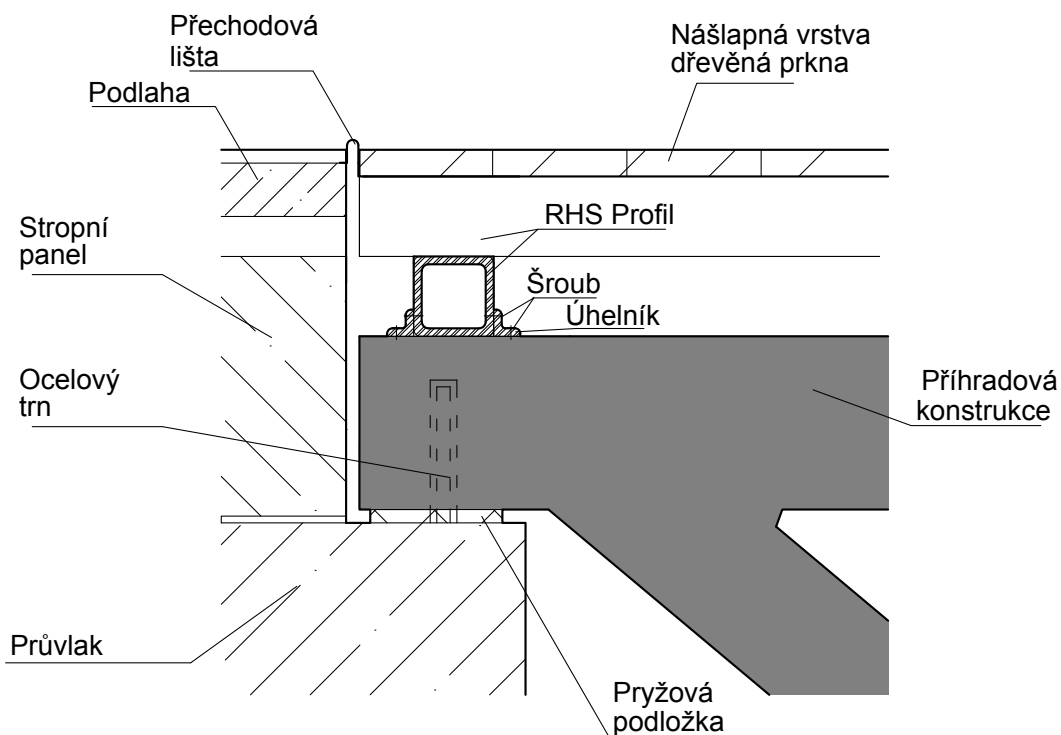
Zatížení	f_k [kN/m]	l [m]	γ [-]	f_d [kN/m]
RHS profily a úhelníky	0,3		1,35	0,405
Nášlapná vrstva dřevěná	7*0,02	1,2	1,35	0,227
Zábradlí	0,5	-	1,35	0,675
Celkem stálé				1,31
Proměnné zatížení	5	1,2	1,5	9
Celkem				10,3

Tabulka 4.1: Liniové zatížení nosníku

Proměnné zatížení uvažuji jako kategorii C3, prostor kde může docházet ke shromažďování osob (5 kN/m). Plošné zatížení jsem převedl na liniové a následně na bodové, které působí v uzlech příhradové konstrukce. Účinky vlastní tíhy vždy rozdělují mezi příslušné uzly, na které působí.

4.2.2 Uložení nosníku

Nosník bude uložen prostě na průvlak. Předpokládám, že průvlaky mají dostatečnou únosnost. Uložení bude provedeno na pryžovou podložku. Ta zabraňuje přenášení dynamických účinků z lávky do zbytku budovy. Stabilita nosníku je zajištěna ocelovým trnem. (viz Obr. 4.3)



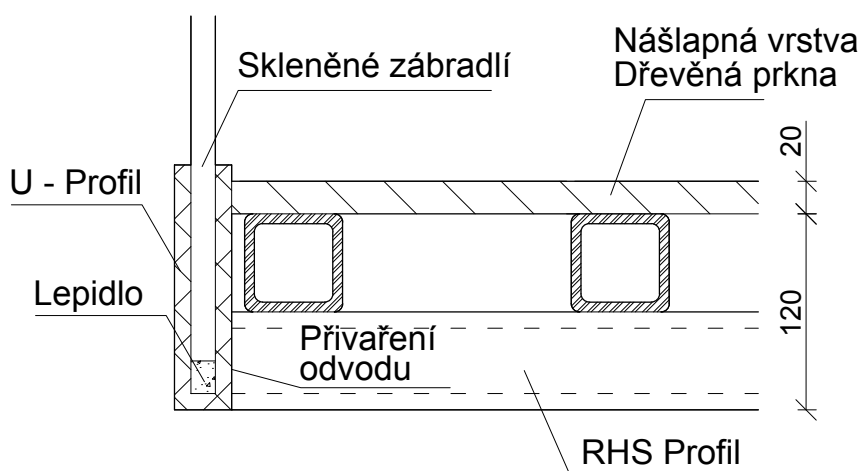
Obrázek 4.3: Schéma uložení příhradové konstrukce.

4.2.3 Pochozí plocha

Nad nosníkem bude proveden rošt, který tvoří obdélníkové duté válcované RHS profily vedené. Poloha RHS profilů je zajištěna úhelníkovými úpaly, které jsou přišroubovány k přehradové konstrukci a k profilu. Na RHS profily bude provedena nášlapná vrstva z dřevěných prken. Prkna budou přišroubována na ocelový profil vruty.

4.2.4 Zábradlí

Zábradlí je navrženo skleněné. Jeho účelem je zabránit pádu osob do prostoru atria. Vzdálenost mezi plochou atria a lávky přesahuje 12 metrů. Z tohoto důvodu je výška zábradlí 1100 mm. Zábradlí je uchycené pomocí u profilu do ocelového roštu. (viz Obr. 4.4)



Obrázek 4.4: Schéma ukotvení konce zábradlí do konstrukce.

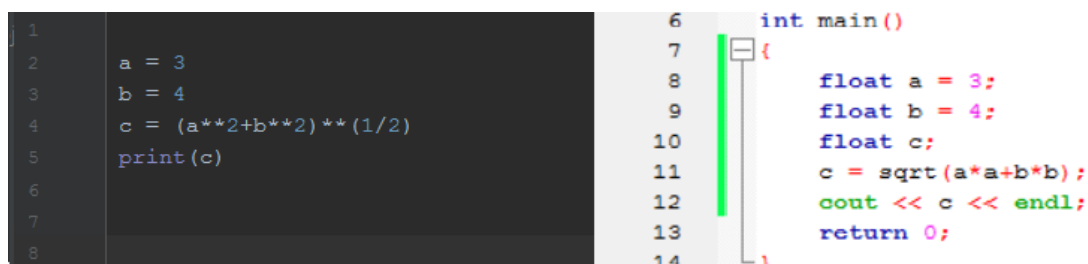
Kapitola 5

Implementace v Pythonu

Výpočet konstrukce je proveden v programovacím jazyce Python. Součástí výstupu mojí práce je program a zdrojový kód. Kromě optimalizace byl vytvořen testovací příklad k ověření funkčnosti algoritmu. K tomu, aby program mohl počítat, bylo nutné zvolit průběhy fitness funkcí proměnných (napětí v tlaku a tahu, průhyb a hmotnost) a hodnotu mutace. (průběh funkce mutace)

5.1 Python

Python je dynamický programovací jazyk. První verze byla vytvořena roku 1991 Guidem van Rossumem. Jedná se jednoduše čitelný programovací jazyk, který podporuje objektově orientované programování. Typ proměnné se oproti jiným programovacím jazykům (např. C++) nemusí deklarovat. (viz Obr. 5.1)[13]



The image shows two side-by-side code snippets. The left snippet is Python code with line numbers 1-8. It defines variables a=3, b=4, and calculates c as the square root of (a squared plus b squared). The right snippet is C++ code with line numbers 6-14. It defines a main function with float variables a=3, b=4, and c, then calculates c using the sqrt function and prints the result.

```
1 a = 3
2 b = 4
3
4 c = (a**2+b**2)**(1/2)
5 print(c)
6
7
8
```

```
6 int main()
7 {
8     float a = 3;
9     float b = 4;
10    float c;
11    c = sqrt(a*a+b*b);
12    cout << c << endl;
13    return 0;
14 }
```

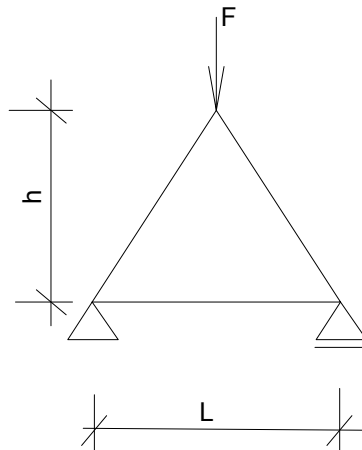
Obrázek 5.1: Příklad výpočtu přepony pravoúhlého trojúhelníka napsaném v Pythonu a c++. Zdroj z Pythonu je vlevo; c++ napravo.

5.1.1 Důvod volby Pythonu

Python jsem si vybral především pro svojí jednoduchost a množství knihoven, které usnadňují práci.

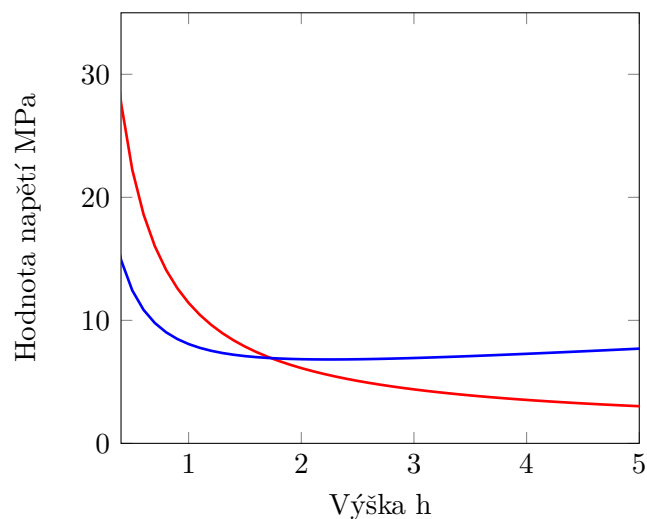
5.2 Ověření funkčnosti

Pro ověření funkčnosti genetického algoritmu jsem vytvořil validaci (viz Obr. 5.2). Validace představuje optimalizaci výšky trojúhelníku zatíženého silou a vlastní vahou. Po-



Obrázek 5.2: Ověření funkčnosti, hledaným parametrem je výška trojúhelníku.

suzovaným parametrem bylo maximální napětí. Posuzuje se absolutní hodnota napětí. Se vzrůstající výškou se zvyšuje výška tlačných vzpěr a tím i tlakové napětí. Naopak klesá tahové namáhání. K nejmenšímu celkovému napětí dochází při rovnosti tahového a tlakového namáhání. (viz Obr. 5.3)



Obrázek 5.3: Hodnota napětí v benchmarku je dána vyšším z napětí.

Tlakové namáhání lze vypočítat:

$$N = (F + A * rho * (h^2 + L^2)^{1/2}) * (h^2 + L^2)^{1/2} / (2 * h) \quad (5.1)$$

Tahové namáhání lze získat:

$$N = (F + A * rho * (h^2 + L^2)^{1/2}) * L / (h) \quad (5.2)$$

Nejmenšímu maximálnímu napětí dochází při rovnostranném trojúhelníku, kdy výška je rovna:

$$h = 0.5 * L * 3^{1/2} \quad (5.3)$$

V mém případě byl tento příklad implementován do algoritmu. Výsledky vyšly přesně na 3 desetina místa.

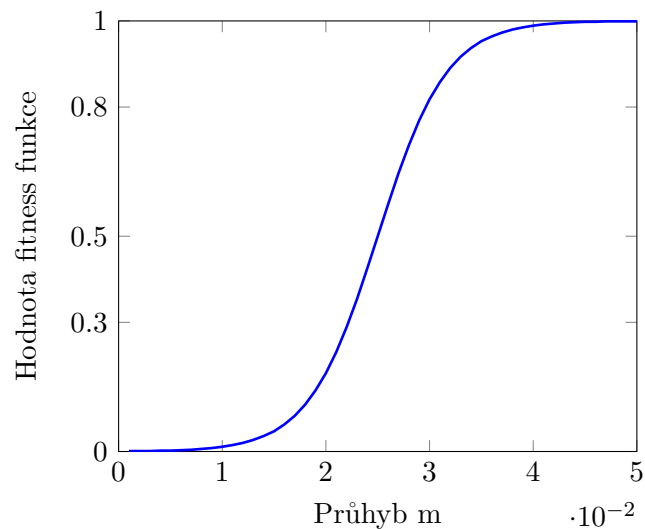
5.3 Zvolené parametry optimalizace

Zde jsou uvedeny fitness funkce jednotlivých kritérií a funkce mutace.

5.3.1 Fitness funkce

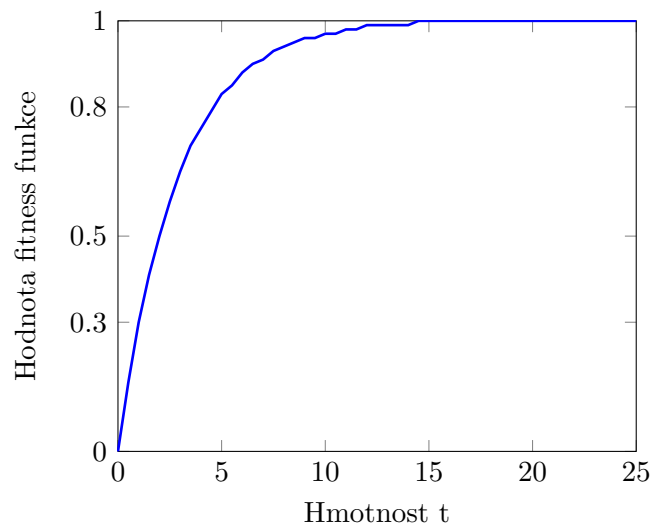
Pro každý hodnotící parametr který vstupuje do výpočtu byla zvolena fitness funkce.

Průhyb je jeden z parametrů ovlivňující optimalizaci konstrukce. Jako fitness funkci jsem zvolil distribuční funkci Gaussova rozdělení. (viz. Obr. 5.4)



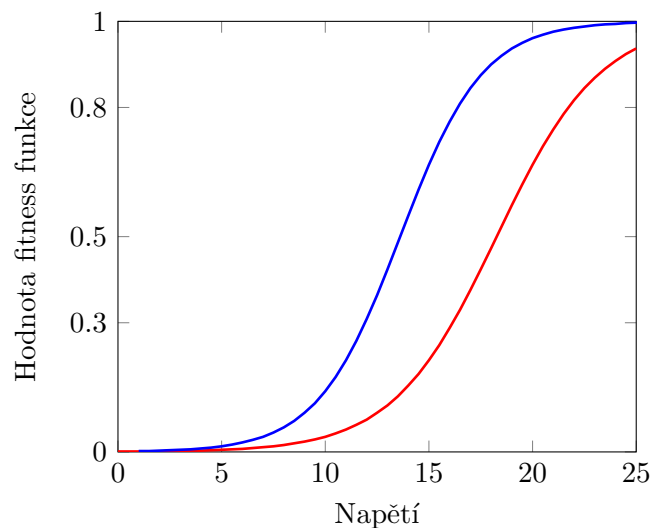
Obrázek 5.4: Tvar a hodnoty fitness funkce pro průhyb konstrukce.

Snažím se o co nejlehčí konstrukci. Fitness funkci jsem zvolil jako distribuční funkci logaritmické rozdělení. (viz Obr. 5.5)



Obrázek 5.5: Tvar a hodnoty fitness funkce pro hmotnost konstrukce.

U napětí zohledňuji tlakové napětí a tahové napětí. Fitness funkce je určena na základě největší hodnoty daného napětí na konstrukci. (viz Obr. 5.6) Modře je znázorněn tah, červeně tlak.



Obrázek 5.6: Tvar a hodnoty fitness funkce pro napětí konstrukce.

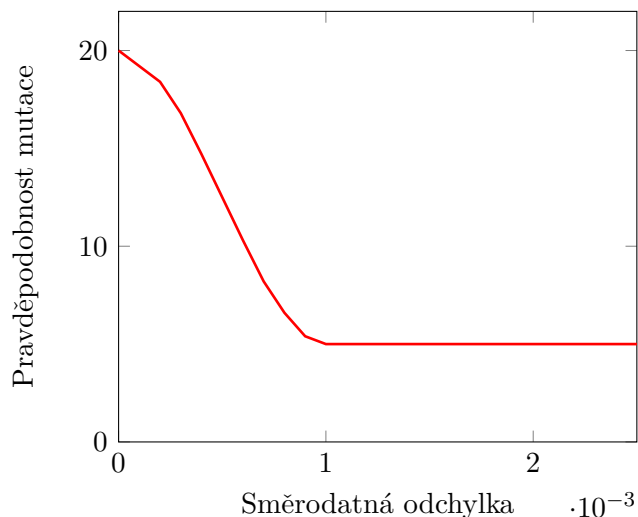
Přikládání vah fitness funkcí je provedeno:

$$f_{sum} = w_1 * f_{tlak} + w_2 * f_{tah} + w_3 * f_{průhyb} + w_4 * f_{hmotnost} \quad (5.4)$$

kde f_{sum} je celková fitness funkce, w_n jsou jednotlivé váhy příslušící k dané fitness funkci proměnné, f_i je hodnota fitness funkce dané proměnné značené v indexu.

5.3.2 Mutace

Funkci mutace jsem zvolil, tak aby plynule přecházela z maximální hodnoty na hodnotu minimální. Jedna z možností bylo proložit body přímkou. Zvolená funkce je kubická funkce, na kterou je od určité hodnoty napojena přímka (viz Obr. 5.7)



Obrázek 5.7: Funkce pravděpodobnosti mutace

5.3.3 Křížení a selekce

Křížení je provedeno pomocí přímkového křížení. Selekcce je prováděna ruletovou selekcí. Selekcční tlak docilují násobením převedení fitness funkce vzorcem:

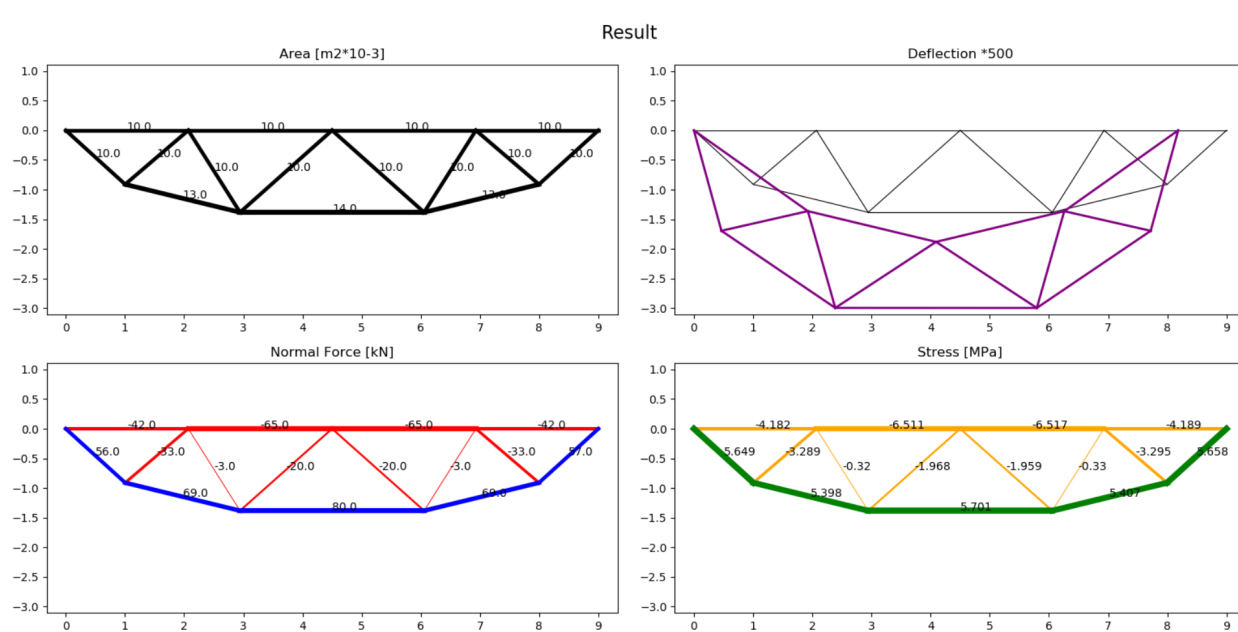
$$f_{new} = A * f_{beg} + B \quad (5.5)$$

kde f_{new} je hodnota nové fitness funkce, f_{beg} je původní hodnota fitness funkce, A a B jsou proměnné, které určují selekcční tlak.

Kapitola 6

Výsledky

K dosažení výsledků jsem provedl výpočet s různými váhami kritérií. (viz. Obr. 6.1, Tab 6.1) Na obrázku je konstrukce se stejnými váhami parametrů.



Obrázek 6.1: Výsledek kdy uvažujeme, že všechny tři kritéria (použitelnost, únosnost a váha) mají stejnou váhu.

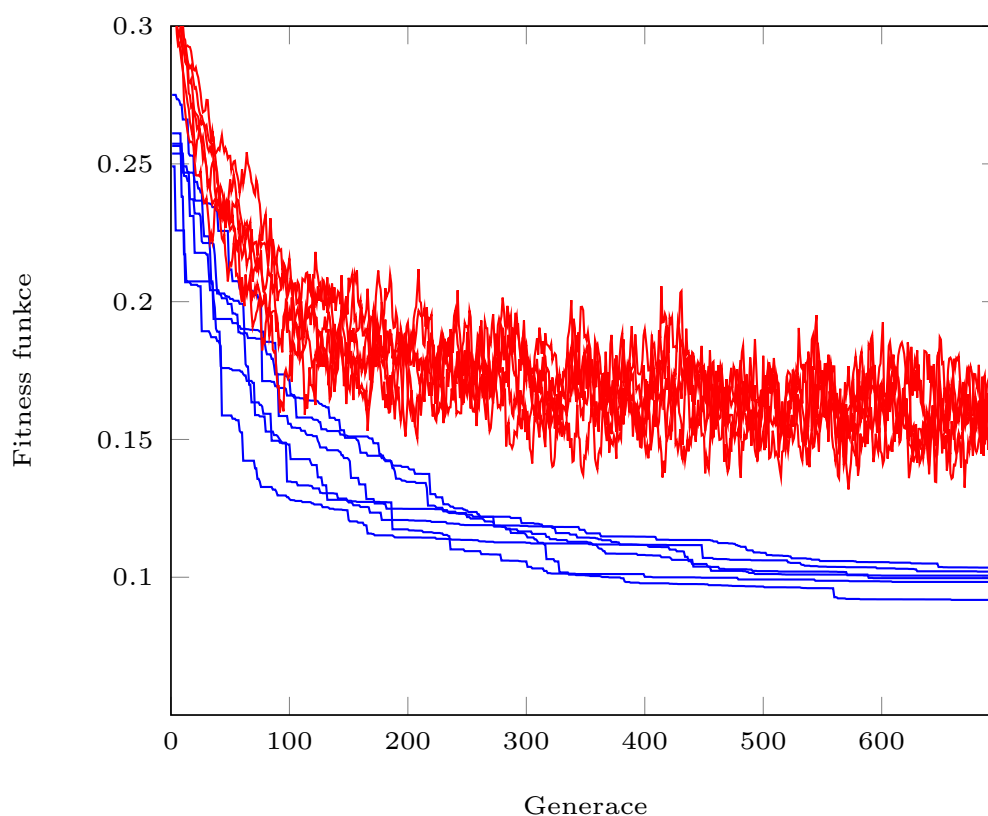
Zvolené váhy	h	n	delta	deltaN	deltaN2	deltah
1/1/1/1	1,74	4	0,2	0,54	0,11	0,26
3/1/1/1	1,76	4	0,08	0,6	0,15	0,22
1/3/1/1	1,62	4	0,05	0,42	0,01	0,48
1/1/3/1	2	4	0,78	0,48	0,12	0,11
1/1/1/3	1,76	4	0,26	0,42	0,045	0,38

Tabulka 6.1: Hodnoty parametrů při různých vahách. Název je odvozen od zvolených vah.

Hodnota δ určuje vodorovnou polohu uzlů horní pásnice. Hodnoty δN a $\delta N2$ značí vodorovnou polohu uzlů dolní pásnice. Hodnoty h a δ určují výšku. Značení vah je v pořadí průhyb/ hmotnost/ tlak/ tah.

6.1 Průběh výpočtu

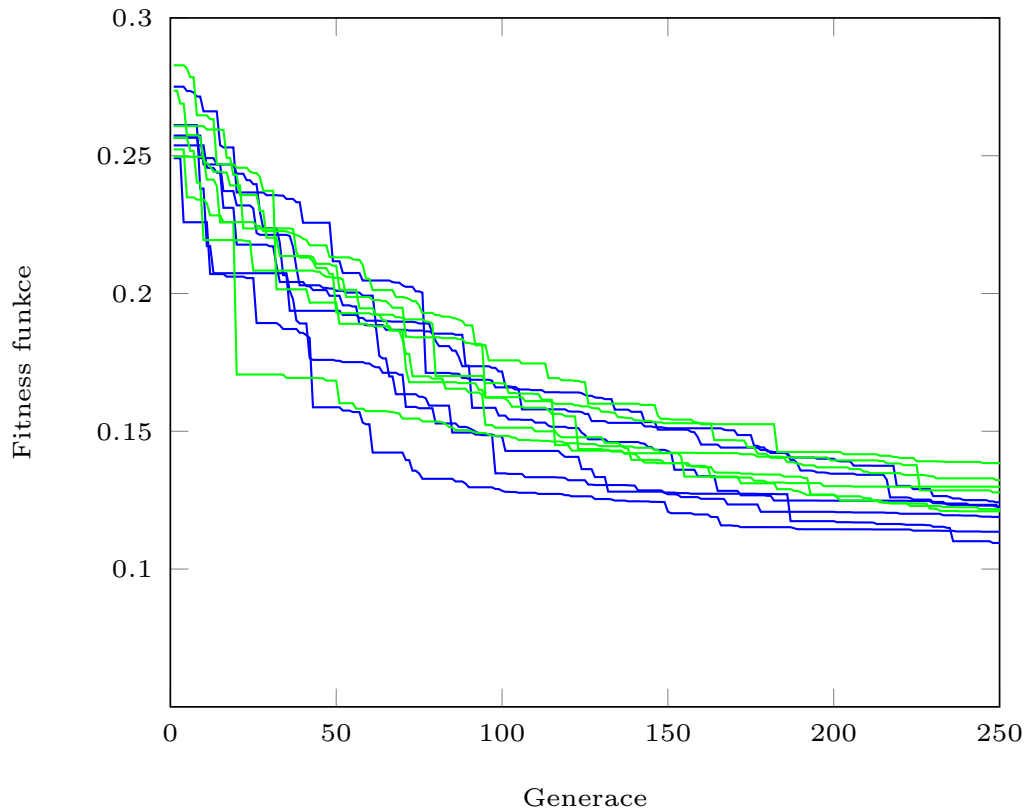
Na grafu níže (viz Obr. 6.2) je znázorněna fitness funkce nejlepšího jedince a průměr fitness funkce všech jedinců, během výpočtu.



Obrázek 6.2: Průběh fitness funkce z šesti výpočtů. Fitness funkce nejlepšího jedince je znázorněna modře. Červeně je znázorněn průměr fitness funkcí jedinců.

6.2 Volba mutace

Mutaci jsem zkoušel jak s konstantní, tak s proměnou hodnotou. Z výsledků lze vyčíst, že zavedení proměnné mutace v průměru zvyšuje účinnost výpočtu. Fitness funkce běhu výpočtu kdy se mutace mění, klesá rychleji než u běhu s mutací nastavenou na konstantní hodnotu. (viz. Obr. 6.3, Tab. 6.2) Modře je vyznačena proměnná mutace, zeleně konstantní.



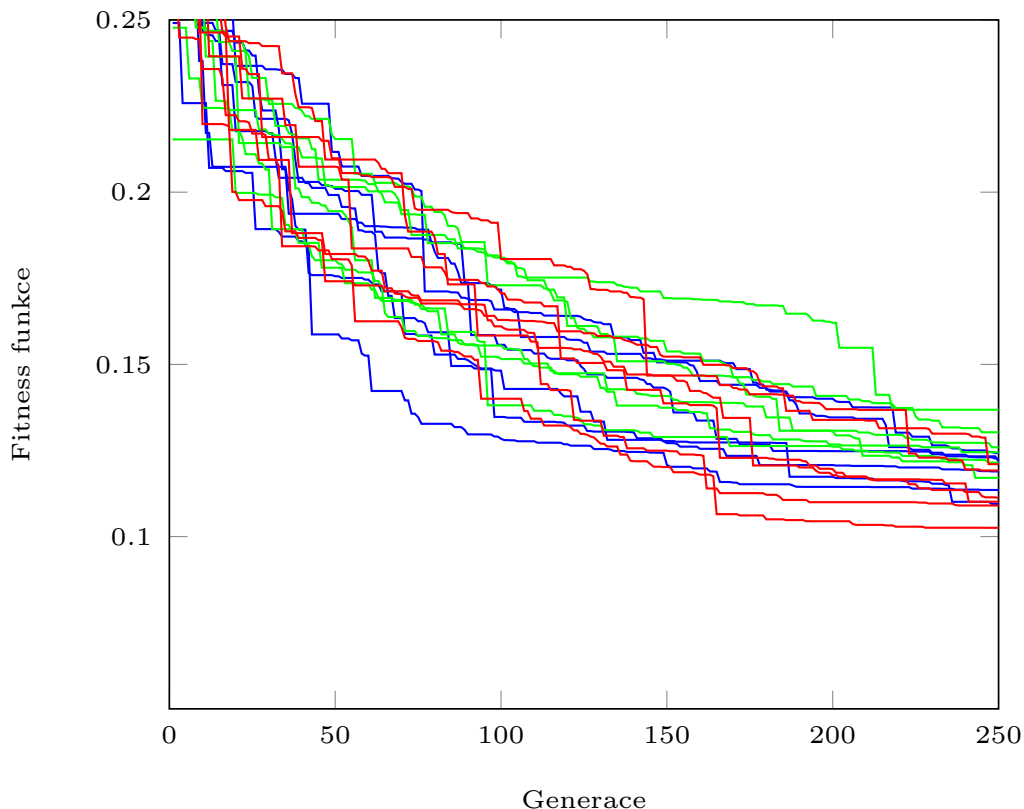
Obrázek 6.3: Porovnání konstantní a proměnné mutace.

Generace	50	100	150	200	250
Proměnná	0,196	0,163	0,138	0,121	0,112
Stálá	0,190	0,151	0,147	0,136	0,126

Tabulka 6.2: Porovnání hodnoty fitness funkce průměru nejlepších jedinců v různých generacích při proměnné a konstantní mutaci.

6.3 Selekcční tlak

Provedl jsem výpočet s různě velkými selekcčními tlaky.(viz Obr. 6.4), kde červeně je vyznačena fitness nejlepšího jedince při nejvyšším selekcčním tlaku. Modře fitness nejlepšího jedince při středním selekcčního tlaku. Zeleně je vyznačen nejnižší selekcční tlak.



Obrázek 6.4: Porovnání tří různých selekcčních tlaků.

Z výsledků není přímo jasné, že by některý selekcční tlak byl vhodnější. (viz. Tab. 6.3)

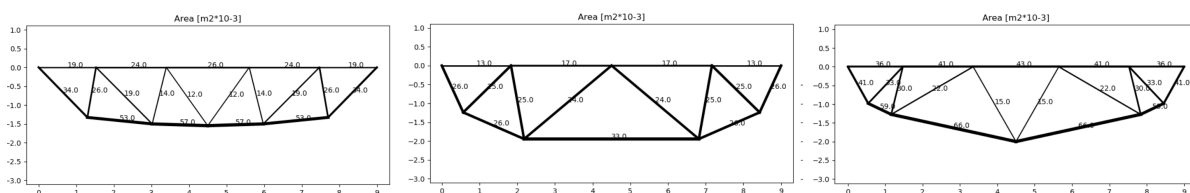
Generace	50	100	150	200	250
Nejnižší	0,195	0,163	0,156	0,147	0,126
Střední	0,190	0,163	0,165	0,137	0,119
Nejvyšší	0,194	0,162	0,160	0,139	0,112

Tabulka 6.3: Průměrné hodnoty fitness funkce nejlepšího jedince v závislosti na selekcčním tlaku.

6.4 Porovnání účinnosti

Porovnání provádím se základní konstrukcí, kterou jsem zvolil na začátku před optimalizací. Tuto konstrukci porovnávám se dvěma konstrukcemi. První z těchto konstrukcí je konstrukce, která je výsledkem procesu optimalizace.

Vzhledem k tomu, že cílem je navrhnout architektonický prvek, tak jsem druhou konstrukci na porovnání zvolil jednu z konstrukcí, kterou jsem subjektivně považoval za tvarově zajímavou. (viz. Obr. 6.5)



Obrázek 6.5: Porovnávané konstrukce.

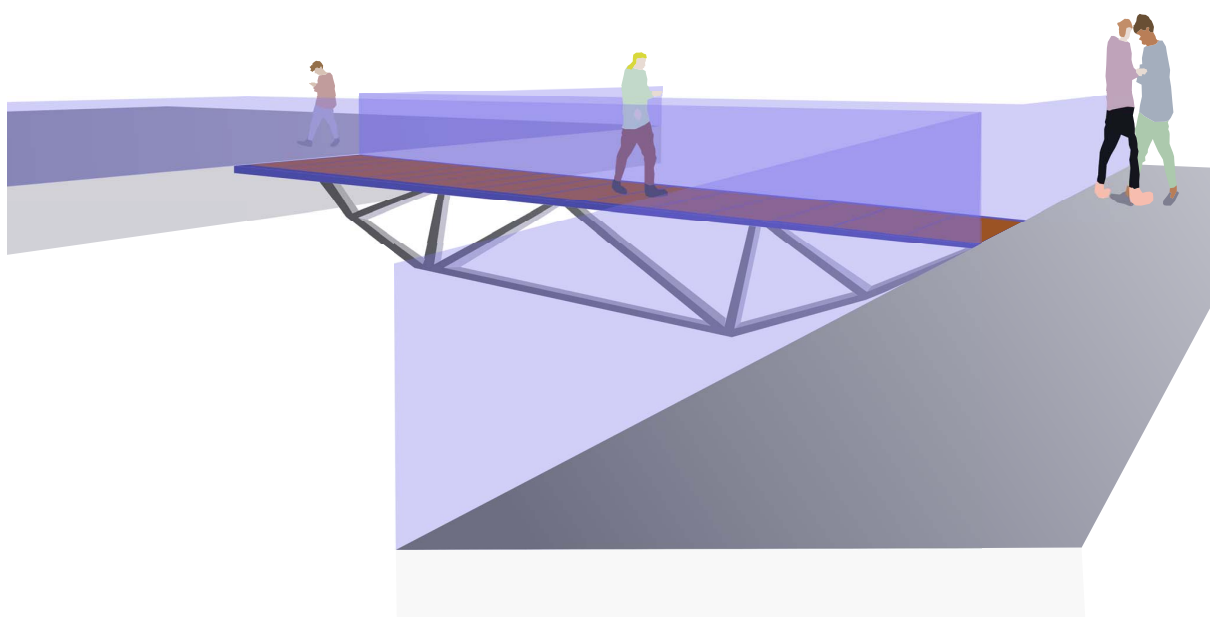
Výsledná konstrukce byla lepší ve všech parametrech. (viz. Tab. 6.4) U tlakových a tahových napětí uvažuji nejvyšší hodnotu. Tvarově zajímavější konstrukce byla výrazně lepší ve všech parametrech, kromě hmotnosti, kde byla výrazně horší.

Konstrukce	Tah	roz.	Tlak	roz.	Průhyb	roz.	Hmotnost	roz.
Jednotky	MPa		MPa	%	mm	%.	t	%
Počáteční	1,94	-	3,028	-	1,45	-	2,32	-
Optimální	1,93	0,5%	2,595	14,3%	1,08	24,5%	1,964	15,4%
Vybraná	1,44	25,7%	1,4	53%	0,6	58%	3,34	-44%

Tabulka 6.4: Porovnání hodnot konstrukcí.

6.5 Zvolená konstrukce

Výpočet probíhal opakovaně s různými vahami a jejím výsledkem je velká skupinu řešení. Z této skupiny jsem jedno vybral (viz. Obr. 6.6)



Obrázek 6.6: Vizualizace zvolené konstrukce.

Kapitola 7

Závěr

Hlavním přínosem bakalářské práce je nalezení optimálních rozměrů konkrétního betonového prvku. Výsledný tvar je lepší než původní výchozí prvek ve všech parametrech, které byly posuzovány (maximální tah, tlak, průhyb a hmotnost). Pomocí adaptivní mutace jsem byl schopen urychlit konvergenci výpočtu. Naopak různé selekční tlaky nevykazovaly znatelný vliv na běh programu.

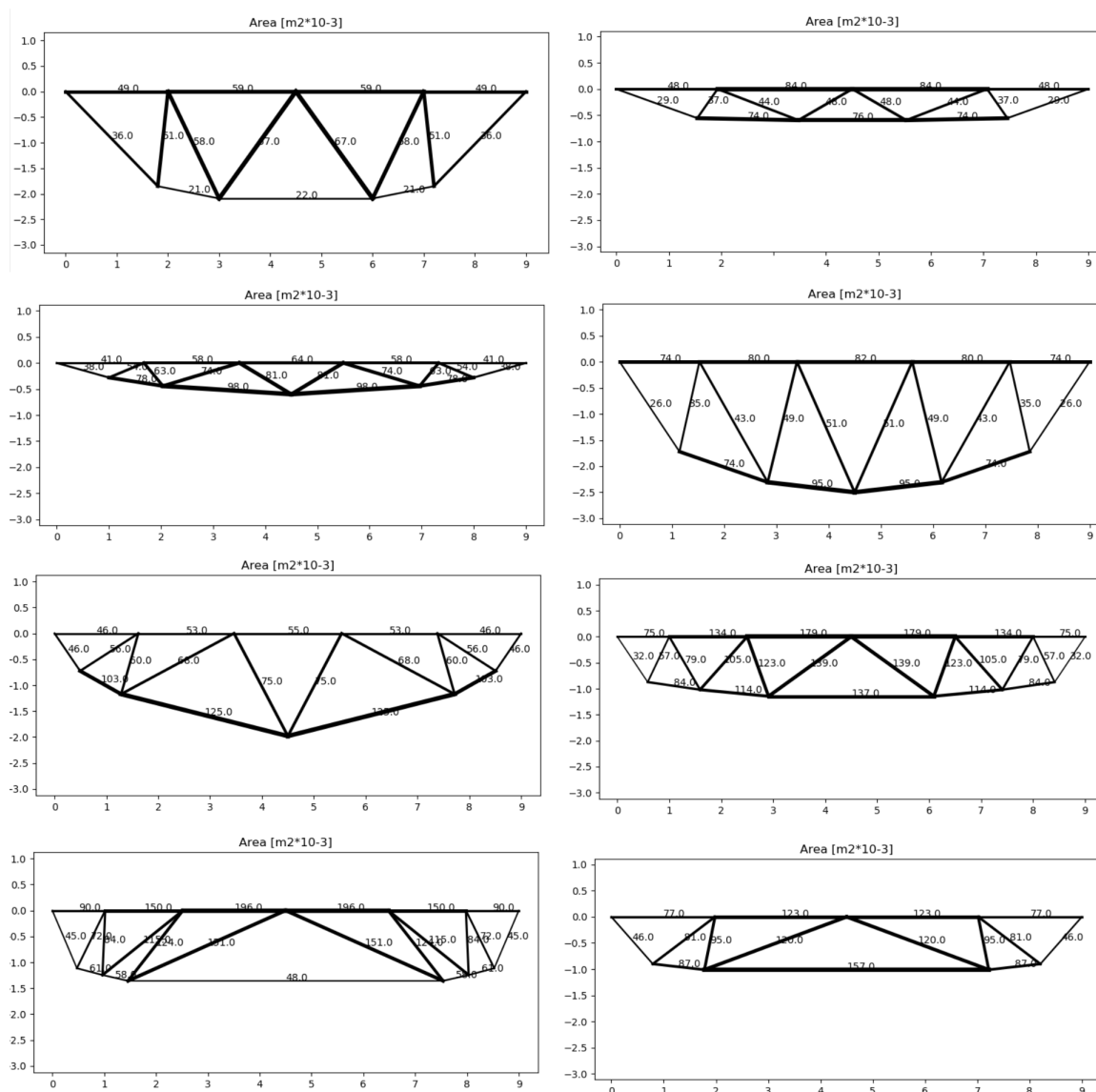
Při vypracování bakalářské práce jsem dospěl k závěru, že by bylo vhodné zahrnout předpětí konstrukce v rámci budoucí práce. Došlo by tak k výrazné redukci tahu. Předpětí bych pravděpodobně aplikoval vložením externích sil.

Bibliografie

- [1] D. V. Hutton, *Fundamentals of finite element analysis*. McGraw-hill, 2004.
- [2] T. G. Zielinski, “Fundamentals of Acoustics”, *Introductory Course on Multiphysics Modelling.[en linea] Last updated December*, roč. 14, 2013.
- [3] D. L. Logan, *A first course in the finite element method*. Cengage Learning, 2011.
- [4] J. Fish a T. Belytschko, *A first course in finite elements*. Wiley, 2007.
- [5] A. Pospíšilová, “Analýza tradičních příkladu rozmerové optimalizace”, *Bakalářská práce, České vysoké učení technické v Praze*, 2010.
- [6] S. Elaoud, T. Loukil a J. Teghem, “The Pareto fitness genetic algorithm: Test function study”, *European Journal of Operational Research*, roč. 177, č. 3, s. 1703–1719, 2007.
- [7] P. Pošík, “Paralelní genetické algoritmy”, *Diplomová práce. české vysoké učení technické, Praha*, 2001.
- [8] S. Sumathi, T Hamsapriya a P Surekha, *Evolutionary intelligence: an introduction to theory and applications with Matlab*. Springer Science & Business Media, 2008.
- [9] W. M. Spears, “Crossover or mutation?”, in *Foundations of genetic algorithms*, sv. 2, Elsevier, 1993, s. 221–237.
- [10] M. Safe, J. Carballido, I. Ponzoni a N. Brignole, “On stopping criteria for genetic algorithms”, in *Brazilian Symposium on Artificial Intelligence*, Springer, 2004, s. 405–413.
- [11] A.-R. Hedar, B. T. Ong a M. Fukushima, “Genetic algorithms with automatic accelerated termination”, *Department of Applied Mathematics and Physics, Kyoto University, Tech. Rep*, roč. 2, 2007.
- [12] E. ČSN, “1-2: Zatížení konstrukcí”, *Obecná zatížení year=1991*,
- [13] G. Van Rossum a F. L. Drake Jr, *Python tutorial*. Centrum voor Wiskunde en Informatica Amsterdam, 1995, sv. 620.

Příloha A

Příloha 1



Obrázek A.1: Příklady konstrukcí z první generace výpočtu