



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV AUTOMOBILNÍHO A DOPRAVNÍHO INŽENÝRTVÍ

DEPARTMENT OF HANDLING AND BUILDING MACHINES

**Optimalizační algoritmus pro příhradové ocelové
konstrukce**

Optimization algorithm for the truss steel structures

TEZE DIZERTAČNÍ PRÁCE

AUTOR PRÁCE

AUTHOR

Ing. Lukáš Zeizinger

ŠKOLITEL

SUPERVISOR

doc. Ing. Jiří Malášek, Ph.D.

BRNO 2020

OBSAH

1	Současný stav řešené problematiky	3
2	Cíl práce	6
3	Zvolené metody zpracování.....	7
3.1.1	Fitness funkce.....	8
3.1.2	Selekční metody	8
3.1.3	Genetické operátory	10
3.1.4	Ukončení algoritmu	11
3.2	Variační metoda metody konečných prvků	11
3.3	Komplexní popis optimalizační metodiky.....	14
3.4	Obecný popis metodiky	14
3.5	Softwarová realizace metodiky.....	15
3.6	Testování algoritmu experimentu	20
3.7	Interpret optimalizačního algoritmu	21
3.7.1	Formulace optimalizačního problému prvního experimentu	21
3.7.2	Formulace optimalizačního problému druhého experimentu.....	23
4	Hlavní výsledky práce.....	24
5	Závěr	26
6	Literatura.....	28
7	Autorovo cv.....	31

1 SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY

Vědeckých publikací týkajících se optimalizace příhradové konstrukce existuje velké množství. Většina prací má souvislost se zkoumanou problematikou dizertační práce, ale v posledních pěti letech došlo k nárůstu prací, které využívají problematiku optimalizace příhradových konstrukcí za účelem otestování vývoje optimalizačních algoritmů. Z tohoto důvodu je bezesporu důležité vnímat informace obsažené ve vědeckých článcích v souvislosti se zkoumanou problematikou.

Jeden z hlavních řešitelů, který ucelenou a systematickou formou popsal optimalizaci příhradových konstrukcí je Martin Bendsøe. V roce 1994 ve svém článku *Optimization methods for truss geometry and topology design* [1] popsal, že správně zvolený přístup k problému optimalizace příhradové konstrukce je klíčovým aspektem k získání požadovaného výsledku. Dále ve svých knihách [6],[7] uvádí, že pokud je potřeba řešit čistě matematický problém, není nutné diskretizovat příčné průřezy, ale naopak je zadat jako spojitou funkci. Z matematického hlediska existují elegantní optimalizační postupy, které využívají vlastností těchto spojitých funkcí. Při této variantě hledání optimálního příčného průřezu je velmi vysoká pravděpodobnost, že příčný průřez prutu nebude standardizovaný. Při praktickém využití získaných výsledků je tudíž nutné upravit optimální návrh tak, aby příčné průřezy prutů odpovídaly výrobním požadavkům. To však přináší odklon od optima a s tím související problémy. Je-li však potřeba uvažovat diskrétní množinu příčných průřezů (např. standardizované příčné profily), je nutné zvolit adekvátní přístup již od počátku analýzy daného problému [4]. Pro optimalizační úlohu, ve které je využíváno diskrétní sady hodnot, je dominantně využito heuristických algoritmů, jak je v této kapitole dále přiblíženo.

Z podstaty dané problematiky vyplývá, že hlavním aspektem při optimalizaci příhradových konstrukcí je zvolení cílové funkce. V převážné většině článků je cílová funkce formulovaná pomocí objemu konstrukce nebo přímo její hmotností. Mnoho autorů (Tezduyar Tayfun, Prabhat Hajela, Niels Olhoff, Aharon Ben-Tal, Julián Norato a další) se soustředí pouze na minimalizaci hmotnosti [4],[6], avšak jejich základním předpokladem jsou obecné velikosti příčných průřezů. Tento krok však logicky vede k řešení, která jsou velice komplikovaná na výrobní nároky (přesnost) nebo využívají nestandardizované příčné průřezy jako třeba v [18].

Na poznatky publikované Martinem Bendsøe navázal Charles Camp, který pro řešení optimalizačního problému minimalizace hmotnosti příhradové konstrukce využil heuristické metody prohledávání stavového prostoru. Camp již v roce 1998 publikoval [9] návrhovou optimalizaci 2D příhradové konstrukce pomocí jednoduchého genetického algoritmu, do kterého zakomponoval modul označující FEAP (*Finite Element Analysis Program*), jenž dokázal analyzovat a kvantifikovat jednotlivé návrhy. Celý algoritmus pojmenovaný FEAPGEN umožňoval základní optimalizovaný návrh 2D příhradové konstrukce. Jednalo se o originální zpracování problému, v němž byly příčné průřezy diskretizovány a zakódovány do genetického kódu jedinců, který byl následně analyzován modulem postaveným na základě metody MKP.

John Bland v roce 2001 navázal na Campem popsaný optimalizační algoritmus a upravil jej, aby pro optimalizaci bylo možné využít metodu mravenčích kolonií [8], jenž byla o deset let dříve poprvé obecně popsána. Pro svou optimalizaci využil stejnou příhradovou konstrukci jako Camp. Jednalo se o příhradovou konstrukci složenou z 25 prutů, která je také použita v této práci a její specifikace lze nalézt v zadání prvního experimentu.

Dále se v roce 2004 Kang Lee nechal inspirovat hudbou a publikoval článek, v němž popisuje základní atributy a chování HS algoritmu (*harmony search algorithm*) [28]. Pro svou aplikaci si taktéž zvolil stejnou příhradovou konstrukci jako Camp a Bland. To umožnilo vzájemné porovnávání jednotlivých vlastností algoritmů, což bylo dalšími autory na poli heuristických algoritmů vnímáno jako vznik tzv. benchmarkové konstrukce. Tato konstrukce dále slouží jako etalon pro kvantifikaci a ověření vlastností heuristických algoritmů.

V následujících letech byla publikovaná celá řada článků, která upravovala již existující heuristické algoritmy a tento trend se drží až do dnešní doby. Pro ověření nových algoritmů, je tato optimalizační úloha velice vhodná, protože se jedná o velice dobře zdokumentovaný soubor zanalyzovaných experimentů. V dnešní době jsou na této konstrukci testovány např. algoritmy opylení květín (*flower pollination algorithm*) [2], nebo vylepšený optimalizační algoritmus kuřecího roje (*Improved Chicken Swarm Optimization Algorithm*) [29], další je algoritmus zrychleného růstu stromů (*Gradient Tree Boosting algorithm*) [40] anebo optimalizace pomocí herního chaosu (*chaos game optimization*) [39]. Všechny tyto algoritmy mají dva společné znaky, prvním je již výše popsaná konstrukce a druhým je účelová funkce, která je definovaná pouze hmotností (objemem).

V kontrastu s těmito publikacemi je velice málo publikací, které se věnují optimalizaci příhradové konstrukce s jinou účelovou funkcí, např. cenou. Martin Bendsøe se této problematice v roce 1995 věnoval a popsal ji v [7]. Podle Martina Bendsøeo je to problém samotné definice úlohy, jak dále popsal v [6]. Hmotnost, nebo je-li uvažován v rámci celé konstrukce jednotný materiál, pak lze uvažovat objem konstrukce, je jednoduše definovaná hodnota funkce, kterou je možné kvantifikovat pomocí plochy průřezů v rámci jednotlivých prutů konstrukce. Na druhou stranu cena již není triviální hodnota účelové funkce, protože je závislá na ceně jednotlivých profilů, u nichž nemusí platit pravidlo “méně materiálu = nižší cena.” Z tohoto důvodu tato oblast optimalizace není tak důkladně prozkoumaná jako v případě hmotnostní optimalizace. V rámci svých prací tento problém potvrdili i další autoři (Sy Wang v [42] a Joel Farkas [17]) a navíc výsledky takové optimalizace nemusí být porovnatelné, protože každý autor ve své práci používá lokálně dostupné profily s místními cenami. Z toho vyplývá, že tato problematika je z globálního pohledu hůře uchopitelná, a proto se jí většina autorů naprosto vyhýbá.

Pro optimalizaci velkých příhradových konstrukcí je často vyžadováno četných analýz pro posouzení proveditelnosti odvozených řešení během procesu optimalizace, které dosahují velké časové náročnosti [10]. V roce 2018 Hy Cao odvodil, že meta-heuristický potažmo i heuristický algoritmus by měl obsahovat jistou formu elitářství při řešení velkých příhradových konstrukcí (od sto prutů) z důvodu rychlejší konvergence výsledků. Dále pak odvodil, že nejrobustnější variantou heuristického algoritmu pro řešení těchto optimalizačních úloh je genetický algoritmus, protože při různých nastaveních nevykazoval ustrnutí v lokálním extrému.

Chceme-li zohlednit dohromady hmotnost a cenu konstrukce, bude nezbytné formulovat cílovou funkci pomocí několika účelových cílů. Pro získání optimálního návrhu konstrukce při zavedení několika účelových cílů je tento problém nutné řešit jako multikriteriální optimalizaci popsanou v [6]. Řešení multikriteriální optimalizace je aktuálním tématem, které si vyžádalo již velkou řadu vědeckých publikací. Předními autory zabývajícími se touto problematikou v mechanice jsou Martin Bendsøe, Niels Olhoff a John Taylor [5], na jejichž základu je vytvořena vícerozměrná optimalizace obsažená v této práci. Hlavním rozdělovacím znakem multikriteriální optimalizace je dominance hledaného řešení, jak je popsáno v [22]. Jedná-li se o dominantní řešení, tak v drtivé většině mluvíme o váhovém kombinování

vícekriteriálního rozhodování, do kterého spadá například všeobecně známá metoda váhových koeficientů. Tato metoda bývá vesměs používána v optimalizaci logistických řetězců nebo při řešení, u kterých jsou předem odhadnutelné dopady jednotlivých koeficientů [36],[37]. Pro univerzální použití je však vhodnější hledat nedominantní řešení, respektive sadu nedominantních řešení.

Problematikou optimalizace více účelových funkcí se zabýval již v roce 1985 Jürgen Guddat, který formuloval vícerozměrnou optimalizaci pomocí parametrické optimalizace, která byla schopna vytvořit sadu dominantních řešení s předem definovanými parametry, jak popsal v [22]. Pokud by bylo možné určit všechna přípustná dominantní řešení daných parametrů, jednalo by se o Pareto hranici, se kterou se dále uvažovalo v [30]. V roce 2008 Jürgen Branke ve své knize popsal, jakým způsobem je možné zapracovat nedominantní optimalizaci do evolučních algoritmů. Na tyto poznatky navázal Keller Andre ve své knize [27], ve které shrnuje aktuální poznatky v problematice vícerozměrné optimalizace. Dále jsou tyto poznatky zpracované například do vícerozměrné optimalizace průhybu a hmotnosti publikované Seyadalem Mirjalilim v [31] nebo Ali Kavehem v [26].

Tato práce se zabývá návrhem metodiky pro optimalizovaný návrh příhradové konstrukce. Pro návrh příčných průřezů prutů je využito normalizovaných řad běžně používaných profilů na území České republiky. Pro tuto optimalizační úlohu, ve které je využíváno diskrétní sady hodnot, je použit genetický algoritmus. Zároveň se jedná o optimalizační problém, jehož metoda je pro řešení dané problematiky velmi aktuálním tématem, jak lze vidět v [3].

Práce se zaměřila na tři hlavní směry optimalizace, a to na optimalizaci hmotnosti, optimalizaci ceny a vícerozměrnou optimalizaci hmotnosti a ceny zároveň. Vstupním předpokladem bude využívání pouze normalizovaných ocelových tyčí různého průřezu a taktéž bude zohledněn minimální odběr daných ocelových tyčí. Předpoklad pro minimální odběr ocelových tyčí bude znamenat započítání ceny jak použitého, tak i zbylého materiálu do celkové ceny konstrukce.

Tato práce propojuje oborové znalosti ze strojírenství, z matematické optimalizace a programování. Přínosem bude hmotnostní, cenová a vícerozměrná optimalizace příhradové konstrukce s použitím normalizovaných profilů jako vstupními daty pro optimalizační algoritmus. Do nynější doby nebyla vypracována podrobná vědecká práce, která by popisovala metodiku optimalizovaného návrhu příhradové konstrukce s použitím normalizovaných profilů a jejich minimální délkový odběr standardizovaných ocelových tyčí. Dále nebyla zatím prokázána změna vlastností rychlosti konvergence genetického algoritmu při použití různě uspořádaných databází vybíraných profilů.

Jedná-li se při optimalizaci o minimalizaci jednorozměrné účelové funkce, lze si to velice dobře představit. Znamená to hledání řešení, které bude mít např. minimální hmotnost nebo cenu. V tomto případě není nutné brát zřetel na jiné aspekty a optimální řešení bude takové, které bude mít minimální hodnotu účelové funkce. V případě, že do této optimalizace vložíme předpoklad minimálního délkového odběru, znamená to, že při optimalizaci se hledá minimální hodnota účelové funkce, ale zároveň je minimalizováno množství použitých profilů. Lze si to představit tak, že při minimalizaci ceny je hledáno takové řešení, které bude mít minimální cenu. Zároveň to však bude takové řešení, které bude mít redukovaný počet profilů, protože budou penalizovány řešení, která by v předchozím případě dosáhla nižší hodnoty účelové funkce.

2 CÍL PRÁCE

Hlavním cílem této práce je vyvinout algoritmus na optimalizaci příhradových ocelových konstrukcí s využitím evolučních algoritmů, který zahrnuje normalizované profily.

Dílčí cíle lze shrnout do následujících bodů:

1. Vybrat a popsat vhodné dílčí metody pro návrh optimálních příhradových konstrukcí transportních strojů.
2. Tvorba řídicího algoritmu.
3. Vybrat vhodný příklad příhradové ocelové konstrukce pro testování navrženého algoritmu.
4. Samostatná funkce softwaru bez použití komerčního softwaru, která využívá maticovou algoritmizaci metody konečných prvků, aby bylo možné spočítat síly v prutech.
5. Simulačně porovnat vlastnosti matematického modelu příhradové ocelové konstrukce s testovací konstrukcí, která je vyřešena v komerčním konečno-prvkovém softwaru.
6. Porovnat vlastnosti databáze evolučního algoritmu s přihlédnutím na rychlost konvergence.

Klíčovou částí v oblasti optimalizace návrhu příhradových ocelových konstrukcí pomocí evolučního algoritmu je zejména vytvoření algoritmu, který bude schopen generovat optimalizovaný návrh příhradové ocelové konstrukce transportních strojů.

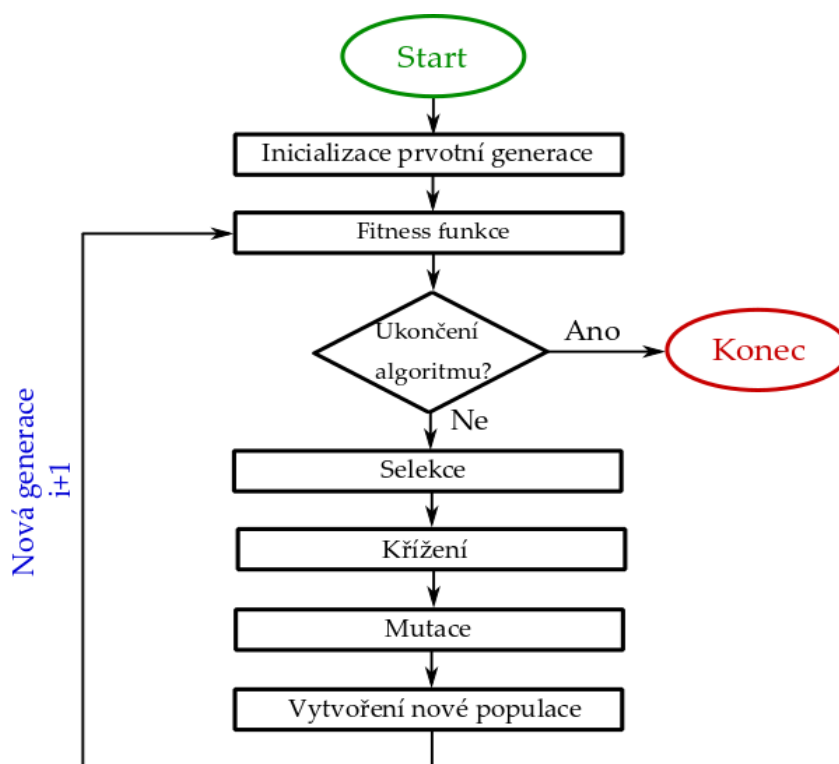
3 ZVOLENÉ METODY ZPRACOVÁNÍ

Genetické algoritmy zažívají v současnosti velkou expanzi do všech vědních oborů [3]. Využívají se pro řešení velkých komplexních úkolů nebo pro zlepšení chování neuronových sítí. Genetický algoritmus přináší již ze své podstaty nemalou míru paralelizace, která je v souladu se současným trendem vývoje moderních počítačových hardwarů [24].

Obecně lze genetické algoritmy jen velice špatně definovat jedním způsobem, protože každý algoritmus využívá jiné funkce a postupy, proto je svým způsobem každý originál. Většinou mají společné alespoň některé znaky a ty jsou dále v této kapitole rozepsané.

Populace, která je pro chod genetického algoritmu zásadní, je složená z velkého množství jedinců. Jedinec populace je zástupným znakem souboru parametrů, které odpovídají jedné možnosti vstupního předpisu. Tyto parametry jsou nazývány geny a charakterizují jedince v populaci. Velikost nebo vlastnost těchto parametrů může nabývat různých hodnot nebo znaků. Chromozom je uspořádaný soubor všech genů utvářejících jedince.

Stejně jako přirozený výběr v přírodě, tak i genetický algoritmus se snaží kombinovat jedince, kteří mají nejlepší pravděpodobnost se na konci procesu stát nejlepšími. Evoluce v genetickém algoritmu spočívá v iterativním zlepšování populace pomocí genetických operátorů a selekčních operátorů, viz Obr. 1 [21].



Obr. 1: Vývojový diagram obecného genetického algoritmu.

Pro návrhový algoritmus pro příhradové konstrukce bude nejlepší vytvořit pro každou proměnnou jeden gen. Zakódování parametru do genů není jednoduché, protože volbou typu a počtem genů můžeme výrazně ovlivnit výkon algoritmu a strukturu prohledávané množiny, taktéž i pořadí genů v chromozomu může posléze ovlivnit rychlost konvergence [21].

3.1.1 FITNESS FUNKCE

Fitness funkce je hodnoticí metoda, která má za úkol posoudit jedince v dané generaci a navrhnout pořadí jedinců. Správně navržená fitness funkce je klíčovým rozhodovacím aparátem genetických algoritmů. Bude-li fitness funkce navržená chybně, může dojít k uvíznutí v lokálním extrému. Ve většině případů se používají dva přístupy, a to ohodnocení pomocí jednoho kritéria anebo pomocí vícerozměrného kritéria [24].

Jednorozměrná klasifikace

Nejjednodušší je jednorozměrný hodnoticí přístup, protože ho lze velice jednoduše zpracovat a interpretovat. Pokud známe idealizovanou (i nereálnou) hodnotu zkoumaného problému, ke které můžeme vztahovat hodnoticí kritérium, tak jej lze kvantifikovat pomocí intervalu $\langle 0; 1 \rangle$, kde v okrajové hodnotě by se měl nacházet globální extrém. Dalším hodnoticím přístupem je neznámá hodnota fitness funkce, která podle cílové funkce může nabývat libovolné hodnoty z oboru reálných čísel a podle zaměření úlohy hledáme jedince s nejmenší nebo největší hodnotou. Jedná-li se o optimalizování hmotnosti konstrukce, lze nalézt případné řešení v následujících člancích [14], [15].

Vícerozměrná klasifikace

Fitness funkce může poskytovat více než jednu hodnotu a v tomto případě se bude jednat o vícerozměrný hodnoticí přístup. Většinou jsou tyto hodnoty poskytovány ve vektorovém zápisu. Hlavním předpokladem vícerozměrného hodnoticího přístupu je prohledání prostoru a nalezení minima funkce $f(x)$, kde $x \in M$, přičemž $f: M \rightarrow R^m$ je vektorová funkce a tedy $f(x)$ je vektor se složkami $(f_1(x), \dots, f_m(x))$, kde M je libovolná množina a R je množina reálných čísel. Nalezení optima je v tomto případě nesmírně náročná operace, protože nelze jednoduše vektory porovnat, a z tohoto důvodu se zavádí pojem eficientní řešení [5].

Jedná-li se o bod $x \in M$ a pro bod $y \in M$ platí, že $f_i(x) > f_i(y)$ pro alespoň jedno $i \in \{1, \dots, m\}$ potom existuje $j \in \{1, \dots, m\}$, takové, že $f_j(x) > f_j(y)$. Bod x je tudíž eficientním řešením a nelze jej v jednom kritériu zlepšit, aniž by došlo ke zhoršení jiného kritéria. Taktéž je toto řešení známo jako Pareto nebo nedominantní řešení [3].

Další možností, jak dosáhnout výsledku vícerozměrné optimalizace, je využití váhových koeficientů pro hodnocení jednotlivých parametrů. Toto řešení se hledá v následujícím tvaru (1):

$$\min_{x \in M} \sum_{i=1}^m \lambda_i f_i(x), \quad (1)$$

pro λ_i platí, že $\lambda_i \geq 0$, kde λ_i jsou váhové koeficienty [1].

3.1.2 SELEKČNÍ METODY

Aby docházelo ke zlepšení populace v průběhu generací, je zapotřebí mít dobře nastavenou fitness funkci, ale taktéž mít dobře zvolenou selekční metodu. Selekční metody se dělí do dvou hlavních skupin podle selekčního tlaku.

- Slabý selekční tlak (*too-weak selection* [24]) umožňuje slabým jedincům v populaci, aby se účastnili dalšího křížení, potažmo mohli přímo přejít do další generace.
- Silný selekční tlak (*too-hard selection* [24]) je přesným opakem slabého selekčního tlaku. Umožňuje pouze silným jedincům v populaci, aby se účastnili dalšího křížení, potažmo mohli přímo přejít do další generace. Slabí jedinci v populaci nemají téměř žádnou šanci účastnit se na produkci další generace.

Pokud použijeme metodu se slabým selekčním tlakem, bude genetický algoritmus prohledávat více i méně perspektivní oblasti. Genetický algoritmus nebude tolik citlivý na přítomnost lokálního minima, na druhou stranu bude rychlost konvergence silně závislá na velikosti populace v generaci. Použijeme-li metodu se silným selekčním tlakem, genetický algoritmus získává na konvergenční rychlosti, ale může uvíznout v lokálním extrému. V některých případech je nejlepším řešením měnit v průběhu výpočtu selekční metody [19].

Selekce úměrná kvalitě jedince

Jedná se o základní selekční metodu, jejíž základ položil John Holland [24]. Je založená na předpokladu bonifikace kvality jedince, takže kvalitnější jedinci by měli produkovat více potomků než jedinci s nízkou kvalitou. V závislosti na kvalitě je v rámci populace rozložena pravděpodobnost výběru pro reprodukci, ta probíhá tolikrát, kolik je jedinců v populaci.

Sigma škálování

Tato metoda mapuje aktuální kvalitu jedinců a vybírá jedince s přibližně podobnou kvalitativní hodnotou, tím se zabrání předčasné konvergenci v lokálním extrému [24]. V pozdější fázi jsou si jedinci velice kvalitativně podobní a pro dobré rozhodování je nutné fitness funkci normovat, aby se zvětšilo pravděpodobnostní rozložení reprodukce.

Elitářství

Elitářství je metoda, která vezme vždy určité procento nejlepších jedinců a tito jedinci jsou automaticky přesunuti do další generace. Tato metoda zajistí, že kvalita populace nebude nikdy klesat [24].

Selekce pomocí pořadí

Tato metoda seřadí jedince v populaci podle kvality. Pravděpodobnost reprodukce se jedincům v populaci přidělí podle předem známého klíče. Tato metoda má předem známý tlak a její výhodou je, že udržuje předem známé rozložení populace [21].

Turnajová selekce

Základní verze této metody porovná náhodně dva jedince v populaci a podle kvality rozhodne, který jedinec je vhodnější. Tento jedinec je poté vybrán jako rodič pro jedince v další populaci a cyklus se opakuje, než je naplněna celá populace.

Velice zajímavou variantou turnajové selekce je tzv. síň slávy. Tato metoda postupuje tak, že porovná ohodnoceného jedince s jedincem, který byl nejlepší v předchozí generaci a následně jsou ohodnoceni [22].

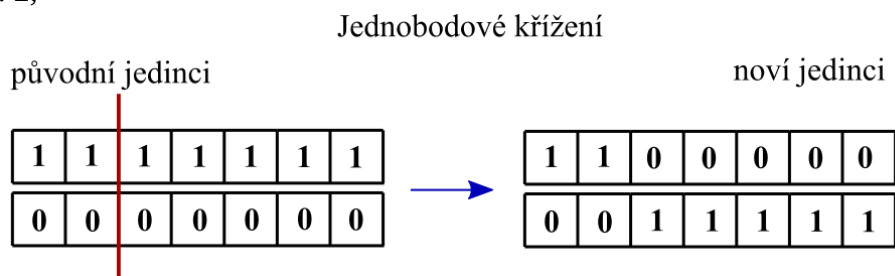
3.1.3 GENETICKÉ OPERÁTORY

Jedná se o reprodukční operátor, který kombinuje geny při tvorbě další generace populace. Hlavním úkolem je křížení a mutace. Může sloužit i k dalším specifickým účelům [24].

Křížení

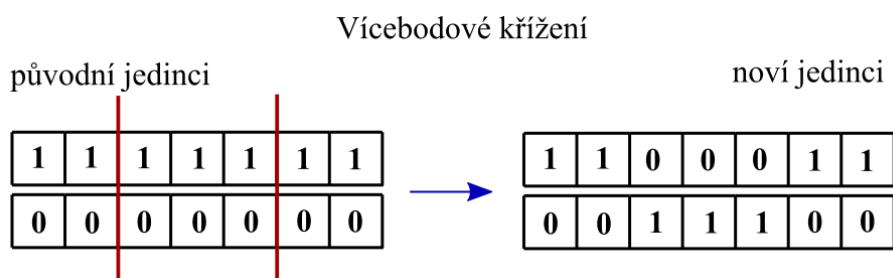
Je to reprodukční operace, při níž dva jedinci poskytnou část svého chromozomu pro tvorbu nových jedinců. Existuje velká řada různých forem křížení a každá aplikace genetického algoritmu vyžaduje jistá specifika. Existují tři základní typy křížení, a to [1]:

- jednobodové křížení – v jednom uzlu dojde ke zkřížení chromozomu, jak je možné vidět na Obr. 2,



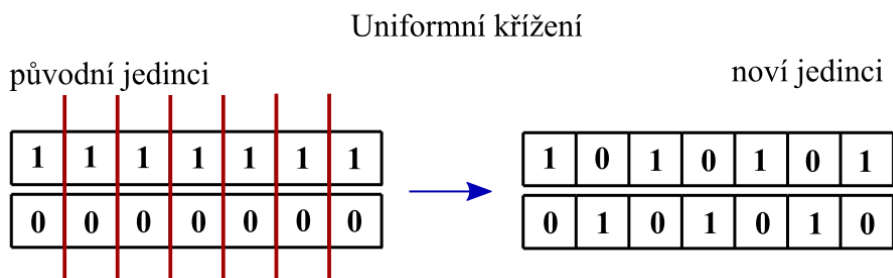
Obr. 2: Schéma jednobodového křížení.

- vícebodové křížení – dojde ke zkřížení chromozomu ve více bodech, jak lze vidět na Obr. 3,



Obr. 3: Schéma vícebodového křížení.

- uniformní křížení – jedná se o speciální případ vícebodového křížení, když dojde k výměně u všech sudých nebo lichých genů, jak je možné vidět na Obr. 4.



Obr. 4: Schéma uniformního křížení.

Mutace

Mutace je operátor, který slouží k výměně informace v genomu. Tento zásah do původního chromozomu jedince může zabránit usednutí populace v lokálním extrému. Pravděpodobnost mutace je velice malá, pohybuje se v řádech setin až tisícín procenta [1], záleží však na velikosti prohledávaného prostoru.

3.1.4 UKONČENÍ ALGORITMU

Tento okamžik jde velice těžko určit, protože je jen velmi malá pravděpodobnost, že genetický algoritmus nalezne přesný bod globálního extrému. Z toho důvodu je nutné mít připravené varianty, pokud by tak nenastalo. Níže je popsáno pět základních důvodů:

- Stanoví se minimální hodnota kvality, kterou musí alespoň jeden jedinec populace dosáhnout.
- Je předem určený čas, po který algoritmus využívá výpočetního výkonu přístroje.
- Je předem určený počet generací, které má algoritmus vyřešit.
- Je předem určený časový interval, za který již nedojde ke zlepšení alespoň jednoho jedince.
- Je předem určený počet generací, za který již nedojde ke zlepšení alespoň jednoho jedince.

Speciálním, ale velice závažným případem pro ukončení algoritmu je degenerace populace. Degenerace v populaci nastává, pokud více jak předem stanovené procento populace má totožnou informaci v genomu. Důvodem, proč nastává degenerace populace je, že všichni členové populace jsou centralizováni do lokálního extrému s neschopností efektivně prohledávat své okolí. Ve většině případů je populace brána za zdegenerovanou, pokud všichni její členové mají shodu ve více než 90 % genů. Existují čtyři hlavní způsoby ochrany populace před degenerováním [24]:

- Navýšení počtu členů v populaci. Čím je populace větší, tím je menší pravděpodobnost degenerace.
- Použití slabé selekční metody. Umožníme slabším jedincům, aby se účastnili reprodukce, a tudíž v populaci uchovali jiný genový materiál.
- Genetická infuze. Jedná se o dočasné zvýšení pravděpodobnosti pro mutaci, často je spjatá s hypermutací. Další používanou možností je udržování subpopulací, kdy je umožněno jedincům, aby se křížili pouze v rámci dané subpopulace. Po určité době pak dojde k promíchání subpopulací.
- Negativní diskriminace podobných jedinců. Jedná se o úpravu fitness funkce, kdy pro jedince s velice podobným chromozomem je snížena pravděpodobnost reprodukce, což přinese bonifikaci jedincům, kteří přinášejí do populace nové geny.

Degenerace populace však nemusí znamenat selhání výpočtu, může se jednat o skutečnost, že populace leží v globálním extrému.

3.2 VARIČNÍ METODA METODY KONEČNÝCH PRVKŮ

Pro řešení příhradových konstrukcí je všeobecně uznávaným standartním řešením variační metoda deformační varianty metody konečných prvků (MKP), jenž využívá Lagrangeův variační princip. Jak uvádí Jindřich Petruška v „*Počítačových metodách mechaniky II: Metoda konečných prvků*“, je tento princip formulován tak, že:

„Mezi všemi funkcemi posuvů, které zachovávají spojitost tělesa a splňují geometrické okrajové podmínky, se realizují ty, které udílejí celkové potenciální energii Π stacionární hodnotu.“ [34]

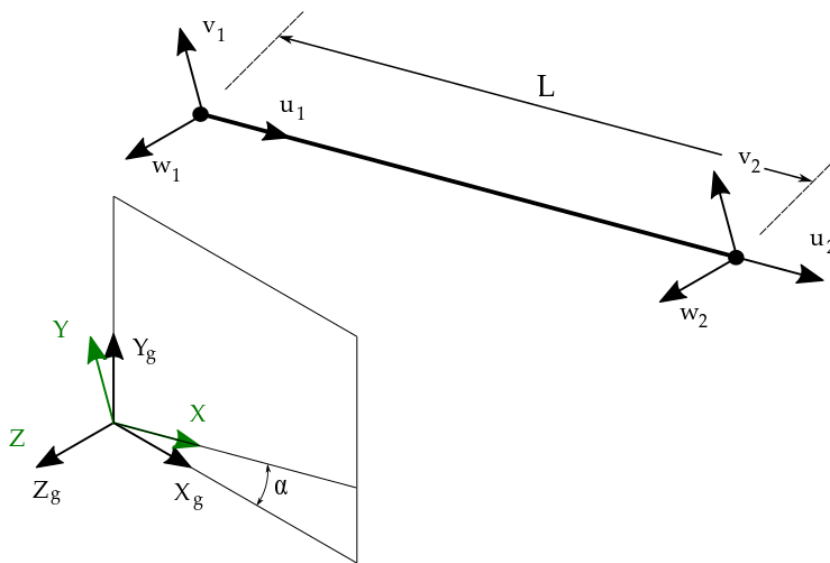
Stacionární hodnota Π je jednoznačná a lze si jí interpretovat jako rovnici (2)

$$\Pi = W - P, \quad (2)$$

kde W je energie napjatosti popisovaného tělesa a P je potenciál vnějšího zatížení. Zároveň představuje minimum Π [33].

Osově zatížené pruty

Odvozená prutová teorie pro prutovou soustavu má základ v lineární pružnosti [35]. Pro popis soustav je nutné sestavit matice tuhosti jednotlivých prutů, které je potřeba následně vyřešit [33]. Prostorový prut lze popsat, jak je vidět na Obr. 5, pomocí délky L a pomocí orientace lokálního souřadného systému vůči globálnímu souřadnému systému. Jedná-li se konkrétně o prut z Obr. 5, tak tento prut má střednici v rovině xy a s osou x globálního souřadného systému svírá úhel α [32].



Obr. 5 Osově namáhání prostorového prutu s pootočeným souřadným systémem o úhel α v rovině xy [32]

Deformační parametry prostorového prutu jsou popsány pomocí u_1 , u_2 , v_1 , v_2 , w_1 , a w_2 , aby byl umožněn obecný posuv uzlových bodů. Matice tuhosti k je popsána následovně

$$k = \frac{ES}{L} \cdot \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (3)$$

kde E je Youngův modul, S je plocha průřezu prutu a L je délka prutu [34]. Deformační parametry jsou vyjádřeny pro hlavní souřadný systém a je tedy nutné provést vzájemné pootočení pomocí známých transformujících vztahů

$$\delta = T \cdot \delta_g, \quad (4)$$

kde δ je vektor deformačních parametrů v pootočeném souřadném systému, δ_g je vektor deformačních parametrů pro globální souřadný systém a T je transformační matice [35]. Vektor deformačních parametrů v pootočeném souřadném systému je podle Obr. 5 dále rozepsaný na

$$\delta = \begin{bmatrix} u_1 \\ v_1 \\ w_1 \\ u_2 \\ v_2 \\ w_2 \end{bmatrix}, \quad (5)$$

pro vektor deformačních parametrů globálního souřadného systému platí analogicky totéž co pro matici (5)

$$\delta_g = \begin{bmatrix} u_{1g} \\ v_{1g} \\ w_{1g} \\ u_{2g} \\ v_{2g} \\ w_{2g} \end{bmatrix}. \quad (6)$$

Transformační matice má tvar

$$T = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}, \quad (7)$$

kde λ je substituce úhlové funkce [32].

Transformační matice v explicitním tvaru lze vyjádřit jako

$$\lambda = \begin{bmatrix} \cos(xx_g) & \cos(xy_g) & \cos(xz_g) \\ \cos(yx_g) & \cos(yy_g) & \cos(yz_g) \\ \cos(zx_g) & \cos(zy_g) & \cos(zz_g) \end{bmatrix}, \quad (8)$$

kde $\cos(xx_g)$ je cosinus úhlu sevřeného příslušnými osami [34]. Z důvodu ortogonality transformační matice T , je možné zapsat rovnici

$$f = T \cdot f_g \quad \Leftrightarrow \quad f_g = T^T \cdot f, \quad (9)$$

kde f je prvková matice vnějšího zatížení podle lokálního souřadného systému, T je transformační matice a f_g je prvková matice vnějšího zatížení orientovaná podle globálního souřadného systému [33].

Prvkovou matici tuhosti k_g lze v globálním souřadném systému zapsat pomocí následující rovnice

$$k_g \cdot \delta_g = f_g, \quad (10)$$

pro které je δ_g vektorem deformačních parametrů v globálním souřadném systému a f_g je prvková matice vnějšího zatížení orientovaná podle globálního souřadného systému.

Při úplném dosazení do rovnice (10) je získán vztah

$$k_g \cdot \delta_g = f_g = T^T \cdot f = T^T \cdot k \cdot \delta = T^T \cdot k \cdot T \cdot \delta_g. \quad (11)$$

Všechny prutové prvky jsou takto formulované za předpokladu lineární pružnosti, a je proto nutné splnit předpoklad namáhání prutů pouze osovými silami [33].

3.3 KOMPLEXNÍ POPIS OPTIMALIZAČNÍ METODIKY

Metodika je zaměřená na optimalizaci příhradové nosné konstrukce transportních strojů. V rámci této metodiky je využíváno variační formulace metody konečných prvků a genetického algoritmu pro nalezení optimální konstrukce podle definovaných kritérií. Pro komplexní popis metodiky je důležité v první řadě stanovit obecný postup a na tomto základě vystavět softwarovou realizaci metodiky. Dále bude přiblíženo testování a verifikace softwarové realizace algoritmu, následně pak bude provedena analýza získaných výsledků.

3.4 OBECNÝ POPIS METODIKY

Obecný popis metodiky je myšlenkovou mapou, která reflektuje blokové uspořádání algoritmu. Na Obr. 6 je možné vidět obecné schéma metodiky, která je v následující kapitole komplexně rozpracována.

Formulace optimalizační úlohy

V rámci formulace optimalizační úlohy bude vytvořený matematický model příhradové konstrukce, taktéž bude formulována doména a omezující podmínky. Pro formulaci úlohy bude klíčové vycházet z Eurokódu 3 a popřípadě z oborových norem (pokud to daná aplikace bude vyžadovat). Klíčovým aspektem je parametrizace matematického modelu.

Testování experimentu genetického algoritmu

Integrací strategie do optimalizačního schématu je dosaženo automatického řízení experimentu. Je využíváno genetického algoritmu pro generování hodnot pro parametrický matematický model, který je následně zkompletován pro strukturální analýzu MKP.

Strukturální analýza MKP

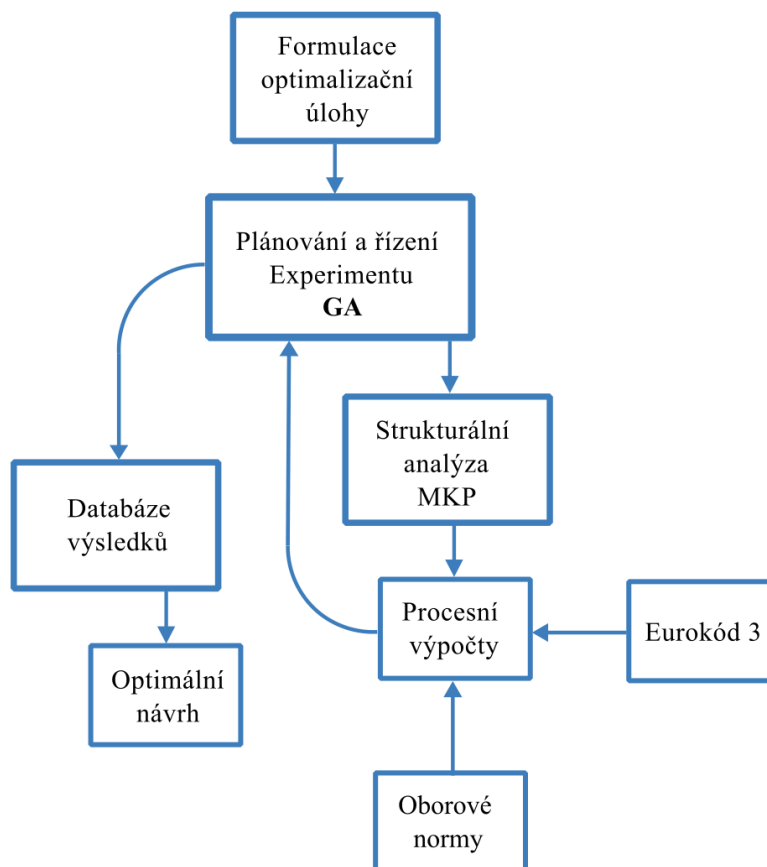
MKP umožňuje řešení strukturálních problémů na obecných tělesech. Pro metodiku je využito pouze zjednodušené formy, pro řešení pouze příhradových konstrukcí. Maticová forma konečno-prvkového modelu umožňuje měnit parametry pro každý výpočet.

Procesní výpočty

Tyto výpočty jsou procesně spjaté s využitím genetického algoritmu a nepatří do MKP. V tomto místě je kvantifikována hodnota účelové funkce a požadavky spjaté s Eurokódem 3, popřípadě s oborovými normami.

Databáze výsledků

Do databáze výsledků jsou zahrnuty všechny výsledky, jež algoritmus vyhodnotí jako relevantní a které jsou řešením vyhovujícím oborovým normám a Eurokódu 3. Taktéž je zde uložena informace identifikující zadání v rámci hodnot parametrů a identifikátory genetického algoritmu.

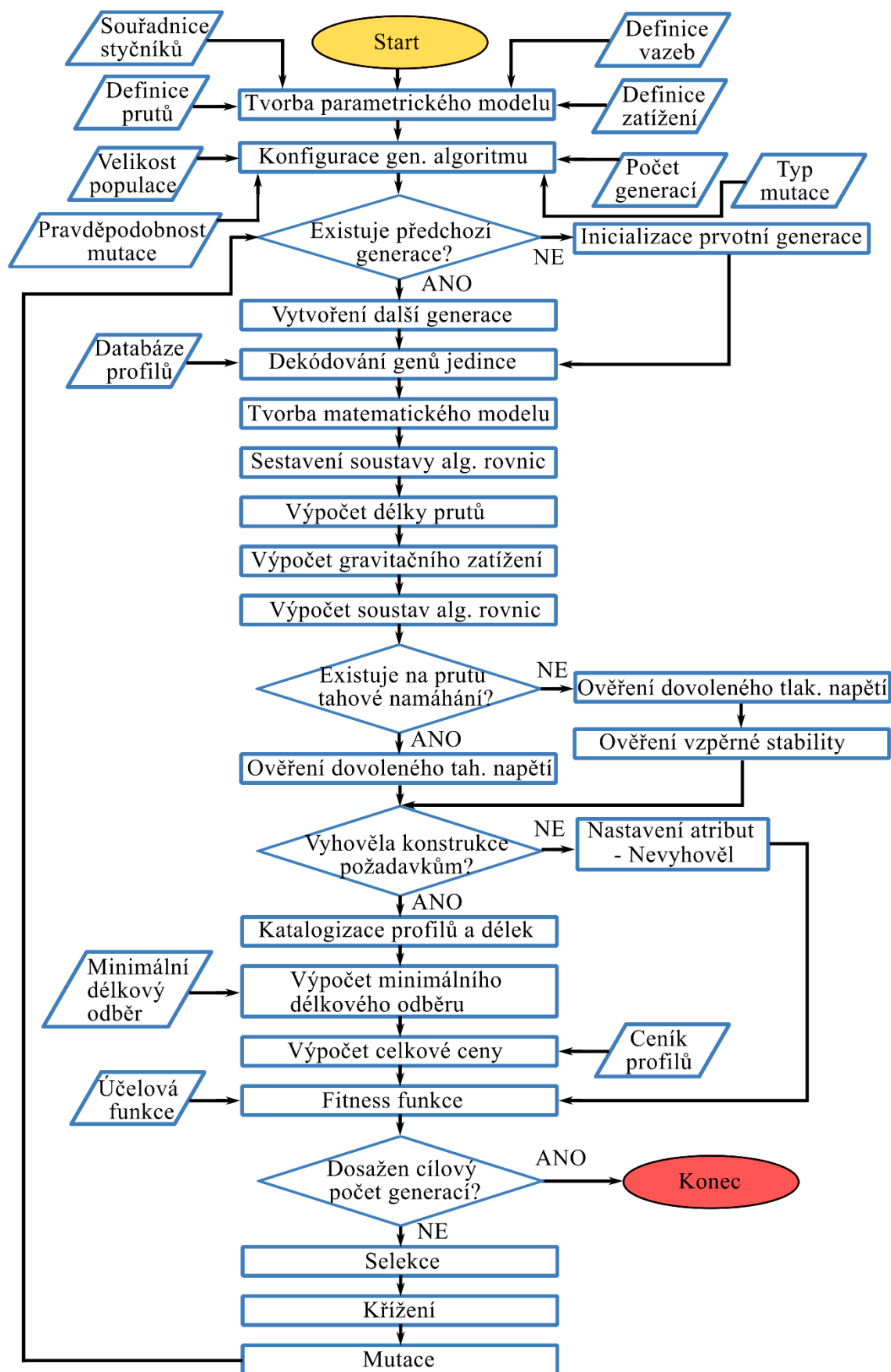


Obr. 6: Obecné schéma metodiky.

3.5 SOFTWAREVÁ REALIZACE METODIKY

Na Obr. 7 je možné vidět vývojový diagram logiky algoritmu, který koresponduje s obecnou metodikou popsanou v předchozí kapitole. Jako první krok po startu algoritmu začne tvorba parametrického modelu, jehož základem jsou souřadnice styčníků v kartézském souřadném systému. Dále jsou definovány pruty, které vždy mají počátek nebo konec v definovaných styčnících. Následně jsou ve vybraných styčnících definované vazby. Pro kompletizaci parametrického modelu už zbývá pouze definovat zatížení, které je zadáno ve formě sil do styčníků.

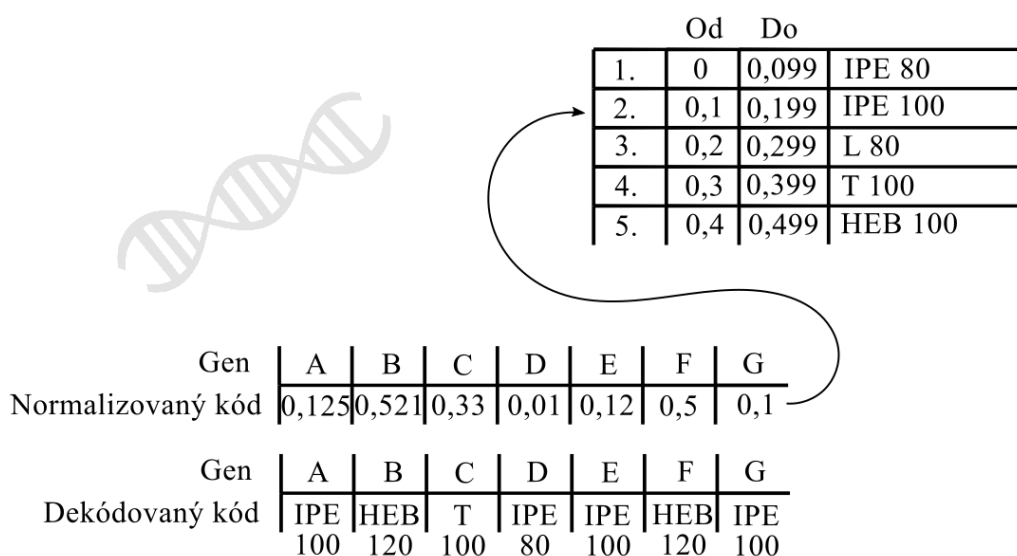
Dalším krokem je konfigurace genetického algoritmu, nejedná se přímo o tvorbu v pravém slova smyslu, ale spíše o nastavení. V první řadě je potřeba nadefinovat velikost populace, která udává počet jedinců pro každou populaci. Následně je pak nutné zadat počet generací, které se mají iterovat, než dojde k ukončení algoritmu. Pro dobrou funkci genetického algoritmu je důležité zvolit vhodnou hodnotu pravděpodobnosti mutace, aby nedošlo k uváznutí v lokálním extrému. Dalším aspektem, který je v této práci zohledněn je tzv. typ mutace. Jedná se buď o náhodnou mutaci nebo o výběr pomocí pravděpodobnostního Gaussova rozdělení. Než však bude popsána mutace, je důležité přiblížit databázi profilů.



Obr. 7: Vývojový diagram softwarové logiky algoritmu.

Databáze profilů obsahuje katalogové informace o tvaru profilu, velikosti a normě, podle které je vyroben. Dále obsahuje materiálovou charakteristiku, kvadratický moment průřezu k osám, modul průřezu v ohybu, plochu průřezu, poloměr setrvačnosti k osám ohybu a součinitel imperfekce α pro obě osy. Jeden z aspektů práce bylo strukturování této databáze profilů a sledování dopadu na rychlost konvergence algoritmu.

První krok algoritmu je inicializace prvotní generace o dané velikosti populace. Každý jedinec je tvořen souborem genů, kde každý gen je generován pomocí pseudonáhodného rozdělení v intervalu $(0, 1)$. Následně dojde k dekódování genu, což znamená zpětný proces normalizace. Pro tento algoritmus bylo použito kódování genů pomocí normalizované hodnoty informace, což znamená, že informace o pozici výběru byly převedeny na interval $(0, 1)$. Z toho vyplývá, že každá položka v databázi má informaci o pozici zakódovanou v intervalu rovném $1 / \text{celkový počet položek v databázi}$, jak lze vidět na Obr. 8.

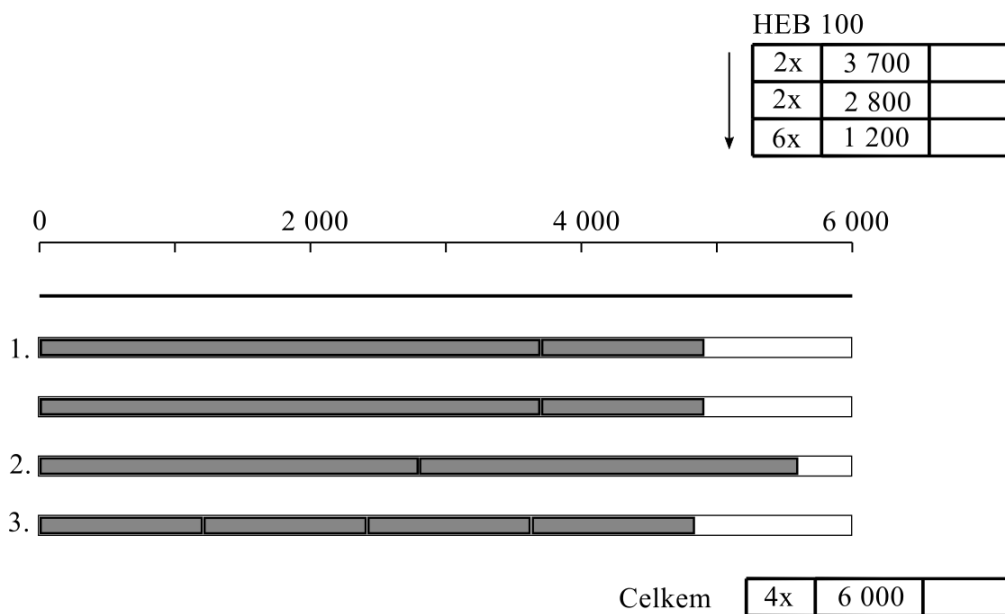


Obr. 8: Grafické znázornění dekódování normalizovaného kódu.

Jakmile jsou zkompletovány informace získané z genů, je možné přiřadit konkrétní profil pro daný prut konstrukce. To umožní vytvořit matematický model, který je vstupem pro konečno-prvkový výpočet. Jako první jsou sestaveny algebraické rovnice a jsou uspořádány do matic, dále pomocí transformačních matic dojde k převedení matic z lokálního souřadného systému do globálního souřadného systému. Následně je vypočítán objem, aby byla dopočítána hmotnost. V návaznosti na tuto informaci se následně na každý prut aplikuje gravitační síla ekvivalentní této hmotnosti. V tuto chvíli má algoritmus připravené všechny algebraické rovnice a pomocí knihovny Numpy naleznou jejich řešení.

Všechny pruty, které jsou v konstrukci namáhané na tlak, jsou podrobeny ověření na vzpěrnou stabilitu s imperfekcí. Jestliže na celé konstrukci existuje alespoň jeden prut, který nevyhověl po stránce vzpěrné stability nebo došlo k překročení dovoleného napětí, celá konstrukce je označena atributem – Nevyhověl a přeskakuje další kroky. Pokud však vyhovuje všem podmínkám, dojde ke katalogizaci profilů a jejich délek do tabulky, kde se vytvoří celkový odběr daného profilu, podle kritéria minimálního délkového odběru tak, aby nebylo nutné pruty svařovat z více dílů, jak je možné vidět na Obr. 9.

Pro každý profil je vytvořena cena podle již zjištěného počtu tyčí, což v sumarizaci pro každý profil dá celkovou cenu konstrukce i s materiálem, který je potřeba nakoupit, ale neslouží čistě jen pro strukturu konstrukce, ale i jako zbytkový materiál po nákupu.

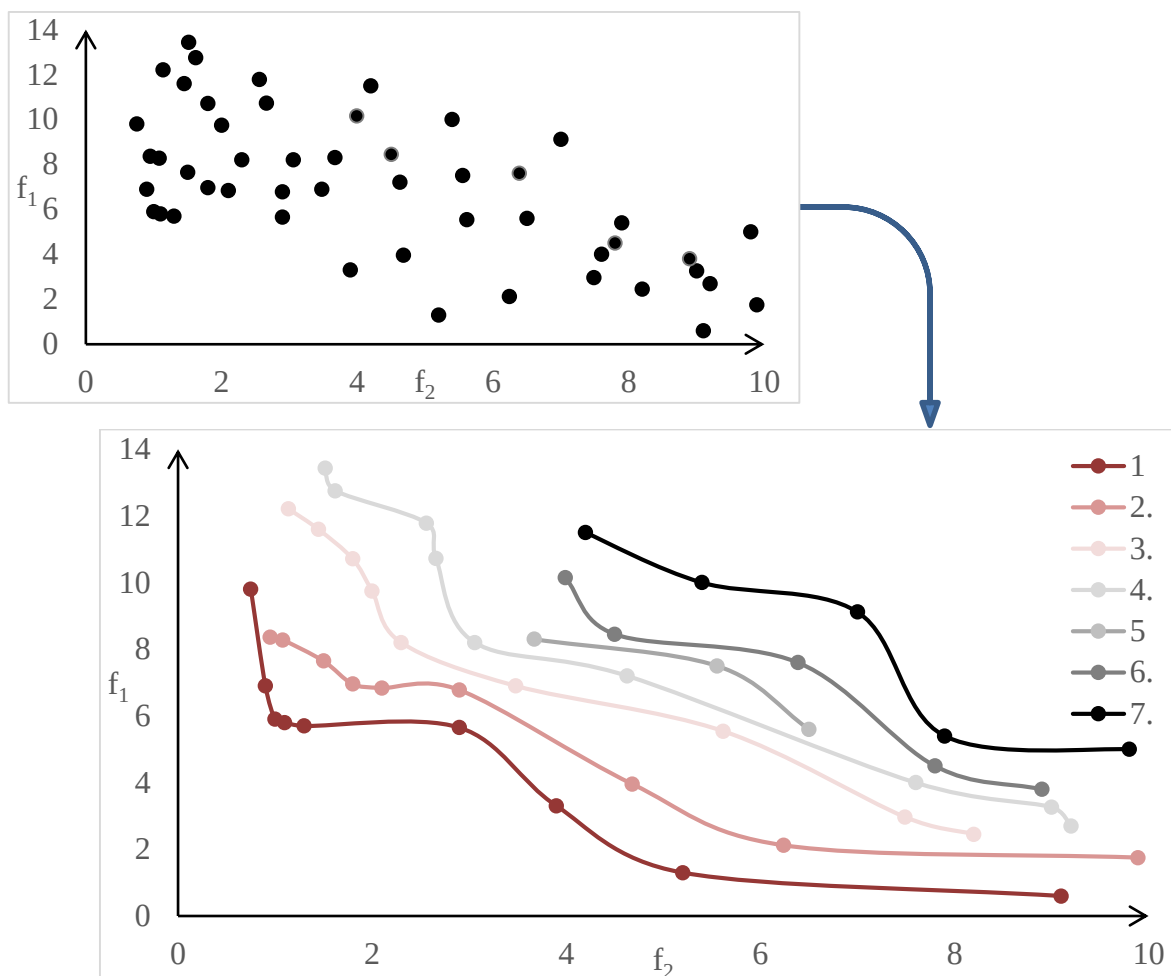


Obr. 9: Grafické znázornění využívání minimální odběrové délky tyčí. Jako první jsou vybrány nejdelší požadované délky, následují je další v pořadí a ty se snaží vyplnit zbylý prostor, pokud není možno doplnit, tak se začíná vyplňovat nový prut.

Daná konfigurace konstrukce si s sebou nese informaci o její hmotnosti a ceně, a proto může být ohodnocena pomocí fitness funkce. V případě jednorozměrné účelové funkce je ohodnocení jednoduché a provede se seřazením podle daného kritéria. Pokud se však jedná o vícerozměrnou účelovou funkci, je celá generace zaznamenána do grafu, ve kterém se určí, které varianty patří do Pareto hranice. Tyto jedinci jsou vyřazeni z grafu a dojde k dalšímu kolu hledání Pareto hranice. Tímto způsobem se poskládají všechny varianty a získá se hodnota fitness funkce podle toho, ve které iteraci Pareto hranice se konstrukce nachází, jak můžete vidět na Obr. 10. Tímto se docílí nedominantního hodnocení v rámci dané generace. Pokud bylo docíleno požadovaného množství iterací genetického algoritmu, dojde k přerušení a vyhodnocení poslední generace.

Pokud se však nejedná o poslední iteraci, takto získané hodnoty fitness funkcí jsou vstupní informací pro selekci, která pomocí turnajové selekce porovná náhodně dva jedince v populaci a podle kvality rozhodne, který jedinec je vhodnější. Tento jedinec je poté vybrán jako rodič pro jedince v další populaci. V návaznosti na takto vybraný soubor jedinců, jakožto rodičů, je provedeno křížení nové generace jedinců [25].

Jedná se o reprodukční operaci, při níž dva jedinci poskytnou část svého chromozomu pro tvorbu nových jedinců. V tomto případě bude použito uniformní křížení, což znamená, že bude využito vícebodového křížení, u kterého se vyměňují všechny sudé nebo liché geny rodičů a vzniká dvojice unikátních nových jedinců.



Obr. 10: Grafické znázornění zpracování dat a roztřídění do Pareto hranic.

Na závěr iteračního procesu genetického algoritmu je každý jedinec v každém genu podroben mutaci s předem danou pravděpodobností. Při mutaci dojde k výběru profilu z databáze a nahrazení jednoho profilu novým. To znamená, že při náhodném výběru seřazení profilů nehraje žádnou roli, proto se výběr řídí pouze jedním pravidlem, a to je identifikace výběru pomocí náhodného čísla. Při použití jiného než náhodného výběru však pořadí jednotlivých profilů v databázi hraje významnou roli. Jako kontrast pro náhodný výběr byl pro mutaci zvolen výběr pomocí pravděpodobnostního Gaussova rozdělení [6], kde Gaussovo rozdělení $N(0, 1)$ má hustotu pravděpodobnosti ve tvaru (12)

$$f(x) = \frac{1}{\sqrt{2 \cdot \pi}} \cdot e^{-\frac{x^2}{2}}. \quad (12)$$

Genetický algoritmus je navržen tak, aby na začátku svého chodu pokryl co možná nejširší pásmo variant, proto je důležité, aby v inicializační fázi algoritmus vytvořil populaci s velkou diverzitou. Na Obr. 11 je možné vidět variabilitu jednotlivých genů jedinců, která je v rámci postupu generací snižována. Mutace obsažená v genetickém algoritmu může způsobit vylepšení, jak je možné vidět u prvního jedince na 8. genu ve 100. generaci.

1. generace

	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10
1.	19	8	46	32	55	33	37	11	16	10
2.	7	1	17	15	48	33	3	55	26	56
3.	43	17	16	44	18	18	46	43	33	12
4.	56	6	44	42	29	24	54	50	6	34
5.	8	14	56	56	35	7	4	8	26	33

100. generace

	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10
1.	43	15	32	32	55	32	32	25	32	32
2.	43	15	32	32	55	32	32	55	32	32
3.	43	15	32	32	55	32	32	55	32	32
4.	43	15	32	32	56	32	32	55	32	32
5.	43	15	32	32	55	32	32	55	32	32

Obr. 11: Porovnání informační variability mezi první a stou generací genetického algoritmu.

Z tohoto důvodu je potřeba zaznamenávat i průměrnou hodnotu účelové funkce, která reflektuje rozmanitost řešení v dané populaci vůči nejlepšímu řešení. To znamená, že pokud se průměrná hodnota účelové funkce v dané generaci bude blížit minimální hodnotě účelové funkce, tak v rámci populace existuje pouze velice málo řešení, která jsou rozdílná.

3.6 TESTOVÁNÍ ALGORITMU EXPERIMENTU

Testování bude provedeno na příhradových konstrukcích, které budou zatíženy silami a definovány vazbami. V rámci těchto experimentů se bude sledovat vývoj hodnoty účelové funkce na parametrech genetického algoritmu, a to v podobě velikosti populace a uspořádanosti databáze profilů.

Experimenty, které budou mít jednorozměrnou účelovou funkci, budou analyzovány v každé páté generaci a bude se ukládat hodnota účelové funkce. Taktéž se bude zaznamenávat průměrná hodnota přípustných řešení pro danou generaci. Pro analýzu a validaci výsledků budou ukládány i hodnoty vedlejších funkcí (jedná se o účelovou funkci, která není aktuálně optimalizovaná, tj. při optimalizaci hmotnosti je to cena). Jakmile se bude provádět experiment s vícerozměrnou účelovou funkcí, bude se analyzovat každá desátá generace z důvodu přehlednosti grafů.

Genetické algoritmy se vyhodnocují pomocí rychlosti konvergence a potřebného času na daný výpočet. Protože je genetický algoritmus postaven na pseudonáhodném generování čísel a heuristickém prohledávání, tak k řešení úlohy není potřeba najít globální extrém, ale přijatelné řešení za rozumný čas, jak již bylo publikováno Antonínem Šťastným v [38].

Strukturální výpočty provedené optimalizačním algoritmem budou verifikovány pro ověření věrohodnosti v komerčním MKP softwaru. Bude se jednat o software MSC Apex a verifikace proběhne na náhodně zvolené konstrukci z daného experimentu.

3.7 INTERPRET OPTIMALIZAČNÍHO ALGORITMU

V průběhu rešerše jsem narazil na fakt, že většina autorů, kteří se zabývají optimalizací nosných konstrukcí a používají ať již genetického algoritmu anebo jinou stochastickou metodu, popisují stejný problém a tím je časová náročnost výpočtu. Tato časová náročnost zajisté koresponduje s výkonnostními možnostmi výpočtových stanic, zároveň však i na vlastnostech interpretačního jazyka, ve kterém je algoritmus napsaný.

V rámci vědecké aplikace je často používáno programovacího jazyka MATLAB (interpretovaný v programu MATLAB), avšak v aktuálních velkých vědeckých projektech se mnohem častěji používá programovací jazyk Python [11]. Je otázkou, který z daných jazyků je rychlejší pro výpočty, ale celkově vždy záleží na zkušenosti programátora, možnostech optimalizace kódu a algoritmech v knihovnách [43].

Python je open-source dynamický interpretovaný vysokoúrovňový skriptovací programovací jazyk a v současné době je často používán pro tvorbu nejenom vědeckých prací, ale je taktéž využíván v široké škále průmyslových aplikací. Svou popularitu si Python získal díky své přehlednosti a srozumitelnosti kódu, velkému počtu jak vlastních, tak také externích knihoven a rozsáhlé komunitě.

Python umožňuje efektivně využívat kód psaný ve svém jazyce, ale taktéž je ho možné použít i jako tzv. wrapper. Tedy pro volání výpočetně náročné části aplikace nebo knihoven, která může být napsána např. v jazyce C++, popř. lze použít Cython, který přeloží v Pythonu napsaný kód do jazyka C a poté jej standardní překladač přeloží do binárního kódu.

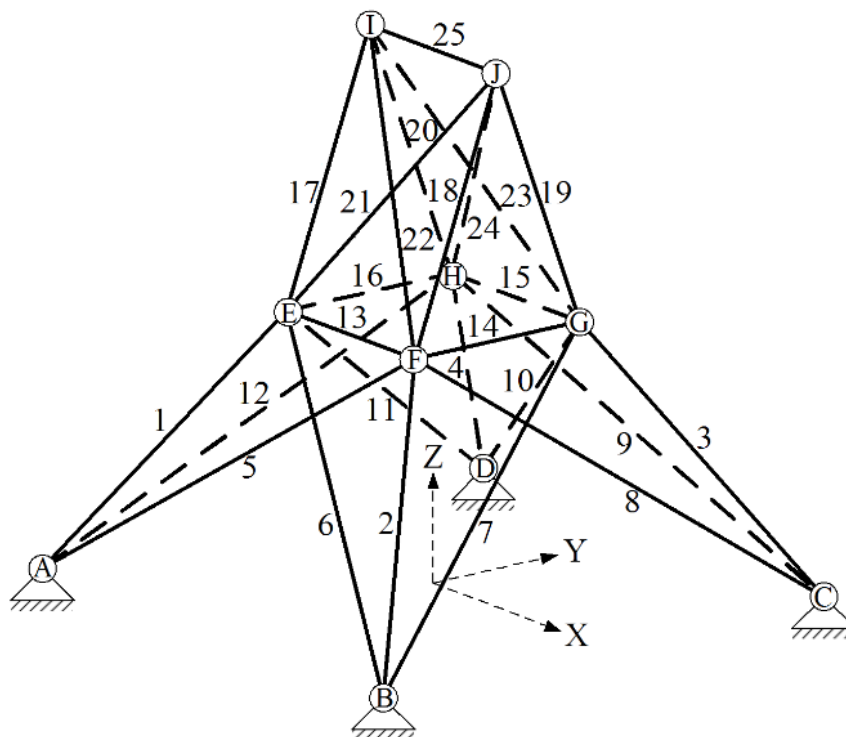
Velkou výhodou Pythonu shledávám v jednoduché možnosti paralelizace výpočtů v rámci jednoho systému, ale i paralelizace na více počítačů/serverů a z tohoto důvodu byl také zvolen pro softwarovou realizaci optimalizačního algoritmu.

3.7.1 FORMULACE OPTIMALIZAČNÍHO PROBLÉMU PRVNÍHO EXPERIMENTU

Pro první aplikaci metodiky byla vybraná konstrukce, která bývá v literatuře označována jako tzv. benchmarková. Jedná se o příhradovou konstrukci složenou z 25 prutů, jak lze vidět na Obr. 12. Styčníky A, B, C a D jsou vetknuté a v těchto styčnicích je omezený translační posun i rotace. Pruty, které se kříží, ale v tomto místě není definovaný styčník, se v tomto místě vzájemně neovlivňují.

Ve styčnicích I a J jsou aplikovány síly užitečného zatížení o velikosti 50 kN v záporném směru osy Z. Dále je aplikována gravitace ve formě síly působící na každý nosník ekvivalentní jeho hmotnosti orientovaný v záporném směru osy Z. Styčníky spojují jednotlivé pruty, jejich souřadnice a profily použité pro návrh této příhradové konstrukce jsou uvedeny v dizertační práci. Pro výpočet vzpěrné stability jednotlivých profilů je využito Eurokódu 3 [13].

Pro správný chod genetického algoritmu je nutné vhodně zvolit velikost populace. Pro tento případ byly zvoleny 3 velikosti populace, a to populace s 250, 2 500 a 25 000 jedinci. Důležitým aspektem je volba nástroje pro tvorbu inicializační populace (nástroj pro vytvoření jedinců), jenž je generována v Pythonu pomocí funkce Random (jedná se o pseudonáhodný výběr).



Obr. 12: Návrh příhradové konstrukce a styčníků v globálním souřadném systému.

Další operátor, jenž bylo nutné zvolit, je křížení a jak již bylo zmíněno v předchozí kapitole, je v tomto experimentu zvoleno uniformní křížení (sudá – lichá). Dalším operátorem je mutace, v první řadě byla nastavena pravděpodobnost mutace, a to na hodnotu 0,78125 % (což je $100 / 128$) zvoleno podle [23].

Pro Gaussovo rozdělení byla databáze profilu seřazena podle:

- 1) plochy průřezu (A),
- 2) ceny za metr délky profilu (M),

rozptyl od původního vybraného profilu je zvolen $\sigma^2 = 5$.

Pro tento experiment byla provedena tato sada výpočtů, viz Tab. 1.

Tab. 1: Tabulka přehledu výpočtů pro všechny řady populace, pro danou cílovou funkci a pro dané seřazení databáze (A je seřazení podle plochy průřezu, M je seřazení podle ceny za metr délky profilu).

Databáze	Náhodná			A			M		
Hmotnost	250	2 500	25 000	250	2 500	25 000	250	2 500	25 000
Cena	250	2 500	25 000	250	2 500	25 000	250	2 500	25 000
Pareto	250	2 500	25 000	250	2 500	25 000	250	2 500	25 000

3.7.2 FORMULACE OPTIMALIZAČNÍHO PROBLÉMU DRUHÉHO EXPERIMENTU

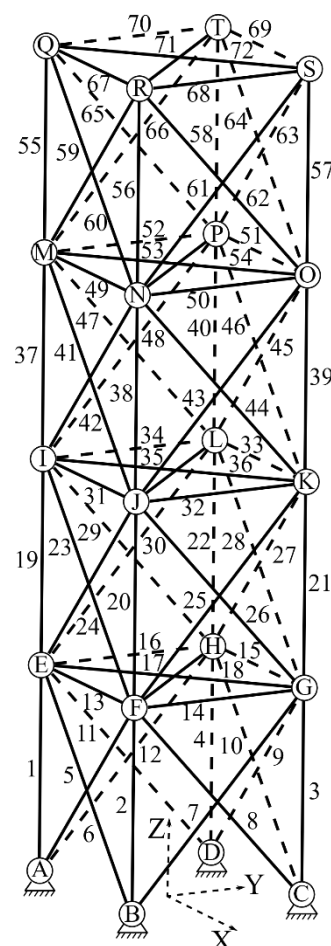
První experiment je v rámci optimalizace příhradových konstrukcí brán jako odrazový můstek, ale v literatuře jsou naznačeny obtíže, které může přinést zvyšování počtu prutů pro výpočet [4],[20],[21],[41]. Tento experiment má ověřit funkčnost výpočtu na složitější konstrukci. Pro tuto aplikaci metodiky byla vybraná konstrukce, která může být základem pro věž například věžového jeřábu. Jedná se o příhradovou konstrukci složenou ze 72 prutů. Styčníky A, B, C a D jsou vetknuté a v těchto styčnicích je omezený translační posun i rotace. Pruty, které se kříží, ale není v tomto místě definovaný styčník, se vzájemně neovlivňují.

Ve styčnicích Q, R, S a T jsou aplikovány síly užitečného zatížení o velikosti 50 kN v záporném směru osy Z. Dále je aplikována gravitace ve formě síly působící na každý nosník ekvivalentní jeho hmotnosti orientovaný v záporném směru osy Z. Styčníky spojují jednotlivé pruty a jejich souřadnice jsou uvedeny v dizertační práci.

Pro tento experiment byl zvolený odlišný soubor velikostí populace, a to populace o velikostech 72, 720 a 7 200 jedinců. Nastavení zbývajících atributů algoritmu zůstalo obdobné jako v předchozím experimentu, jenž je možné nalézt v předchozí podkapitole. V Tab. 2 je uvedený přehled veškerých sad výpočtů řazených podle účelové funkce a následně pak podle seřazení databáze profilů a velikostí populace.

Tab. 2: Tabulka přehledu výpočtů pro všechny řady populace, pro danou cílovou funkci a pro dané seřazení databáze (A je seřazení podle plochy průřezu, M je seřazení podle ceny za metr délky profilu).

Databáze	Náhodná		
Hmotnost	72	720	7 200
Cena	72	720	7 200
Pareto	72	720	7 200
Databáze	A		
Hmotnost	72	720	7 200
Cena	72	720	7 200
Pareto	72	720	7 200
Databáze	M		
Hmotnost	72	720	7 200
Cena	72	720	7 200
Pareto	72	720	7 200



Obr. 13: Návrh druhé příhradové konstrukce a styčniců v globálním souřadném systému.

4 HLAVNÍ VÝSLEDKY PRÁCE

Výsledkem optimalizačního algoritmu je pro danou generaci výsledek, který má minimální hodnotu účelové funkce. V případě, že se jedná o jednorozměrnou účelovou funkci, lze usuzovat, že pro každou generaci existuje výsledek, který lze považovat za dominantní, a tudíž lze u něj předpokládat, že reprezentuje účelovost tohoto algoritmu.

V rámci této práce byla vyslovena teze, existence závislosti mezi optimalizováním ceny a optimalizováním hmotnosti. Proto byla pro každý dominantní výsledek vypočítána Pearsonova korelace. Výsledky jsou brány stejným způsobem, jako tomu bylo v celé práci, a to pro každou pátou generaci. Získané Pearsonovy korelační koeficienty a intervaly spolehlivosti jsou uvedeny v Tab. 3.

Z výsledku vidíme, že na interval spolehlivosti 95 % pro hmotnost a cenu bylo dosaženo 0,86 až 0,98, což podle [16] znamená velmi silnou závislost. Avšak pokud již výsledky zkombinujeme a budeme společně analyzovat výsledky z účelové funkce hmotnosti a ceny dohromady zjistíme, že existuje silná závislost mezi optimalizováním ceny a hmotnosti.

Při výpočtu prvního experimentu bylo dosaženo minimální hmotnosti 521 kg při výpočtu, který trval 9 866 s. Pokud bychom chtěli propočítat všechny možné varianty, tak jednoduchou kombinatorikou bychom přišli na to, že by bylo potřeba spočítat 63^{25} variant, což odpovídá zhruba číslu $9,628 \cdot 10^{44}$. Průměrný čas, který byl potřeba pro výpočet jedné konstrukce, trval algoritmu přibližně 0,062 s, což by znamenalo, že verifikace všech konstrukcí by trvala $5,969 \cdot 10^{43}$ s. Pro porovnání stáří vesmíru je asi $13,799 \cdot 10^9$ let, a to odpovídá asi $4,352 \cdot 10^{17}$ s. Z toho vyplývá, že pokud by algoritmus začal vyhodnocovat jednotlivé konstrukce od počátku vzniku vesmíru na popsané výpočetní stanici, stále by mu tato doba vůbec nestačila. Verifikační výpočet konstrukce počítaný v softwaru MSC Apex trval přibližně 1,43 s, což je zhruba 23 krát pomalejší.

V případě druhého experimentu byla dosažena minimální hmotnost 1 137 kg při výpočtu, který trval 250 650 s. Pokud bychom chtěli propočítat všechny možné varianty, tak by bylo potřeba spočítat 63^{72} variant, což odpovídá zhruba číslu $3,569 \cdot 10^{129}$.

Tab. 3: Shrnutí korelačních koeficientů na intervalu spolehlivosti 95 %.

	Zkratka	Experiment první (25 prutů)			Experiment druhý (72 prutů)		
Sada výsledků pro účelovou funkci		Hmotnost	Cena	Společná	Hmotnost	Cena	Společná
Počet měření	n	188	188	376	188	188	376
Pearsonův korelační koeficient	r	0,96	0,96	0,69	0,98	0,9	0,67
Transformace na náhodnou veličinu	W_t	1,95	1,94	0,85	2,23	1,45	0,81
95 % interval spolehlivosti	α_s	0,05	0,05	0,05	0,05	0,05	0,05
Kvantil standardizovaného normálního rozdělení	$Z_{(1-\alpha/2)}$	1,96	1,96	1,96	1,96	1,96	1,96
Interval spolehlivosti pro střední hodnotu náhodné veličiny – dolní	h^*	2,09	2,08	0,95	2,37	1,59	0,91
Interval spolehlivosti pro střední hodnotu náhodné veličiny – horní	d^*	1,81	1,8	0,74	2,08	1,3	0,71
Interval spolehlivosti dolní	d	0,95	0,95	0,63	0,97	0,86	0,61
Interval spolehlivosti horní	h	0,97	0,97	0,74	0,98	0,92	0,72

Z výše uvedeného tedy vyplývá, že přínosem použití algoritmu je, že není potřeba procházet všechny možné varianty, abychom našli optimální řešení. Postačí udělat výpočet v rozumně krátké době, abychom dostali přijatelně optimalizovanou konstrukci v dosažitelném čase. Další výhodou je automatizace zadávání výpočtu oproti komerčním programům, do kterých je nutné ručně zadávat geometrii konstrukce, nadefinovat profily a průřezové charakteristiky, nebo ručně vymodelovat profily, což značně prodlužuje dobu nutnou pro získání výsledků.

5 ZÁVĚR

Dizertační práce byla zaměřená na algoritmus optimalizující návrh příhradové konstrukce stavebních strojů. V rámci práce byly podrobně rozebrány aspekty spjaté s touto problematikou. V rámci první kapitoly byl vytvořen krátký historický vhled do vývoje výpočetních metod příhradových konstrukcí. Na tuto kapitolu navazoval popis problematiky a analýza vědeckého poznání v rámci tohoto oboru. Následovala analýza dílčích metod, která kladla důraz převážně na rozbor optimalizačních metod a posléze na genetický algoritmus a jeho možné uplatnění pro řešení tohoto problému. Taktéž zde byl zdůvodněn výběr výpočetní metody pro řešení příhradových konstrukcí, na něj navázala podkapitola o vzpěrné stabilitě.

Hlavním cílem této práce bylo vytvoření algoritmu na optimalizaci příhradové ocelové konstrukce s využitím evolučního algoritmu, který zahrnuje normované profily, což je důkladně zdokumentováno v páté kapitole. Na toto navazuje aplikace algoritmu při řešení dvou experimentálních návrhů konstrukcí. Jako první byla zvolena konstrukce, která byla složena z 25 prutů a měla užité zatížení 2 krát 50 kN. V rámci řešení byla vytvořena databáze profilů, která obnášela 63 různých standardizovaných položek. Jako jednorozměrné účelové funkce byly zvoleny hmotnost a cena konstrukce, dále pak byla vybrána vícerozměrná účelová funkce, která zahrnovala jak hmotnost, tak i cenu. Pro experimentální řešení byly zvoleny tři velikosti populace, a to 250, 2 500 a 25 000 jedinců. Experiment byl dále ještě rozšířen o varianty seřazení databáze pro každou populaci, to mělo důsledek na rychlost konvergence.

V rámci minimalizování účelové funkce hmotnosti bylo zjištěno, že nejrychleji konverguje varianta s 25 000 jedinci a databází seřazenou podle plochy profilu, avšak při sledování času výpočtu se nejlépe jevila varianta s 2 500 jedinci a databází seřazenou podle ceny za metr délky prutu. Při minimalizaci účelové funkce ceny byla nejlépe konvergující varianta s 25 000 jedinci a databází seřazenou podle plochy profilu, i když z časového hlediska výpočtu trvala nejdéle ze všech počítaných variant. Po těchto variantách následovalo řešení vícerozměrné účelové funkce, při němž dvě varianty dosáhly téměř totožných výsledků. První byla varianta s velikostí populace 2 500 jedinci a databází seřazenou podle plochy a druhá byla varianta s 25 000 jedinci a taktéž stejně seřazenou databází, avšak téměř s deset krát větším výpočtovým časem. Při verifikaci výsledků byla prokázána shoda matematického modelu algoritmu s MKP řešením.

Pro druhý experimentální návrh konstrukce byla zvolena 72 prutová příhradová konstrukce, pro jejíž řešení byly zvoleny tři velikosti populace, a to 72, 720 a 7 200 jedinců. Při minimalizaci účelové funkce hmotnosti bylo zjištěno, že nejlépe konvergují varianty se 7 200 jedinci a nejlepšího výsledku bylo docíleno se seřazenou databází podle plochy. Pro minimalizaci účelové funkce ceny byl výsledek obdobný jako v předchozím případě, a to tak, že nejlépe konvergovala varianta se 7 200 jedinci a databází seřazenou podle ceny za délku profilu. V případě minimalizace vícerozměrné účelové funkce bylo docíleno nejlepších výsledků při použití největší populace.

Závěry plynoucí z experimentů jsou, že zvolení populace o velkém počtu jedinců přináší lepší výsledky, avšak ekvivalentně tomu narůstá výpočtový čas. Při zvolení vhodné velikosti populace se může docílit kvalitních výsledků za minimálně nutný výpočtový čas. Experimenty byly počítané na výpočtové stanici, jež umožňovala paralelizaci výpočtu na 16 vláken, což s výhledem na rychle se rozvíjející procesorový segment přináší příslib větší výpočtové kapacity, s čímž dojde i ke zkrácení výpočtové doby při použití velké populace.

Pro tuto práci byly záměrně vybrány účelové funkce zaměřené na hmotnost a cenu z důvodu kontrastu, který byl již zmíněn v úvodu. Z toho vyplývala otázka, zda může existovat souvislost mezi optimalizací hmotnosti a ceny. Pomocí již zmíněných dvou experimentů byla vytvořena sada výsledků, ze kterých lze usuzovat, že na intervalu spolehlivosti 95 % je možné říct, že existuje silná závislost mezi optimalizováním ceny a optimalizováním hmotnosti příhradové konstrukce při využití standardizovaných profilů.

Při celkovém shrnutí lze usuzovat, že algoritmus vyvinutý a uvedený v této práci splnil veškeré požadavky, které na něj byly kladeny. Výsledky práce jsou také přímo využitelné v oboru ocelových konstrukcí pozemních staveb. Další výzkum v této oblasti spočívá v možnosti zapracování styčnickových spojů a dalších konstrukčních a technologických aspektů do optimalizačního algoritmu. Další možností výzkumu je pak zefektivnění genetického algoritmu a jeho komponent, zároveň s tím využití pokročilejších možností paralelizace například využití distribuovaných výpočtů na grafických kartách.

6 LITERATURA

- [1] ASSIMI, Hirad, Ali JAMALI a Nader NARIMAN-ZADEH. Multi-objective sizing and topology optimization of truss structures using genetic programming based on a new adaptive mutant operator. *Neural Computing and Applications*. DOI: 10.1007/s00521-018-3401-9. ISSN 0941-0643. Dostupné také z: <http://link.springer.com/10.1007/s00521-018-3401-9>
- [2] BEKDAŞ, Gebrail, Sinan Melih NIGDELI a Xin-She YANG. Sizing optimization of truss structures using flower pollination algorithm. *Applied Soft Computing*. 2015, 37, 322-331. DOI: 10.1016/j.asoc.2015.08.037. ISSN 15684946. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S1568494615005384>
- [3] BEKDAŞ, Gebrail, Sinan Melih NIGDELI, Aylin Ece KAYABEKIR a Xin-She YANG. Optimization in Civil Engineering and Metaheuristic Algorithms: A Review of State-of-the-Art Developments. *Computational Intelligence, Optimization and Inverse Problems with Applications in Engineering*. Cham: Springer International Publishing, 2019, 2019-09-26, 111-137. DOI: 10.1007/978-3-319-96433-1_6. ISBN 978-3-319-96432-4. Dostupné také z: http://link.springer.com/10.1007/978-3-319-96433-1_6
- [4] BENDSØE, Martin P. Optimization methods for truss geometry and topology design. *Structural optimization*. Berlin: Springer-Verlag, 1994, (1), 141-159. DOI: BF01742459. ISSN 1615-1488.
- [5] BENDSØE, Martin P., Niels OLHOFF a John E. TAYLOR. A Variational Formulation for Multicriteria Structural Optimization*. *Journal of Structural Mechanics*. 2007, 11(4), 523-544. DOI: 10.1080/03601218308907456. ISSN 0360-1218. Dostupné také z: <http://www.tandfonline.com/doi/abs/10.1080/03601218308907456>
- [6] BENDSØE, Martin, Ole SIGMUND. *Topology optimization: theory, methods, and applications*. Second edition, corrected printing. Berlin: Springer-Verlag, [2004]. ISBN 978-3-642-07698-5.
- [7] BENDSØE, Martin. *Optimization of Structural Topology, Shape, and Material*. 1995. DOI: 10.1007/978-3-662-03115-5.
- [8] BLAND, J. A. Optimal structural design by ant colony optimization. *Engineering Optimization*. 2001, 33(4), 425-443. DOI: 10.1080/03052150108940927. ISSN 0305-215X. Dostupné také z: <http://www.tandfonline.com/doi/abs/10.1080/03052150108940927>
- [9] CAMP, Charles, Shahram PEZESHK a Guozhong CAO. Optimized Design of Two-Dimensional Structures Using a Genetic Algorithm. *Journal of Structural Engineering*. 1998, 124(5), 551-559. DOI: 10.1061/(ASCE)0733-9445(1998)124:5(551). ISSN 0733-9445. Dostupné také z: [http://ascelibrary.org/doi/10.1061/\(ASCE\)0733-9445\(1998\)124:5\(551\)](http://ascelibrary.org/doi/10.1061/(ASCE)0733-9445(1998)124:5(551))
- [10] CAO, Hongyou, Xudong QIAN, Yunlai ZHOU, Jean-Marc DROUET a Julio Rodríguez-Vigil JUNCO. Large-scale structural optimization using metaheuristic algorithms with elitism and a filter strategy. *Structural and Multidisciplinary Optimization*. 2018, 57(2), 799-814. DOI: 10.1007/s00158-017-1784-3. ISSN 1615-147X. Dostupné také z: <http://link.springer.com/10.1007/s00158-017-1784-3>
- [11] CASTELVECCHI, Davide. Black hole pictured for first time — in spectacular detail. *Nature*. 2019, 568(7752), 284-285. DOI: 10.1038/d41586-019-01155-0. ISSN 0028-0836. Dostupné také z: <http://www.nature.com/articles/d41586-019-01155-0>
- [12] COELLO, C.A. a A.D. CHRISTIANSEN. *Multiobjective optimization of trusses using genetic algorithms*. 2000, 75(6), 647-660. DOI: 10.1016/S0045-7949(99)00110-8. ISSN 00457949. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S0045794999001108>
- [13] ČSN EN 1993-1-1 (731401). *Eurokód 3: Navrhování ocelových konstrukcí – Část 1-1: Obecná pravidla a pravidla pro pozemní stavby*. 1. Praha: Český normalizační institut, 2006.
- [14] DEB, Kalyanmoy a Surendra GULATI. Design of truss-structures for minimum weight using genetic algorithms. *Finite Elements in Analysis and Design*. 2001, 37(5), 447-465. DOI:

- 10.1016/S0168-874X(00)00057-3. ISSN 0168874X. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S0168874X00000573>
- [15] DEDE, Tayfun, Serkan BEKIROĞLU a Yusuf AYVAZ. Weight minimization of trusses with genetic algorithm. *Applied Soft Computing*. 2011, 11(2), 2565-2575. DOI: 10.1016/j.asoc.2010.10.006. ISSN 15684946. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S1568494610002619>
- [16] EVANS, James R. *Business Analytics*. 2nd Edition. Vineland: Pearson, 2016. ISBN 9780321997821.
- [17] FARKAS, J. a K. JÁRMAI. Volume and Cost Minimization of a Tubular Truss with Displacement-Constraint. *Advances in Structural Engineering*. 2012, 15(10), 1729-1737. DOI: 10.1260/1369-4332.15.10.1729. ISSN 1369-4332. Dostupné také z: <http://journals.sagepub.com/doi/10.1260/1369-4332.15.10.1729>
- [18] FARSHCHIN, Mohammad, Mohsen MANIAT, Charles V. CAMP a Shahram PEZESHK. School based optimization algorithm for design of steel frames. *Engineering Structures*. 2018, 171, 326-335. DOI: 10.1016/j.engstruct.2018.05.085. ISSN 01410296. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S0141029617308787>
- [19] FRIEBELOVÁ, Jana. *Vícekriteriální rozhodování za jistoty [online]*. Ekonomická fakulta Jihočeské univerzity v Českých Budějovicích [cit. 2019-09-03]. Kapitola Část I, s. 1. Dostupné online.
- [20] GENTLE, J. Elon. Computational Statistics. *International Encyclopedia of Education*. Elsevier, 2010, 2010, 93-97. DOI: 10.1016/B978-0-08-044894-7.01316-6. ISBN 9780080448947. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/B9780080448947013166>
- [21] GREINER, David. *Computer Methods in Applied Mechanics and Engineering*. 2004, 193(33-35). DOI: 10.1016/j.cma.2004.02.001. ISSN 00457825. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S0045782504001410>
- [22] GUDDAT, Jürgen. *Multiobjective and stochastic optimization based on parametric optimization*. Berlin: Akademie Verlag, 1985. Mathematical Research. ISBN 0138-3019-13399882.
- [23] HAJELA, P. a E. LEE. Genetic algorithms in truss topological optimization. *International Journal of Solids and Structures*. 1995, 32(22), 3341-3357. DOI: 10.1016/0020-7683(94)00306-H. ISSN 00207683. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/002076839400306H>
- [24] HOLLAND, John H. Genetic Algorithms and Adaptation. *Adaptive Control of Ill-Defined Systems*. Boston, MA: Springer US, 1984, 1984, 317-333. DOI: 10.1007/978-1-4684-8941-5_21. ISBN 978-1-4684-8943-9. Dostupné také z: http://link.springer.com/10.1007/978-1-4684-8941-5_21
- [25] HYNEK, Josef. *Genetické algoritmy a genetické programování*. Praha: Grada, 2008. Průvodce (Grada). ISBN 978-80-247-2695-3.
- [26] KAVEH, A. a M. ILCHI GHAZAAN. A new VPS-based algorithm for multi-objective optimization problems. *Engineering with Computers*. 2020, 36(3), 1029-1040. DOI: 10.1007/s00366-019-00747-8. ISSN 0177-0667. Dostupné také z: <http://link.springer.com/10.1007/s00366-019-00747-8>
- [27] KELLER, Andre. *Multi-Objective Optimization in Theory and Practice I: Classical Methods*. Sharjah: Bentham Science Publishers, 2017. ISBN 978-1-68108-569-2.
- [28] LEE, Kang Seok a Zong Woo GEEM. A new structural optimization method based on the harmony search algorithm. 2004, 82(9-10), 781-798. DOI: 10.1016/j.compstruc.2004.01.002. ISSN 00457949. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S004579490400029X>
- [29] LI, Yancang, Shiwen WANG a Muxuan HAN. Truss Structure Optimization Based on Improved Chicken Swarm Optimization Algorithm. *Advances in Civil Engineering*. 2019, 2019, 1-

16. DOI: 10.1155/2019/6902428. ISSN 1687-8086. Dostupné také z: <https://www.hindawi.com/journals/ace/2019/6902428/>
- [30] MIETTINEN, Kaisa. *Nonlinear Multiobjective Optimization*. 1998. Boston: Springer, DOI: 10.1007/978-1-4615-5563-6, ISBN 978-1-4613-7544-9.
- [31] MIRJALILI, Seyedali, Pradeep JANGIR a Shahrzad SAREMI. Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems. *Applied Intelligence*. 2017, 46(1), 79-95. DOI: 10.1007/s10489-016-0825-8. ISSN 0924-669X. Dostupné také z: <http://link.springer.com/10.1007/s10489-016-0825-8>
- [32] MOAVENI, Saeed. *Finite element analysis: theory and application with ANSYS*. Upper Saddle River, N.J.: Prentice Hall, 1999. ISBN 01-378-5098-0.
- [33] PETRUŠKA, Jindřich. MKP v inženýrských výpočtech. *MKP v inženýrských výpočtech*, 2011.
- [34] PETRUŠKA, Jindřich. Počítačové metody mechaniky II, Metoda konečných prvků: výukový text. *VUT v Brně, Fakulta strojního inženýrství*, 2003.
- [35] PETRUŠKA, Jindřich. Počítačové metody mechaniky II. *Základní veličiny a rovnice obecné pružnosti*. [online]. [cit. 2018-03-01]. Dostupné z: <http://www.kvm.tul.cz/getFile/id>, 2003, 2499.
- [36] QUYEN, Vu Thi Bich, Tran Thi THUY VAN a Cao Quoc KHANH. A new algorithm for size optimization of the truss structures using finite element method. *IOP Conference Series: Materials Science and Engineering*. 2018, 365. DOI: 10.1088/1757-899X/365/4/042055. ISSN 1757-8981. Dostupné také z: <https://iopscience.iop.org/article/10.1088/1757-899X/365/4/042055>
- [37] RAHMAN, Izaz Ur, Muhammad ZAKARYA, Mushtaq RAZA a Rahim KHAN. An n-state switching PSO algorithm for scalable optimization. *Soft Computing*. 2020, 24(15), 11297-11314. DOI: 10.1007/s00500-020-05069-2. ISSN 1432-7643. Dostupné také z: <http://link.springer.com/10.1007/s00500-020-05069-2>
- [38] ŠŤASTNÝ, Antonín. *Modelování silových účinků působících na dopravní a manipulační zařízení s cílem jejich optimalizace*. Brno, 2015. Dizertační práce. FSI VUT v Brně. Vedoucí práce Doc. Ing. Miroslav Škopán, Csc.
- [39] TALATAHARI, Siamak, Mahdi AZIZI, Huu-Tai THAI a Manh-Hung HA. Optimization of constrained mathematical and engineering design problems using chaos game optimization. *Advances in Engineering Software*. 2020, 145, 1-16. DOI: 10.1016/j.cie.2020.106560. ISSN 03608352. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S0360835220302941>
- [40] TRUONG, Viet-Hung, Quang-Viet VU, Huu-Tai THAI a Manh-Hung HA. A robust method for safety evaluation of steel trusses using Gradient Tree Boosting algorithm. *Advances in Engineering Software*. 2020, 147, 1-16. DOI: 10.1016/j.advengsoft.2020.102825. ISSN 09659978. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S0965997819311585>
- [41] TUSHAJ, Ersilio. Size optimization of steel trusses using a genetic algorithm in MATLAB. *JOURNAL OF SOFTWARE ENGINEERING & INTELLIGENT SYSTEMS*. Rawalpindi, Pakistan: Caomei, 2017, 2(2), 10. ISSN 2518-8739.
- [42] WANG, S.Y. a K. TAI. *Graph representation for structural topology optimization using genetic algorithms*. 2004, 82(20-21), 1609-1622. DOI: 10.1016/j.compstruc.2004.05.005. ISSN 00457949. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S0045794904001750>
- [43] WEBER, Bryan. MATLAB vs Python: Why and How to Make the Switch. *Realpython* [online]. Connecticut: University of Connecticut, 2019, 09.09.2019 [cit. 2020-06-20]. Dostupné z: <https://realpython.com/matlab-vs-python/#matlab-vs-python-comparing-features-and-philosophy>

7 AUTOROVO CV

Jméno, příjmení, titul: Lukáš Zeizinger, Ing.
Datum a místo narození: 05. listopadu 1991, Šternberk
Státní občanství: Česká republika

Vzdělání:

01/09/2016 – Doktorské studium – Konstrukční a procesní inženýrství
 do současnosti FSI, VUT v Brně, Brno
 Optimalizační algoritmus pro příhradové ocelové konstrukce

25/11/2016 – Specializační studium technického znalectví pro pracovníky
 23/08/2017 znaleckých ústavů na VUT v Brně, ÚSI, VUT v Brně, Brno

01/09/2014 – Magisterské studium – Odbor transportních a stavebních strojů
 31/08/2016 ÚADI, FSI, VUT v Brně, Brno

01/07/2011 – Bakalářské studium – Strojní inženýrství
 31/08/2014 FSI, VUT v Brně, Brno

Pracovní zkušenosti:

01/11/2019 – Technickohospodářský pracovník – ÚADI – Odbor spalovacích
 do současnosti motorů
 ÚADI, VUT v Brně, Brno

01/10/2016 – Projektant ocelových konstrukcí
 31/03/2019 Noen a.s., Praha
 Výpočtář a projektant ocelových konstrukcí kolesových rypadel

01/06/2014 – Projektový technik aditivních technologií
 31/12/2015 RSmodels s.r.o., Rýmařov
 Projektový technik pro inovaci prototypové aditivní výroby

Výuka:

-Vedení cvičení z předmětů: Nosné konstrukce strojů I (QN1), Nosné konstrukce strojů II (QN2), Počítačové modelování strojních konstrukcí (QMK), Dopravní a manipulační zařízení (EDO), Hydraulické převody strojů (FHP), Počítačové navrhování strojů NX (QN0-A), Bakalářský projekt (FDS)

-Vedení bakalářských prací

Projekty:

2020 - Nové technologie pro nízkoemisní mobilitu, FSI-S-20-6267, řešitel

2018 - Modelování a simulace virtuálních dynamických silových účinků na nosné konstrukce stavebních strojů, FV 18–05, řešitel

2017 - Modelování silových účinků hydraulických systémů na nosné konstrukce mobilních pracovních strojů, FV 17-36, spoluřešitel

- Výzkum výpočtových a experimentálních metod v oblasti mobilních systémů, FSI-S-17-4104, řešitel

Ocenění vědeckou komunitou:

Získal cenu za příspěvek "Návrhová optimalizace příhradových konstrukcí" v soutěži o nejlepší práci mladého účastníka konference.

Dynamika tuhých a deformovatelných těles, Ústí nad Labem, 12. října 2017.