

České vysoké učení technické v Praze,  
Fakulta stavební

Czech Technical University in Prague,  
Faculty of Civil Engineering

Ing. Matěj Lepš, Ph.D.

Metody evoluční optimalizace a jejich  
aplikace ve stavebním inženýrství

Evolutionary Optimization Methods and  
Their Applications in Civil Engineering

# Summary

The current development in computing technologies enables shift from optimization of small structural parts to the optimization of whole structures. Traditional mathematical programming methods based on the fastest descent to local extremes can have problems to solve big optimization tasks demanded by engineering practice. Therefore, there is an aim to develop robust optimization algorithms, which are able to find an improvement of the original design on an arbitrary optimization problem. Studies show that evolutionary algorithms are capable to fulfill these demands.

Therefore, the first part of this work presents the zero order methods, that do not utilize a knowledge of derivatives and even more, do not need a continuity or convexity of the given optimization problem. The bigger part is devoted to algorithms that are inspired by nature, both by physical and evolutionary processes. Evolutionary algorithms are then one of the most popular examples of zero order methods.

During the research project in the optimization area performed in collaboration between company Nemetschek Scia and Faculty of Civil Engineering CTU, a prototype of optimization module EOT (Engineering Optimization Toolbox) has been developed, which is connected to the Scia Engineer program. The purpose of this work is to get a tool, which is able to optimize any structure with properties defined by any set of parameters, which means, to be a general optimization tool for usage in civil engineering practice. The important goal is to optimize structures with respect to the final design code and construction rules. The second part of this work presents applications of this tool to several problems encountered in civil engineering practice.

# Souhrn

Vývoj počítačové techniky v posledních letech umožnil posun od optimalizace jednotlivých konstrukčních prvků k optimalizaci celých konstrukcí. Ukazuje se, že tradiční matematické metody založené na vyhledávání lokálních extrémů mají problémy při řešení optimalizačních úloh požadovaných inženýrskou praxí. Nejen z tohoto důvodu je snaha nalézt robustní optimalizační metody, které budou schopny zlepšit původní návrh na libovolném optimalizačním problému. Ukazuje se, že zástupci evolučních algoritmů jsou schopni tyto požadavky splnit.

První část této práce představuje metody přímého vyhledávání, které k nalezení optimu nevyužívají znalostí derivací a zároveň ani nevyžadují po optimalizačním problému jeho spojitost či konvexitu. Větší část je věnována algoritmům, které našly svůj předobraz v přírodě, a to jak ve fyzikálních zákonech, tak zejména v evolučních procesech. Evoluční algoritmy jsou pak jedny z nejznámějších zástupců celé oblasti metod přímého vyhledávání.

Výsledkem spolupráce na poli optimalizačních algoritmů mezi firmou Nemetschek Scia a Fakultou stavební, ČVUT v Praze, je prototyp optimalizačního nástroje EOT (Engineering Optimization Toolbox), který je spojen s návrhovým a statickým softwarem Scia Engineer. Cílem tohoto projektu je vytvoření nástroje, který je schopen optimalizovat libovolnou konstrukci, jejíž vlastnosti jsou definovány zadanou sadou parametrů. Cílem je poskytnout obecný optimalizační nástroj pro běžnou inženýrskou praxi. Nedílnou součástí tohoto nástroje je zohlednění jak normových, tak i konstrukčních požadavků. Druhá část této práce tak představuje aplikaci tohoto nástroje na některé problémy z inženýrské praxe.

## **Klíčová slova**

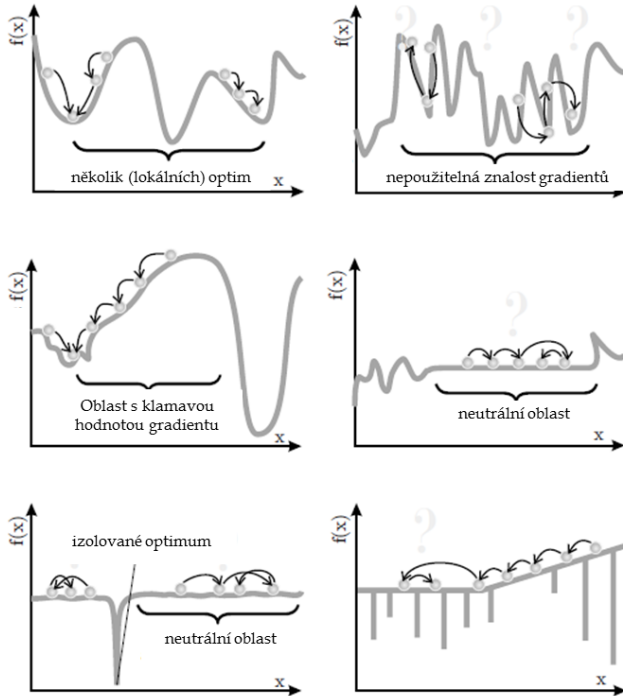
optimalizace; evoluční algoritmy; genetické algoritmy; návrh konstrukcí; železobetonové konstrukce; ocelové konstrukce

## **Keywords**

optimization; evolutionary algorithms; genetic algorithms; structural design; reinforced concrete; steel structures

# Obsah

<b>1</b>	<b>Úvod</b>	<b>6</b>
<b>2</b>	<b>Metody přímého vyhledávání</b>	<b>7</b>
2.1	Simplexový algoritmus . . . . .	8
2.2	Simulované žíhání . . . . .	9
2.3	Genetické algoritmy . . . . .	11
2.4	Diferenciální evoluce . . . . .	11
2.5	Modifikované simulované žíhání . . . . .	12
<b>3</b>	<b>Aplikace evolučních algoritmů</b>	<b>12</b>
3.1	Optimalizační nástroj EOT . . . . .	15
3.2	Příklad návrhu ocelové haly . . . . .	16
3.3	Příklad návrhu železobetonového skeletu . . . . .	19
<b>4</b>	<b>Závěr</b>	<b>20</b>

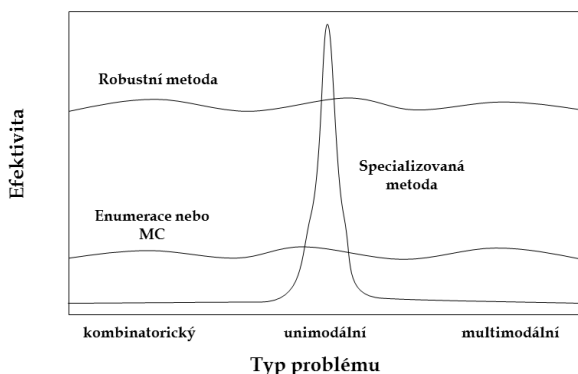


Obrázek 1: Nástin problémů globální optimalizace a několik příkladů, kde gradientní metody selhávají; reprodukováno z [9].

## 1 Úvod

Při optimalizaci velkých inženýrských problémů často selhávají tradiční metody matematického programování, které vycházejí ze znalosti dané funkce (Obr. 1). V mnoha případech takovéto postupy vyžadují splnění určitých předpokladů, například spojitosti, konvexity atp., které nejsme schopni zajistit. To v historii vedlo k nedůvěře mnoha inženýrů k matematickým metodám. Hledaly se tudíž optimalizační algoritmy, které by nebyly citlivé na typ optimalizačního problému, počet a typ proměnných atd. Cílem tedy byla tzv. robustní optimalizační metoda, která by pro široké spektrum úloh vykazovala výsledky lepší než slepé či enumerativní algoritmy. Tyto výzkumné snahy je možné dokumentovat na značně zjednodušeném Obr. 2, kdy robustní metoda vítězí na

poli široké škály problémů, nicméně pro daný jeden konkrétní problém, např. konvexní funkci, nikdy nebude schopna dosahovat výsledků specializovaných metod. U komplexních inženýrských optimalizačních problémů je tak zapotřebí prohledat prostor možných řešení. Na toto pole nastupují metaheuristiky, u nichž je zaručeno 100% nalezení minima za předpokladu nekonečného počtu iterací. Úloha se tak v podstatě redukuje z hledání minima cílové funkce na minimalizaci počtu iterací zvolené stochastické metody.

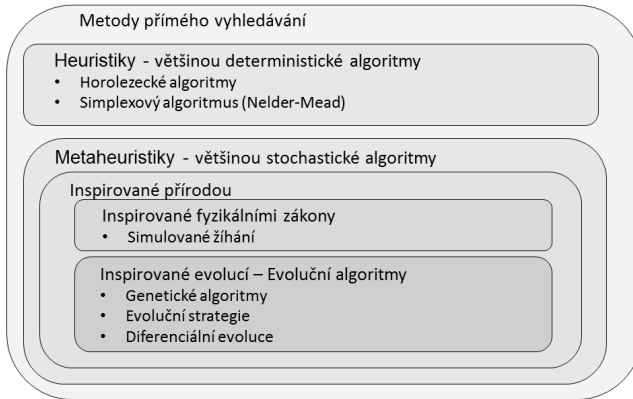


Obrázek 2: Naivní pohled na hledání robustní optimalizační metody; reprodukováno z [2].

Práce je zorganizována následovně: Nejprve jsou představeny metody přímého vyhledávání, jejíž dominantní složkou jsou právě evoluční algoritmy. Následně jsou představeni nejznámější reprezentatni jednotlivých podoblastí metody přímého vyhledávání. Využití těchto algoritmů je následně představeno na aplikaci optimalizačního nástroje EOT, na kterém se autor podílel. Praktické ukázky optimalizace jsou pak představeny na návrhu ocelové haly a návrhu železobetonového skeletu.

## 2 Metody přímého vyhledávání

Optimalizační oblast metod přímého vyhledávání zahrnuje různorodou skupinu algoritmů, které k nalezení optimálního řešení využívají pouze funkční hodnoty v daném bodě. Hrubé dělení zobrazené na Obr. 3 zahrnuje oblast heuristických algoritmů, které využívají znalosti vlast-



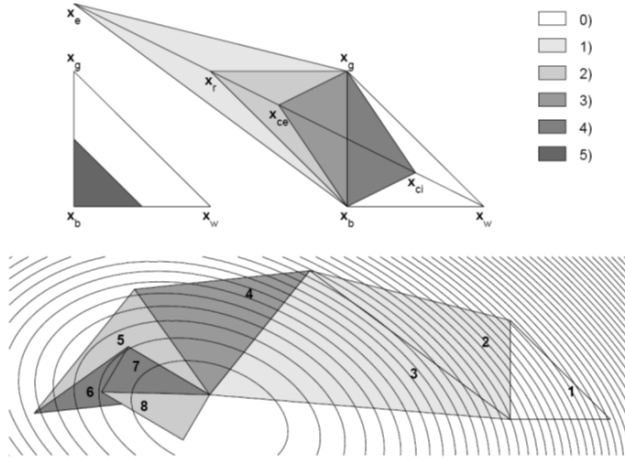
Obrázek 3: Hrubé dělení metod přímého vyhledávání.

ností optim k nalezení podobných řešení. Nejznámějšími jsou různé varianty horolezeckých algoritmů či zde představovaná metoda simplexová. Většina heuristik jsou deterministické algoritmy, tzn. mají pevně zadanou strukturu jednotlivých kroků. Naopak, metaheuristiky jsou algoritmy imitující jiné procesy. Většina těchto algoritmů v průběhu algoritmu využívá generátory náhodných čísel. Tímto se tyto metody řadí mezi stochastické algoritmy, které vykazují náhodné chování a tudíž je zapotřebí jejich konvergenční charakteristiky studovat z několika nezávislých spuštění.

## 2.1 Simplexový algoritmus (Nelder-Mead)

Přestože je řada důkazů, že Nelderova-Meadova metoda známá též jako metoda simplexů na mnoha optimalizačních případech selhává [6], patří tato metoda mezi heuristickými metodami mezi nejvíce oblíbené. Základem je simplex, tedy nejjednodušší spojení  $N + 1$  bodů v  $N$ -dimenzionálním prostoru. V každé iteraci se pozice simplexu přemístí do nové pozice a tím vytvoří opět simplex. Posun a výběr bodu (bodů) je dán šesti jednoduchými pravidly, viz Obr. 4, které jsou vybírány de facto na základě aproximace gradientu ze znalosti funkčních hodnot v bodech simplexu.

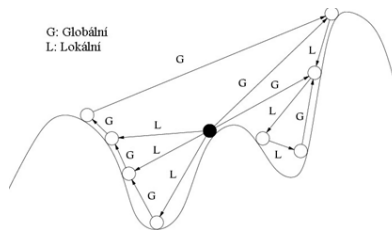




Obrázek 4: Nelderova-Meadova metoda: 0) originální trojúhelník, 1) expanze, 2) reflexe, 3) vnější kontrakce, 4) vnitřní kontrakce, 5) redukce; reprodukováno z [4].

## 2.2 Simulované žíhání

Metoda simulovaného žíhání je založena na fyzikálních principech, kdy optimalizační algoritmus efektivně kopíruje proces žíhání v pevných látkách. Při žíhání je nejprve teplota nastavena na vysokou hodnotu a pak je postupně pomalu snižována tak, aby jednotlivé krystaly materiálu měly čas zaujmout stav s minimální energií na dané teplotní hladině. Jak teplota klesá dolů, klesá i celková energie materiálu až dosáhne své minimální hodnoty. Algoritmus simulovaného žíhání pracuje na stejném principu - je náhodně vytvořeno počáteční řešení, vir-



Obrázek 5: Princip simulovaného žíhání.

tuální parametr nazývaný *teplota* je nastaven na startovací hodnotu a je náhodně vytvořeno další řešení. Pokud je z pohledu cílové funkce nové řešení lepší než staré, je automaticky nové přijato. Pokud je horší, stále zůstává šance, že nahradí staré řešení - tato pravděpodobnost je závislá na aktuální teplotě, která umožní řešení uniknout z lokálního minima (Obr. 5) a to na základě pravděpodobnosti, která se řídí Boltzmannovým rozdělením [6]

$$Pr(E) \sim e^{\left(\frac{-E}{k_B T}\right)}. \quad (1)$$

Uvedený vztah popisuje rozdělení energie systému, který je v tepelné rovnováze s teplotou  $T$ , mezi všemi rozdílnými energetickými stavy, kde  $k_B$  je Boltzmannova konstanta. Proces je opakován několikrát při konstantní teplotě, která je postupně snižována do předepsaného minima. Tato verze algoritmu byla poprvé představena v [3] a nezávisle v [8].

---

```

1  t = 0
2  vytvoř populaci P0, ohodnot P0
3  while (not zastavovací - podmínka) {
4      t = t + 1
5      vyber Mt z Pt-1                (aplikace výběru)
6      změň Mt                        (aplikace genetických operátorů)
7      vytvoř Pt z Mt a ohodnot Pt  (vlození nových jedinců do Pt)
8  }
```

---

Obrázek 6: Princip genetického algoritmu.

šablona	0 1 0 0 1 1 1 0 0 1
rodič 1	1 1 1 0 1 0 0 1 0 0      1 1 1 0 1 0 0 1 0 0
potomek 1,2	1 1 1 0 1 0 0 1 1 0      1 0 1 0 0 1 0 1 0 1
rodič 2	1 0 1 0 0 1 0 1 1 1      1 0 1 0 0 1 0 1 1 1

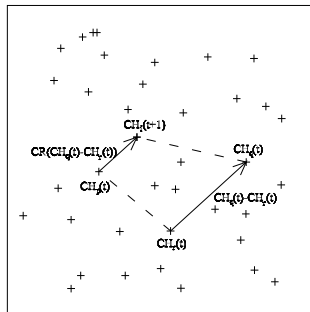
Obrázek 7: Uniformní křížení.

## 2.3 Genetické algoritmy

Genetické algoritmy jsou formulovány na základě analogie s biologickým vývojem v přírodě a ve své podstatě vychází z principů Darwinovy evoluční teorie. Názvosloví pak pochází z genetiky, odkud jsou převzaty i některé procesy. Základní charakteristikou tohoto algoritmu je využití populace jedinců, jejichž postupným vývojem dochází k prohledávání dostupného prostoru paralelně, což zajišťuje tomuto algoritmu požadovanou robustnost. Stejně jako v přírodě, i zde spolu jednotlivá idividua soupeří v zájmu vlastního přežití tak, že silnější jedinci přežívají do dalších generací, zatímco slabší obvykle zahynou. Tento proces lze stručně popsat algoritmem na Obr. 6. Algoritmus zahrnuje základní kroky cyklu jednoduchého genetického algoritmu: reprodukci (#5), rekombinační fázi (#6) složenou z rekombinačních operátorů křížení (Obr. 7) a mutace (Obr. 8) a z výběru nové populace (#7). Jeden průchod cyklem (#3) se pak nazývá generace. Podrobný popis a význam jednotlivých částí lze nalézt např. v [2]

1 1 1 0 1 0 0 1 1 0                      1 0 1 0 1 0 1 1 1 0

Obrázek 8: Uniformní mutace.



Obrázek 9: Diferenční operátor diferenciální evoluce.

## 2.4 Diferenciální evoluce

Diferenciální evoluce je zástupce populačních evolučních algoritmů, které pracují přímo s reálnými vektory [7]. Zavádí diferenční operátor,

který nahrazuje binární operátor křížení v genetickém algoritmu. Za předpokladu, že  $CH_i(t)$  je  $i$ -té řešení v generaci  $t$ ,

$$CH_i(t) = (ch_{i1}(t), ch_{i2}(t), \dots, ch_{in}(t)), \quad (2)$$

kde  $n$  je počet proměnných, a  $\Lambda$  je náhodná podmnožina  $\{1, 2, \dots, n\}$ , pak pro každé  $j \in \Lambda$

$$\begin{aligned} ch_{ij}(t+1) = ch_{ij}(t) &+ F_1(ch_{pj}(t) - ch_{qj}(t)) \\ &+ F_2(ch_{\text{best}j}(t) - ch_{ij}(t)), \end{aligned} \quad (3)$$

a pro každé  $j \notin \Lambda$  je nové řešení po křížení

$$ch_{ij}(t+1) = ch_{ij}(t), \quad (4)$$

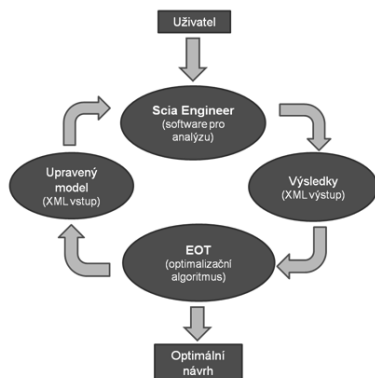
Zde jsou  $ch_{pj}$  a  $ch_{qj}$   $j$ -té souřadnice dvou náhodně zvolených řešení z populace a  $ch_{\text{best}j}$  je  $j$ -tá souřadnice nejlepšího řešení v populaci v daném čase  $t$ .  $F_1$  a  $F_2$  jsou parametry algoritmu z intervalu  $(0, 1)$ . Obr. 9 zobrazuje geometrický význam operátoru. Oproti klasickému genetickému algoritmu chybí operátor mutace a nová řešení nahrazují pouze své rodiče. Přestože algoritmus diferenciální evoluce využívá výhod populace, zkušenosti s tímto algoritmem ukazují nižší robustnost vykoupenou vyšší rychlostí konvergence oproti genetickým algoritmům.

## 2.5 Modifikované simulované žhání (MSA)

V díle [5] byla představena efektivní kombinace genetických algoritmů a metody simulovaného žhání. Využívá základy genetických algoritmů (populace chromozomů a genetické operátory) spolu s principem vyhledávání řešení směrem ke stavům s minimální energií. Ve fázi začlenění nového řešení do populace (krok #7 genetického algoritmu) je využito principu simulovaného žhání, kdy jsou horší řešení přijata s pravděpodobností danou rovnicí (1). V následujícím textu je tato metoda označena zkratkou MSA z anglického *Modified Simulated Annealing*.

## 3 Aplikace evolučních algoritmů ve stavebním inženýrství

Výsledkem spolupráce na poli optimalizačních algoritmů mezi firmou Nemetschek Scia a Fakultou stavební, ČVUT v Praze, je prototyp optimalizačního nástroje EOT (Engineering Optimization Toolbox), který je spojen s návrhovým a statickým softwarem Scia Engineer a spolu vytvářejí programový nástroj Scia Engineer MOOT pro



Obrázek 10: Schéma komunikace nástroje EOT se softwarem SCIA Engineer.

obecnou optimalizaci stavebních konstrukcí. Je tvořen kombinací široce používaného programu pro výpočty a posudky Scia Engineer a optimalizačního řešiče EOT (EOT - Engineering Optimization Tool). Integrace těchto dvou programů nabízí unikátní řešení pro mnohostranně použitelnou optimalizaci všech typů staveb [1]:

**Scia Engineer** je komplexní programový balík pro výpočty a posudky stavebních konstrukcí. Integraci Scia Engineer do procesu obecné optimalizace umožňují jeho nadstandardní vlastnosti programu, jako jsou:

**Parametrizace modelu:** číselné hodnoty jednotlivých vlastností modelu mohou být nahrazeny parametry. Parametry lze zadávat a upravovat přímo ve Scia Engineer nebo pomocí dalších programů přes otevřené komunikační rozhraní.

**AutoDesign:** automatické vyhledání minimálního návrhu jednotlivých částí modelu - např. vyhledání minimálně nutného ocelového průřezu nebo minimální nutné výztuže pro železobetonový průřez na základě spočtených vnitřních sil.

**Rozhraní XML:** otevřené rozhraní pro komunikaci mezi různými aplikacemi.

**EOT** je optimalizační řešič, který umožňuje uživatelům zadávat optimalizační funkci, vztahy mezi parametry a vybrat vhodnou optimalizační metodu. Řešič vyhledává optimální řešení podle za-

daných požadavků při snaze o co nejnižší počet vyžadovaných kroků.

Pomůcka k výběru optimalizační metody					
Metoda	Vlastnosti metody		Vlastnosti projektu		Příklady projektů
	Rychlost konvergence	Robustnost	Počet parametrů	Rozložení parametrů	
GS	Velmi rychlá	--	Jeden	jen spojité	Umístění podpory, zatížitelnost
N-M	Středně rychlá	Nižší	Malý	jen spojité	Příhradové konstrukce - průřezy
SQP	Rychlá	Velmi nízká	Středně velký	jen spojité	Rámové konstrukce
DE	Pomalá	Vyšší	Vysoký	Ize diskrétní	Alternativně železobeton
MSA	Velmi pomalá	Vysoká	Vysoký	Ize diskrétní	Železobeton, rozsáhlé projekty

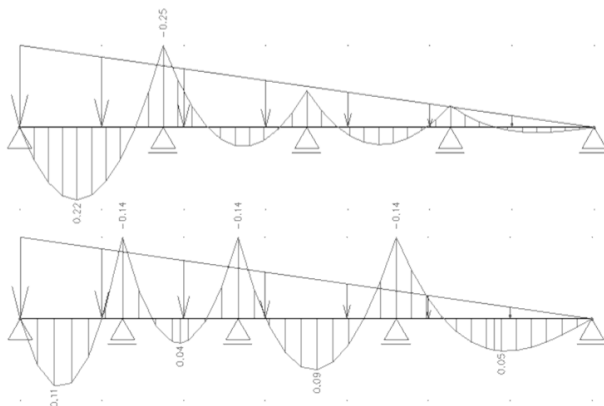
Obrázek 11: Hrubé rozdělení optimalizačních metod dostupných v programu EOT. Klíč: GS = metoda zlatého řezu, N-M = simplexový algoritmus, SQP = sekvenční kvadratické programování, DE = diferenciální evoluce, MSA = modifikované simulované žíhání.

Proces optimalizace je znázorněn na Obr. 10. Jsou-li zadána vstupní data, tzn. model počítané konstrukce je vytvořen a upraven pro optimalizaci, potom vyhledání optimálního řešení probíhá zcela automaticky bez zásahu uživatele. Pro reálné konstrukce může být nalezeno více potenciálně vhodných řešení, která se z matematického hlediska mohou lišit pouze minimálně, ale konstrukční řešení se mohou podstatně lišit. Proto je na uživateli, aby se rozhodl, které z nabízených řešení bude nakonec vybráno.

Celý proces tvorby optimalizační úlohy se tedy skládá z následujících kroků:

- vytvoření modelu a jeho parametrizace,
- zadání optimalizované (cílové) funkce a výběr metody,
- optimalizačního cyklu a
- vyhodnocení nalezených optim,

kdy jádro optimalizace se skrývá v programu EOT.



Obrázek 12: Jednoduchý optimalizační příklad, kdy jsou hledány pozice třech vnitřních podpor tak, aby byly minimalizovány ohybové momenty. Počáteční (nahore) a optimální řešení (dole).

### 3.1 Optimalizační nástroj EOT

Optimalizační nástroj EOT (Engineering Optimization Toolbox) nabízí v současné době několik optimalizačních metod:

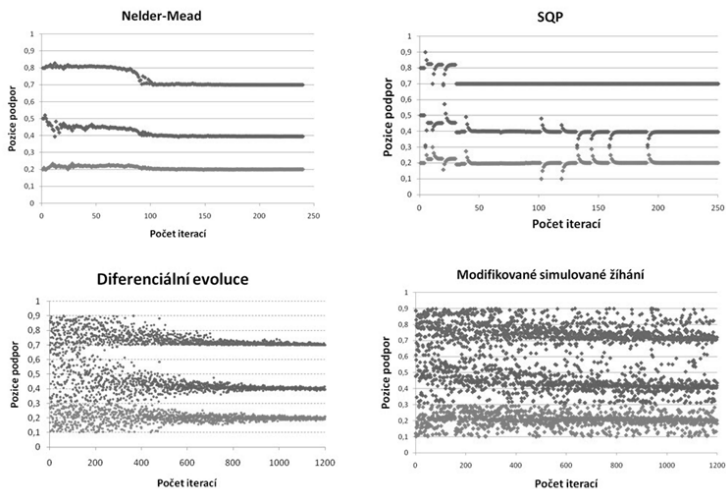
**Jednparametrickou metodu**, konkrétně metodu zlatého řezu (GS),

**Gradientní metodu**, konkrétně sekvenční kvadratické programování (SQP),

**Heuristickou metodu**, simplexový algoritmus (N-M), viz kapitola 2.1,

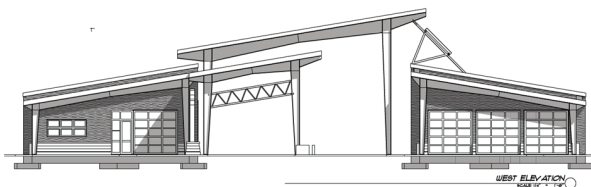
**Evoluční algoritmy**, jmenovitě diferenciální evoluci (DE), viz kapitola 2.4, a modifikované simulované žíhání (MSA), viz kapitola 2.5.

Výběr vhodné metody je nejvíce ovlivněn typem a množstvím parametrů, které popisují daný optimalizační problém. Doporučené rozdělení metod je uvedeno na Obr. 11. Porovnání rychlosti konvergence a způsobů prohledávání daného prostoru lze nejlépe dokumentovat na jednoduchém příkladu. Obr. 12 představuje čtyřpolový spojitý nosník zatížený trojúhelníkovým spojitým zatížením. V tomto případě je cílem nepodmíněná minimalizace maximálních hodnot ohybových momentů. Průběh konvergence víceparametrických optimalizačních metod je pak zobrazen na Obr. 13. Charakter konvergence je pro jednotlivé metody typický,



Obrázek 13: Ukázka konvergence čtyř algoritmů na jednoduchém příkladu z Obr. 12. Jednotlivé barvy na svislé ose představují pozice třech vnitřních podpor v průběhu optimalizace.

od nejrychlejšího kvadratického programování až po binární modifikované simulované žihání. Další část textu bude věnována praktickým příkladům z inženýrské praxe.

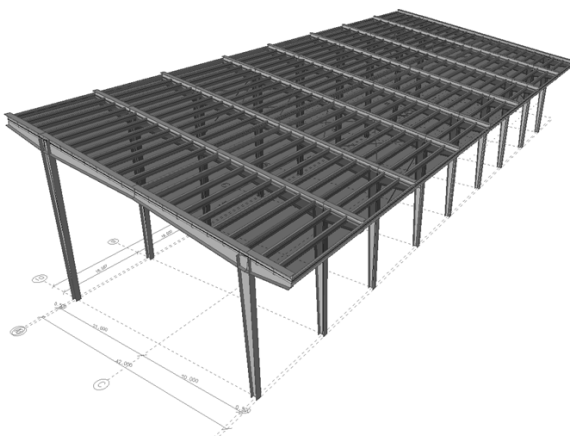


Obrázek 14: Ocelová otevřená konstrukce mezi dvěma existujícími budovami.

### 3.2 Příklad návrhu ocelové haly

Prvním příkladem je návrh otevřené ocelové haly mezi dvěma existujícími budovami, viz Obr. 14. Konkrétně se jedná o devět rámových výseků, viz taktéž model konstrukce na Obr. 15. Rozteč sloupů je 12,5 m

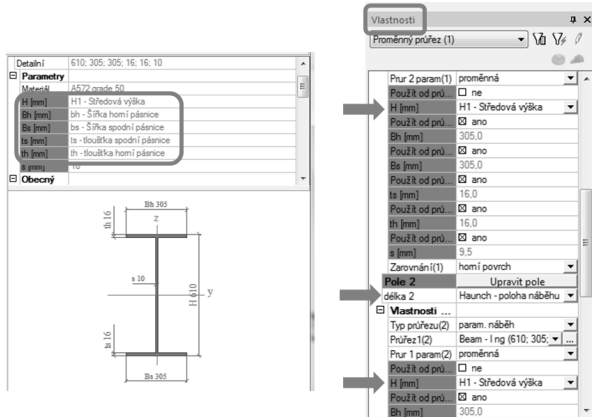




Obrázek 15: 3D model konstrukce.

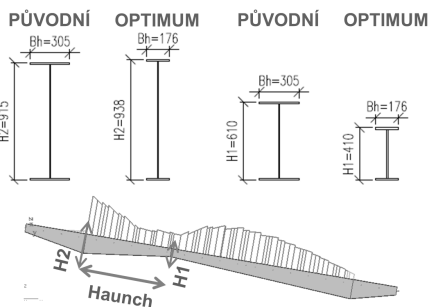
x 5,08 m a zatížení zahrnuje vlastní tíhu a stálé zatížení, zatížení větrem a sněhem. Cílovou funkcí byla hmotnost konstrukce a omezujícími podmínkami splnění prvního i druhého mezního stavu. Byl optimalizován jak jeden rámový výsek, tak i celá konstrukce. Dále byly prozkoumány různé kombinace parametrů konstrukce a jejich vliv na výslednou cenu. Zadáání jednotlivých parametrů průřezu je patrné z Obr. 16, konstrukčních parametrů pak na Obr. 17. Tamtéž je představen příklad porovnání dvou studií. Je patrné, že postupným přidáváním proměnných je možné docílit drobných úspor. Nicméně čím je víc parametrů, tím delší je potřebný čas nutné optimalizace.

Výsledná optimalizace celé konstrukce zabrala přes 10 hodin výpočetního času. Z praktických důvodů byly ve výsledku použity zaokrouhlené hodnoty parametrů blízko nalezeného optima. Konečná úspora proti původnímu návrhu tak dosáhla téměř 10 %.

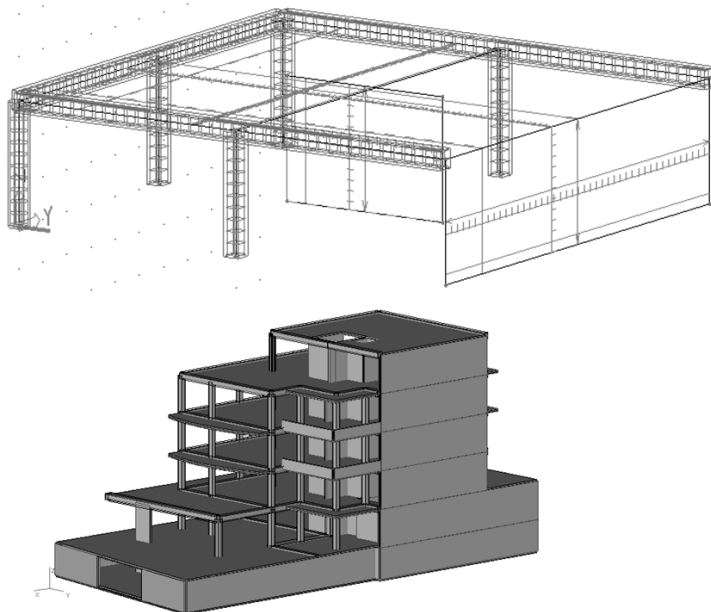


Obrázek 16: Přiřazení parametrů pro optimalizaci rámového výseku.

	PARAMETRY							Hmotnost [kg]	Úspora [%]
	Haunch [relativ.]	H1 [mm]	H2 [mm]	Bh [mm]	Bs [mm]	th [mm]	ts [mm]		
Původní	0,5	610	915	305	305	16	16	20758	
Optimum 1	0,5	<b>514</b>	<b>873</b>	<b>260</b>	<b>260</b>	16	16	19114	<b>7,92</b>
Optimum 2	<b>0,416</b>	<b>480</b>	<b>888</b>	<b>282</b>	<b>341</b>	<b>13</b>	<b>10</b>	18509	<b>10,8</b>



Obrázek 17: Porovnání výsledků dvou optimalizačních studií; v dolní části definice dalších proměnných.

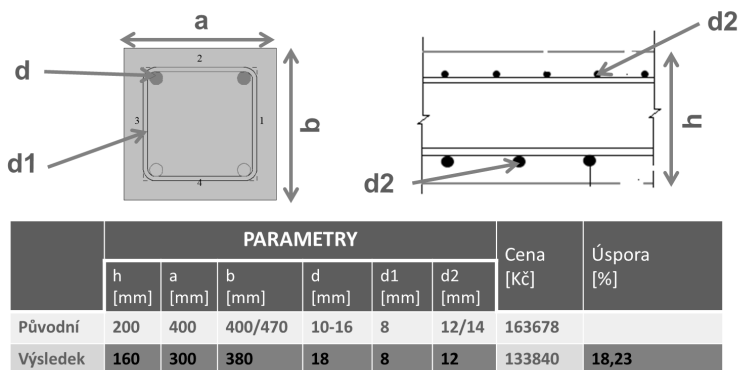


Obrázek 18: Návrh výztuže v horním patře železobetonového skeletu (nahore), 3D model železobetonového skeletu (dole).

### 3.3 Příklad návrhu železobetonového skeletu

Cílem následujícího příkladu je optimalizace návrhu železobetonové konstrukce. Jedná se o bytový dům v proluce, s jedním podzemním a pěti nadzemními patry, viz Obr. 18. Zatížení zahrnuje vlastní tíhu, stálé i užitné zatížení a zatížení větrem a sněhem. Železobetonový skelet je doplněn stěnami v suterénu, stěnami na straně původní zástavby a stěnami schodišťového jádra. Byly provedeny dva typy optimalizace, detailní návrh posledního patra a hrubý návrh celé budovy. Cílovou funkcí byla cena hlavních materiálů vyjádřená v korunách a omezujícími podmínkami splnění prvního i druhého mezního stavu. Parametry rozměrů prutových prvků a označení parametrů podélné a smykové výztuže je naznačeno na Obr. 19. V rámci optimalizace detailního návrhu posledního patra byla optimalizována i výztuž v deskových prvcích, kdy je z výsledků na Obr. 19 jasně patrné, že rozhodujícím parametrem byla tloušťka desky umožňující až 18% úsporu. Při hrubém návrhu celé

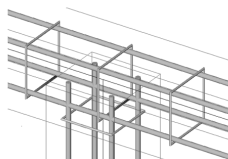
budovy byl optimalizován pouze železobetonový skelet. Jednotlivé parametry jsou odstupňovány podle výšky budovy do dvou skupin, jak je patrné na Obr. 20. Celková 15% úspora na celkové ceně skeletu pak byla vykoupena více než 60 hodinami výpočetního času.



PŮVODNÍ



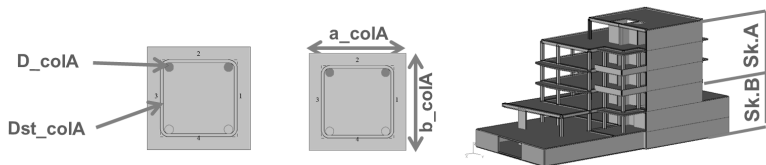
OPTIMUM



Obrázek 19: Definice některých proměnných při optimalizaci horního patra železobetonového skeletu (nahore) a výsledky optimalizace (dole).

## 4 Závěr

O potřebě optimalizací v současné době není potřeba diskutovat. S přibývajícím výpočetním výkonem a s nástrojem EOT představeným v této práci je možné dosáhnout praktických výsledků na širokém spektru stavebních konstrukcí. Efektivitu aplikace optimalizace lze zhodnotit z dosavadních zkušeností, které ukazují míru úspor v mezích od 5 do 20 %. Je nutné podotknout, že při vhodné úvaze je optimalizace vhodná i na složitější konstrukce. Představené evoluční algoritmy pak poskytují robustní alternativu ke klasickým matematickým metodám.



	PARAMETRY PRŮŘEZŮ					
	a_beam [mm]	b_beam [mm]	a_colA [mm]	b_colA [mm]	a_colB [mm]	b_colB [mm]
Původní	400-470	200-400	400-600	400-600	400-600	400-600
Optimum	180	340	320	370	320	420

	PARAMETRY VÝTZUŽE						Cena [Kč]	Úspora [%]
	D_beam [mm]	Dst_beam [mm]	D_colA [mm]	Dst_colA [mm]	D_colB [mm]	Dst_colB [mm]		
Původní	10-16	8-12	16	8-12	16	8-12	250258	
Optimum	22	10	20	12	26	8	213019	14,88

Obrázek 20: Definice některých proměnných při optimalizaci celého železobetonového skeletu (nahore) a výsledky optimalizace (dole).

**Poděkování** Na tomto místě bych rád poděkoval kolegům z katedry mechaniky, Fakulty stavební ČVUT v Praze, konkrétně J. Zemanovi, A. Kučerové, J. Noskovi a D. Šedlbauerovi, a taktéž kolegům z firmy Nemetschek Scia, konkrétně L. Dlouhému, J. Kabeláčovi, R. Blažkovi a M. Novákovi, za podporu, rady a pomoc při práci na projektu EOT. Projekt EOT byl podpořen z prostředků státního rozpočtu prostřednictvím Ministerstva průmyslu a obchodu (projekt MPO FT-TA4/100).

## Reference

- [1] Kolektiv autorů. Scia Engineer MOOT: Automatická optimalizace stavebních konstrukcí. Nemetschek Scia, White Paper, <http://www.scia-online.com/cs/engineering-white-papers.html>, to be published, 2011.
- [2] D. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989.

- [3] S. Kirkpatrick, C.D. Gelatt, Jr., and M. P. Vecchi. Optimization by Simulated Annealing. *Science*, 220:671–680, 1983.
- [4] R. Hlavička L. Čermák. Numerické metody. Ústav matematiky, Fakulta strojíňho inženýrství, Vysoké učení technické v Brně, <http://mathonline.fme.vutbr.cz>, 2006.
- [5] S.W. Mahfoud and D. E. Goldberg. A genetic algorithm for parallel simulated annealing. In R. Manner and B. Manderick, editors, *Parallel Problem from Nature*, volume 2, pages 301–310. North-Holland, Amsterdam, 1992.
- [6] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C*. Cambridge University Press, 2nd edition, 1992.
- [7] R. Storn and K. Price. Differential Evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, University of Berkeley, 1995.
- [8] V. Černý. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *J. Opt. Theory Appl.*, 45:41–51, 1985.
- [9] T. Weise, M. Zapf, R. Chiong, and A. J. Nebro. *Nature-Inspired Algorithms for Optimisation*, volume SCI 193, chapter Why Is Optimization Difficult?, pages 1–50. Springer-Verlag Berlin Heidelberg, 2009.

# Ing. Matěj Lepš, Ph.D.

## Osobní data

Jméno: Matěj Lepš  
Narozen: 9. Prosince 1976, Benešov, Česká Republika  
Národnost: česká  
E-mail: leps@cml.fsv.cvut.cz

## Vzdělání

2000–2005 České Vysoké Učení Technické v Praze (Ph.D.)  
Fakulta stavební  
obor Konstrukce a dopravní stavby  
Téma disertační práce: Single and Multi-objective Optimization in Civil Engineering with Applications  
1995–2000 České Vysoké Učení Technické v Praze (Ing.)  
Fakulta stavební  
obor Konstrukce a dopravní stavby  
Téma diplomové práce: Optimalizace železobetonového spojitého nosníku

## Praxe

2006– Odborný asistent  
České Vysoké Učení Technické v Praze – Fakulta stavební  
2004–2006 Výzkumný pracovník  
České Vysoké Učení Technické v Praze – Fakulta stavební  
2000–2005 Student doktorského studia  
České Vysoké Učení Technické v Praze – Fakulta stavební  
1996–2000 Pomocná vědecká síla  
České Vysoké Učení Technické v Praze – Fakulta stavební

## Výzkumná činnost

Globální metody optimalizace  
Inverzní problémy  
Optimalizace konstrukcí  
Paralelní výpočty  
4 publikace na ISI Web of Science®  
59 citací na ISI Web of Science®  
(bez autocitací), *h*-index 3.  
Řešitel 4 grantových projektů (GAČR, FRVŠ, ESA)  
Jmenovaný spolupracovník 11 grantových projektů

## Výuka

### **Přednášky v předmětech:**

Moderní metody optimalizace

Stavební mechanika 1A

Zatížení a spolehlivost

### **Cvičení v předmětech:**

Stavební mechanika 1, 1A, 2, 2A, 3, 31, 40, 50

Pružnost a pevnost

Analýza konstrukcí

Numerická analýza konstrukcí 1

Zatížení a spolehlivost

Školitel **7** diplomových prací, **1** úspěšného

a **3** současných studentů doktorského studia

## Monografie

- [1] Lepš, M.: *Single and Multi-Objective Optimization in Civil Engineering*, In: Evolutionary Algorithms and Intelligent Tools in Engineering Optimization. WIT Press, 2005, p. 320-341. ISBN 1-84564-038-1.
- [2] Lepš, M.: *Identification of material parameters for concrete from simulations at different scales*, In: Science and Supercomputing in Europe, report 2007. CINECA, 2008, p. 641-645. ISBN 978-88-86037-21-1.
- [3] Lepš, M.: *Soft-Computing Methods in Material Parameters Identification: State of the Art and Open Issues*, In: Evolutionary and Deterministic Methods for Design, Optimization and Control. Barcelona: CIMNE, 2008, p. 317-322. ISBN 978-84-96736-45-0.
- [4] Lepš, M. - Vitingerová, Z.: *Towards the multi-objective identification of material parameters from indentation test*, In: Evolutionary and Deterministic Methods for Design, Optimization and Control. Barcelona: CIMNE, 2008, p. 335-340. ISBN 978-84-96736-45-0.

## Články v impaktovaném časopise

- [1] Hrstka, O. - Kučerová, A. - Lepš, M. - Zeman, J.: *A Competitive Comparison of Different Types of Evolutionary Algorithms*, In: Computers and Structures. 2003, vol. 81, no. 18-19, p. 1979-1990. ISSN 0045-7949.
- [2] Lepš, M. - Šejnoha, M.: *New Approach to Optimization of Reinforced Concrete Beams*, In: Computers and Structures. 2003, vol. 81, no. 18-19, p. 1957-1966. ISSN 0045-7949.
- [3] Kuráž, M. - Mayer, P. - Lepš, M. - Trpkošová, D.: *An adaptive time discretization of the classical and the dual porosity model of Richards' equation*, In: Journal of Computational and Applied Mathematics, vol. 233, p.3167–3177, 2010. ISSN: 0377-0427.