



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STAVEBNÍ
ÚSTAV STAVEBNÍ MECHANIKY

FACULTY OF CIVIL ENGINEERING
INSTITUTE OF STRUCTURAL MECHANICS

OPTIMALIZACE BETONOVÝCH KONSTRUKCÍ STOCHASTICKÝMI METODAMI OPTIMALIZACE

TITLE

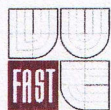
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

ONDŘEJ SLOWIK

VEDOUCÍ PRÁCE
SUPERVISOR

prof. Ing. DRAHOMÍR NOVÁK, DrSc.



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ FAKULTA STAVEBNÍ

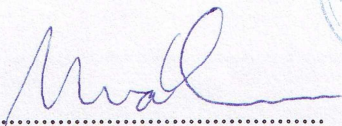
Studijní program B3607 Stavební inženýrství
Typ studijního programu Bakalářský studijní program s prezenční formou studia
Studijní obor 3608R001 Pozemní stavby
Pracoviště Ústav stavební mechaniky

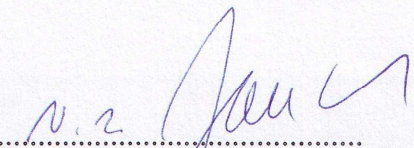
ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student Ondřej Slowik
Název Optimalizace betonových konstrukcí
stochastickými metodami optimalizace
Vedoucí bakalářské práce prof. Ing. Drahomír Novák, DrSc.
**Datum zadání
bakalářské práce** 30. 11. 2011
**Datum odevzdání
bakalářské práce** 25. 5. 2012

V Brně dne 30. 11. 2011




prof. Ing. Drahomír Novák, DrSc.
Vedoucí ústavu


prof. Ing. Rostislav Drochytka, CSc.
Děkan Fakulty stavební VUT

Podklady a literatura

Sborníky konferencí ICOSSAR a ICASP.

FREET - User's and Theory Guides, Brno/Červenka Consulting, V 1.5, 2011.

SARA - User's and Theory Guides, Červenka Consulting, 2010.

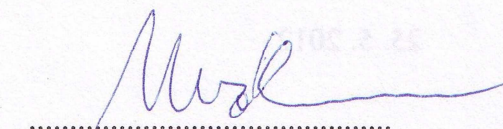
ATENA - User's and Theory Guides, Červenka Consulting, 2010.

Zásady pro vypracování

Student nejdříve nastuduje a vyhodnotí stochastické metody optimalizace vhodné k implementaci v rámci stratifikované simulace typu Monte Carlo. Práce je založena na využití pravděpodobnostního software FREET. Budou ověřeny možnosti implementace některých metod, vybraná metoda bude dle možností naprogramována a částečně implementována do softwarového systému. Ověření bude nejdříve provedeno na jednoduchých optimalizačních příkladech, pak aplikované rovněž na konkrétní betonovou konstrukci řešenou programovým systémem SARA - ATENA.

Předepsané přílohy

Licenční smlouva o zveřejňování vysokoškolských kvalifikačních prací



prof. Ing. Drahomír Novák, DrSc.
Vedoucí bakalářské práce

ABSTRAKT

Práce si klade za cíl přiblížit čtenáři smysl optimalizace a její význam pro stavební praxi. Naznačuje principy některých používaných optimalizačních metod a zabývá se možností spojení některého z typů metody LHS s jinou optimalizační metodou. Výsledkem tohoto spojení je nový typ optimalizační metody pracovně pojmenovaný Nested LHS, popsáný v textu třetí kapitoly. V páté kapitole se aplikují některé ze získaných poznatků na řešení praktické optimalizační úlohy – železobetonové mostní konstrukce řešené nelineární metodou konečných prvků s užitím pseudostochastické optimalizace metodou LHS mean.

KLÍČOVÁ SLOVA

Optimalizace, Simulované žíhání, Genetické algoritmy, LHS, Nested LHS, Monte Carlo, Latin Hypercube Sampling, Optimalizační metody, Nelineární řešení, Metoda konečných prvků

ABSTRACT

The thesis focuses the reader on the sense of optimization and its importance for civil engineering. It outlines the principles used by some optimization methods and discusses the possibility of combination of any type of methods LHS with other optimization method. The result is a new type of optimization method named Nested LHS described in the text of the third chapter. The fifth chapter applies some of the lessons learned to solve practical optimization problem - reinforced concrete bridge solved by nonlinear finite element analysis using pseudostochastic optimization method LHS mean.

KEYWORDS

Optimization, Simulated Annealing, Genetic Algorithms, LHS, LHS Nested, Monte Carlo, Latin Hypercube Sampling, Optimization Methods, Nonlinear Solution, Finite Element Method

BIBLIOGRAFICKÁ CITACE

Slowik, O. Optimalizace betonových konstrukcí stochastickými metodami optimalizace. Brno: Vysoké učení technické v Brně, Fakulta stavební, 2012. 69 s. Vedoucí diplomové práce prof. Ing. Drahomír Novák, DrSc.

Prohlášení:

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně a že jsem uvedl všechny použité informační zdroje.

V Brně dne 23.5. 2012

.....

Ondřej Slowik

PROHLÁŠENÍ O SHODĚ LISTINNÉ A ELEKTRONICKÉ FORMY VŠKP

Prohlašuji, že elektronická forma odevzdané bakalářské práce je shodná s odevzdanou listinnou formou.

V Brně dne 23. 5. 2012

Ondřej Slowik

PODĚKOVÁNÍ

Mé poděkování patří prof. Ing. Drahomíru Novákovi, DrSc. za cenné rady a doporučení během tvorby této práce. Dále pak škole, jež mi poskytla zázemí a potřebné prostředky a v neposlední řadě rodině a známým, kteří mne po celou dobu studia podporovali.

OBSAH

BRNO UNIVERSITY OF TECHNOLOGY	1
Úvod	11
1. Optimalizace	12
1.1 Metody deterministické	12
1.2 Metody stochastické	13
1.3 Pravděpodobnostní optimalizace	13
2. Vybrané stochastické metody	14
2.1 Historie	14
2.2 Metody typu Monte Carlo	15
2.3 Metody LHS (Latin Hypercube Sampling)	16
2.3.1 Lattice sampling by Patterson (LHS median)	16
2.3.2 LHS-random from here on	16
2.3.3 LHS mean	17
2.4 Metody evolučních strategií (genetické algoritmy)	17
2.5 Metoda simulovaného žíhání (simulated annealing)	19
3. Výběr vhodné kombinace metod pro small sample analysis (SSA)	21
3.1 Testovací funkce	21
3.1.1 Six-hump camelback function	21
3.1.2 Branins's rcos function	22
3.1.3 Goldstein-price's function	23
3.1.4 Easom's function	24
3.2 Srovnání optimalizačních metod	25
3.2.1 Kombinace LHS a simulovaného žíhání	25
3.2.2 Nested LHS	34
3.2.3 Vyhodnocení	43
4. Programové prostředky	43
4.1 Freet	43
4.2 Atena	44
4.3 Sara	44
5. Praktická úloha	44
5.1 Zadání	45
5.2 Řešení	47
5.2.1 Tvorba idealizovaného 2D modelu	47
5.2.2 Randomizace parametrů	48
5.2.3 Funkce ceny materiálu	51

5.2.4	Posouzení vygenerovaných variant	53
5.2.5	Vyhodnocení.....	56
Závěr		60
Seznam použitých zkratek a symbolů		62
Seznam příloh		63

Úvod

Efektivní využívání zdrojů a vynaložených prostředků je základním předpokladem úspěchu lidské činnosti. Rozvoj konkurenčního prostředí zapříčinil zvýšení požadavků na kvalitu výroby za současné minimalizace výrobních nákladů. Nalezení optimálního řešení průmyslových problémů, jež splňuje požadavky na kvalitu s minimálními náklady na výrobu, se stalo hlavním cílem většiny výrobců.

Současná matematika umožňuje modelovat celou řadu reálných problémů a procesů. Optimalizací takového matematického modelu je možné teoreticky nalézt nejlepší konfiguraci vstupních hodnot vedoucích k optimálnímu řešení ještě před vlastní výrobou. Problém představuje samotné nalezení optimálního řešení. Modelované problémy se podle své složitosti dělí na deterministické (snadno řešitelné deterministickými metodami optimalizace) a nedeterministické polynomiální problémy (v současnosti řešené heuristicky stochastickými optimalizačními metodami). Dobře známým nedeterministickým polynomiálním (dále jen NP) problémem je například tzv. problém obchodního cestujícího, jehož laická formulace zní:

Mějme n měst (uzlů) vzájemně spojených silnicemi o známých délkách. Úkolem je nalézt nejkratší cestu procházející všemi městy a končící ve výchozím bodě.

V současné době existuje jen jeden způsob nalezení správného řešení – prozkoumání všech možných tras. Nedostatkem takového postupu je, že časová náročnost výpočtu narůstá velice rychle s rostoucím počtem měst a problém se stává výpočetně extrémně náročný již při řádově desítkách uzlů. Analogické problémy lze nalézt v mnoha vědních a průmyslových oborech. Příkladem může být řešení ideální architektury procesorů, stanovení optimálního množství skladovaných výrobků či nalezení nejkratší trasy v navigačních systémech.

Stavebnictví je průmysl zabývající se individuální výrobou mnohdy velice složitých a nákladných konstrukcí. Vzhledem k ohromnému objemu investovaných prostředků je zde potřeba maximální efektivity vynaložených zdrojů znatelná. Ve stavební praxi převažoval dříve deterministický přístup k navrhování konstrukcí. V takovém případě záleželo na zkušenosti, úsilí, ale někdy i na náhodě, zda daná konstrukce bude zcela využita či nehospodárně předimenzována. Rozvoj výpočetní techniky umožnil modelovat konstrukce metodou konečných prvků, jež umožňuje řešit složité modely závislé na velkém množství vstupních parametrů se složitými vzájemnými vztahy. Vyčíslení takovýchto vztahů je samo o sobě výpočetně náročné a stanovení optimální kombinace vstupních hodnot tak, aby daná konstrukce plnila účel, k němuž byla navržena s minimálními náklady na její realizaci, je mnohdy problémem srovnatelným svou složitostí s problémem obchodního cestujícího. Stochastické optimalizační metody nám umožní „zrychlení“ heuristického řešení, je však nutné vzdát se nároku na přesné vyčíslení funkčního extrému a spokojit se s výsledkem, jež extrému s určitou pravděpodobností odpovídá. U složitých problémů běžných ve stavební praxi navíc není z časových důvodů možné provádět řádově tisícovky iterací, jež potřebují některé optimalizační metody (simulované žíhání, genetické algoritmy) ke spolehlivému nalezení extrému. Činnost projektanta se proto soustřeďuje na nalezení co nejlepšího řešení s řádově desítkami iterací. Tento postup se nazývá analýza s malým počtem vzorků (small sample analysis) a je předmětem této práce.

1. OPTIMALIZACE

Optimalizace funkce (matematického modelu) znamená nalezení minima či maxima této funkce. Je zřejmé, že abychom mohli problém optimalizovat, musíme nejdříve vytvořit matematický model tohoto problému a stanovit omezující podmínky optimalizace v podobě omezujících funkcí, jež definují stav, při němž je optimalizace přípustná.

Příklad optimalizace:

Mějme funkci závislosti ceny konstrukce na objemu a druhu použitého materiálu. Je nutné stanovit požadavky na konstrukci a z nich vyplývající omezující funkce (například se požaduje, aby geometrické parametry byly kladné a v rámci reálných mezí). Tyto omezující podmínky stanoví hranice přípustné minimalizace objemu konstrukce (či jiného parametru). Je tedy v našem zájmu minimalizovat cenu tak, aby konstrukce stále plnila účel, k němuž je navržena. Podobně lze optimalizovat libovolný parametr konstrukce.

Optimalizační metody můžeme rozdělit na deterministické a stochastické.

1.1 METODY DETERMINISTICKÉ

Modely optimalizované deterministickými metodami mají předvídatelný výsledek a jednotlivé proměnné vstupující do modelu a vztahy mezi nimi jsou deterministicky určeny (pevně dány). Tyto modely nezohledňují náhodnost proměnných vstupujících do modelu či náhodnost vztahů samotných.

Deterministickou optimalizaci můžeme formulovat takto:

Minimalizujme:

$$f_{(X)}, \quad X \in E_n \quad (1.1.1)$$

Za předpokladu existence omezujících podmínek:

$$c_{i(X)} \geq 0, \quad i = 1, 2, \dots, m \quad (1.1.2)$$

$f_{(X)}$ je cílovou funkcí vektoru X složeného z n návrhových proměnných $X=(x_1;x_2;\dots;x_n)$. Ani jedna z návrhových proměnných není náhodná. Funkce $c_{i(X)}$ vyjadřují omezující podmínky optimalizace. Popisují fyzikální realitu existence proměnných a vazeb mezi nimi (např. geometrické parametry musí být kladné a odpovídající reálným mezím).

Deterministické optimalizační metody lze rozdělit v závislosti na druhu informace, kterou potřebují na:

- metody vyhledávací, vyžadující pouze hodnoty cílové funkce $f(x)$
- gradientní metody, využívající první parciální derivace $f_{(x)}$ (tzv. strategie prvního řádu)
- Newtonovy metody, využívající také druhé parciální derivace $f_{(x)}$ (tzv. strategie druhého řádu)

Nevýhodou výše uvedených deterministických metod je, že žádná z nich nevede s absolutní jistotou k nalezení globálního minima optimalizované funkce.

1.2 METODY STOCHASTICKÉ

Modely optimalizované stochastickými metodami obsahují jednu nebo více proměnných či vztahů, jež mají charakter náhodné veličiny či náhodného procesu. Tyto proměnné se generují náhodně na základě jejich pravděpodobnostního rozdělení. Modely pracující s náhodnými proměnnými či operátory lépe odpovídají reálným procesům, v nichž je nahodilá složka většinou přítomna. Takové aproximace pak s určitou pravděpodobností odpovídají reálnému stavu.

Stochastické metody jsou založeny na iteračních postupech, kdy jsou generovány náhodné vektory $X (x_1; x_2; \dots; x_n)$, na jejichž základě jsou vyčíslovány funkční hodnoty $f_{(x)}$. Tyto funkční hodnoty jsou dále zpracovávány běžnými postupy matematické statistiky či speciálními porovnávacími algoritmy (viz dále). Výpočtová náročnost stochastického modelu proto roste s počtem náhodných proměnných (náhodných procesů) a se složitostí vztahů mezi nimi. Současná výpočetní technika umožňuje provádění ohromného počtu operací v krátkém čase, stále však existují úlohy, jejichž optimalizace je časově velmi náročná. Mezi tyto patří například optimalizace modelů založených na metodě konečných prvků, jež pracuje s modely s velkým počtem náhodných veličin a složitými vzájemnými vztahy.

Příklady stochastických metod:

- metody typu Monte Carlo
- metody evolučních strategií
- metoda simulovaného žíhání
- metody typu LHS (Latin Hypercube Sampling)
- pokročilé simulační metody (Importance sampling, Adaptive sampling, Directional sampling či Line sampling)

Některým z těchto metod se věnuje kapitola 2.

1.3 PRAVDĚPODOBNOSTNÍ OPTIMALIZACE

Stochastickou optimalizační úlohu se zohledněním pravděpodobnosti poruchy můžeme formulovat:

1. Minimalizujeme funkci $f(x)$ za předpokladu existence omezujících podmínek (1.1.2) a podmínky:

$$p_{f(x)} \leq p_{f,d} \quad (1.2.1)$$

Kde $p_{f(x)}$ je teoretická pravděpodobnost poruchy a $p_{t,d}$ je návrhová pravděpodobnost poruchy. Tento zápis je sám jednou z omezujících podmínek $c_{i(x)}$ (1.1.2) a lze přepsat:

$$c_{i(x)} = p_{t,d} - p_{f(x)} \quad (1.2.2)$$

2. Minimalizujeme přímo teoretickou pravděpodobnost poruchy p_f , při zohlednění omezujících podmínek (1.1.2).

Pravděpodobnostní optimalizace je nejpokročilejší formou posuzování spolehlivosti stavebních konstrukcí

2. VYBRANÉ STOCHASTICKÉ METODY

2.1 HISTORIE

Pravděpodobně první popsáný případ užití stochastické metody pro řešení matematického problému se datuje do roku 1777 (problém Buffonovy jehly). Francouzský matematik Georges-Louis Leclerc Comte de Buffon se pokoušel odhadnout hodnotu Ludolfova čísla π náhodným vrháním jehly na linkovaný papír. Pravděpodobnost jevu, kdy jehla délky rovné vzdálenosti mezi linkami po dopadu na papír zůstane ležet na papíře tak, že protíná některou z linek, je rovna $p = \frac{2}{\pi}$. Po statistickém vyhodnocení počtu úspěšných a neúspěšných realizací tohoto náhodného jevu bylo možné určit přibližně hodnotu čísla π . Přesnost takto získané hodnoty narůstá s rostoucím počtem realizací náhodného jevu.

Tato metoda byla poprvé systematicky použita kolem roku 1930 Enrico Fermim při generování náhodných čísel k výpočtu vlastností neutronu. Později byla pojmenována Stanislawem Ulamem Monte Carlo. Rozvoj Výpočetní techniky během 20. století vytvořil prostor pro zdokonalování a vývoj optimalizačních algoritmů. V 60. letech byly vytvořeny teoretické předpoklady k optimalizačním metodám napodobujícím přírodní vývoj tzv. metody evolučních strategií. Na konci 70. let se objevila Metoda Latin Hypercube Sampling (dále jen LHS). V 80. letech představili nezávisle na sobě Scott Kirkpatrick, C. Daniel Gelatt, Mario P. Vecchi (1983) a Vlado Černý (1985) metodu Simulovaného žíhání (simulate annealing) [9]. V průběhu století byly rozvíjeny i simulační algoritmy vycházejících z metody Monte Carlo. Došlo k vytvoření celé řady pokročilých simulačních metod jako: Importance sampling, Adaptive sampling, Directional sampling či Line sampling. [9]

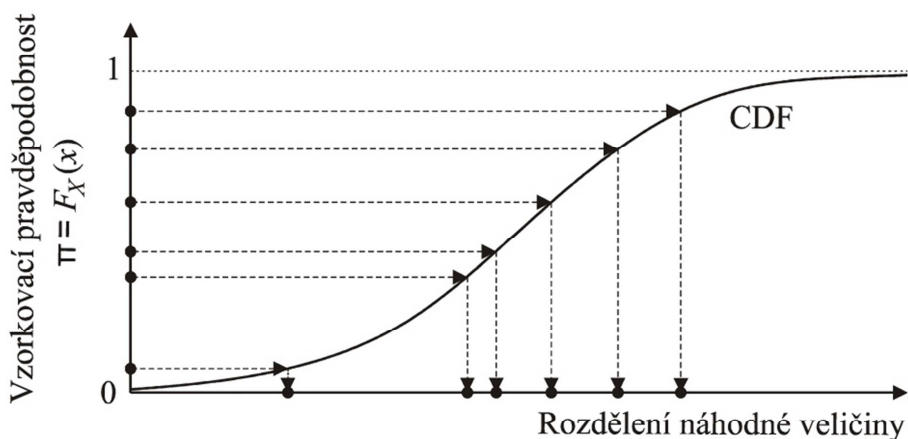
2.2 METODY TYPU MONTE CARLO

Pro účely numerické simulace pomocí některého z typů metody Monte Carlo je nutné nadefinovat matematický model problému a zajistit generování proměnných hodnot náhodných veličin v závislosti na jejich teoretickém modelu rozdělení pravděpodobnosti. K tomu je nezbytné užití generátoru náhodných čísel rovnoměrně rozdělených na intervalu (0;1) (generátory náhodných čísel na intervalu 0 až 1 jsou dnes standardně součástí mnoha programovacích jazyků). Na základě vygenerovaného pseudonáhodného čísla $u_{i,j}$ se generuje realizace náhodné veličiny s daným pravděpodobnostním rozdělením tak, že se hledá realizace x_i v j -té simulaci, pro niž platí, že funkční hodnota distribuční funkce jejího pravděpodobnostního rozdělení v x_i je rovna vygenerovanému pseudonáhodnému číslu $u_{i,j}$ (viz obrázek 2.2.1 a vztah 2.2.1). Vygenerovaná čísla nejsou zcela náhodná, neboť proces generování lze kdykoliv zopakovat se stejnými vstupními podmínkami a řada vygenerovaných čísel se bude s určitou periodou opakovat, hodnota této periody musí být tudíž vysoká. [1]

$$x_{i,j} = \Phi_{X_i}^{-1}(u_{i,j}), \quad (2.2.1)$$

kde $\Phi_{X_i}^{-1}(u_{i,j})$ je inverzní distribuční funkce náhodné veličiny X_i .

Na základě takto vygenerovaných náhodných vektorů $X (x_1; x_2; \dots; x_n)$ se vypočtou funkční hodnoty $f(x)$. Soubor takto získaných funkčních hodnot je zpracován metodami matematické statistiky.

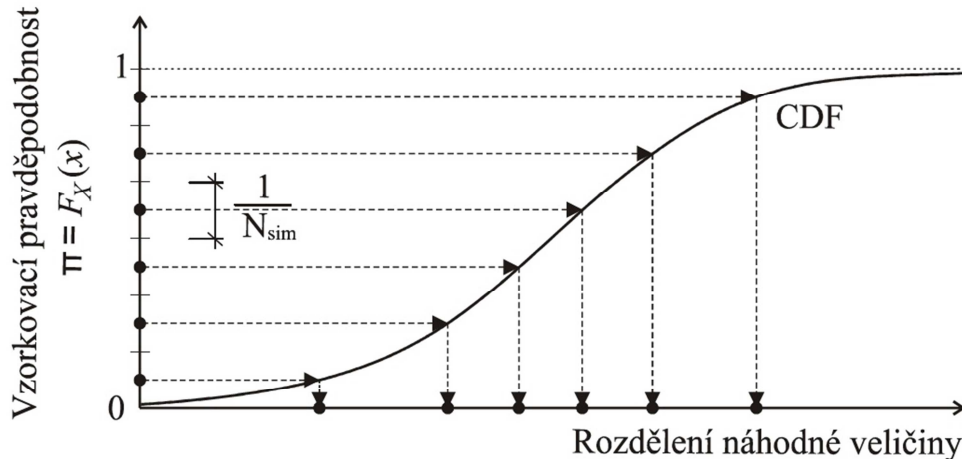


Obr. (2.2.1) Princip výběru realizací náhodných veličin metodou Monte Carlo [6]

Výhodou těchto metod je jejich transparentnost a srozumitelnost. Nevýhodou pak může být vysoký počet iterací, jež je činí obtížně použitelné pro optimalizaci výpočetně náročných úloh.

2.3 METODY LHS (LATIN HYPERCUBE SAMPLING)

LHS jsou speciálními typy simulací Monte Carlo využívající pravidelné rozvrstvení oboru hodnot pravděpodobnostní distribuční funkce náhodné proměnné na intervaly se stejnou pravděpodobností $1/N_{sim}$, kde N_{sim} je počet realizací náhodné veličiny (viz obrázek 2.3.1).



Obr. (2.3.1) Princip výběru realizací náhodných veličin metodou LHS [6]

Z každého sub intervalu spojitě distribuční funkce ($j = 1, \dots, N_{sim}$) je vybrána právě jedna realizace náhodné proměnné $x_{i,j}$. LHS zajišťuje rovnoměrné pokrytí prostoru s mnoha náhodnými proměnnými pouze s minimálním počtem vzorků. V závislosti na způsobu výběru hodnoty $x_{i,j}$ ze sub intervalu rozlišujeme několik typů LHS:

2.3.1 LATTICE SAMPLING BY PATTERSON (LHS MEDIAN)

Realizace náhodné proměnné $x_{i,j}$ je zde generována jako:

$$x_{i,j} = F_i^{-1} \left(\frac{\pi_i(j) - 0,5}{N_{sim}} \right) \quad (2.3.1)$$

Kde $\pi_i(1), \dots, \pi_i(N_{sim})$ je náhodná permutace z $1, \dots, N_{sim}$. F_i^{-1} je inverzní distribuční funkce této náhodné proměnné.

2.3.2 LHS-RANDOM FROM HERE ON

Tato metoda je téměř analogická s metodou (2.3.1) s tím rozdílem, že reprezentant není vybírán jako střed sub intervalu, ale může být vybrána jakákoli realizace náhodné veličiny s funkční hodnotou distribuční funkce v mezích sub intervalu. Toto se projeví na vztahu (2.3.1) výměnou hodnoty 0,5 za hodnotu U_j^i (náhodně vygenerované číslo z intervalu 0 až 1).

Nevýhodou centrické verze LHS je soustředění vzorků na středy sub intervalů to vede k redukci variability vzorků z okrajových částí (tails) funkce hustoty pravděpodobnosti (dále jen PDF- probability density function), které nejvíce ovlivňují šikmost a špičatost rozdělení veličin.

2.3.3 LHS MEAN

Tato metoda odstraňuje problém variability vzorků z okrajových částí PDF výběrem vzorku odpovídajících střední hodnotě sub intervalu PDF.

Realizace náhodných veličin jsou tak generovány podle vztahu:

$$x_{i,j} = \frac{\int_{\xi_{i,j-1}}^{\xi_{i,j}} x f_i(X) dx}{\int_{\xi_{i,j-1}}^{\xi_{i,j}} f_i(X) dx} = N_{sim} \int_{\xi_{i,j-1}}^{\xi_{i,j}} x f_i(X) dx \quad (2.2.2)$$

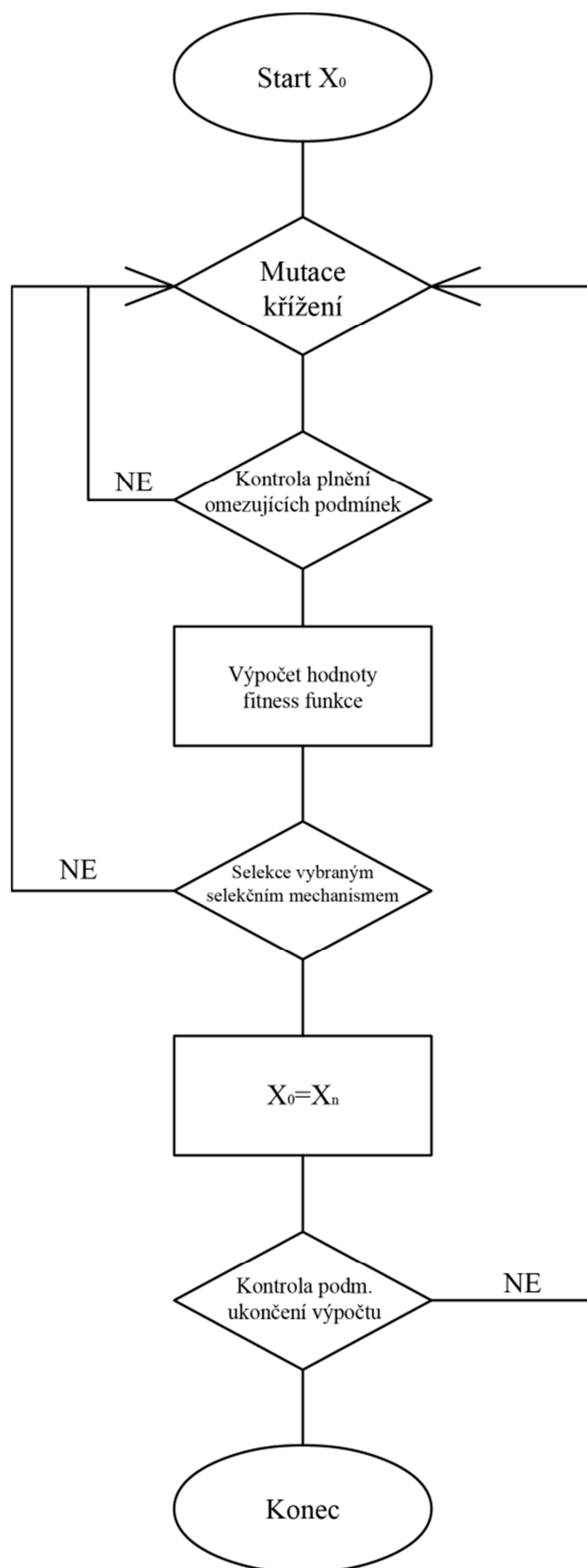
$f_i(X)$ je funkce hustoty pravděpodobnosti neznámé X_i . Hranice integrační oblasti $\xi_{i,j} = F_i^{-1}(j/N_{sim})$ pro $j=1, \dots, N_{sim}$

Tuto simulační metodu lze s výhodou používat u proměnných s funkcí hustoty pravděpodobnosti, jež je snadno integrovatelná. V případech, kdy je analytické řešení náročné nebo nemožné je nutné přistoupit k numerické integraci (tím vzroste výpočetní náročnost mnohdy nad únosnou mez). Vzorky vybírané dle vztahů (2.2.1) a (2.2.2) jsou téměř identické vyjma reprezentantů sub intervalů na krajích PDF.

Proti klasické metodě Monte Carlo poskytují metody LHS uspokojivou přesnost i při malém počtu simulací. Jsou proto vhodné pro optimalizaci výpočetně náročných problémů s malým množstvím provedených simulací (small sample analysis) běžných ve stavební praxi.

2.4 METODY EVOLUČNÍCH STRATEGIÍ (GENETICKÉ ALGORITMY)

Metody evolučních strategií využívají algoritmů, jež imitují přírodní výběr k heuristickému řešení výpočetně náročných problémů. Proces pracuje s náhodnými vektory X_o v tzv. generaci rodičů (první generaci X_{o1} je nutné zadat) a vektory X_n tzv. generace potomků. Tato je získána mutací či vzájemnou kombinací či křížením z generace rodičů. Pro náhodné vektory X_n je následně vypočtena jejich fitness funkce (určuje šanci vektoru X_n na přijetí do další generace). Poté je provedena selekce řešení podle zvoleného selekčního mechanismu (blíže ve [3]). Náhodné vektory X_n jež mají nejlepší předpoklady přežít (nejlepší hodnoty fitness funkce vybrané zvoleným selekčním mechanismem) jsou v následujícím kroku přijaty do generace rodičů. Celý proces se iterativně opakuje a řešení se postupně zlepšují, dokud není splněna podmínka ukončení výpočtu. Obecný algoritmus evolučních strategií je na obr. (2.4.1)



Obr. (2.4.1) algoritmus metody evolučních strategií

2.5 METODA SIMULOVANÉHO ŽÍHÁNÍ (SIMULATED ANNEALING)

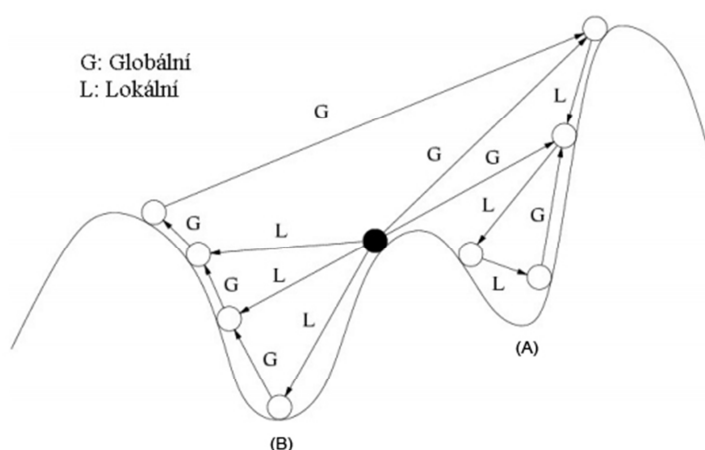
Metoda simulovaného žíhání je přibližný stochastický algoritmus vycházející z Boltzmannova pravděpodobnostního rozdělení (vztah 2.5.1).

$$P_r(E) \sim e^{(-E/k_B T)} \quad (2.5.1)$$

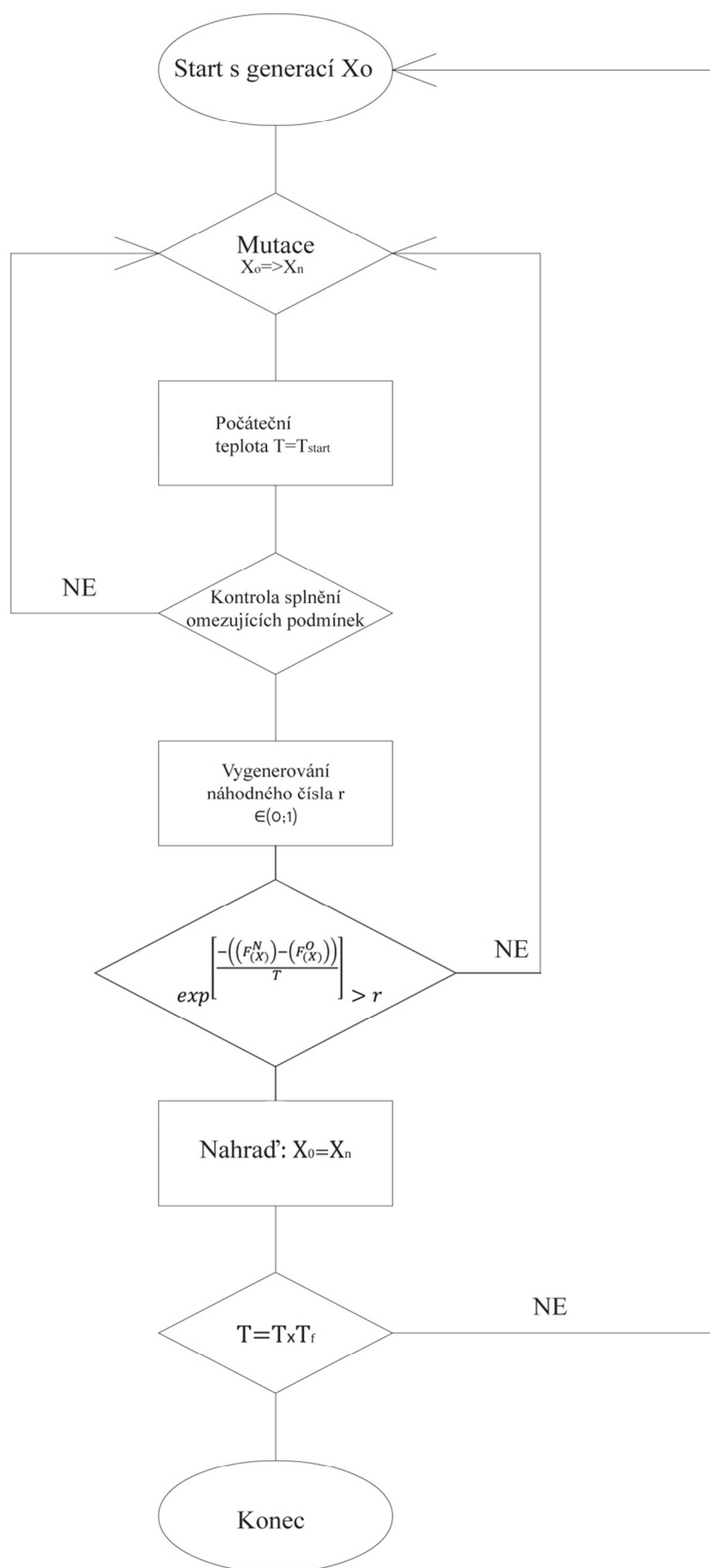
Kde E je energetická hladina, T je teplota systému a k_B je Boltzmannova konstanta jež zavádí souvislost mezi teplotou a energií.

Tento algoritmus byl převzat z metalurgie a vychází z předpokladu, že systém v teplotní rovnováze o dané teplotě T má veškerou svou tepelnou energii rozloženou pravděpodobnostně mezi všechny různé energetické hladiny E . S určitou pravděpodobností (velmi malou) může být systém i při nízkých teplotách lokálně ve vyšším energetickém stavu. Tuto vlastnost můžeme využít při rozhodování, zda nově vygenerovaný (některým z algoritmů mutace) náhodný vektor X_n přijmeme v následující generaci jako vektor rodičovský X_o či nikoli. Aplikace metody simulovaného žíhání umožňuje vyhledávacímu procesu „vyskočit“ z lokálního minima funkce a pokračovat směrem k minimu globálnímu.

Podstatu simulovaného žíhání lze demonstrovat na základě energetického grafu (obrázek. 2.5.1). Na obrázku je znázorněno minimum lokální (A) a globální (B). Najde-li vyhledávací algoritmus lokální minimum, pak v případě deterministických metod v něm vyhledávání většinou končí. U metody simulovaného žíhání existuje určitá pravděpodobnost (daná vztahem 2.5.1), že vyhledávání „vyskočí“ z minima lokálního a nasměruje se k minimu globálnímu. Systém však musí být dostatečně excitován (mít dost vysokou teplotu), aby energie potřebná k „vyskočení“ byla dostatečná. Algoritmus metody simulovaného žíhání je na obrázku (2.5.2).



Obr. (2.5.1) energetický graf



Obr. (2.5.2) algoritmus metody simulovaného žíhání

$F_{(x)}^N$ je funkční hodnota v nově vygenerovaném bodu návrhového prostoru, $F_{(x)}^O$ je funkční hodnota rodičovského bodu návrhového prostoru. Počáteční teplota T_{start} se podle [5] nastavuje heuristicky tak, aby po provedení $n \times 10$ iterací byl poměr úspěšných ku všem iteracím cca 0,5. Snížení teploty T teplotním redukčním faktorem T_f se provádí pravidelně po stanoveném počtu iterací.

3. VÝBĚR VHODNÉ KOMBINACE METOD PRO SMALL SAMPLE ANALYSIS (SSA)

3.1 TESTOVACÍ FUNKCE

Pro ověření účinnosti testovaných optimalizačních metod je nutné zvolit vhodné testovací funkce. Vybrané funkce by měly být spojité na svém definičním oboru, dobře popsané se známou hodnotou extrémů. Pro dobrou představu o průběhu funkce a možnost její vizualizace je vhodné volit testovací funkce jako dvourozměrné. Přílišná jednoduchost funkcí by však snižovala vypovídací hodnotu provedených pokusů. Kvůli možnosti srovnání jednotlivých optimalizačních metod je vhodné, aby se vybrané funkce vzájemně typově odlišovaly (optimalizační metody mohou vykazovat rozdílnou účinnost pro různé typy funkcí). Pro účely testování byly zvoleny tyto testovací funkce:

- Six-hump camel back function
- Branins's rcos function
- Goldstein-Price's function
- Easom's function

3.1.1 SIX-HUMP CAMELBACK FUNCTION

Je jednoduchá funkce dvou proměnných. V ohraničené oblasti se nachází šest lokálních extrémů (minima) z toho dva jsou zároveň extrémy globálními. Tato funkce je vhodná pro ověření schopnosti optimalizačního algoritmu najít globální minimum mezi minimy lokálními.

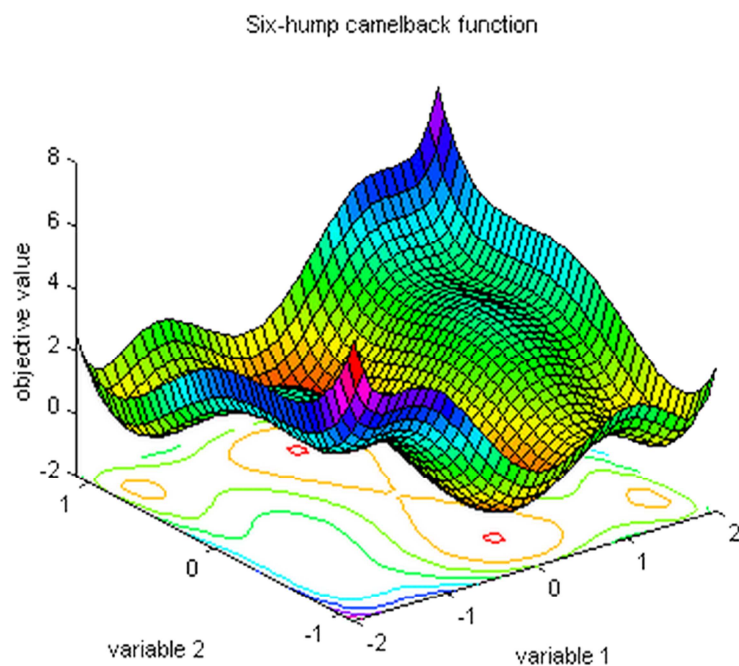
$$f_{(x,y)} = \left(4 - 2,1x^2 + x^{\frac{4}{3}}\right)x^2 + xy + (-4 + 4y^2)y^2 \quad (3.1.1)$$

Definiční obor:

$$x \in (-3; 3), y \in (-2; 2)$$

Globální minima:

$$f_{(x,y)} = -1,0316; (x,y) = (-0,0898; 0,7126), (0,0898; -0,7126)$$



Obr. (3.1.1) Six-hump camelback function [4]

3.1.2 BRANINS'S RCOS FUNCTION

Je funkcí dvou proměnných se třemi globálními minimy, běžně užívaná k testování optimalizačních algoritmů. Tato funkce nemá minima lokální. Je definována vztahem (3.2.1).

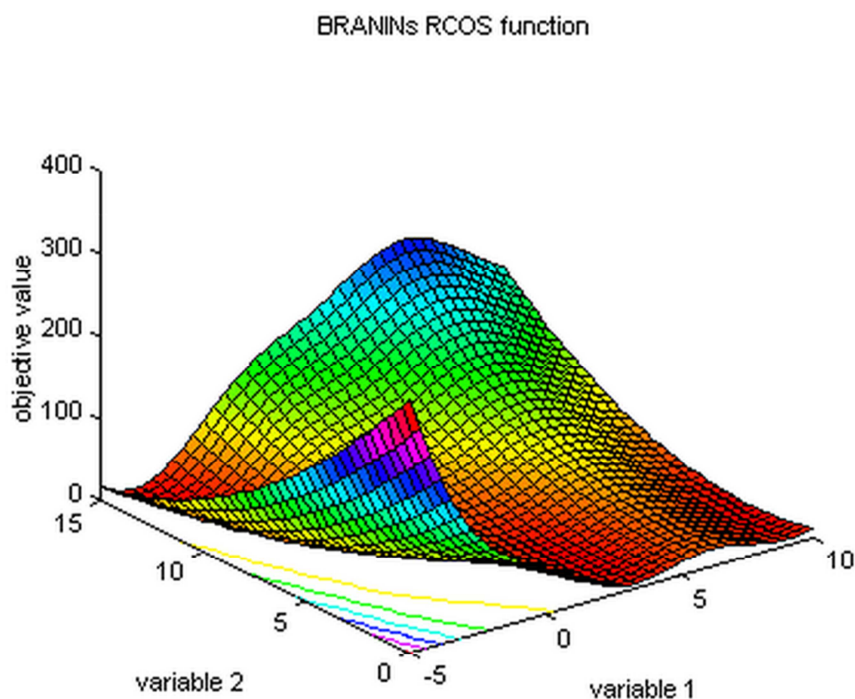
$$f_{(x,y)} = \left(y - \frac{5,1}{4\pi^2} y + \frac{5}{\pi} x - 6 \right) + 10 \left(1 - \frac{1}{8\pi} \right) \cos(x) + 10 \quad (3.2.1)$$

Definiční obor:

$$x \in (-5; 10), y \in (0; 15)$$

Globální minima:

$$f_{(x,y)} = 0,397887; (x,y) = (-\pi; 12,275), (\pi; 2,275), (9,42478; 2,475).$$



Obr. (3.2.1) Branins's rcos function [4]

3.1.3 GOLDSTEIN-PRICE'S FUNCTION

Je funkce dvou proměnných s jedním globálním minimem ve výrazně ploché oblasti údolí. Je charakteristická větším rozsahem oboru hodnot. Goldstein – Price's function je definována vztahem (3.1.3).

$$f_{(x,y)} = a \times b \quad (3.1.3)$$

$$a_{(x,y)} = [1 + (x + y + 1)^2(19 - 14x + 3x^2 - 6xy + 3y^2)]$$

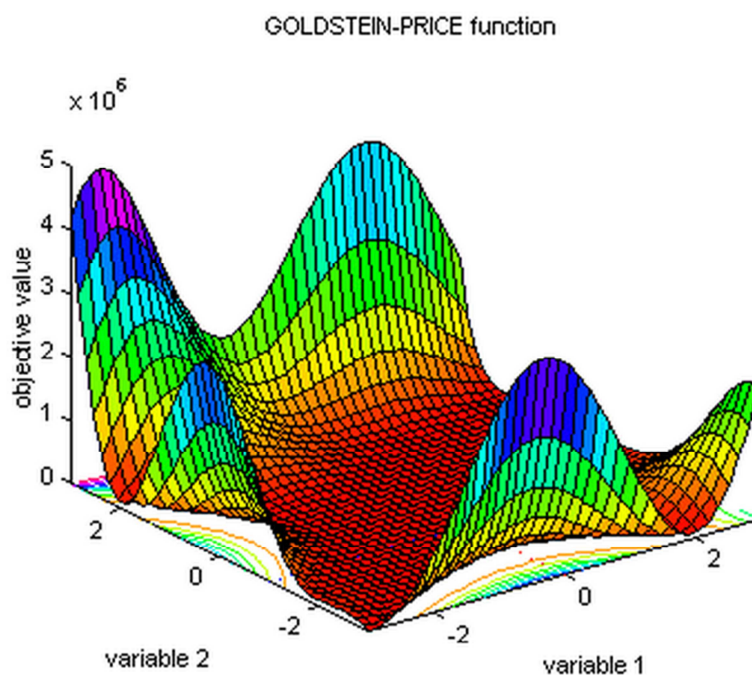
$$b_{(x,y)} = [30 + (2x - 3y)^2(18 - 32x + 12x^2 + 48y - 36xy + 27y^2)]$$

Definiční obor:

$$x \in (-2; 2), y \in (-2; 2)$$

Globální minimum:

$$f_{(x,y)}=3; (x,y)=(0;-1)$$



Obr. (3.1.3) Goldstein-Price's function [4]

3.1.4 EASOM'S FUNCTION

Je funkcí dvou proměnných s jedním globálním minimem soustředěným v poměrně malé oblasti vzhledem k definičnímu oboru. Pro některé typy optimalizačních metod (LHS) je proto nalezení jejího globálního minima nepravděpodobné. Funkce je definovaná vztahem (3.1.4).

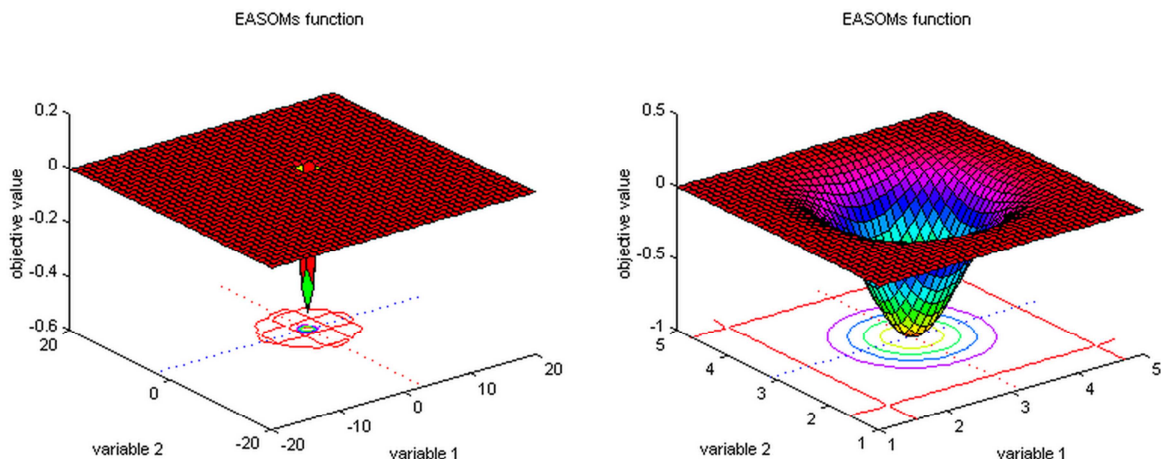
$$f_{(x,y)} = -\cos(x) \cos(y) \exp(-(x - \pi)^2 + (y - \pi)^2) \quad (3.1.4)$$

Definiční obor:

$$x \in (-100; 100), y \in (-100; 100)$$

Globální minimum:

$$f(x,y) = -1; (x,y) = (\pi, \pi).$$



Obr. (3.1.4) Easom's function [4]

3.2 SROVNÁNÍ OPTIMALIZAČNÍCH METOD

Metody typu LHS se ukázaly být vhodnější pro počáteční výběr vzorků k SSA než ostatní metody Monte Carlo, neboť vzorky generované LHS pokrývají návrhový prostor rovnoměrně a pravděpodobnost nalezení extrému je při malém počtu generovaných vzorků vyšší než u běžných typů metody Monte Carlo. Základní myšlenkou bylo využít některou z metod LHS pro vygenerování počáteční sady vzorků. Poté byl nejlepší vzorek přijat jako výchozí pro optimalizaci některým z vyhledávacích algoritmů (genetické algoritmy či simulované žíhání). Tento postup by měl zkrátit optimalizační proces na úroveň použitelnou ve stavební praxi (přijatelné jsou řádově stovky iterací). Další zvažovanou možností bylo použít „pseudooptimalizační” metodu pracovně pojmenovanou nested (vnořená) LHS. Tato metoda předpokládá vygenerování sady vzorků metodou LHS. Poté se zúží definiční obor okolo nejlepšího vzorku, jehož vektor $X(x_1, x_2, \dots, x_n)$ by představoval vektor středních hodnot pro generování další sady vzorků na zmenšeném definičním oboru. Tento postup je opakován až do nalezení uspokojivé minimální hodnoty. V rámci bakalářské práce byly srovnány dvě kombinace metod:

- metoda LHS zkombinovaná s metodou simulovaného žíhání
- tzv. metoda nested LHS, tedy kombinace LHS s postupným přesnějším cílením kolem nejlepšího vzorku

3.2.1 KOMBINACE LHS A SIMULOVANÉHO ŽÍHÁNÍ

U každé z testovacích funkcí bylo provedeno třicet simulací metodou LHS mean pomocí softwaru FReET. Z těchto vzorků byla vybrána nejlepší realizace náhodného vektoru $X(x, y)$ a tato byla stanovena jako počáteční bod pro start metody simulovaného žíhání. Pro aplikaci simulovaného žíhání byl použit algoritmus zapsaný v programu MATLAB [6]. Výsledný počet iterací nutný k nalezení globálního minima je závislý na kvalitě odhadu metodou LHS a na nastavení parametrů simulovaného žíhání. Zásadní vliv má především nastavení počáteční a konečné teploty. Interval mezi počáteční (T_{start}) a konečnou (T_{stop})

teplotou ovlivňuje schopnost algoritmu nalézt globální minimum i počet iterací nutných k nalezení optima. Neméně důležitým parametrem je i počet iterací s konstantní teplotou (tedy množství iterací, po kterém dojde ke snížení teploty T teplotním redukčním faktorem T_f). Vhodná kombinace těchto parametrů umožní algoritmu dosáhnout uspokojivého optima s množstvím iterací řádově v desítkách až stovkách. Nalezení vhodné kombinace parametrů je obtížná úloha řešená převážně heuristicky. V případě nesprávného nastavení vstupů je počet nutných iterací příliš vysoký nebo nedojde k nalezení optima vůbec (malý interval mezi T_{start} a T_{stop}). Během testování se postupně snižovala teplota geometrickou řadou s kvocientem $q=0,1$ první člen měl hodnotu 1 a poslední $1e-11$. Současně byl sledován počet iterací nutných k nalezení minima funkce. Náhodné proměnné měly ve všech testovaných případech rektangulární rozdělení.

Ukázka algoritmu metody simulovaného žíhání zapsaného v programu MATLAB je v příloze P1. Program umožňuje sledovat výpočetní náročnost. V řádku závěrečné zprávy Number of function calls je zobrazen počet vyčíslení funkční hodnoty, jež je vhodným měřítkem výpočetní náročnosti. Při statistickém hodnocení kombinace metody LHS mean a simulovaného žíhání byla sledována závislost počtu iterací na zvolené počáteční teplotě.

LHS + SIMULOVANÉ ŽÍHÁNÍ SIX-HUMP CAMELBACK FUNCTION

Řádkový zápis funkce: $(4-2.1*x_1^2+x_1^4/3)*x_1^2+x_1*x_2+(-4+4*x_2^2)*x_2^2$

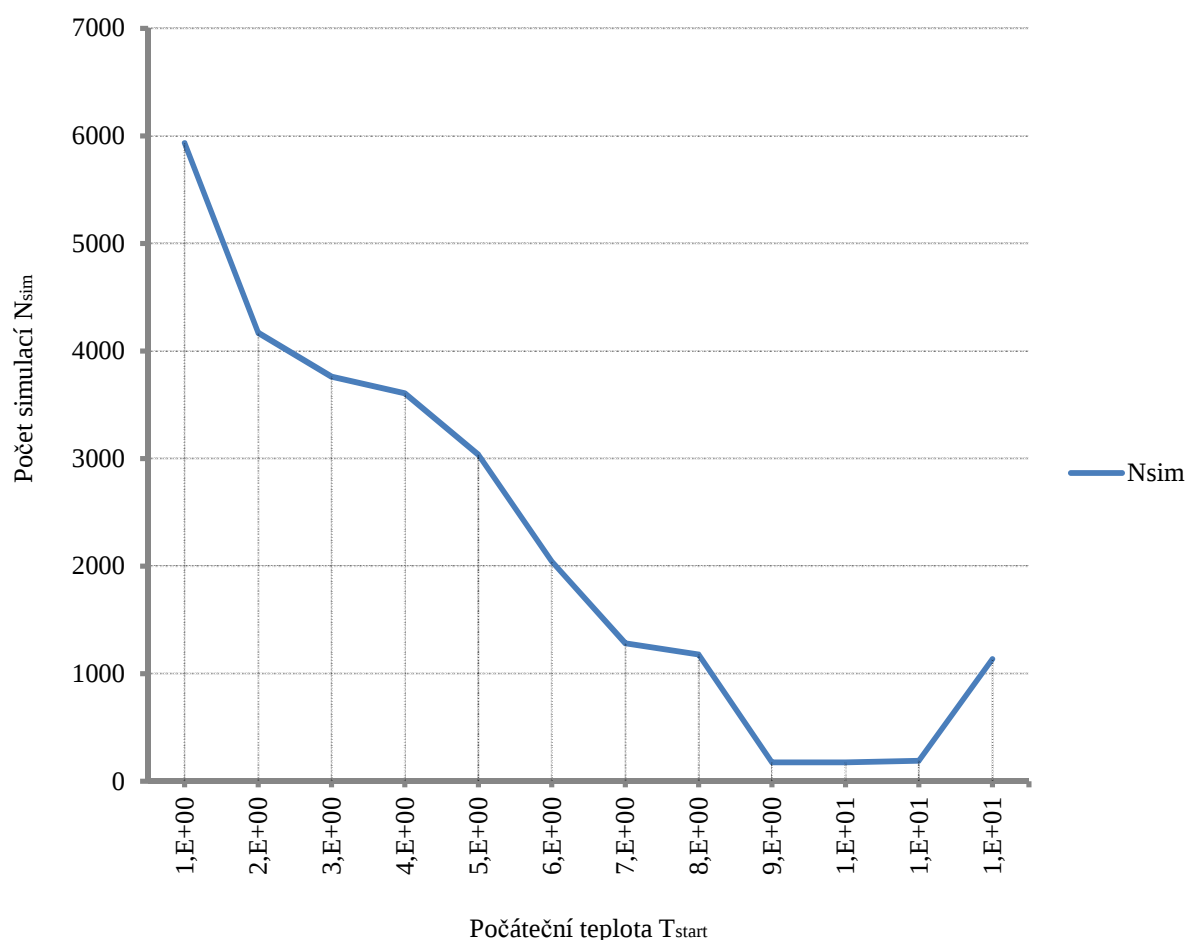
Výchozí bod po výběru 30 vzorků metodou LHS mean:

$$f(x) = -0,88117 \quad x_1 = -0,1 \quad x_2 = -0,73333$$

Tab. (3.2.1.1) Six-hump camelback function závislost počtu simulací na počáteční teplotě

Aplikace SA na min. po provedení LHS				T=1	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
5904	-1,03163	-0,08980	0,71260		
Aplikace SA na min. po provedení LHS				T=0,1	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
4138	-1,03163	0,08980	-0,71280		
Aplikace SA na min. po provedení LHS				T=0,01	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
3732	-1,03163	0,09000	-0,71280		
Aplikace SA na min. po provedení LHS				T=0,001	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
3578	-1,03163	0,08960	-0,71260		
Aplikace SA na min. po provedení LHS				T=1e-4	T _{stop} =1e-8

N_{sim}	$f_{(x1-x2) \min}$	x_1	x_2		
3005	-1,03163	0,08980	-0,71260		
Aplikace SA na min. po provedení LHS				$T=1e-5$	$T_{stop}=1e-8$
N_{sim}	$f_{(x1-x2) \min}$	x_1	x_2		
2015	-1,03163	0,08970	-0,71260		
Aplikace SA na min. po provedení LHS				$T=1e-6$	$T_{stop}=1e-8$
N_{sim}	$f_{(x1-x2) \min}$	x_1	x_2		
1253	-1,03163	0,08980	-0,71270		
Aplikace SA na min. po provedení LHS				$T=1e-7$	$T_{stop}=1e-8$
N_{sim}	$f_{(x1-x2) \min}$	x_1	x_2		
1149	-1,03163	0,08980	-0,71260		
Aplikace SA na min. po provedení LHS				$T=1e-8$	$T_{stop}=1e-8$
N_{sim}	$f_{(x1-x2) \min}$	x_1	x_2		
145	-1,03158	0,08970	-0,71030		
Aplikace SA na min. po provedení LHS				$T=1e-9$	$T_{stop}=1e-9$
N_{sim}	$f_{(x1-x2) \min}$	x_1	x_2		
146	-1,03162	0,09060	-0,71320		
Aplikace SA na min. po provedení LHS				$T=1e-10$	$T_{stop}=1e-10$
N_{sim}	$f_{(x1-x2) \min}$	x_1	x_2		
162	-1,03163	0,08990	-0,71250		
Aplikace SA na min. po provedení LHS				$T=1e-11$	$T_{stop}=1e-11$
N_{sim}	$f_{(x1-x2) \min}$	x_1	x_2		
1106	-1,03163	0,08980	-0,71270		



Graf (3.2.1.1) Závislost počtu simulací na počáteční teplotě T_{start}

Z tabulky (3.2.1.1) a grafu (3.2.1.1) je patrné, že optimální nastavení teplot je přibližně: $T_{start}=1e-9$; $T_{stop}=1e-9$, kdy bylo pro nalezení minima potřeba pouhých 176 simulací (146+30 na výchozí LHS) tento nízký počet je však způsoben pozicí výchozího bodu pro simulované žíhání, jež je sama blízko hledanému optimu při opakování pokusu není dosažení srovnatelného výsledku pravděpodobné. Algoritmus simulovaného žíhání byl schopen u dané funkce nalézt minimum pro všechny testované počáteční teploty.

LHS+SIMULOVANÉ ŽÍHÁNÍ BRANIN'S RCOS FUNCTION

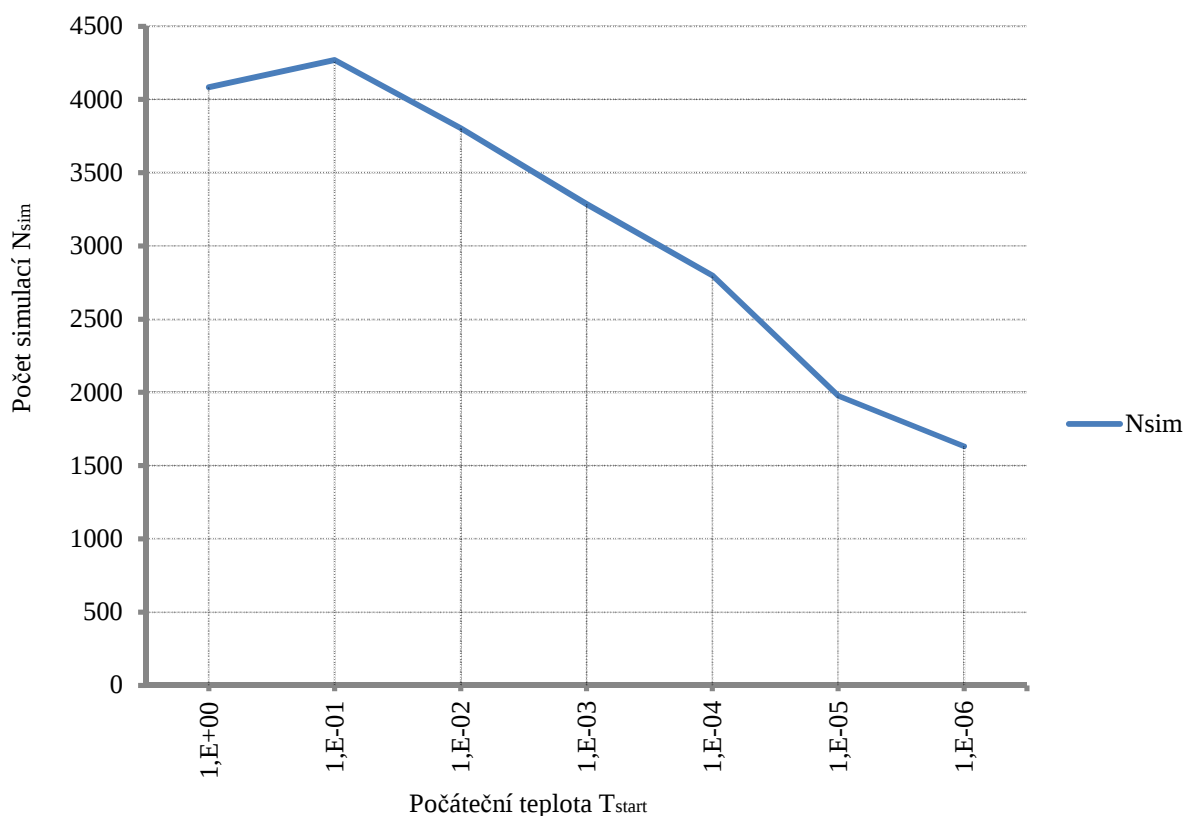
Řádkový zápis funkce: $(x_2 - (5.1/(4 \cdot \pi^2)) \cdot x_1^2 + 5 \cdot x_1/\pi - 6)^2 + 10 \cdot (1 - 1/(8 \cdot \pi)) \cdot \cos(x_1) + 10$

Výchozí bod po výběru 30 vzorků metodou LHS mean:

$$f(x) = 2,5335 \quad x_1 = 3,25 \quad x_2 = 0,75$$

Tab. (3.2.1.2) Branin's rcos function závislost počtu simulací na počáteční teplotě

Aplikace SA na min. po provedení LHS				T=1	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
4054	0,39790	3,14170	2,27490		
Aplikace SA na min. po provedení LHS				T=0,1	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
4240	0,39790	3,14170	2,27510		
Aplikace SA na min. po provedení LHS				T=0,01	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
3775	0,39790	3,14160	2,27480		
Aplikace SA na min. po provedení LHS				T=0,001	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
3256	0,39790	3,14170	2,27500		
Aplikace SA na min. po provedení LHS				T=1e-4	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
2769	0,39790	3,14150	2,27500		
Aplikace SA na min. po provedení LHS				T=1e-5	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
1947	0,39790	3,14140	2,27510		
Aplikace SA na min. po provedení LHS				T=1e-6	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
1601	0,39790	3,14160	2,27500		
Aplikace SA na min. po provedení LHS				T=1e-7	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
640	0,48150	3,18510	1,96840		
Aplikace SA na min. po provedení LHS				T=1e-8	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
115	1,89160	3,31910	0,98180		
Aplikace SA na min. po provedení LHS				T=1e-9	T _{stop} =1e-9
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
94	2,02430	3,32620	0,92580		
Aplikace SA na min. po provedení LHS				T=1e-10	T _{stop} =1e-10
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
92	2,05950	3,33100	0,91130		
Aplikace SA na min. po provedení LHS				T=1e-11	T _{stop} =1e-11
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
102	2,04480	3,32940	0,91730		



Graf (3.2.1.2) Závislost počtu simulací na počáteční teplotě T_{start}

Nejllepší dosažené nastavení taplot ($T_{start}=1e-6$; $T_{stop}=1e-8$) zajišťující nalezení minima vyžadovalo 1631 simulací. V posledních pěti případech (v tabulce vyznačeny červenými odstíny) sice poklesl počet simulací, ale nedošlo k nalezení minima funkce. Při zvětšení rozsahu intervalu (T_{stop} ; T_{start}) došlo k opětovnému nárůstu N_{sim} nad hodnotu 1000, minimum však bylo nalezeno. Graf (3.2.1.2) zahrnuje jen případy, kdy bylo optimum úspěšně nalezeno.

LHS+SIMULOVANÉ ŽIHÁNÍ GOLDSTEIN-PRICE'S FUNCTION

Řádkový zápis funkce:

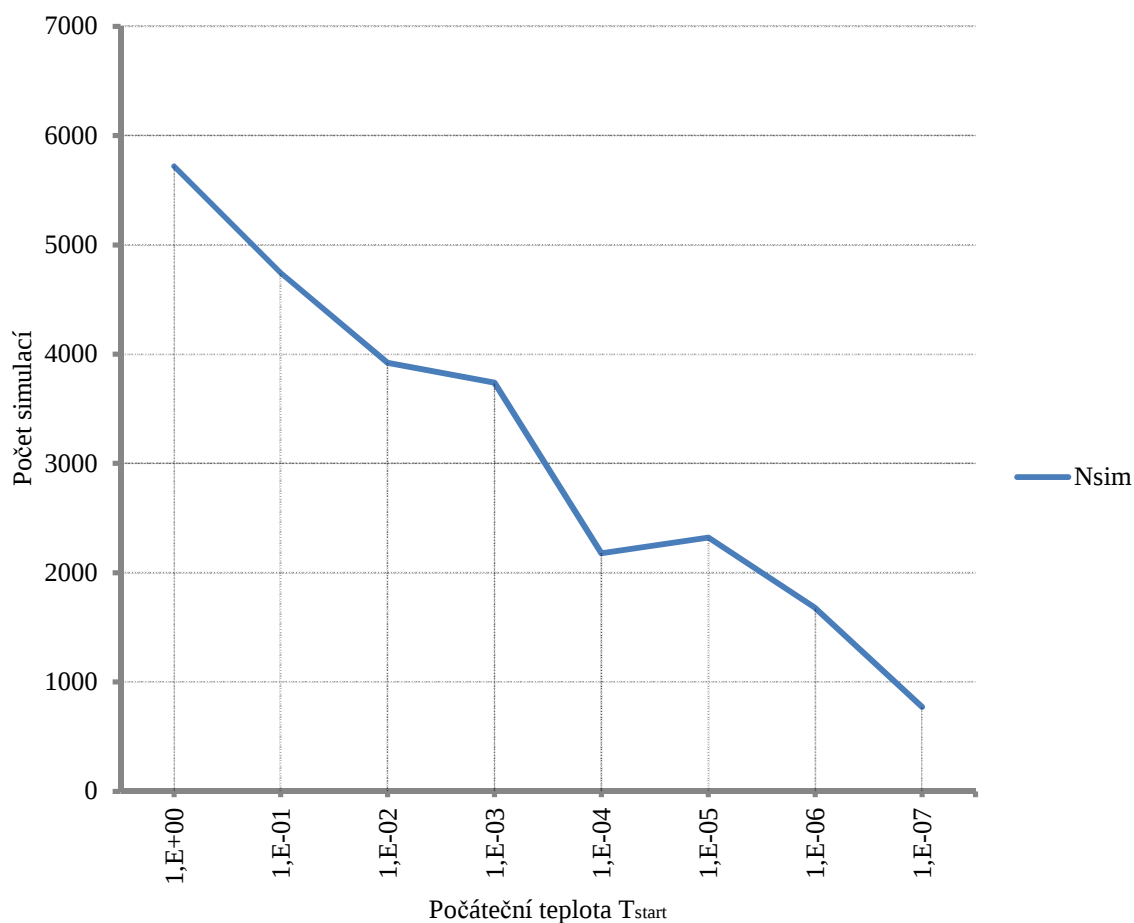
$$(1+(x_1+x_2+1)^2*(19-14*x_1+3*x_1^2-14*x_2+6*x_1*x_2+3*x_2^2))*(30+(2*x_1-3*x_2)^2*(18-32*x_1+12*x_1^2+48*x_2-36*x_1*x_2+27*x_2^2))$$

Výchozí bod po výběru 30 vzorků metodou LHS mean:

$$f(x) = 78,365 \quad x_1 = 0,73333 \quad x_2 = -0,46667$$

Tab. (3.2.1.3) Goldstein-Price's function závislost počtu simulací na počáteční teplotě

Aplikace SA na min. po provedení LHS				T=1	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
5688	3,00000	0,00000	-1,00000		
Aplikace SA na min. po provedení LHS				T=0,1	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
4713	3,00000	0,00000	-1,00000		
Aplikace SA na min. po provedení LHS				T=0,01	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
3892	3,00000	0,00000	-1,00000		
Aplikace SA na min. po provedení LHS				T=0,001	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
3709	3,00000	0,00000	-1,00000		
Aplikace SA na min. po provedení LHS				T=1e-4	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
2149	3,00000	0,00000	-1,00000		
Aplikace SA na min. po provedení LHS				T=1e-5	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
2292	3,00000	0,00000	-1,00000		
Aplikace SA na min. po provedení LHS				T=1e-6	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
1647	3,00000	0,00000	-1,00000		
Aplikace SA na min. po provedení LHS				T=1e-7	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
742	3,00000	0,00000	-1,00000		
Aplikace SA na min. po provedení LHS				T=1e-8	T _{stop} =1e-8
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
93	63,54860	0,62140	-0,57770		
Aplikace SA na min. po provedení LHS				T=1e-9	T _{stop} =1e-9
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
95	60.5134	0,60100	-0,59530		
Aplikace SA na min. po provedení LHS				T=1e-10	T _{stop} =1e-10
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
116	66,93950	0,65910	-0,56000		
Aplikace SA na min. po provedení LHS				T=1e-11	T _{stop} =1e-11
N _{sim}	f _{(x1-x2) min}	x ₁	x ₂		
125	65,54120	0,64810	-0,57590		



Graf (3.2.1.3) Závislost počtu simulací na počáteční teplotě T_{start}

U této testovací funkce vykazoval algoritmus simulovaného žíhání podobnou účinnost jako u Branin's rcos function. Nejlepšího výsledku bylo dosaženo při nastavení teplot ($T_{start}=1e-7$; $T_{stop}=1e-8$) kdy bylo na nalezení minima potřeba 772 simulací.

LHS+SIMULOVANÉ ŽÍHÁNÍ EASOM'S FUNCTION

Řádkový zápis funkce: $(x_2 - (5.1/(4 \cdot \pi^2)) \cdot x_1^2 + 5 \cdot x_1 / \pi - 6)^2 + 10 \cdot (1 - 1/(8 \cdot \pi)) \cdot \cos(x_1) + 10$

Výchozí bod po výběru 30 vzorků metodou LHS mean:

$$f(x)=0 \quad x_1=-3,3333 \quad x_2=-3,3333$$

Tab. (3.2.1.4) Easom's function závislost počtu simulací na počáteční teplotě

Aplikace SA na min. po provedení LHS				T=1	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
1681	0,00000	-2,89520	-2,90000		
Aplikace SA na min. po provedení LHS				T=0,1	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
1481	0,00000	-3,66560	-3,28430		
Aplikace SA na min. po provedení LHS				T=0,01	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
1261	0,00000	-3,24430	-3,11450		
Aplikace SA na min. po provedení LHS				T=0,001	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
1061	0,00000	-3,09700	-3,00580		
Aplikace SA na min. po provedení LHS				T=1e-4	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
861	0,00000	-3,40130	-3,14460		
Aplikace SA na min. po provedení LHS				T=1e-5	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
641	0,00000	-3,26400	-3,40530		
Aplikace SA na min. po provedení LHS				T=1e-6	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
441	0,00000	-3,46570	-3,37710		
Aplikace SA na min. po provedení LHS				T=1e-7	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
241	0,00000	-3,32000	-3,45630		
Aplikace SA na min. po provedení LHS				T=1e-8	T _{stop} =1e-8
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
41	0,00000	-3,32870	-3,35730		
Aplikace SA na min. po provedení LHS				T=1e-9	T _{stop} =1e-9
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
41	0,00000	-3,38140	-3,41050		
Aplikace SA na min. po provedení LHS				T=1e-10	T _{stop} =1e-10
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
41	0,00000	-3,29700	-3,32260		
Aplikace SA na min. po provedení LHS				T=1e-11	T _{stop} =1e-11
N _{sim}	f _(x1-x2) min	x ₁	x ₂		
41	0,00000	-3,32720	-3,38410		

Easom's function je konstantní (až na minimální fluktuace) po celém svém definičním oboru vyjma malé oblasti okolo globálního minima. Metoda simulovaného žíhání po určitém množství neúspěšných iterací zastaví vyhledávání (například: algoritmus nenajde po 1000 iteracích lepší hodnotu => je pravděpodobné, že současná hodnota je hledaným optimem). Toto množství je možné nastavit. V případě, že je funkce na většině svého definičního oboru konstantní, je při jakémkoli nastavení vstupních parametrů simulovaného žíhání nalezení jejího optima nepravděpodobné. U většiny problémů z praxe navíc situaci komplikuje neznalost charakteristik zkoumané funkce. Účinné nastavení vstupních parametrů metody simulovaného žíhání je pak téměř nemožné.

3.2.2 NESTED LHS

Je nově navržená metoda využívající některý z typů metod LHS k výběru vzorků návrhového prostoru. Po vygenerování daného počtu vzorků se určí nejlepší z nich (vzorek s maximální či minimální funkční hodnotou podle toho jaké optimum požadujeme). Realizace náhodných veličin $x_1; x_2; \dots; x_n$ náhodného vektoru $X = (x_1; x_2; \dots; x_n)$ s nejlepší funkční hodnotou $f(x)$ jsou určeny jako střední hodnoty dále vygenerovaných náhodných veličin. Zároveň je omezen návrhový prostor (omezí se rozptyl náhodných veličin, nebo se stanoví přímo interval hodnot, z něhož jsou realizace vybírány). V každém dalším cyklu je „zacílení“ kolem nejlepšího bodu z cyklu minulého užší. Výběr metodou LHS zároveň zajistí rovnoměrné pokrytí nově omezeného návrhového prostoru. Výhodou této metody je minimální množství uživatelem zadávaných parametrů. Ty jsou redukovány pouze na volbu typu metody LHS (mean, median, atd.) a stanovení „rychlosti“ zužování výběrového prostoru. Výběrový prostor je vhodné zmenšovat geometrickou řadou, jejíž kvocient stanoví „rychlost“ zaměřování. Hodnota kvocientu by se navíc mohla během procesu měnit. Na začátku je vhodné volit kvocient vyšší (kvocient je vybírán z intervalu (0;1), neboť prostor výběru je třeba zúžit), což umožní algoritmu „vyskočit“ z případného lokálního minima podobně jako vyšší teplota u metody simulovaného žíhání. V průběhu výpočtu by se poté mohl kvocient snížit pro přesnější lokalizaci funkčního optima (zde je opět vidět podobnost s teplotou u metody simulovaného žíhání). Konvergence geometrické řady může být při vhodně zvoleném kvocientu optimální během celého procesu (v takovém případě není nutné kvocient během výpočtu měnit vůbec). V optimálním případě je tedy nutné nastavit pouze jediný parametr. Během procesu může dojít k situaci, kdy nově určený výběrový prostor zasahuje mimo definiční obor funkce. Tento případ musí algoritmus umět rozpoznat a nastavit rozsah výběru kryjící se s původním definičním oborem.

Pro účely testování byl stanoven limit 150 simulací (vyšší počty nemají u časově náročnějších výpočtů smysl). U každé z testovaných funkcí bylo tedy provedeno 150 simulací v 5 cyklech, tedy 30 simulací v každém cyklu. Pro účely „cílení“ kolem nejlepší hodnoty z cyklu minulého byl zvolen kvocient geometrické řady 0,5. Jako první člen této řady byla stanovena délka definičního oboru PDF jednotlivých náhodných veličin. Užití náhodné veličiny mají pro názornost rektangulární rozdělení. Celý proces byl simulován pomocí

softwaru FReET jeho manuálním přenastavením před každým dalším cyklem. Budoucím cílem je proces automatizovat pomocí vhodné úpravy softwaru.

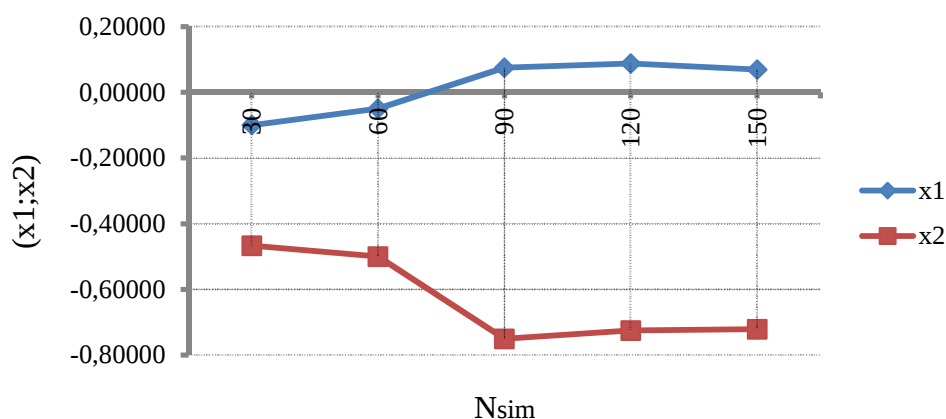
NESTED LHS SIX-HUMP CAMELBACK FUNCTION

Tab. (3.2.2.1) Six-hump camelback function průběh LHS nested

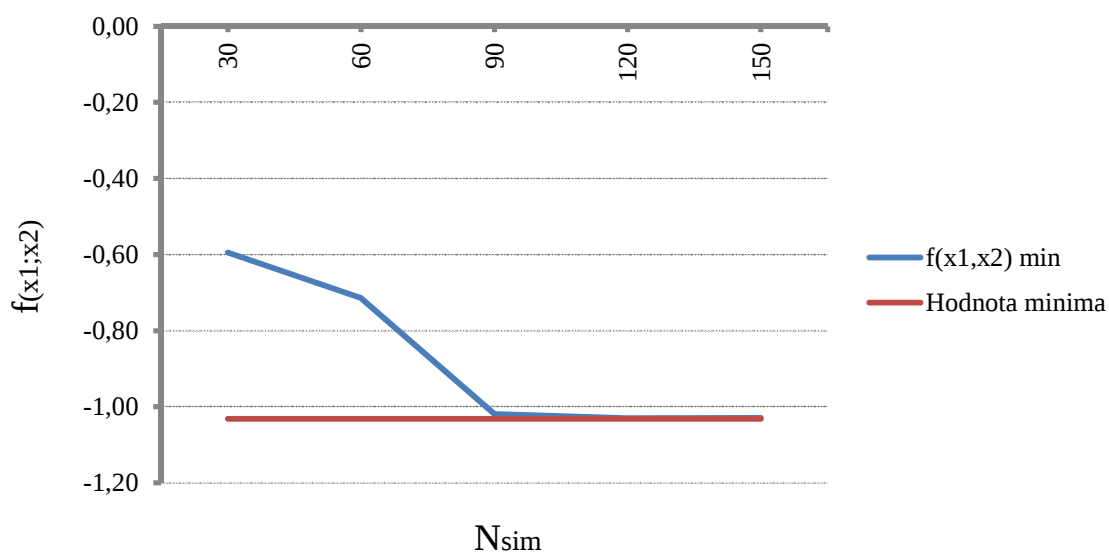
1.-30 sim		E=0	E=0	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$-3 \leq X_1 \leq 3$	rozsah: $E \pm 3$
30	-0,59495	-0,10000	-0,46667	$-2 \leq X_2 \leq 2$	rozsah: $E \pm 2$
2.-30 sim		E=-0,1	E=-0,46667	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$-1,6 \leq X_1 \leq 1,4$	rozsah: $E \pm 1,5$
60	-0,71502	-0,05000	-0,50000	$-1,46667 \leq X_2 \leq -0,53333$	rozsah: $E \pm 1$
3.-30 sim		E=-0,05	E=-0,5	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$-0,8 \leq X_1 \leq 0,7$	rozsah: $E \pm 0,75$
90	-1,01910	0,07500	-0,75000	$-1 \leq X_2 \leq 0$	rozsah: $E \pm 0,5$
4.-30 sim		E=0,075	E=-0,75	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$-0,3 \leq X_1 \leq 0,45$	rozsah: $E \pm 0,375$
120	-1,03030	0,08750	-0,72500	$-1 \leq X_2 \leq -0,5$	rozsah: $E \pm 0,25$
5.-30 sim		E=0,0875	E=-0,725	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$-0,1 \leq X_1 \leq 0,275$	rozsah: $E \pm 0,1875$
150	-1,02920	0,06875	-0,72083	$-0,85 \leq X_2 \leq -0,6$	rozsah: $E \pm 0,125$

Rozdíl mezi skutečným a nalezeným minimem: 0,00130

V oblasti okolo minima funkce může docházet k drobným výkyvům (hodnota po 150 simulacích je mírně horší než hodnota po 120 simulacích) v takovém případě by algoritmus použil jako výchozí bod pro další cyklus dosavadní nejlepší bod. Průběh procesu je znázorněn na obrázku (3.2.2.1.). Hodnota E uvedená v tabulkách (3.2.2.1-4) představuje střední hodnotu pravděpodobnostního rozdělení veličin x_1 a x_2 .

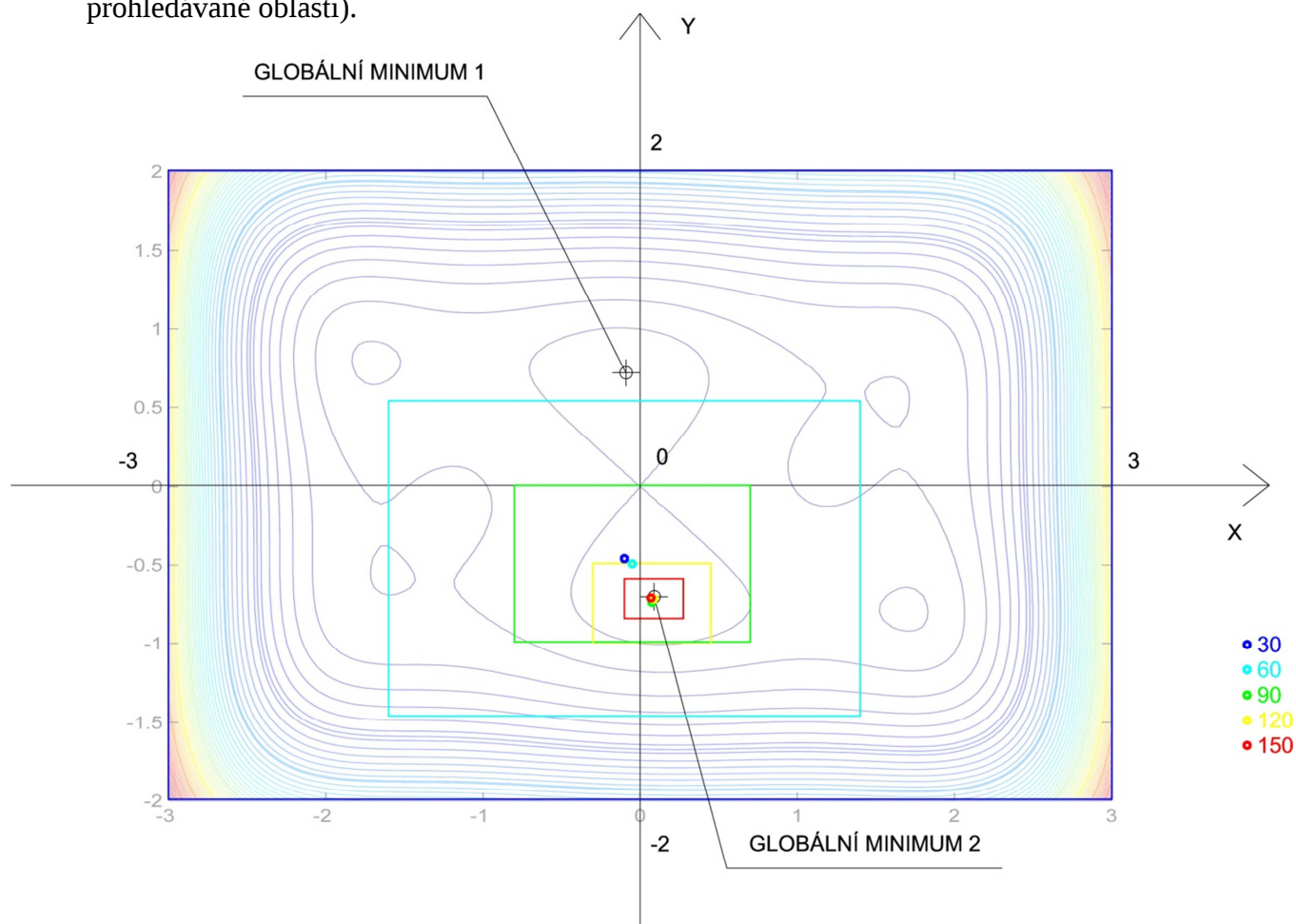


Graf (3.2.2.1) Vývoj hodnot náhodných veličin v průběhu procesu



Graf (3.2.2.2) Závislost funkční hodnoty $f(x)$ na počtu simulací

Na obrázcích (3.2.2.1-4) je vidět průběh procesu nested LHS. Jednotlivé cykly „cílení“ jsou od sebe barevně odlišeny (barevné hranice představují postupné zmenšování prohledávané oblasti).



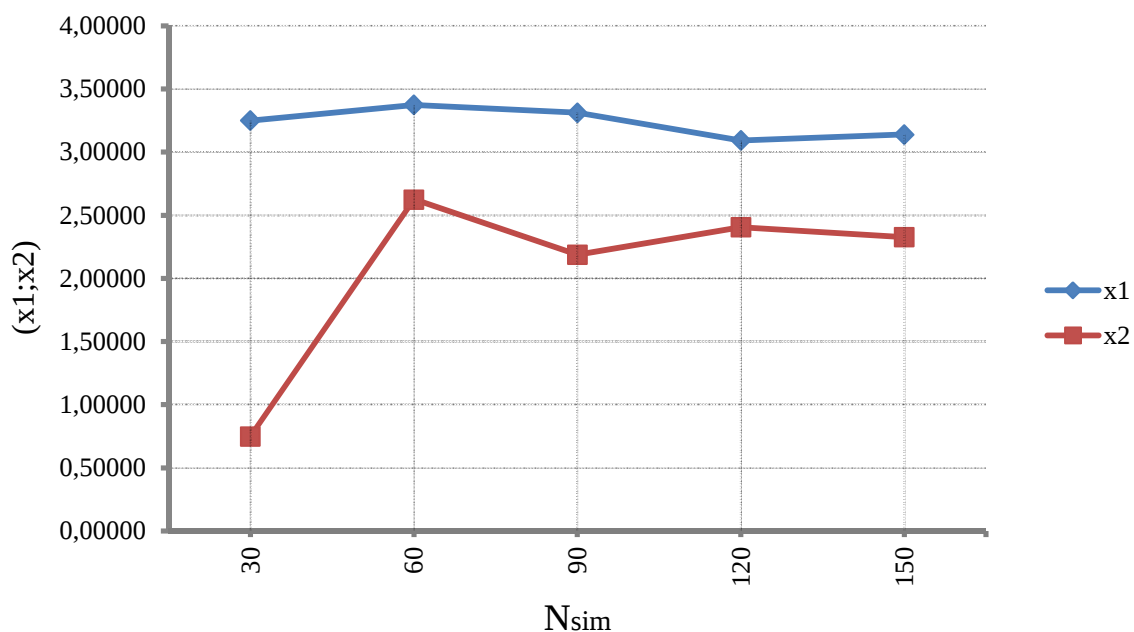
Obr. (3.2.2.1) průběh LHS nested

NESTED LHS BRANIN'S RCOS FUNCTION

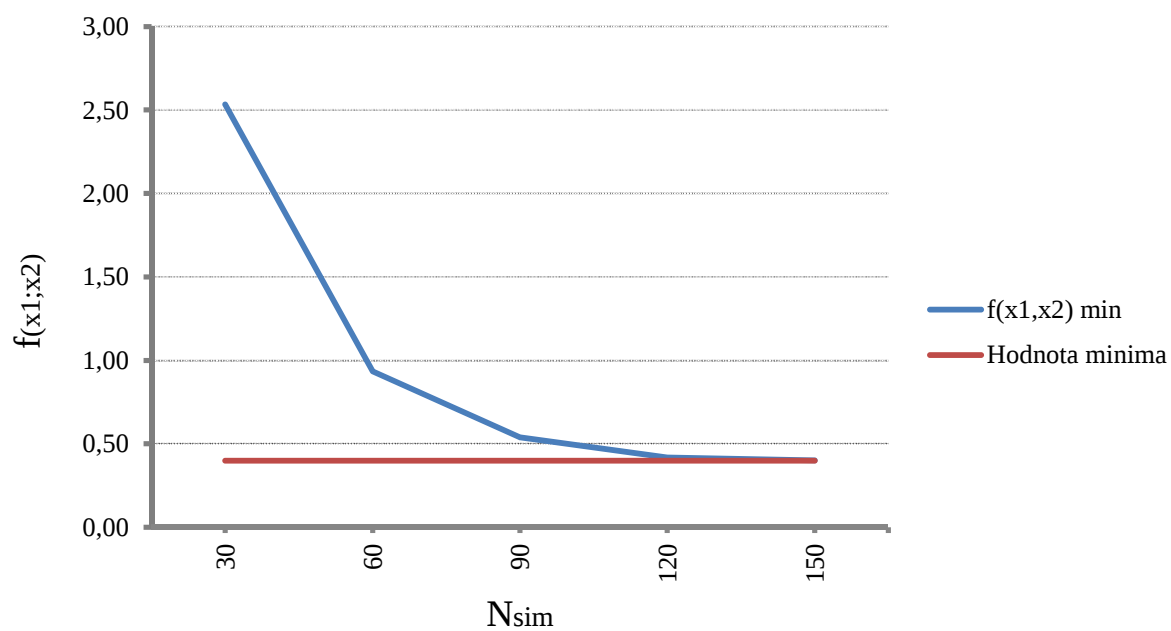
Tab. (3.2.2.2) Branin's rcos function průběh LHS nested

1.-30 sim		E=2,5	E=7,5	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$-5 \leq X_1 \leq 10$	rozsah: $E \pm 7,5$
30	2,53350	3,25000	0,75000	$0 \leq X_2 \leq 15$	rozsah: $E \pm 7,5$
2.-30 sim		E=3,25	E=0,75	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$-0,5 \leq X_1 \leq 7$	rozsah: $E \pm 3,75$
60	0,93387	3,37500	2,62500	$0 \leq X_2 \leq 4,5$	rozsah: $E \pm 3,75$
3.-30 sim		E=3,375	E=2,625	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$1,5 \leq X_1 \leq 5,25$	rozsah: $E \pm 1,875$
90	0,53955	3,31250	2,18750	$0,75 \leq X_2 \leq 4,5$	rozsah: $E \pm 1,875$
4.-30 sim		E=3,3125	E=2,1875	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$2,3745 \leq X_1 \leq 4,2505$	rozsah: $E \pm 0,938$
120	0,41764	3,09360	2,40610	$1,2495 \leq X_2 \leq 3,125$	rozsah: $E \pm 0,938$
5.-30 sim		E=3,0936	E=2,4061	meze:	
N_{sim}	$f_{(x_1,x_2)} \min$	x_1	x_2	$2,6246 \leq X_1 \leq 3,5626$	rozsah: $E \pm 0,469$
150	0,40061	3,14050	2,32790	$1,9371 \leq X_2 \leq 2,8751$	rozsah: $E \pm 0,469$

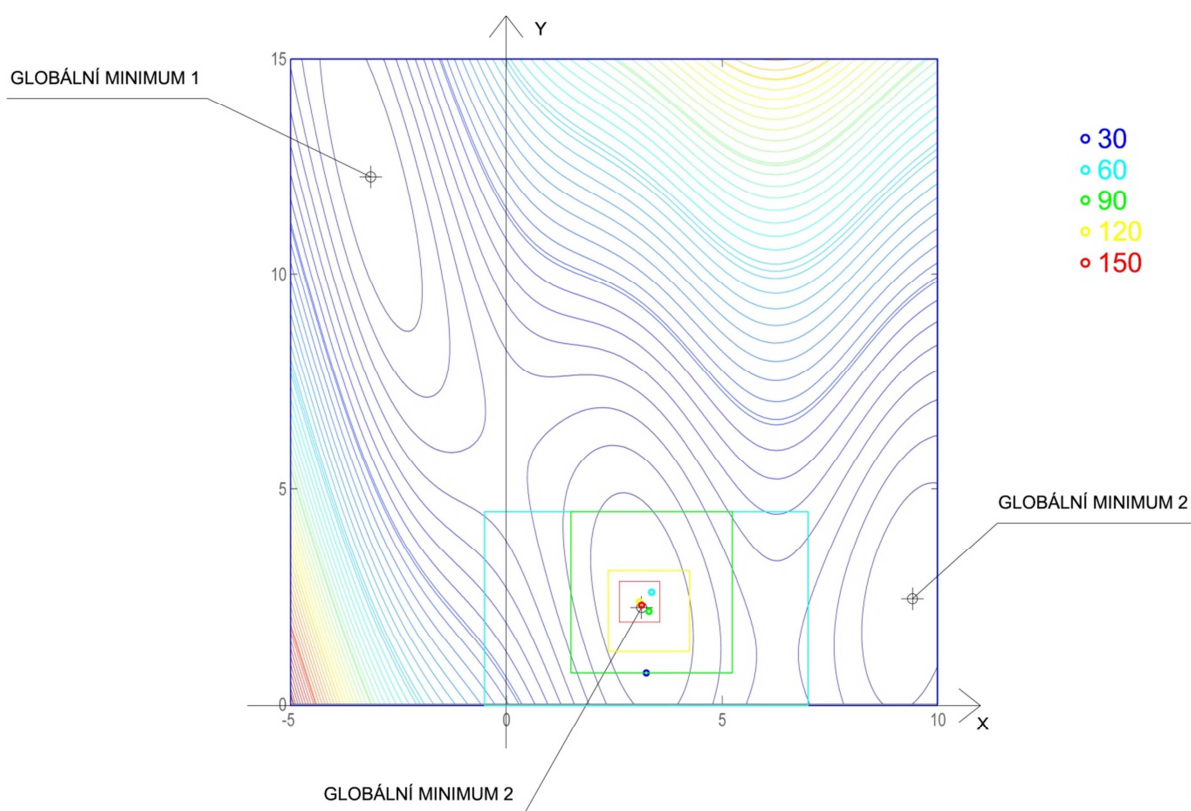
Rozdíl mezi skutečným a nalezeným minimem: 0,00272



Graf (3.2.2.3) Vývoj hodnot náhodných veličin v průběhu procesu



Graf (3.2.2.4) Závislost funkční hodnoty $f(x)$ na počtu simulací



Obr. (3.2.2.2) průběh LHS nested

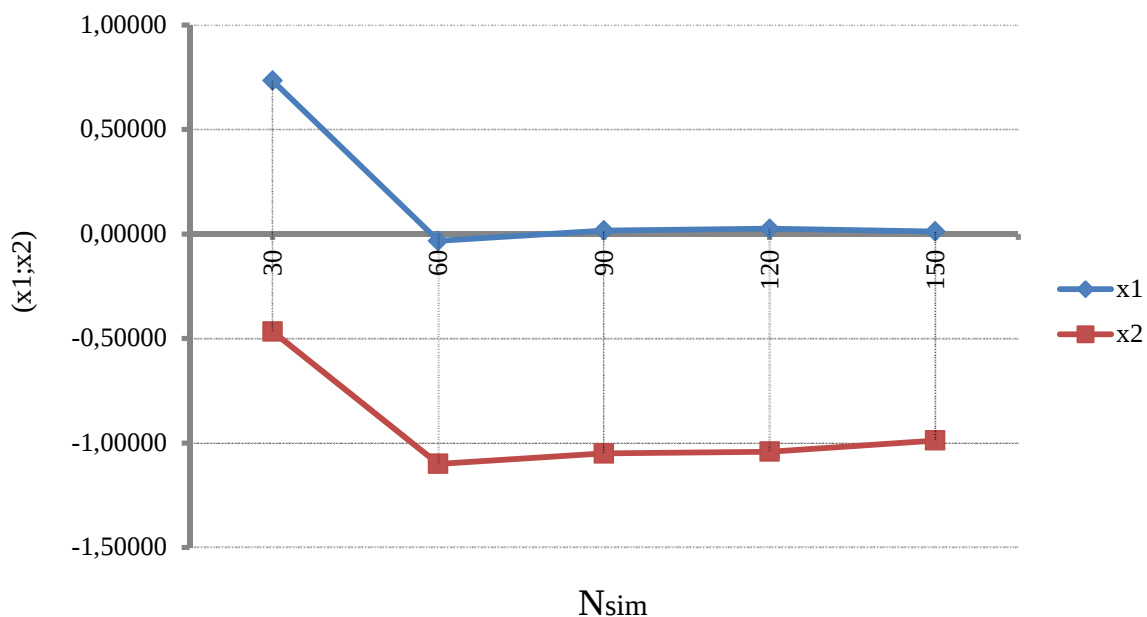
NESTED LHS GOLDSTEIN-PRICE'S FUNCTION

Tab. (3.2.2.3) Goldstein-Price's function průběh LHS nested

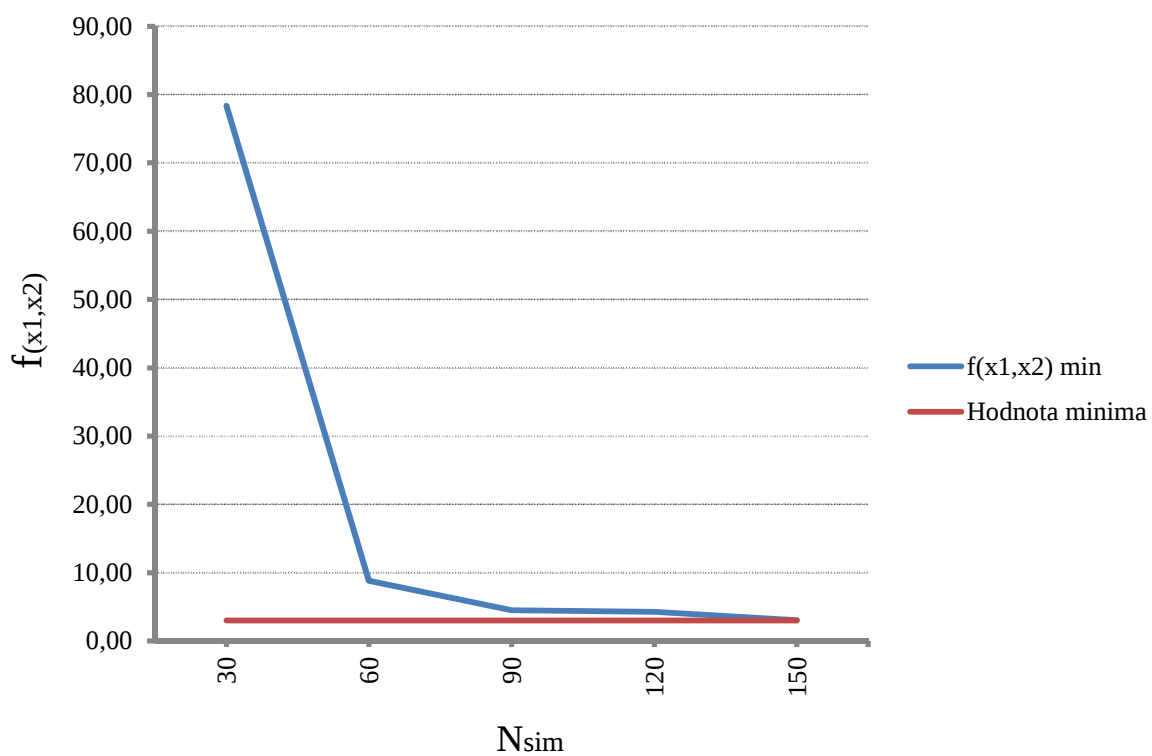
1.-30 sim		E=0	E=0	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-2 \leq X1 \leq 2$	rozsah: $E \pm 2$
30	78,36500	0,73333	-0,46667	$-2 \leq X2 \leq 2$	rozsah: $E \pm 2$
2.-30 sim		E=0,73333	E=-0,46667	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-0,26667 \leq X1 \leq 1,73333$	rozsah: $E \pm 1$
60	8,81870	-0,03334	-1,10000	$-1,46667 \leq X2 \leq 0,53333$	rozsah: $E \pm 1$
3.-30 sim		E=-0,03334	E=-1,1	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-0,533337 \leq X1 \leq 0,466663$	rozsah: $E \pm 0,5$
90	4,51320	0,01666	-1,05000	$-1,6 \leq X2 \leq -0,6$	rozsah: $E \pm 0,5$
4.-30 sim		E=0,016663	E=-1,05	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-0,233337 \leq X1 \leq 0,266663$	rozsah: $E \pm 0,25$
120	4,25490	0,02500	-1,04170	$-1,3 \leq X2 \leq -0,8$	rozsah: $E \pm 0,25$
5.-30 sim		E=0,024996	E=-1,0417	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-0,1 \leq X1 \leq 0,149996$	rozsah: $E \pm 0,125$
150	3,07200	0,01250	-0,98753	$-1,1667 \leq X2 \leq -0,9167$	rozsah: $E \pm 0,125$

Rozdíl mezi skutečným a nalezeným minimem: 0,072

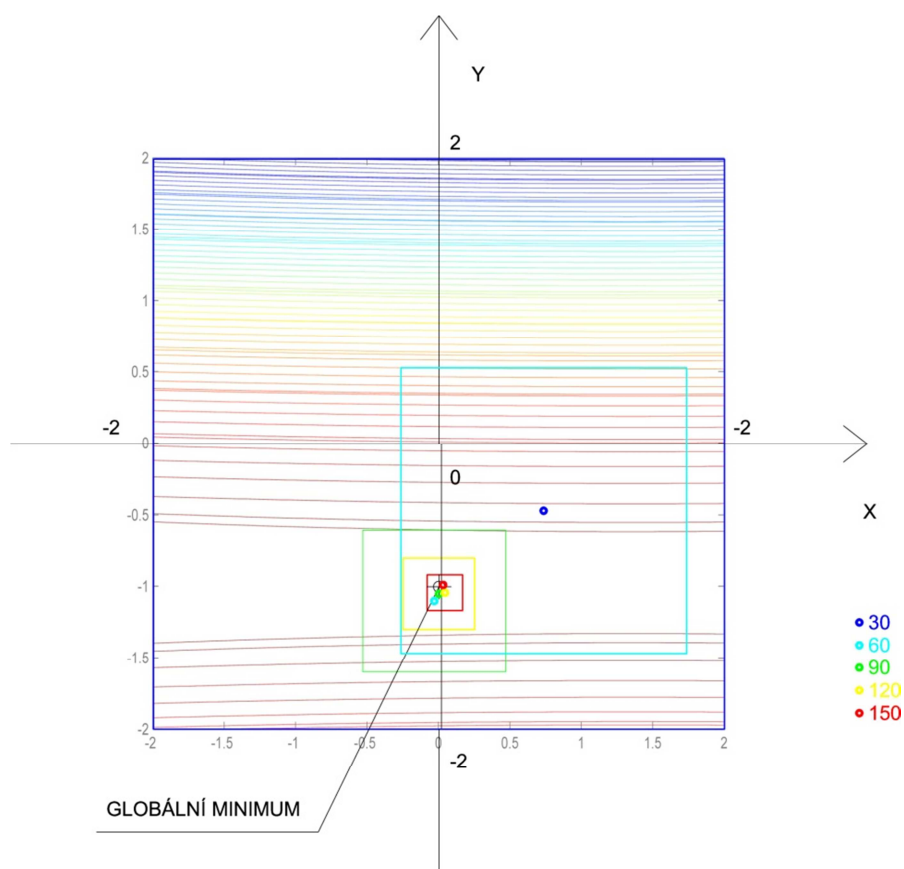
Nalezená hodnota minima je vzhledem k rozsahu oboru hodnot funkce uspokojivá.



Graf (3.2.2.5) Vývoj hodnot náhodných veličin v průběhu procesu



Graf (3.2.2.6) Závislost funkční hodnoty $f(x)$ na počtu simulací



Obr. (3.2.2.3) průběh LHS nested

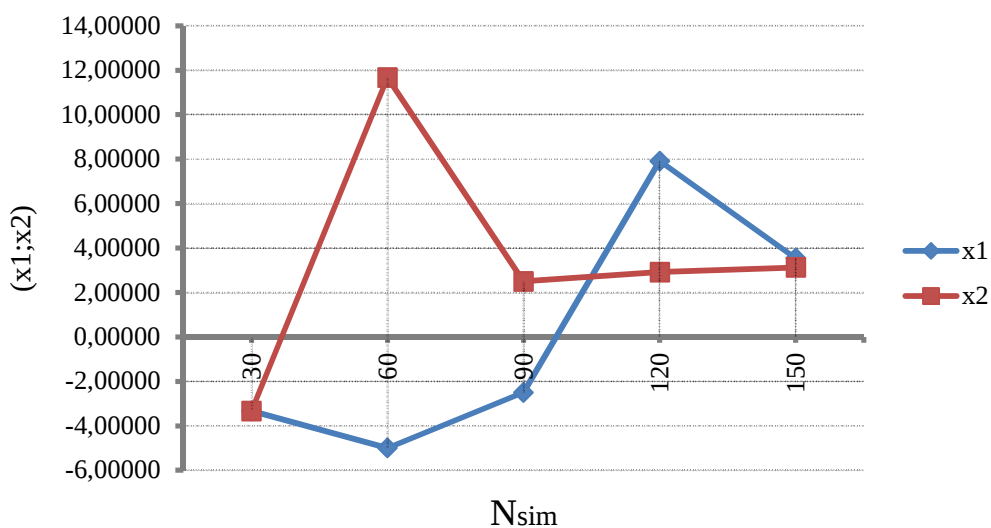
NESTED LHS EASOM'S FUNCTION

Tab. (3.2.2.4) Easom's function průběh LHS nested

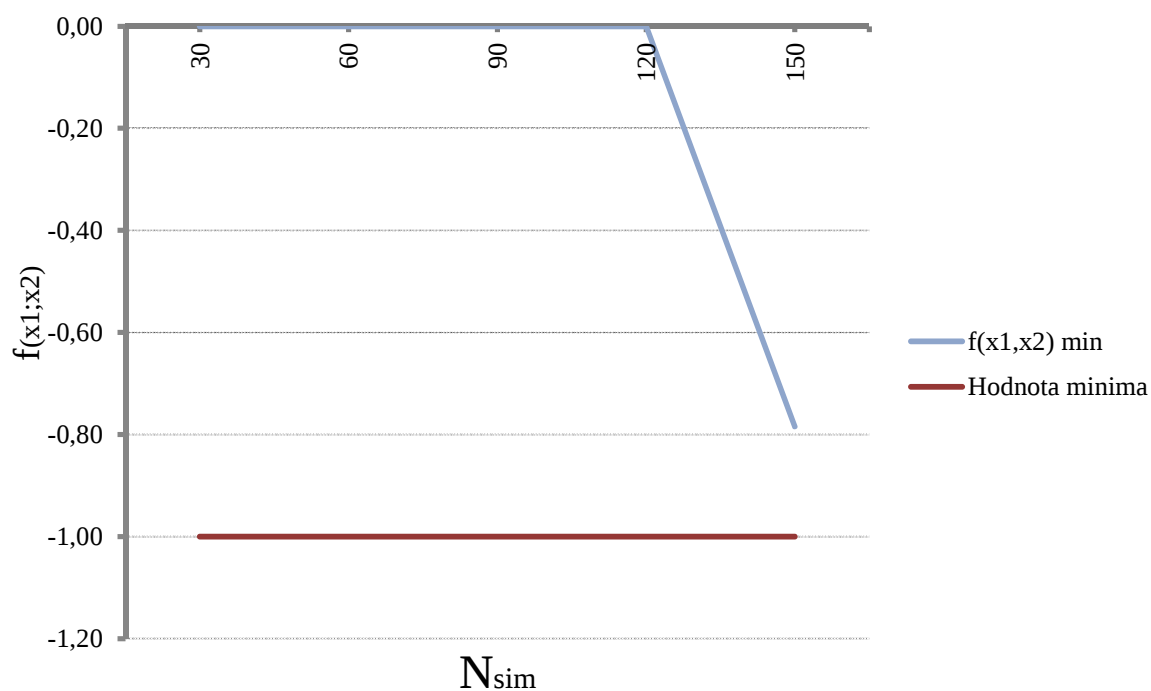
1.-30 sim		E=0	E=0	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-100 \leq X1 \leq 100$	rozsah: $E \pm 100$
30	0,00000	-3,33330	-3,33330	$-100 \leq X2 \leq 100$	rozsah: $E \pm 100$
2.-30 sim		E=-3,3333	E=-3,3333	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-53,3333 \leq X1 \leq 46,6667$	rozsah: $E \pm 50$
60	0,00000	-5,00000	11,66700	$-53,3333 \leq X2 \leq 46,6667$	rozsah: $E \pm 50$
1.-30 sim		E=-5	E=11,667	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-30 \leq X1 \leq 20$	rozsah: $E \pm 25$
90	0,00000	-2,50000	2,50030	$-13,333 \leq X2 \leq 36,667$	rozsah: $E \pm 25$
2.-30 sim		E=-2,5	E=2,5003	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$-15 \leq X1 \leq 10$	rozsah: $E \pm 12,5$
120	0,00000	7,91670	2,91640	$-15,0003 \leq X2 \leq 9,9997$	rozsah: $E \pm 12,5$
1.-30 sim		E=7,9167	E=2,9164	meze:	
N_{sim}	$f_{(x1,x2)} \min$	x_1	x_2	$1,6667 \leq X1 \leq 14,1667$	rozsah: $E \pm 6,25$
150	-0,78444	3,54170	3,12470	$-3,3336 \leq X2 \leq 9,1664$	rozsah: $E \pm 6,25$

Rozdíl mezi skutečným a nalezeným minimem: 0,21556

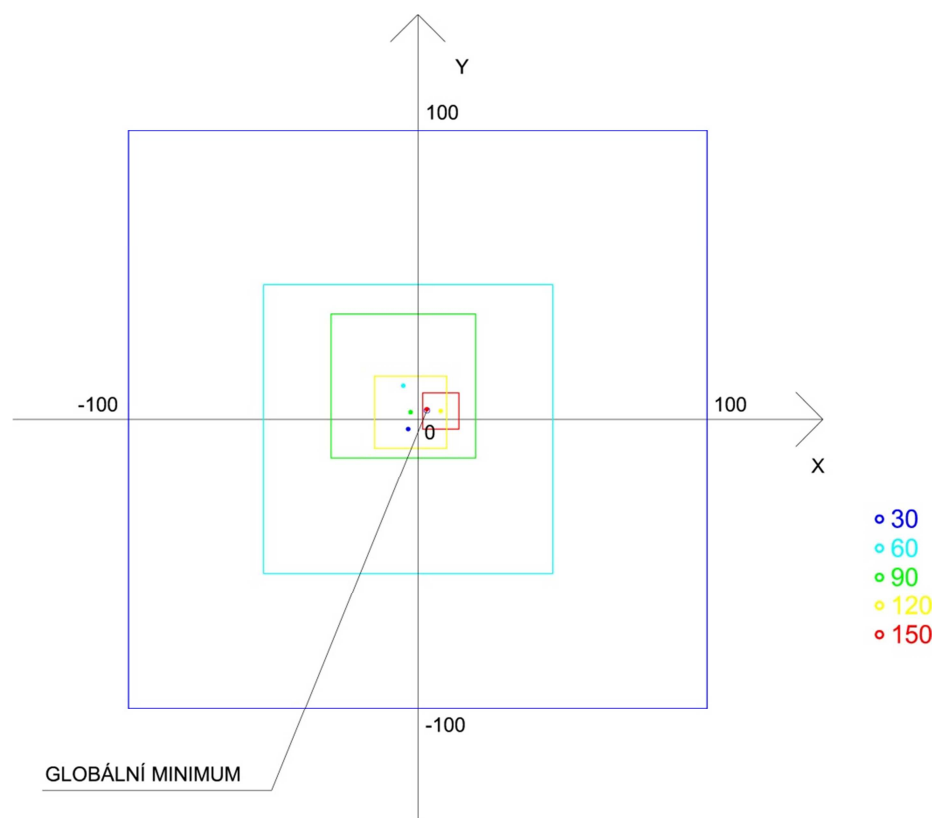
Vzhledem k charakteru funkce bylo nalezení správného směru k funkčnímu minimu nepravděpodobné. Přesto byla oblast minima lokalizována a je zřejmé, že pokud by proces pokračoval, uspokojivé minimum by bylo nalezeno.



Graf (3.2.2.7) Vývoj hodnot náhodných veličin v průběhu procesu



Graf (3.2.2.8) Závislost funkční hodnoty $f(x)$ na počtu simulací



Obr. (3.2.2.4) průběh LHS nested

3.2.3 VYHODNOCENÍ

Metoda simulovaného žíhání v kombinaci s metodou LHS je schopna (při optimálním nastavení všech parametrů) s velkou pravděpodobností nalézt optimum dané funkce (do určité složitosti). Nalezení optimálního nastavení algoritmu simulovaného žíhání je však velice složitý problém. Správné nastavení simulovaného žíhání je navíc závislé na poloze počátečního bodu vzhledem k hledanému optimu (touto informací uživatel většinou vůbec nedisponuje). Počáteční bod pro simulované žíhání je generován metodou LHS (je proto náhodného charakteru) a při dalším opakování téhož procesu bude jako začátek simulovaného žíhání přijat bod jiný. Nastavení optimální teploty bude v takovém případě odlišné. Z těchto důvodů není prakticky možné počítat u této kombinace metod s nalezením minima při provedení řádově stovek simulací (pravděpodobnost optimálního nastavení všech parametrů bez předchozího heuristického vyšetřování je malá). Pro úspěšnou optimalizaci vyžaduje tato kombinace nejčastěji tisícovky simulací. Výpočetní náročnost zkoumané kombinace metod je pro účely výpočtů mechaniky kontinua vysoká.

Metoda LHS nested byla schopna najít uspokojivé minimum u všech testovaných funkcí s užitím minimálního počtu simulací. Její výhodou je minimum nastavovaných parametrů oproti metodě simulovaného žíhání. LHS nested vykazovala u všech testovaných funkcí vyšší efektivitu než kombinace LHS a simulovaného žíhání. Zdá se proto být vhodným kandidátem pro další studium v rámci analýzy s malým počtem vzorků. Provedené testy bude třeba v budoucnu zopakovat a rozšířit (není jasné, zda metoda bude vykazovat stejnou efektivitu i u vícerozměrných funkcí). Další zkoumanou variantou může být například kombinace LHS a genetických algoritmů. Uvedené možnosti nebyly vzhledem k rozsahu této práce prozkoumány. Prokáže-li algoritmus LHS nested svou schopnost řešit náročnější problémy, mohl by být naprogramován a implementován v některém ze stávajících programů.

4. PROGRAMOVÉ PROSTŘEDKY

4.1 FREET

FReET (Feasible Reliability Engineering Tool) je víceúčelový pravděpodobnostní software určený pro statistickou analýzu, analýzu senzitivity a analýzu spolehlivosti při řešení technických problémů. Umožňuje simulovat úlohy na úrovni náhodných proměnných. K výběru vzorků používá klasickou metodu Monte Carlo, či některou z metod LHS. Pro zavedení požadované korelace mezi vygenerovanými náhodnými proměnnými využívá FReET metodu simulovaného žíhání. Náhodným veličinám je možné přidělit některé z předdefinovaných rozdělení. [7]

4.2 ATENA

Program ATENA (Advanced Tool for Engineering Nonlinear Analysis) byl vyvinut firmou Červenka Consulting z Prahy pro počítačovou simulaci železobetonových konstrukcí. Program je založen na deformační metodě konečných prvků a jeho hlavní charakteristikou je použití nelineárních modelů materiálu, což umožňuje analyzovat chování stavební konstrukce v kritických podmínkách, kdy dochází k jejímu porušování. Základem jsou přitom pokročilé nelineární materiálové modely pro beton a betonářskou výztuž. Během vývoje programu byly vytvořeny další materiálové modely vhodné pro simulování cihelného či kamenného zdiva, vláknobetonu, zvláštních druhů výztuže (např. uhlíkových lamel nebo aramidových tyčí) nebo podloží. Tyto modely lze snadno kombinovat pro řešení spřažených konstrukcí z různých materiálů. [8]

4.3 SARA

Programový systém nazvaný SARA (Structural Analysis and Reliability Assessment) kombinuje nelineární výpočet s pravděpodobnostními metodami a degradačními modely. To umožňuje předpovědět pravděpodobnost poruchy stavby a vývoj této pravděpodobnosti v čase. Pomocí tohoto systému lze například vybrat optimální návrh rekonstrukce stavebního objektu z hlediska jeho spolehlivosti a životnosti. SARA studio je program zajišťující spolupráci programů ATENA a FReET. [8]

5. PRAKTICKÁ ÚLOHA

Doposud byla v této práci pozornost zaměřena na optimalizaci pouze teoretických funkcí, u nichž bylo cílem najít minimum či maximum dané funkce. Reálné optimalizační úlohy vyskytující se ve stavební praxi jsou však většinou mnohem složitější než uvedené teoretické příklady a jejich řešením bývá spíše než nalezení jednoho konkrétního optima, stanovení optimálního poměru sledovaných veličin (například cena/únosnost). Mezi jednotlivými parametry reálného modelu existuje zpravidla různě silná statistická závislost. Randomizujeme-li jeden z těchto parametrů, pak musíme randomizovat i všechny zbylé na něm závislé parametry. Například se změnou pevnosti betonu v tlaku se mění také jeho pevnost v tahu, lomová energie a Youngův modul pružnosti. Pro výstižný popis reality je zpravidla nutné použít modely s velkým počtem náhodných proměnných s vhodně zvolenou korelační maticí, jež stanoví správnou míru závislosti jednotlivých parametrů. Randomizovaným veličinám je nutné v závislosti na požadavcích optimalizace přidělit vhodná pravděpodobnostní rozdělení. Ukázka jednoduché optimalizace praktické úlohy je v následujícím textu.

5.1 ZADÁNÍ

U stávající železobetonové konstrukce silničního mostu s dvěma jízdními pruhy je nutné provést nadbetonování na obou stranách komunikace pro zvýšení jeho únosnosti. Cílem optimalizace bylo najít vhodnou kombinaci výšky nadbetonování (šířka nadbetonování je dána šířkou mostu a vozovky) a kvality použitého betonu tak, aby cena materiálu byla co nejnížší a únosnost konstrukce co nejvyšší.

Geometrie, způsob vyztužení a uložení desky mostu je na schématech v přílohách P2 až P4. Beton bude dovážen z nedaleké betonárky, jež si účtuje ceny za 1 m³ betonu v závislosti na jeho kvalitě dle tabulky (5.1.1) [10].

Tab. (5.1.1) Část ceníku společnosti SPH stavby s.r.o.

Třída pevnosti betonu	cena za 1 m ³ směsi
C8/10	1495
C12/15	1620
C16/20	1665
C20/25	1800
C25/30	1915
C30/37	2240
C35/45	2410
C40/50	2495

Byly stanoveny následující omezující podmínky optimalizace:

1) Mostní konstrukce musí být schopna bezpečně přenést zatížení od vlastní tíhy a rovnoměrně rozložené zatížení odpovídající šesti vozidlům s hmotností 7 tun vyskytujícím se v jeden okamžik současně na mostní komunikaci.

Výpočet reakce v podpoře od zatížení vozidly:

$$R_{cn} = \frac{Q_n \times g \times m \times n}{2 \times 1000} = \frac{1,5 \times 10 \times 7000 \times 6}{2 \times 1000} = 315 \text{ kN} \quad (5.1.1)$$

Kde Q_n je součinitel bezpečnosti pro nahodilé zatížení (zde uvažován roven 1,5). Idealizace bude dvourozměrná, proto při rovnoměrném rozložení zatížení od vozidel je reakce v jedné ze dvou krajních podpor rovna polovině celkové síly vyvolané tímto zatížením. Úloha bude řešena metodou konečných prvků s užitím softwaru ATENA 2D. Jím vypočtená zatěžovací křivka reprezentuje závislost velikosti reakce v podpoře (při symetrickém rozložení zatížení na idealizovaném nosníku) na hodnotě deformace uprostřed idealizovaného nosníku. Z tohoto

důvodu je vhodné zavést první omezující podmínku jako minimální hodnotu reakce v podpoře při zatížení na mezi únosnosti R_{cc} .

První omezující podmínka je stanovena takto:

$$R_{cc} \geq R_{cs} + R_{cn} = R_{cs} + 315 \text{ [kN]} \quad (5.1.2)$$

Kde R_{cc} je celková hodnota reakce v podpoře na mezi únosnosti, R_{cs} je část reakce v podpoře odpovídající vlastní tíze a R_{cn} je část reakce připadající na zatížení vozidly. Levá strana představuje výsledek nelineárního výpočtu pomocí softwaru ATENA, pravá strana pak mezní předepsanou hodnotu.

2) Druhá omezující podmínka stanovuje maximální přípustnou cenu použitého materiálu a je zvolena takto:

$$C \leq 35000 \text{ [Kč]} \quad (5.1.3)$$

Kde C je cena konstrukce.

3) Třetí omezující podmínky se týká výšky nadbetonování h :

$$h \in (0; 0,5) \quad (5.1.4)$$



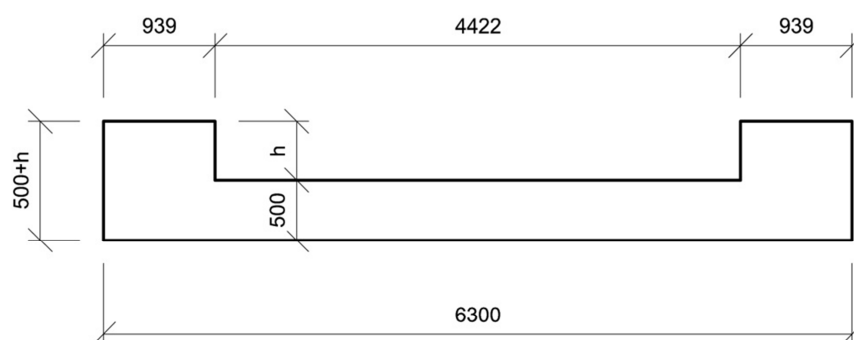
Obr. (5.1.1) Fotografie analyzované konstrukce

5.2 ŘEŠENÍ

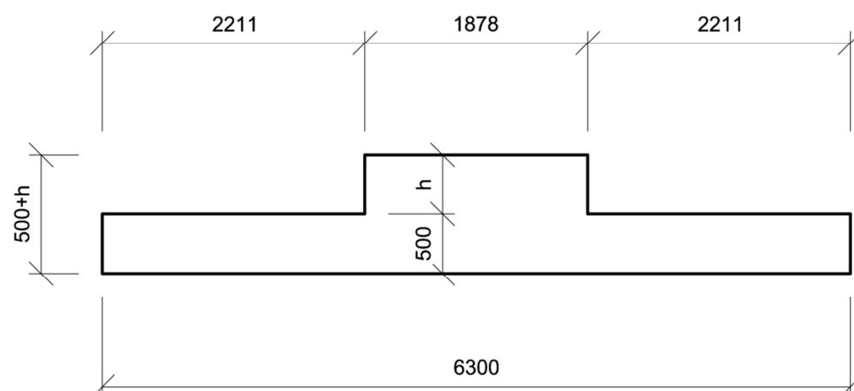
5.2.1 TVORBA IDEALIZOVANÉHO 2D MODELU

Skutečnou konstrukci je pro účely výpočtu nutné aproximovat co možná nejpřesnějším matematickým modelem. Vzhledem k jednoduché geometrii zadané konstrukce postačí k simulaci úlohy pouze dvourozměrný model vytvořený v programu ATENA 2D.

Železobetonová deska mostu má tvar pravidelného kosodélníku s délkou podstavy 15,583 m (viz. Příloha P2). Most je uložen na čtyřech podporách v rozích půdorysu (uložení viz. Příloha P4). Vyztužení desky je patrné z výkresu v příloze P3. Skutečný tvar průřezu je znázorněn na obrázku (5.2.1). Takovýto průřez je nutné v programu ATENA 2D nahradit průřezem o totožné průřezové ploše se stejným momentem setrvačnosti vzhledem k vodorovné ose. Substituční průřez je vykreslen na obrázku (5.2.2).



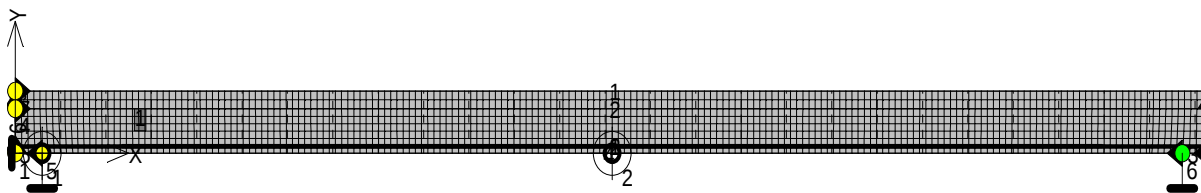
Obr. (5.2.1) Skutečný průřez mostu



Obr. (5.2.2) Substituční průřez mostu pro účely 2D modelování

Hodnota h na obrázcích představuje hledanou optimální výšku nadbetonování.

Výpočtovým modelem je v našem případě prostý nosník dlouhý 15,583 m složený ze dvou makroprvků. Jeden makroprvek představuje původní železobetonovou desku a druhý nadbetonování. Skutečné vyztužení bylo v modelu nahrazeno pomocí dvou prutů rozdílné délky z oceli R 10 505. U prvního prutu byla uvažována délka 15,583 m a průřezová plocha $1.005\text{E-}02 \text{ m}^2$. U druhého prutu byla uvažována délka 12,783 m a průřezová plocha $3.927\text{E-}03 \text{ m}^2$. Smyková a konstrukční výztuž byla nahrazena rozptýlenou výztuží zaujímající 3% objemu materiálu. Pro vygenerování sítě konečných prvků byla stanovena délka strany prvky 0,08 m. Byly použity dva monitory: první pro monitorování reakce v pravé podpoře a druhý pro monitorování svislého posunu na spodním okraji ve středu nosníku. Všechny nastavené zatěžovací kroky byly násobeny koeficientem 0.1. První dva výpočtové kroky měly nastaveny parametry výpočtu: standartní Newton-Raphson, zbylé pak: standartní Arc Length. Idealizovaný model konstrukce vytvořený v programu ATENA 2D je zachycen na obrázku (5.2.3).



Obr. (5.2.3) Idealizovaný model konstrukce vytvořený v programu ATENA 2D

5.2.2 RANDOMIZACE PARAMETRŮ

Randomizace proběhla v rámci programu SARA studio. SARA studio je řídicím programem zajišťujícím spolupráci mezi pravděpodobnostním Softwarem FReET a programem ATENA 2D. Analýza výsledků byla z časových důvodů zpracována ručně pomocí programu Microsoft Excel. SARA studio sice dokáže zpracovat analýzu výsledků automaticky, neumožňuje však v rámci jednoho projektu spustit více souběžných výpočtů. Vzhledem k množství provedených simulací bylo výhodnější použít studio SARA pouze k vygenerování jednotlivých modelů, jež byly poté spuštěny programem ATENA 2D mimo SARA studio a spočteny zvlášť.

Cílem optimalizace byla minimalizace nákladů na materiál a dosažení co nejvyšší únosnosti konstrukce. Cena konstrukce závisí na množství a kvalitě použitého betonu. Z tohoto důvodu je vhodné randomizovat právě tyto dva parametry.

Objem použitého materiálu je závislý na jediné náhodné proměnné h (výška nadbetonování). Půdorysný rozměr nadbetonovaných lemů mostní komunikace je pevně stanoven, neboť je omezen šířkou silnice a rozměry mostu. Změna výšky nadbetonování je realizována změnou svislé polohy horních styčníků. Mezi jejich pohybem je nutné zavést korelaci s korelačním koeficientem 1 (0,9999 – korelační matice musí být pozitivně definitivní) [9].

Randomizace kvality betonu je poněkud složitější. Kvalita dostupných betonových směsí je diskrétně oddělena. Není možné pořídit si například beton s krychelnou pevností v tlaku 23 MPa, k dostání je pouze směs s 5 % kvantilem krychelné pevnosti v tlaku 20 MPa nebo 25 MPa. Program FReET nedokáže takto diskrétně rozložené vzorky generovat. Po nalezení vhodného optima je tedy nutné konstatovat, že místo nalezeného optima bude použita běžně dostupná směs s materiálovými parametry co nejbližšími nalezenému optimu. Dalším problém může být stanovení vhodné korelace mezi jednotlivými materiálovými parametry betonu. Změní-li se pevnost betonu v tlaku dojde i ke změně pevnosti v tahu, lomové energie a Youngova modulu pružnosti, proto musí být randomizovány (s vhodnou statistickou závislostí) všechny tyto parametry. V našem případě byla použita korelační matice v podobě:

Tab. (5.2.2.1) Korelační matice betonu nadbetonování [11]

Proměnná	E_c	f_t	f_c	G_f
E_c	1	0.7	-0.9	0.5
f_t	0.7	1	-0.8	0.9
f_c	-0.9	-0.8	1	-0.6
G_f	0.5	0.9	-0.6	1

Pro 30 vzorků (30 variant idealizovaného nosníku s různou výškou nadbetonování a různou kvalitou betonu) byla využita metoda LHS mean (vztah 2.2.2). Podoba náhodných vektorů pro jednotlivé simulace je uvedena v tabulce (5.2.2.2). Pro lokalizaci minima funkce ceny (viz. další podkapitola) je důležité rovnoměrné pokrytí návrhového prostoru. Všechny náhodné proměnné jsou proto uvažovány s rektangulárním rozdělením. Definiční obory náhodných proměnných byly v souladu s tab. (5.1.1) definovány takto:

$$E \in (26000; 35000) [MPa]$$

$$f_c \in (-50; -8) [MPa]$$

$$f_t \in (1,3; 3,5) [MPa]$$

$$G_f \in (50; 150) [kJ]$$

Rozsah výšky nadbetonování byl volen v souladu s podmínkou optimalizace (5.1.4)

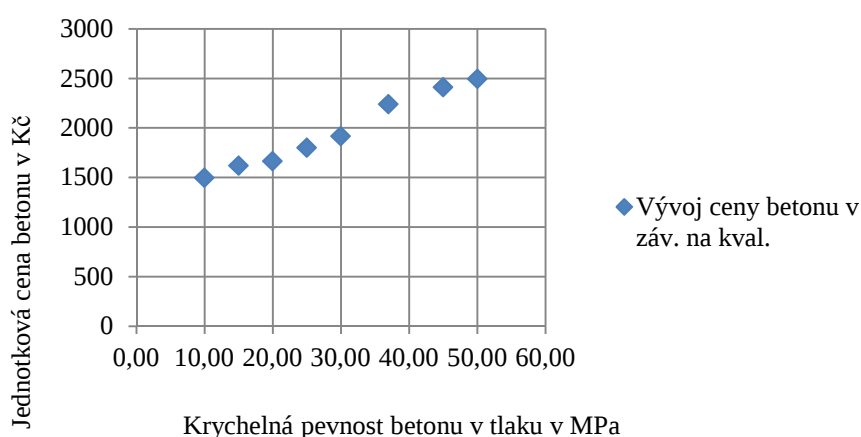
Tab. (5.2.2.2) Vygenerované náhodné vektory

N	E [MPa]	ft [MPa]	fc [MPa]	Gf [kJ]	Styčník 7 y [m]	Styčník 8 y [m]
1	26150	1.33	-10.1	68.333	0.69	0.70
2	26450	1.33	-11.5	75	0.59	0.60
3	26750	1.33	-17.1	91.667	0.89	0.89
4	27050	1.41	-14.3	51.667	0.86	0.84
5	27350	1.63	-15.7	61.667	0.81	0.78
6	27650	2.33	-24.1	118.33	0.68	0.68
7	27950	1.67	-12.9	81.667	0.96	0.98
8	28250	1.67	-19.9	71.667	0.83	0.81
9	28550	2.67	-8.7	105	0.64	0.64
10	28850	1.85	-21.3	58.333	0.94	0.94
11	29150	2.29	-25.5	88.333	0.73	0.69
12	29450	2.7	-18.5	78.333	0.54	0.56
13	29750	3.33	-33.9	148.33	0.71	0.73
14	30050	2.67	-22.7	101.67	0.84	0.86
15	30350	2.95	-32.5	145	0.56	0.53
16	30650	2.67	-29.7	108.33	0.51	0.51
17	30950	2.33	-28.3	128.33	0.61	0.63
18	31250	2.1433	-38.1	65	0.74	0.76
19	31550	1.67	-26.9	55	0.58	0.58
20	31850	3.0967	-35.3	131.67	0.99	0.99
21	32150	3.33	-47.9	138.33	0.88	0.88
22	32450	1.67	-31.1	85	0.925	0.925
23	32750	3.17	-46.5	135	0.79	0.83
24	33050	3.39	-40.9	141.67	0.98	0.96
25	33350	2.51	-45.1	95	0.63	0.59
26	33650	2.73	-43.7	111.67	0.53	0.54
27	33950	2.33	-36.7	125	0.91	0.91
28	34250	2.67	-39.5	98.333	0.76	0.79
29	34550	3.0233	-49.3	115	0.66	0.66
30	34850	3.67	-42.3	121.67	0.78	0.74

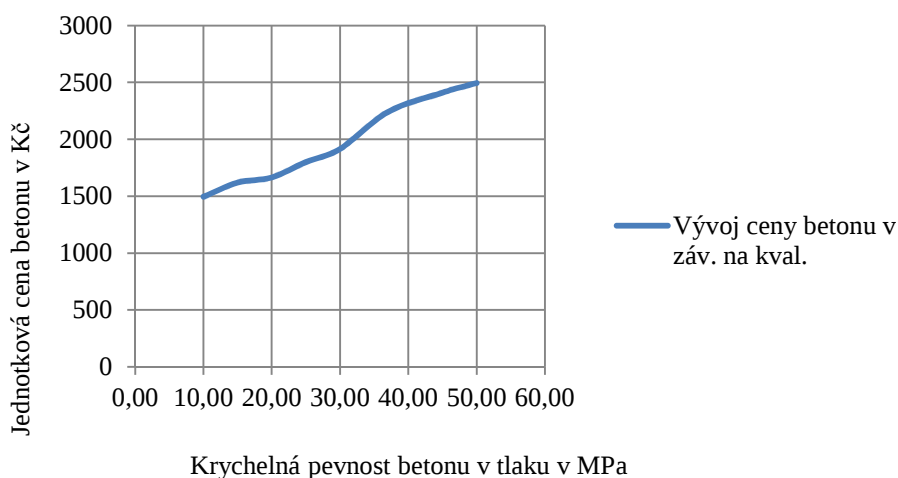
Odchyly polohy styčníku 7 a 8 ve směru Y jsou způsobeny nepřesnostmi v korelační matici vygenerované programem FReET. Velikost těchto odchylek však nepřesahuje 4 cm na rozpětí 15,583 m dosažená přesnost je tak pro účely jednoduché optimalizace postačující.

5.2.3 FUNKCE CENY MATERIÁLU

Funkcí ceny materiálu se rozumí součin objemu použitého materiálu a jeho jednotkové ceny v závislosti na kvalitě betonu (vztah 5.2.3.1). Jednotková cena materiálu je výrobcem zadána diskrétně v důsledku diskrétního rozřídění betonu podle kvality. Závislost jednotkové ceny betonu (stanovené zvoleným výrobcem) a jeho třídy kvality zobrazuje graf (5.2.3.1). Hodnoty pevností v tlaku jednotlivých vzorků vygenerované metodou LHS mean neodpovídají diskrétnímu rozdělení pevnostních tříd betonu. Pro výpočet ceny vygenerovaných materiálů je proto třeba stanovit přibližnou funkci jednotkové ceny materiálu $J_{(fc)}$ tu získáme spojením bodů grafu (5.2.3.1). Výsledný průběh přibližné funkce jednotkové ceny materiálu pak zobrazuje graf (5.2.3.2). Konkrétní hodnotu ceny v závislosti na vygenerované pevnosti v tlaku získáme lineární interpolací mezi nejbližšími diskrétně určenými body.



Graf (5.2.3.1) Vývoj ceny betonu v závislosti na krychelné pevnosti v tlaku



Graf (5.2.3.2) Průběh funkce jednotkové ceny materiálu $J_{(fc)}$

Funkci ceny materiálu můžeme tedy stanovit takto:

$$C_{(h, fc)} = V_{(h)} \times J_{(fc)} \quad (5.2.3.1)$$

Kde $V_{(h)}$ je funkcí objemu použitého materiálu v závislosti na výšce nadbetonování h definovaná takto:

$$V_{(h)} = A \times h \quad (5.2.3.2)$$

$A = 1,878 \times 15,583 = 29,265 \text{ m}^2$ - plocha půdorysu nadbetonování

Odtud:

$$V_{(h)} = 29,265 \times h \quad [\text{m}^3] \quad (5.2.3.3)$$

Funkce $J_{(fc)}$ je definována úsečkami tvořícími spojnice bodů grafu (5.2.3.1). Její přesnou funkční hodnotu pro danou pevnost v tlaku získáme lineární interpolací mezi známými body.

S určeným průběhem funkce ceny materiálu $C_{(h, fc)}$ není problém stanovit cenu materiálu pro jednotlivé varianty vygenerované metodou LHS mean. Cena stanovená pro jednotlivé vzorky je uvedena v tabulce (5.2.3.1).

Tab. (5.2.3.1) přehled cen vygenerovaných vzorků

N_{sim}	f_c [MPa]	$J_{(f_c)}$ [Kč]	Styčník 8 y [m]	h [m]	V [m ³]	$C_{(h, f_c)}$ [Kč]
1	10,1	1497,5	0,7083	0,208	6,097	9130
2	11,5	1532,5	0,6083	0,108	3,170	4858
3	17,1	1638,9	0,892	0,392	11,462	18785
4	14,3	1602,5	0,842	0,342	9,999	16023
5	15,7	1629,3	0,775	0,275	8,048	13112
6	24,1	1775,7	0,675	0,175	5,121	9094
7	12,9	1567,5	0,975	0,475	13,901	21790
8	19,9	1664,1	0,808	0,308	9,023	15016
9	8,7	1300,6	0,642	0,142	4,146	5392
10	21,3	1700,1	0,942	0,442	12,925	21975
11	25,5	1811,5	0,692	0,192	5,609	10161
12	18,5	1651,5	0,558	0,058	1,707	2819
13	33,9	2096,1	0,725	0,225	6,585	13802
14	22,7	1737,9	0,858	0,358	10,487	18225
15	32,5	2031,1	0,525	0,025	0,732	1486
16	29,7	1908,1	0,508	0,008	0,244	465
17	28,3	1875,9	0,625	0,125	3,658	6862
18	38,1	2263,4	0,758	0,258	7,560	17111
19	26,9	1843,7	0,575	0,075	2,195	4047

20	35,3	2161,1	0,992	0,492	14,389	31095
21	47,9	2459,3	0,875	0,375	10,974	26989
22	31,1	1966,1	0,925	0,425	12,438	24454
23	46,5	2435,5	0,825	0,325	9,511	23164
24	40,9	2322,9	0,958	0,458	13,413	31157
25	45,1	2411,7	0,592	0,092	2,683	6470
26	43,7	2382,4	0,542	0,042	1,219	2905
27	36,7	2226,1	0,908	0,408	11,950	26601
28	39,5	2293,1	0,792	0,292	8,536	19573
29	49,3	2483,1	0,658	0,158	4,634	11506
30	42,3	2352,6	0,742	0,242	7,072	16639

Omezující podmínka 5.1.3 stanovující maximální dovolenou cenu materiálu nebyla žádnou vygenerovanou variantou porušena.

5.2.4 POSOUZENÍ VYGENEROVANÝCH VARIANT

Varianty vygenerované metodou LHS mean je nutné posoudit z hlediska únosnosti a následně je konfrontovat s omezující podmínkou (5.1.2). Ke stanovení únosnosti jednotlivých variant poslouží zatěžovací křivky nosníků vypočtené programem ATENA 2D. Vzhledem k rozmístění monitorů a jejich nastavení v rámci programu ATENA (viz podkapitola 5.2.1) budou výsledné křivky reprezentovat závislost mezi reakcí v levé podpoře a svislým posunem uprostřed nosníku. Rozborem zatěžovacích křivek určíme reakci v podpoře na mezi únosnosti. Od takto získané reakce odečteme část vyvolanou vlastní tíhou konstrukce R_{cs} (vypočteno dle vztahu 5.2.4.3) a výsledek porovnáme s hodnotou R_{cn} stanovenou vztahem (5.1.1). Varianty nesplňující požadavky kladené omezujícími podmínkami při dalším postupu ignorujeme. Srovnání zatěžovacích křivek jednotlivých variant je zobrazeno v grafu v příloze P5. Výsledky rozboru únosnosti ze zatěžovacích křivek jsou uspořádány v tabulce (5.2.4.1).

Výpočet hodnoty R_{cs} :

$$R_{cs} = \frac{V_{(h)} \times g_{zb}}{2} \quad (5.2.4.1)$$

Kde $V_{(h)}$ je funkcí objemu konstrukce v závislosti na výšce nadbetonování h

$$V_{(h)} = 6,3 \times 0,5 \times 15,583 + 15,583 \times 1,878 \times h = 49,086 + 29,265 \times h \quad [\text{m}^3] \quad (5.2.4.2)$$

g_{zb} je objemová tíha železobetonu v našem případě uvažovaná:

$$g_{zb} = 25 \frac{\text{kN}}{\text{m}^3}$$

Betony různé kvality se mohou svou objemovou tíhou lišit. Změna objemové tíhy v závislosti na kvalitě betonu je však minimální. V našem modelu proto tento fakt zanedbáme.

Dosažením této hodnoty a vztahu (5.2.4.2) do rovnice (5.2.4.1) získáme:

$$R_{cs} = \frac{(49,086 + 29,265 \times h) \times 25}{2} = 613,575 + 365,813h \quad [\text{kN}] \quad (5.2.4.3)$$

Tab. (5.2.4.1) Výsledky rozboru únosnosti ze zatěžovacích křivek

N _{sim}	f _c [MPa]	h [m]	R _{cs} [kN]	R _{cn} [kN]	R _{cs} +R _{cn}	R _{cc} [kN]	plnění podmínky 5.1.2
1	10,1	0,208	689,664	315,000	1004,664	954,000	Nevyhoví
2	11,5	0,108	653,083	315,000	968,083	879,000	Nevyhoví
3	17,1	0,392	756,974	315,000	1071,974	1460,000	Vyhoví
4	14,3	0,342	738,683	315,000	1053,683	1290,000	Vyhoví
5	15,7	0,275	714,173	315,000	1029,173	1210,000	Vyhoví
6	24,1	0,175	677,592	315,000	992,592	1040,000	Vyhoví
7	12,9	0,475	787,336	315,000	1102,336	1500,000	Vyhoví
8	19,9	0,308	726,245	315,000	1041,245	1310,000	Vyhoví
9	8,7	0,142	665,520	315,000	980,520	893,000	Nevyhoví
10	21,3	0,442	775,264	315,000	1090,264	1590,000	Vyhoví
11	25,5	0,192	683,811	315,000	998,811	1130,000	Vyhoví
12	18,5	0,058	634,792	315,000	949,792	872,000	Nevyhoví
13	33,9	0,225	695,883	315,000	1010,883	1200,000	Vyhoví
14	22,7	0,358	744,536	315,000	1059,536	1420,000	Vyhoví
15	32,5	0,025	622,720	315,000	937,720	896,000	Nevyhoví
16	29,7	0,008	616,502	315,000	931,502	883,000	Nevyhoví
17	28,3	0,125	659,302	315,000	974,302	958,000	Nevyhoví
18	38,1	0,258	707,955	315,000	1022,955	1270,000	Vyhoví
19	26,9	0,075	641,011	315,000	956,011	907,000	Nevyhoví
20	35,3	0,492	793,555	315,000	1108,555	1860,000	Vyhoví
21	47,9	0,375	750,755	315,000	1065,755	1610,000	Vyhoví
22	31,1	0,425	769,045	315,000	1084,045	1690,000	Vyhoví
23	46,5	0,325	732,464	315,000	1047,464	1450,000	Vyhoví
24	40,9	0,458	781,117	315,000	1096,117	1840,000	Vyhoví
25	45,1	0,092	647,230	315,000	962,230	990,000	Vyhoví
26	43,7	0,042	628,939	315,000	943,939	891,000	Nevyhoví
27	36,7	0,408	762,827	315,000	1077,827	1640,000	Vyhoví
28	39,5	0,292	720,392	315,000	1035,392	1350,000	Vyhoví
29	49,3	0,158	671,373	315,000	986,373	1120,000	Vyhoví
30	42,3	0,242	702,102	315,000	1017,102	1330,000	Vyhoví

Omezující podmínka optimalizace (5.1.2) byla porušena u realizací vektorů náhodných proměnných číslo: 1, 2, 9, 12, 15, 16, 17, 19 a 26 (v tabulce vyznačeny červeně). Zbylé realizace vektoru náhodných veličin nejsou v rozporu s omezujícími podmínkami optimalizace a můžeme s nimi i nadále počítat.

Nyní máme stanovenou únosnost i cenu při jednotlivých realizacích. Vybraná varianta nadbetonování by měla být co nejehospodárnější a její únosnost co nejvyšší. Ponecháme-li oběma zmiňovaným parametrům stejnou váhu při konečném rozhodování, pak se jako nejvhodnější kritérium výběru optimální varianty jeví srovnání poměrů cena – únosnost jednotlivých případů. Je vhodné stanovit pro všechny varianty „efektivní“ únosnost tj. únosnost stanovená na základě maximální reakce v podpoře snižená o část vyvolanou vlastní tíhou konstrukce. Efektivní únosnost se tedy stanoví rozdílem:

$$R_u = R_{cc} - R_{cs} \quad (5.2.4.4)$$

Význam užitých symbolů viz kapitola 5.1

Hledáme tedy takovou variantu, u níž bude hodnota poměru cena - efektivní únosnost nejnižší. Ceny, únosnosti a jejich vzájemný poměr jsou pro jednotlivé realizace náhodného vektoru uvedeny v tabulce (5.2.4.2).

Tab. (5.2.4.2) Ceny, únosnosti a jejich vzájemný poměr

N_{sim}	f_c [MPa]	h [m]	R_{cs} [kN]	R_{cc} [kN]	R_u [kN]	$C_{(h, f_c)}$ [Kč]	$C_{(h, f_c)} / R_u$
3	17,1	0,392	756,974	1460,000	703,027	18785	26,720
4	14,3	0,342	738,683	1290,000	551,317	16023	29,063
5	15,7	0,275	714,173	1210,000	495,827	13112	26,445
6	24,1	0,175	677,592	1040,000	362,408	9094	25,093
7	12,9	0,475	787,336	1500,000	712,664	21790	30,575
8	19,9	0,308	726,245	1310,000	583,755	15016	25,723
10	21,3	0,442	775,264	1590,000	814,736	21975	26,972
11	25,5	0,192	683,811	1130,000	446,189	10161	22,773
13	33,9	0,225	695,883	1200,000	504,117	13802	27,379
14	22,7	0,358	744,536	1420,000	675,464	18225	26,981
18	38,1	0,258	707,955	1270,000	562,045	17111	30,444
20	35,3	0,492	793,555	1860,000	1066,445	31095	29,158
21	47,9	0,375	750,755	1610,000	859,245	26989	31,410
22	31,1	0,425	769,045	1690,000	920,955	24454	26,553
23	46,5	0,325	732,464	1450,000	717,536	23164	32,283
24	40,9	0,458	781,117	1840,000	1058,883	31157	29,424
25	45,1	0,092	647,230	990,000	342,770	6470	18,876
27	36,7	0,408	762,827	1640,000	877,174	26601	30,326
28	39,5	0,292	720,392	1350,000	629,608	19573	31,088
29	49,3	0,158	671,373	1120,000	448,627	11506	25,647
30	42,3	0,242	702,102	1330,000	627,898	16639	26,500

Tabulka (5.2.4.2) již neobsahuje vzorky vyřazené z důvodu porušení omezujících podmínek optimalizace. Po porovnání všech zbylých variant pomocí hodnoty poměru cena – únosnost se jako nejefektivnější (v rámci stanovených optimalizačních podmínek) ukázala být varianta náhodného vektoru číslo 25 (v tabulce vyznačena červeně). Této variantě odpovídá cena 6470 Kč a výška nadbetonování 0,092 m při užití betonu s hodnotou krychelné pevnosti v tlaku rovnou 45,1 MPa.

Uvedená úloha je pouze ilustrativní a výsledná cena materiálu není příliš vysoká. V praxi by byl pravděpodobně vyšší význam přiřazen únosnosti konstrukce. Pro hodnocení výsledků s rozdílnou důležitostí posuzovaných parametrů je možné zavést „váhový systém“ kdy je každé z vygenerovaných realizací přiděleno množství bodů odpovídající jejímu umístění podle výsledné ceny a únosnosti. Maximální množství bodů získané za jednotlivé parametry odpovídá počtu realizací (v našem případě 21 realizací splňujících omezující podmínky), minimální množství bodů přidělených za parametr je 1. Body za cenu a únosnost se poté sečtou a realizace s nejvyšším počtem bodů je prohlášena za hledané optimum. Důležitost jednotlivých parametrů (cena, únosnost) je možné dle požadavků investora zvýraznit či snížit vhodným koeficientem. Je-li například únosnost konstrukce o 10% důležitější než cena, může výpočet celkového množství bodů vypadat takto:

$$C = A + 1,1B \quad (5.2.4.5)$$

Kde C je celkový počet bodů. A je počet bodů přidělených za cenu. B je počet bodů přidělených za únosnost.

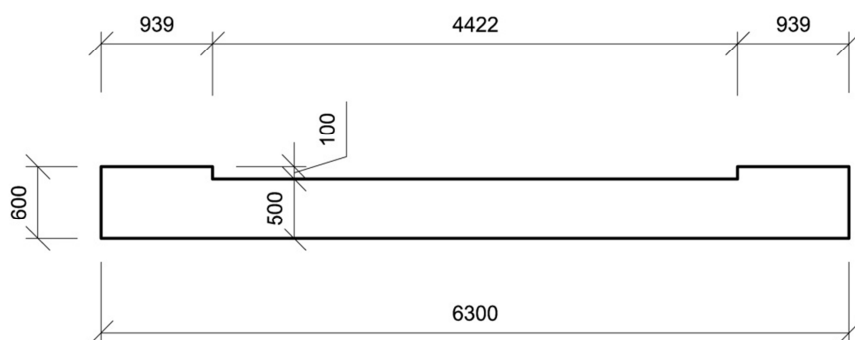
Po aplikaci tohoto postupu na námi optimalizovanou úlohu s využitím kritéria (5.2.4.5) by nejvyššího bodového ohodnocení (25,9 bodů) dosáhla realizace číslo 22 s příslušnou hodnotou pevnosti v tlaku $f_c = 31,1$ MPa, výškou nadbetonování $h = 0,425$ m a cenou 24 454 Kč.

Způsob hodnocení je silně závislý na požadavcích investora a bez další specifikace požadavků budeme dále považovat za optimální řešení variantu 25.

5.2.5 VYHODNOCENÍ

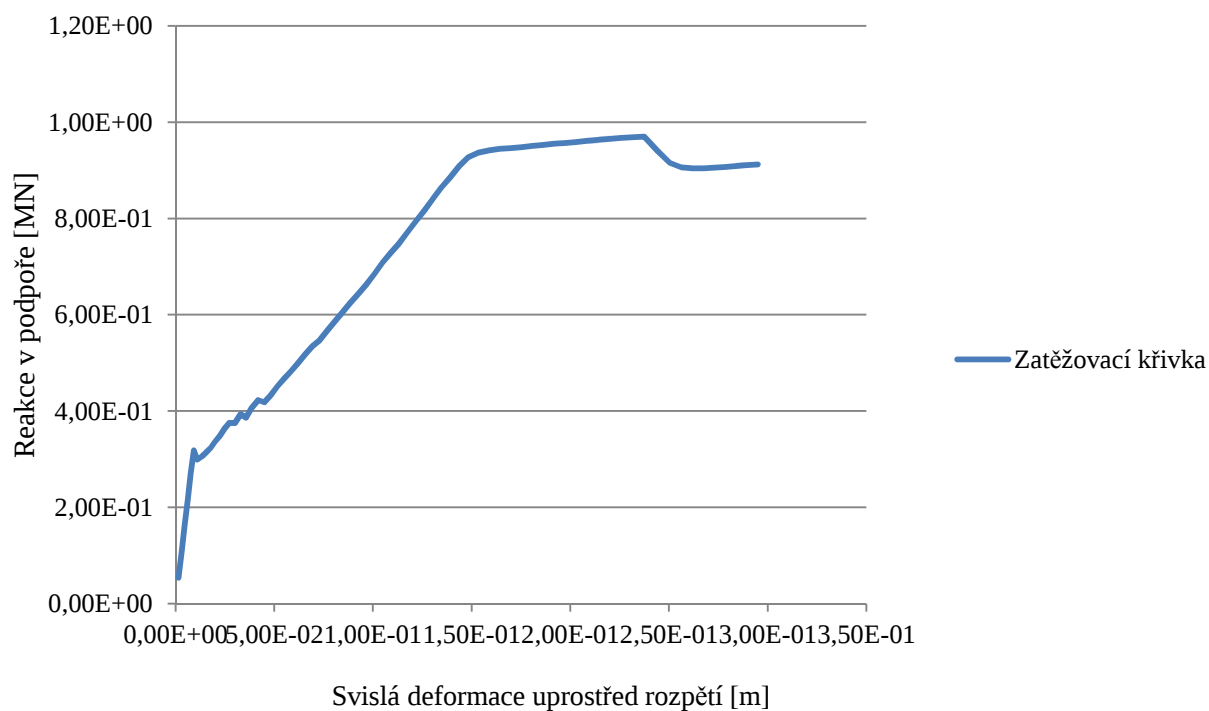
Nalezené optimum neodpovídá svou pevností v tlaku kvalitativnímu roztřídění prodáváných betonových směsí. Pro účely praktického návrhu proto musíme použít beton o třídě pevnosti co nejbližze optimálnímu řešení. Zvolená varianta vykazuje pevnost v tlaku 45,1 MPa. Nejbližší třída pevnosti betonu je tedy C40/50 které odpovídá jednotková cena 2495 Kč/m³. Také výšku nadbetonování je z realizačních důvodů vhodné zaokrouhlit na hodnotu: $h=100$ mm. Doporučený tvar průřezu je zachycen na obrázku (5.2.5.1). Výsledná cena materiálu se stanoví dle vztahu (5.2.3.1) takto:

$$C_{(h, f_c)} = 29,265 \times 0,1 \times 2495 = 7302 \text{ Kč}$$



Obr. (5.2.5.1) Výsledný průřez

Pro účel posouzení byl v programu ATENA 2D vytvořen model navržené konstrukce. Principy tvorby tohoto modelu zůstaly shodné s postupem uvedeným v kapitole 5.2.1 při použití navržených konstrukčních a materiálových parametrů (beton C40/50, výška nadbetonování 100 mm). Křivka zatížení - přetvoření navržené konstrukce je zachycena v grafu (5.2.5.1).



Graf (5.2.5.1) Křivka zatížení - přetvoření navržené konstrukce

Rozborem zatěžovací křivky určíme maximální hodnotu reakce v podpoře. K porušení konstrukce došlo v 62. výpočtovém kroku při svislé deformaci $2,95\text{E-}01$ m. Tomuto bodu odpovídá reakce v podpoře $9,70\text{E-}01$ MN. Reakce odpovídající vlastní tíze mostu se vypočte podle vztahu (5.2.4.3) takto:

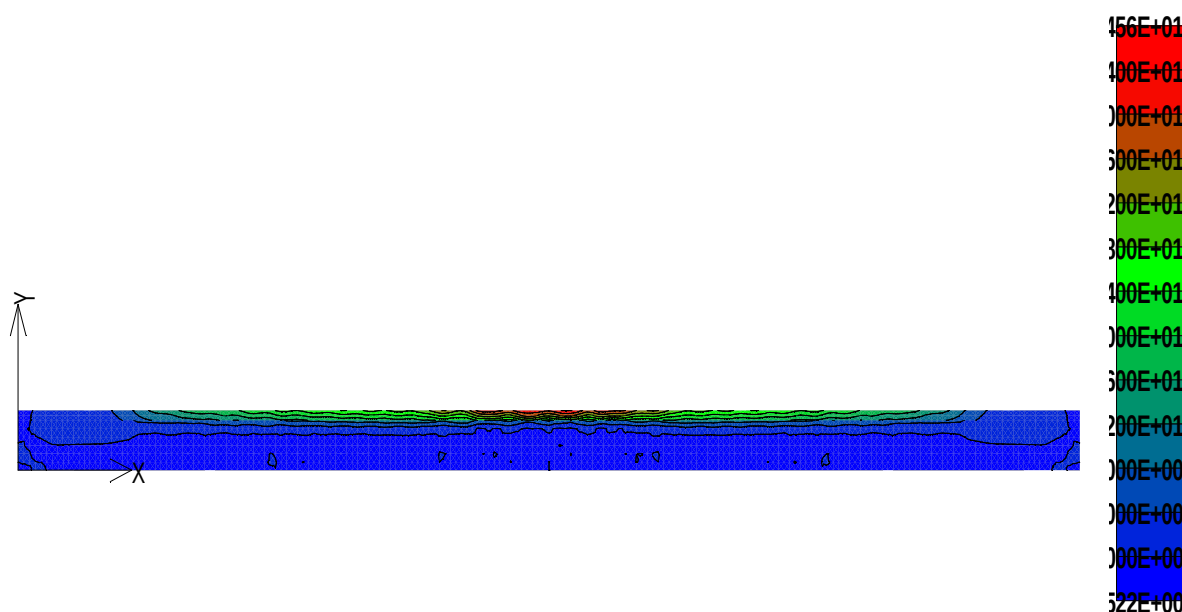
$$R_{cs} = \frac{(49,086 + 29,265 \times 0,1) \times 25}{2} = 613,575 + 365,813 \times 0,1 = 650,156 \text{ kN}$$

Posouzení plnění omezující podmínky optimalizace (5.1.2):

$$R_{cc} = 970 \text{ kN} \geq R_{cs} + R_{cn} = 650,156 + 315 = 965,156 \text{ kN} \quad \underline{\text{Splněno}}$$

Je vidět že únosnost konstrukce je při splnění všech omezujících podmínek téměř zcela využita. Rezerva únosnosti činí pouhých 0,5%. Návrh je tedy velice hospodárný.

Průběh napětí v konstrukci při dosažení meze pevnosti je zobrazen na obrázku (5.2.5.2). Deformovaná konstrukce se zobrazením trhlin je zachycena na obrázku (5.2.5.3).



Obr. (5.2.5.2) Průběh napětí v konstrukci při dosažení meze pevnosti



Obr. (5.2.5.3) Deformovaná konstrukce se zobrazením trhlin

Uvedená úloha je pouze ilustrativní a slouží k demonstraci základních optimalizačních postupů při použití metody LHS mean. Nalezené řešení odpovídá stanoveným vstupním požadavkům. V praxi řešený problém by vyžadoval z důvodu zajištění vyšší bezpečnosti přísnější nastavení omezujících podmínek optimalizace. V rámci zadaných mezí se však podařilo nalézt nejvhodnější řešení.

ZÁVĚR

Využití optimalizačních metod je ve stavebním průmyslu stále okrajovou záležitostí. Tvorba matematického modelu reálné konstrukce naráží na řadu omezení, jímž musí projektant čelit, ať už se jedná o problém randomizace materiálových parametrů, či volbu vhodného optimalizačního algoritmu. Úspora finančních prostředků či materiálu navíc u konstrukcí menšího rozsahu nevyváží zpravidla ani náklady vynaložené na samotnou optimalizaci. Užití optimalizačních metod má proto opodstatnění zejména u prefabrikovaných konstrukcí, u nichž je množství prostředků vynaložených na optimalizaci prvku kompenzováno úsporami při opakované výrobě daného prvku. Další možností budoucího využití optimalizačních metod ve stavebnictví je navrhování rozsáhlých a náročných konstrukcí, pro které bude důležitá úspora hmotnosti, optimální vynaložení nákladů a zajištění požadované spolehlivosti. Rozvoj výpočetní techniky a softwarového vybavení zjednoduší budoucím projektantům práci s optimalizačními algoritmy, jež se snad stanou (tak jako tomu je už dnes v jiných oborech) běžnou součástí stavební praxe. Tato práce studuje a naznačuje možnosti tzv. pseudostochastické optimalizace vhodné pro výpočtově náročnější optimalizační úlohy mechaniky kontinua.

POUŽITÉ INFORMAČNÍ ZDROJE

- [1] TEPLÝ, B. a D. NOVÁK. Spolehlivost stavebních konstrukcí: teorie, numerické metody, navrhování, software : skriptum FAST VUT. Vyd. 2., opr. Brno: CERM, 2004, 89 s. Učební texty vysokých škol (Vysoké učení technické v Brně. Stavební fakulta). ISBN 80-214-2577-6.
- [2] SILBRNÍK, V. Pravděpodobnostní optimalizace nosných stavebních konstrukcí metodou evolučních strategií. brno, 9.6. 1994. Diplomová práce. Vysoké učení technické v Brně. Vedoucí práce Ing. Drahomír NOVÁK, CSc
- [3] MAŘÍK, V., O. ŠTĚPÁNKOVÁ, J. LAŽANSKÝ a kol. Umělá inteligence (3). 1. vyd. Praha: Academia, 2001, 328 s. ISBN 80-200-0472-6.
- [4] GEATbx [online]. [cit. 2012-03-27]. Dostupné z: <http://www.geatbx.com/>
- [5] VOŘECHOVSKÝ, M. a D. NOVÁK. Correlation control in small-sample Monte Carlon type simulations: A simulated annealing approach. ELSVIER: Probabilistic Engineering Mechanics. 2009, s. 11.
- [6] ČERNÍK, F. *Ověření výkonnosti metody HSLHS pro odhad statistik funkcí náhodných vektorů v úlohách mechaniky*. Brno, 2009. Studentská vědecká odborná činnost. Vysoké učení technické v Brně Fakulta stavební. Vedoucí práce doc. Ing. Miroslav Vořechovský, Ph.D.
- [7] ČERVENKA CONSULTING LTD. *FReET* [online]. [cit. 2012-03-27]. Dostupné z: <http://www.freet.cz/>
- [8] ČERVENKA CONSULTING LTD. *ATENA, SARA* [online]. [cit. 2012-03-27]. Dostupné z: <http://www.cervenka.cz/>
- [9] KREJSA, M. Přednášky z předmětu Pravděpodobnostní posuzování konstrukcí. In: *Katedra stavební mechaniky Vysoká škola báňská - Technická univerzita Ostrava* [online]. [cit. 2012-05-12]. Dostupné z: <http://fast10.vsb.cz/krejsa/posuzovani.htm>
- [10] SPH STAVBY. *Sph-stavby* [online]. 2002 [cit. 2012-05-12]. Dostupné z: http://www.sph-stavby.cz/pdf/cenik%20-%20nova%20betonarna_%20sph_1.pdf
- [11] PUKL, R., V. ČERVENKA, A. STRAUSS, K. BERGMEISTER a D. NOVÁK. An advanced engineering software for probabilistic-based assessment of concrete structures using nonlinear fracture mechanics. s. 7.

SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

E	[MPa]	Youngův modul pružnosti
f_c	[MPa]	pevnost betonu v tlaku
f_t	[MPa]	pevnost betonu v tahu
G_f	[kJ]	lomová energie

SEZNAM PŘÍLOH

P1 – Ukázka algoritmu metody simulovaného žíhaní zapsaného v programu MATLAB

P2 - Schéma tvaru konstrukce

P3 – Schéma vyztužení

P4 – Schéma podepření mostu

P5 – Graf závislosti reakce v podpoře na deformaci v polovině rozpětí vygenerovaných Variant

PŘÍLOHA P1:

Ukázka algoritmu metody simulovaného žíhání zapsaného v programu MATLAB:

```
def = struct(...
    'CoolSched',@(T) (.8*T),...
    'Generator',@(x) (x+(randperm(length(x))==length(x))*randn/100),...
    'InitTemp',1e-8,...
    'MaxConsRej',1000,...
    'MaxSuccess',20,...
    'MaxTries',100,...
    'StopTemp',1e-9,...
    'StopVal',-Inf,...
    'Verbosity',1);

% kontrola vstupů
if ~nargin %V případě, že uživatel použije předdefinované nastavení program
    ho vrátí a zastaví se
    minimum = def;
    return
elseif nargin<2, %uživatel zadal pouze objektivní funkci, vypsát chybu
    error('MATLAB:anneal:noParent','You need to input a first guess.');
```

elseif nargin<3, %uživatel nezadal vlastní nastavení, použije se předdefinované
options=def;

else %uživatel zadal všechny vstupy, kontrola kompletnosti vstupů
if ~isstruct(options)
 error('MATLAB:anneal:badOptions',...
 'Input argument 'options' is not a structure')

end
fs = {'CoolSched','Generator','InitTemp','MaxConsRej',...
 'MaxSuccess','MaxTries','StopTemp','StopVal','Verbosity'};

for nm=1:length(fs)
 if ~isfield(options,fs{nm}), options.(fs{nm}) = def.(fs{nm}); end
end
end

% hlavní nastavení
newsol = options.Generator;
Tinit = options.InitTemp; % počáteční teplota
minT = options.StopTemp; % konečná teplota
cool = options.CoolSched; % harmonogram žíhání
minF = options.StopVal;
max_consec_rejections = options.MaxConsRej;
max_try = options.MaxTries;
max_success = options.MaxSuccess;
report = options.Verbosity;
k = 1; % boltzmannova konstanta

itry = 0;
success = 0;
finished = 0;
consec = 0;
T = Tinit;
initenergy = loss(parent);
oldenergy = initenergy;
total = 0;
if report==2, fprintf(1,'\n T = %7.5f, loss = %10.5f\n',T,oldenergy); end

while ~finished;

```
itry = itry+1; % počítadlo iterací
current = parent;

% % zastavovací / kritéria snižující teplotu
if itry >= max_try || success >= max_success;
    if T < minT || consec >= max_consec_rejections;
        finished = 1;
        total = total + itry;
        break;
    else
        T = cool(T); % snížení teploty dle ochlazovacího harmonogramu
        if report==2, % output
            fprintf(1, ' T = %7.5f, loss = %10.5f\n', T, oldenergy);
        end
        total = total + itry;
        itry = 1;
        success = 1;
    end
end

newparam = newsol(current);
newenergy = loss(newparam);

if (newenergy < minF),
    parent = newparam;
    oldenergy = newenergy;
    break
end

if (oldenergy-newenergy > 1e-6)
    parent = newparam;
    oldenergy = newenergy;
    success = success+1;
    consec = 0;
else
    if (rand < exp( (oldenergy-newenergy)/(k*T) ));
        parent = newparam;
        oldenergy = newenergy;
        success = success+1;
    else
        consec = consec+1;
    end
end

end

minimum = parent;
fval = oldenergy;

if report;
    fprintf(1, '\n Initial temperature: \t%g\n', Tinit);
    fprintf(1, ' Final temperature: \t%g\n', T);
    fprintf(1, ' Consecutive rejections: \t%i\n', consec);
    fprintf(1, ' Number of function calls:\t%i\n', total);
    fprintf(1, ' Total final loss: \t%g\n', fval);
end
```

Vysvětlivky zadávaných parametrů:

'CoolSched'	- Harmonogram ochlazování
'Generator'	- Generátor nového řešení ze starého
'InitTemp'	- Počáteční teplota
'MaxConsRej'	- Maximální počet po sobě následujících zamítnutí vygenerovaného vzorku
'MaxSuccess'	- Maximální počet po sobě následujících úspěšných iterací se stejnou teplotu
'MaxTries'	- Maximální počet iterací beze změny teploty
'StopTemp'	- Konečná teplota
'StopVal'	- Konečná hodnota minimalizace (uživatel stanovený uspokojivý výsledek)
'Verbosity'	- Volba množství zobrazovaných parametrů: 0-potlačí všechny výstupy 1-zobrazí jen závěrečnou zprávu 2-zobrazí změny teploty a závěrečnou zprávu

Příklad zadání (Six-hump camelback function):

```
>>camel = @(x,y)(4-2.1*x.^2+x.^4/3).*x.^2+x.*y+4*(y.^2-1).*y.^2;  
loss = @(p)camel(p(1),p(2));  
[x f] = ANNEAL(loss,[0 0])           - zadání počátečního bodu pro SA
```

Po potvrzení dostaneme závěrečnou zprávu:

```
Initial temperature:      1  
Final temperature:       3.21388e-007  
Consecutive rejections:  1027  
Number of function calls: 6220  
Total final loss:        -1.03163
```

```
x =  
-0.0899    0.7127
```

```
f =  
-1.0316
```

PŘÍLOHA P2:

Schéma tvaru konstrukce

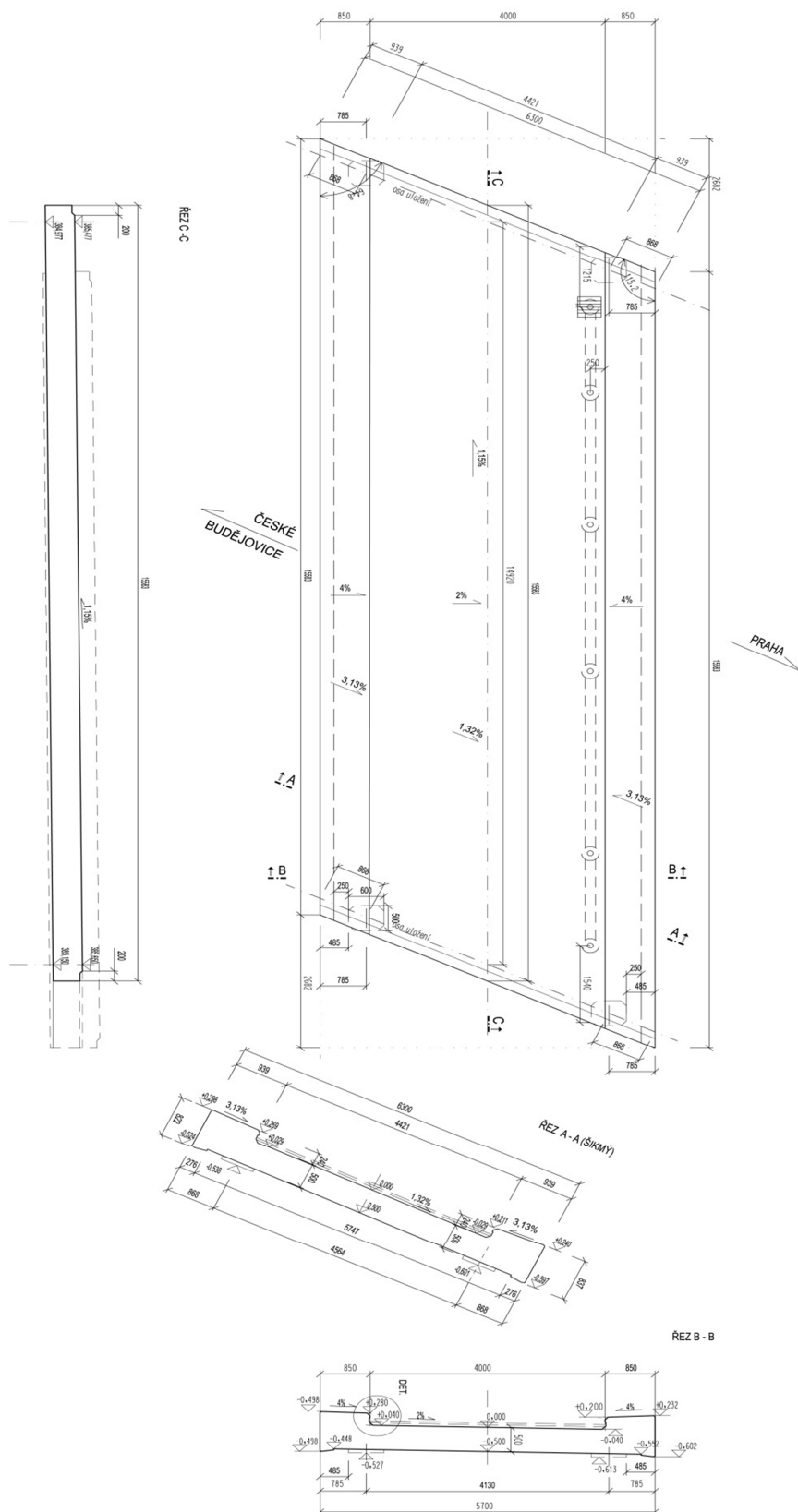
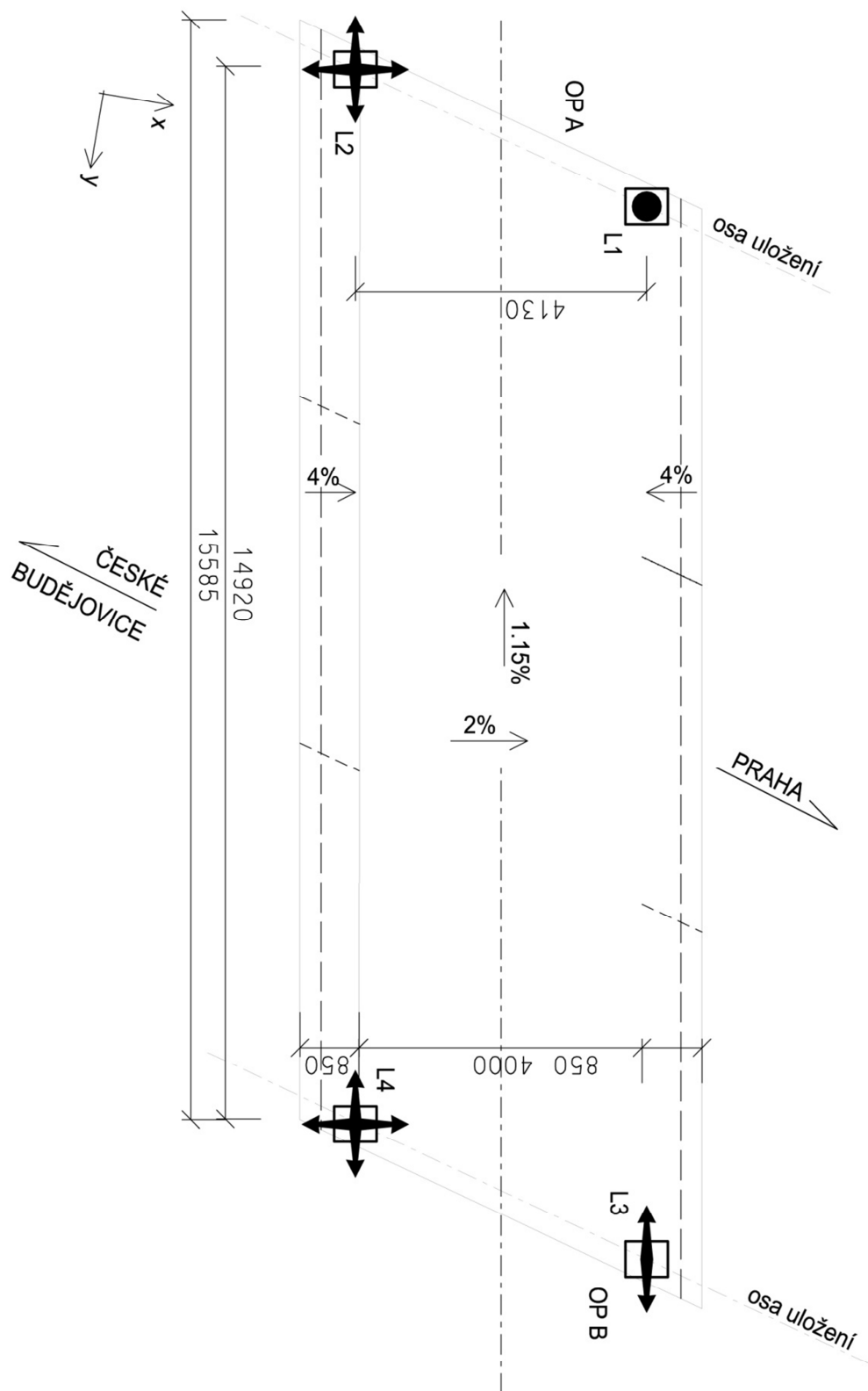


Schéma vyztužení



PŘÍLOHA P4:

Schéma podepření mostu



PŘÍLOHA P5:

Graf závislosti reakce v podpoře na deformaci v polovině rozpětí vygenerovaných variant

