

## **AUTOMATED DESIGN OF REINFORCED CONCRETE FRAMES**

**M. Lepš\*, J. Zeman\*, Z. Bittnar\***

**Summary:** *This paper presents discrete optimization of reinforced concrete structures based on an efficient combination of deterministic and stochastic optimization strategies. The deterministic optimization algorithm is used for the detailing of a reinforced concrete cross-section for a given combination of internal forces. The stochastic optimization algorithm, based on the Augmented Simulated Annealing, is used for the optimization of a whole structure in terms of basic structural characteristics like types of materials, dimensions of elements or profiles of steel bars. The implicit parallelization of genetic algorithm is implemented to reduce the overall computational time.*

### **1. Úvod**

Snaha vytvořit efektivní algoritmus na návrh železobetonových konstrukcí je patrná již v dřívějších epochách stavebního inženýrství. Prudký rozvoj železobetonových konstrukcí v posledních letech tuto snahu ještě podporuje, zejména v kandidátských zemích Evropské Unie. Jednou z nejčastějších realizací železobetonových staveb jsou rámové skelety. Proto jsme se soustředili na vytvoření návrhového nástroje, který by jednoduše a spolehlivě navrhl a zoptimalizoval železobetonovou rámovou konstrukci. V předchozích pracích [5, 6, 7, 9] jsme vyzkoušeli několik typů návrhových algoritmů spolu s několika optimalizačními metodami založenými na principech genetických algoritmů. Bylo by jistě pěkné, kdyby se dal nastíněný problém řešit vcelku jako jeden optimalizační problém, ale množina všech možných řešení je příliš velká. Ukazuje se, že je nevyhnutelné úlohu rozdělit na dvě části – na návrh na úrovni jednotlivých průřezů a na optimalizaci celé konstrukce na úrovni jejích globálních vlastností, jako jsou typy materiálů, dimenze dílčích stavebních elementů nebo průřezy výztužných profilů.

V první části je snahou navrhnout výztuž daného průřezu tak, aby příslušný interakční diagram pokryl odezvu konstrukce na všechny zatěžovací stavy. K tomu je zapotřebí rychlá metoda výpočtu interakčního diagramu. Algoritmus navržený R. Vondráčkem v [13] je schopen analytickou integrací spočítat interakční diagram libovolného polygonálního průřezu ze znalosti pracovních diagramů použitých materiálů. Návrh výztuže příčného řezu se pak zjednoduší na pouhé nalezení vhodné kombinace výztužení.

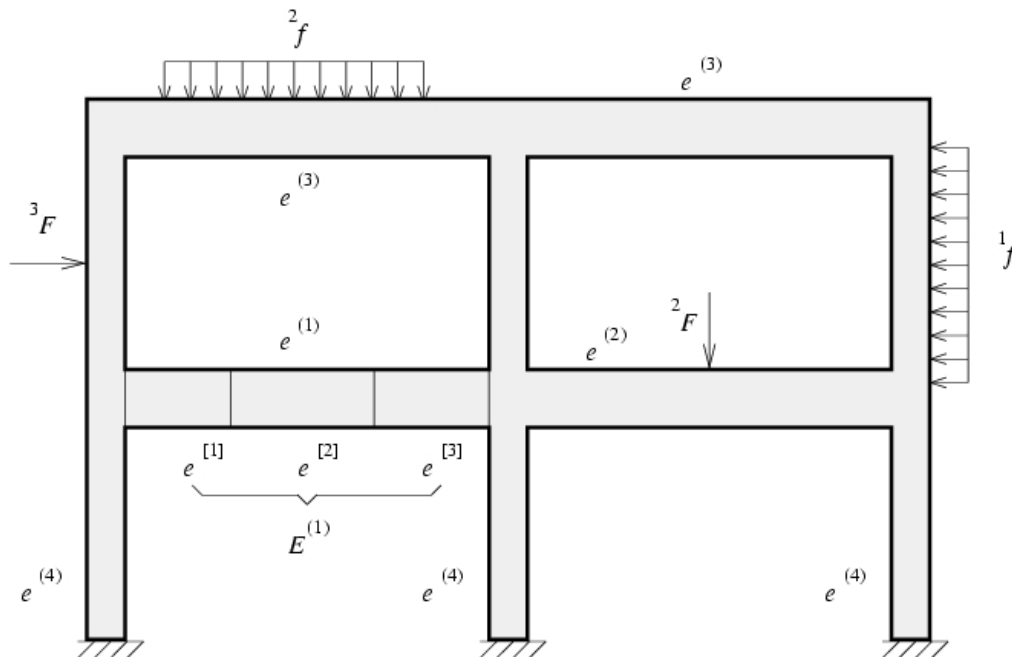
---

\* Ing. Matěj Lepš, Ing. Jan Zeman, Prof. Ing. Zdeněk Bittnar, DrSc.: Katedra stavební mechaniky, Stavební fakulta ČVUT, Thákurova 7, 166 29, Praha 6; tel.: +420-2-2435-5417, fax: +420-2-2431-0775; e-mail: [leps@cml.fsv.cvut.cz](mailto:leps@cml.fsv.cvut.cz)

Druhá část „automatického“ návrhu železobetonového rámu se zabývá návrhem proporcí jednotlivých konstrukčních elementů, jako jsou rozměry příčného řezu, výběrem profilu podélné a smykové výztuže apod. Z matematického hlediska se jedná o hledání optimální kombinace známých vstupů za omezujících podmínek jako jsou požadavky mezních stavů únosnosti a použitelnosti či různých konstrukčních omezení. Naše předchozí zkušenost [6, 7, 9] ukazuje, že stochastické optimalizační algoritmy, zejména tzv. genetické algoritmy, jsou schopny tuto úlohu efektivně řešit. Námi navržená metoda, celočíselné rozšířené simulované žíhání s diferenciálním operátorem, v předchozích testech na podobné úloze vykázala slibné výsledky [5]. Základními principy této metody jsou evoluční teorie přežití silnějších jedinců (analogicky optimalita jednotlivých řešení) spolu se simulovaným žíháním, celočíselným kódováním, diferenciálním křížením a mutací s Gaussovým náhodným rozložením.

Nevýhodou téměř všech optimalizací je výpočtová náročnost jak samostatné optimalizace, tak zejména analýzy metodou konečných prvků. S výhodou zde využíváme implicitní paralelizace vlastního genetického algoritmu [8], kde jsou jednotlivá potenciální řešení na sobě nezávislá a tudíž oddělitelná. Program je tedy rozdělen na optimalizační a výpočetní část a v této podobě je naprogramován na clusteru osobních počítačů na řešitelském pracovišti.

Zbývající část příspěvku je rozdělena následovně: v kapitole 2 je zformulována úloha a jsou představeny optimalizované proměnné. Optimalizace příčného řezu je nastíněna v kapitole 3 a použitá stochastická optimalizační metoda je popsána v kapitole 4. V kapitole 5 se stručně zmíníme o použitém paralelním schématu a příspěvek zakončíme příklady v kapitole 6.



Obrázek 1: Příklad rámové konstrukce

## 2. Formulace problému

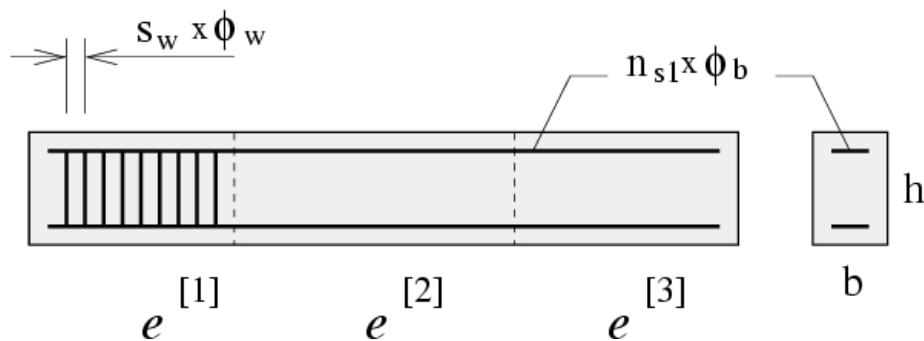
Jak již bylo předesláno v úvodu, snažíme se navrhnout rámovou konstrukci, jejíž pořizovací hodnota bude minimální, ale zároveň bude splňovat všechny (vybrané) podmínky podle

zvolené návrhové normy. V našem případě jsme si vybrali normu EUROCODE 2 (EC2) [4]. Pro zjednodušení se zabýváme pouze řešením problému ve 2D a uvažujeme pouze obdélníkové průřezy. To znamená, že se konstrukce nalézá v rovině  $xz$  a naše pozornost se tak zaměřuje pouze na vnitřní síly v této rovině – ohybový moment  $M_y$ , normálovou sílu  $N_x$  a smykovou sílu  $V_z$ . Rozšíření na obecný trojrozměrný případ je pak vcelku přímočaré a bude uvažováno v blízké budoucnosti.

Jak z konstrukčního, tak i z optimalizačního hlediska se zdá praktické rozdělit celou konstrukci na několik  $n_d$  návrhových elementů (viz Obr. 1). Tyto uživatelem definované prvky mohou představovat ty části konstrukce, které sdílí z hlediska optimalizace shodné parametry, jako jsou např. průměry výztužných profilů apod. Dále předpokládáme, že je konstrukce rozdělena na  $n_e$  konečněprvkových elementů, které jsou zapotřebí k získání vnitřních sil z numerické analýzy. Podle tohoto rozdělení budeme označovat jakoukoliv veličinu  $X$  příslušnou k  $i$ -tému návrhovému elementu jako  $X^{(i)}$  a k  $i$ -tému konečněprvkovému elementu jako  $X^{[i]}$ , tj. hodnoty příslušné k návrhovým elementům jsou uvedeny s kulatými závorkami, zatímco vlastnosti konečněprvkových elementů jsou značeny hranatými závorkami. Symboly  $e^{(i)}$  a  $e^{[i]}$  značí  $i$ -tý návrhový a konečněprvkový element, zatímco  $E^{(i)}$  označuje množinu konečných prvků příslušných k  $i$ -tému návrhovému elementu, nebo-li

$$E^{(i)} = \left\{ e^{[j]} : e^{[j]} \cap e^{(i)} \neq \emptyset, j = 1, \dots, n_e \right\}. \quad (1)$$

Nakonec předpokládáme, že je konstrukce zatížena  $n_c$  zatěžovacími stavy. Veličinu příslušnou k  $i$ -tému návrhovému elementu a k  $c$ -tému zatěžovacímu stavu pak budeme označovat  ${}^cX^{(i)}$ . Poznamenejme, že analýza konstrukce je prováděna pomocí C++ MKP programu SIFEL, který je vyvíjen na Katedře stavební mechaniky, Fakulty stavební ČVUT.



Obrázek 2: Příklad návrhového elementu

Jak již bylo dříve řečeno, návrhový element slouží k definici základních optimalizačních parametrů (viz Obr. 2). Těmi jsou rozměry příčného řezu  $b$  a  $h$ , profil ohybové výztuže  $\phi_b$ , počet výztužných vložek u dolního a horního povrchu  $n_{s1}$  a  $n_{s2}$ , průměr smykové výztuže  $\phi_w$  a vzdálenost třímínků  $s_w$ . Dále předpokládáme, že rozměry příčného řezu a vzdálenosti třímínků mohou nabývat pouze určitých hodnot, odstupňovaných např. po 25 mm, zatímco  $\phi_b$  a  $\phi_w$  jsou vybrány ze seznamu dostupných profilů.

Omezující podmínky vystupující v optimalizačním problému jsou určeny vybranou návrhovou normou EC 2 [4]. Z obecného hlediska je lze rozdělit do dvou základních skupin: (i) požadavky vyplývající z mezního stavu únosnosti, (ii) požadavky vyplývající z mezního stavu použitelnosti. První typ podmínek je přímo zahrnut do návrhového algoritmu a jeho

detailní diskuzi lze nalézt v kapitole 3. Požadavky vyplývající z mezního stavu spolehlivosti jsou zahrnuty do cílové funkce ve formě penalt.

Prvním požadavkem vyplývajícím z mezních stavů použitelnosti je omezení maximálního průhybu konstrukce, formálně reprezentované podmínkou

$$\delta^{(i)} - \delta_{\text{lim}}^{(i)} \leq 0, i = 1, \dots, n_d, \quad (2)$$

kde  $\delta^{(i)}$  je maximální průhyb  $i$ -tého návrhového prvku, neboli,

$$\delta^{(i)} = \max_{j \in E^{(i)} e=1, \dots, n_e} \max^c |\delta^{[j]}|, \quad (3)$$

a  $\delta_{\text{lim}}^{(i)}$  je mezní průhyb  $i$ -tého návrhového prvku. Poznamenejme, že hodnota průhybu může být určena pomocí pružné analýzy konstrukce, např. za platnosti podmínky (4), případně pomocí zpřesněné nelineární analýzy zohledňující zvolený pracovní diagram [13].

Jako jeden z dalších cílů optimalizace může být uvažováno vyloučení nebo omezení tahových napětí, která jsou obecně nepříznivá pro betonové konstrukce. Tuto podmínku můžeme formalizovat jako

$$\frac{M_{Sk}^{(i)}(h^{(i)} - a_g^{(i)})}{I^{(i)}} - f_{ctm} \leq 0, \quad i = 1, \dots, n_d, \quad (4)$$

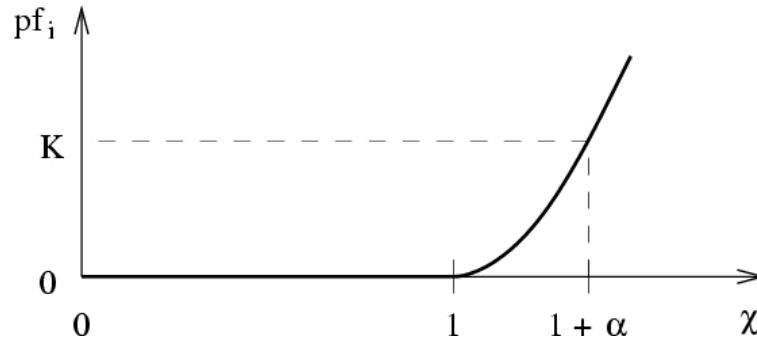
kde  $M_{Sk}^{(i)}$  je charakteristická hodnota momentu pro  $i$ -tý návrhový prvek určená na základě vztahu podobnému k (3),  $a_g^{(i)}$  je vzdálenost mezi horním lícem konstrukce a těžištěm ideálního průřezu,  $I^{(i)}$  je moment setrvačnosti ideálního průřezu a  $f_{ctm}$  je průměrná hodnota tahové pevnosti betonu. Detailní diskuzi uvedených vztahů lze nalézt např. v [11].

Pomocí dříve uvedených parametrů můžeme nyní popsat množinu možných konfigurací konstrukce a ocenit kvalitu jednotlivých prvků této množiny. V souladu s předchozí diskuzí pro tento účel volíme celkovou cenu konstrukce

$$f(X) = V_c P_c + W_s P_s + A_c P_{Ac} + \sum_{i=1}^2 p f_i, \quad (5)$$

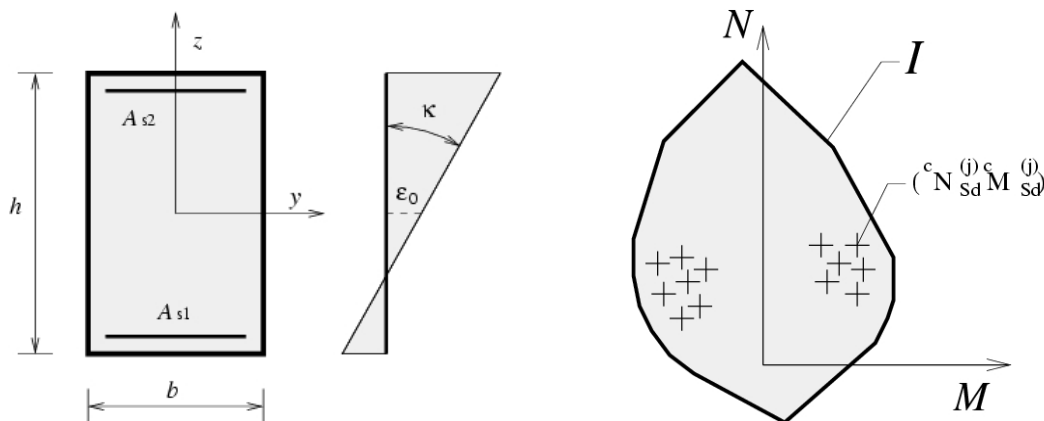
kde  $X$  je vektor návrhových proměnných,  $V_c$  je objem betonu,  $W_s$  je váha oceli a  $A_c$  je plocha bednění. Dále jsou  $P_c$  a  $P_s$  jednotkové ceny za 1 m<sup>3</sup> betonu a za 1 kg oceli,  $P_{Ac}$  značí cenu za 1 m<sup>2</sup> bednění.

Omezující podmínky (2,4) lze zahrnout do uvažovaného problému ve formě penalt. Z matematického hlediska můžeme penaltu  $p f_i$  chápat jako vzdálenost řešení od množiny přípustných (tj. nepenalizovaných) řešení, popřípadě jako cenu vynaloženou na splnění podmínky  $i$ . V současné implementaci uvažujeme penalizační funkci ve formě znázorněné na Obr.3; dodatečné informace lze nalézt např. v [5,7,9].

Obrázek 3: Příklad penalizační funkce  $pf_i$ .

### 3. Optimalizace průřezu

V dřívější práci [6] byla optimalizace výztuže průřezu prováděna současně s určením geometrických parametrů konstrukce. Tento přístup bohužel není aplikovatelný pro rozsáhlejší konstrukční systémy, jelikož nevyhnutelně vede k velkému počtu optimalizovaných proměnných, což prakticky znemožňuje jeho řešení v reálném čase dokonce i na moderních výkonných počítačích. Proto k řešení tohoto problému využijeme jednoduchého, nicméně velmi efektivního, algoritmu pro návrh výztuže průřezu namáhaného obecnou kombinací momentů a normálové síly. Postup je založen na rychlém výpočtu vnitřních sil dle návrhu R. Vondráčka [13]. Obdobný přístup je použit při návrhu smykové výztuže.



Obr. 4 Schéma průřezu, (a) rovina deformace, (b) interakční diagram

Nejprve uvedme krátký popis postupu použitého pro výpočet vnitřních sil v této práci. Uvažujme obecný polygonální průřez s lineárním průběhem deformace  $\varepsilon_x$  po výšce průřezu,

$$\varepsilon_x(z) = \varepsilon_0 + z \kappa, \quad (6)$$

kde  $\varepsilon_0$  je deformace v těžišťové ose a  $\kappa$  křivost ve směru osy  $z$ . Odezva materiálu je popsána konstitutivním zákonem

$$\sigma_x = \sigma(\varepsilon_x). \quad (7)$$

Vnitřní síly  $N_x$  a  $M_y$  pak plynou ze známých vztahů

$$N_x = \int_A \sigma_x dA, \quad M_y = \int_A \sigma_x z dA. \quad (7)$$

Převedením plošných integrálů (7) na hraniční pomocí Gauss-Greenovy věty a po zohlednění faktu, že geometrie průřezu je popsána po částech lineárními funkcemi, získáme po několika úpravách vztahy [13]

$$N_x = -\frac{1}{\kappa^2} \sum_{i=0}^{n_p-1} \frac{1}{k^i} (ss(\varepsilon^{(i+1)}) - ss(\varepsilon^{(i)})), \quad (8)$$

$$M_u = -\frac{1}{\kappa^3} \sum_{i=0}^{n_p-1} \frac{1}{k^i} [(\varepsilon - \varepsilon_0)ss(\varepsilon) - 2ss(\varepsilon) - 2sss(\varepsilon)]_{\varepsilon^{(i)}}^{\varepsilon^{(i+1)}} \quad (9)$$

kde  $n_p$  je počet stran polygonu,  $k_i$  je směrnice  $i$ -tého segmentu,  $\varepsilon^i$  je hodnota deformace pro  $i$ -tý vrchol polygonu a funkce  $ss()$  a  $sss()$  jsou definovány rekurzemi

$$sss(\varepsilon) = \int_0^\varepsilon ss(\varepsilon) d\varepsilon = \int_0^\varepsilon \int_0^\varepsilon s(\varepsilon) d\varepsilon = \int_0^\varepsilon \int_0^\varepsilon \int_0^\varepsilon \sigma(\varepsilon) d\varepsilon d\varepsilon d\varepsilon.$$

Podrobné odvození a diskuse prezentovaných vztahů spolu s ošetřením singularit ( $\kappa=0$  nebo  $k_i=0$ ) lze opět nalézt v práci [13].

Uvedeného postupu lze s výhodou použít pro konstrukci interakčního diagramu  $I$  pro daný průřez (viz Obr. 4b) na základě výpočtu velikosti normálové síly  $N_x$  a ohybového momentu  $M_y$  pro danou množinu extrémních rovin deformace [11,13]. Posuzovaný průřez je pak schopen přenést danou návrhovou normálovou sílu  $N_{sd}$  a ohybový moment  $M_{sd}$ , je-li splněna následující podmínka:

$$(N_{sd}, M_{sd}) \in I. \quad (11)$$

Pro účely návrhu výztuže předpokládejme, že jsou zadány rozměry průřezu  $b$  a  $h$  spolu s průměrem výztužných vložek  $\phi_b$ . Ze zvolené návrhové normy dále vyplývají minimální a maximální plochy výztuže  $A_{s1}$  a  $A_{s2}$ , které můžeme díky znalostem o profilu výztuže snadno převést na minimální a maximální počet výztužných vložek  $n_{s,min}$  a  $n_{s,max}$ . Optimální vyztužení daného návrhového prvku pak určíme z podmínky

$$(N_{sd}, M_{sd}) = ({}^c N_{sd}^{[j]}, {}^c M_{sd}^{[j]}), \quad j \in E^{(i)}, \quad i = 1, \dots, n_d, \quad c = 1, \dots, n_c. \quad (11)$$

I přes značnou jednoduchost je uvedený návrhový postup dobře aplikovatelný pro prezentovaný problém. Této skutečnosti je dosaženo díky velmi efektivní implementaci integrace vnitřních sil a možnosti vyloučení některých kombinací výztužných prvků v případě, že bylo nalezeno alespoň jedno přípustné řešení. Jako další výhodu navrženého postupu lze uvést skutečnost, že vždy nalezneme optimální vyztužení průřezu.

V principu stejného, nicméně implementačně podstatně jednoduššího postupu, lze použít i pro návrh smykové výztuže. Vzdálenost třmínek lze určit na základě standardní příhradové analogie se sklonem tlačných diagonál  $45^\circ$ . Návrhová posouvající síla v tomto případě nesmí překročit únosnost tlačných diagonál  $V_{Rd2}$ . Celková smyková únosnost betonu a třmínek plyne ze vztahu

$$V_{Rd3} = V_{cd} + V_{wd}, \quad (12)$$

kde  $V_{cd}$  značí hodnotu posouvající síly přenášené tlačenu betonovou částí průřezu a  $V_{wd}$  je smyková únosnost třmínků. Podrobnosti o prezentované návrhové metodě lze opět nalézt v normě EC 2 [4]. Výsledný návrh pak musí splňovat nerovnosti

$$|V_{Sd}| - V_{Rd2} \leq 0, \quad |V_{Sd}| - V_{Rd3} \leq 0, \quad (13)$$

spolu s jistými omezeními vyplývajícími z konstrukčních zásad a minimální požadované plochy smykové výztuže. Obdobně jako při návrhu ohybové výztuže, optimální vzdálenosti mezi třmínky plynou z přímého výpočtu a z porovnání návrhových sil a únosností konstrukce pro daný návrhový prvek.

#### 4. Stochastický optimalizační algoritmus

V okamžiku, kdy byl vyřešen návrh výztužných prvků procedurou popsanou v předchozí kapitole, zůstávají jedinými neznámými optimalizačními parametry geometrické rozměry konstrukce, tj. šířky  $b^{(i)}$  a výšky  $h^{(i)}$  jednotlivých návrhových prvků a profily výztužných prvků  $\phi_b^{(i)}$  a  $\phi_w^{(i)}$ ,  $i = 1, \dots, n_d$ , spolu s globálními parametry konstrukce jako je třída betonu nebo výztužné oceli. Jelikož ve shodě s návrhovou praxí předpokládáme, že rozměry konstrukce mohou nabývat pouze jistých diskrétních hodnot, návrh těchto rozměrů představuje kombinatorický problém definovaný na množině přirozených čísel. Algoritmus schopný úspěšně řešit tuto třídu úloh je představen v následujícím odstavci.

Ještě před vlastním popisem optimalizační metody nejprve představíme zobrazení mezi reprezentačním a vyhledávacím prostorem pro jednotlivé optimalizované proměnné. Buď  $\mathbf{X} = \{x_1, x_2, \dots, x_n\}$  vektor  $n$  celých, případně reálných, čísel uvažovaných na uzavřené množině

$$\min_i \leq x_i \leq \max_i.$$

Dále předpokládejme, že každá z proměnných  $x_i$  má být reprezentována sjistou požadovanou přesností  $p_i$ , definovanou jako nejmenší jednotka, které může proměnná  $x_i$  nabýt. Za těchto předpokladů může být každá z proměnných  $x_i$  transformována na nezáporné celé číslo  $y_i$  pomocí vztahu

$$y_i = \left\lfloor \frac{x_i - \min_i}{p_i} \right\rfloor, \quad (14)$$

kde symbol  $\lfloor a \rfloor$  označuje celou část z  $a$  (celočíselné oříznutí). Inverzní vztah je pak dán předpisem

$$x_i = y_i p_i + \min_i. \quad (15)$$

Poznamenejme, že s transformovaným vektorem  $\mathbf{Y}$  můžeme zacházet buď jako s binárním řetězcem pomocí nízkourovňových bitových operací nebo jako s vektorem celých čísel.

Na základě našich zkušeností se zkoumaným optimalizačním problémem byla otestována řada globálních stochastických optimalizačních algoritmů s genetickými algoritmy jako významnými reprezentanty těchto metod. Nejlepších výsledků bylo dosaženo metodou rozšířeného simulovaného žihání (AUSA) [8]. Tento algoritmus efektivně kombinuje základní principy genetických algoritmů (*populace* možných řešení namísto jednoho bodu v optimalizačním prostoru) s ideou metody *simulovaného žihání* založené na analogii s postupem pro minimalizaci defektů v pevných látkách. Základem metod založených na genetických algoritmech je proces rekombinace, kdy dvě řešení kombinují své dobré vlastnosti při produkci svých potomků. V genetických algoritmech je tento proces simulován pomocí operátorů *mutace* a *křížení*.

V předchozích příspěvcích na toto téma [7,9] jsme představili implementaci binární verze popsaného algoritmu; tato metoda je více náchylná k předčasné konvergenci do lokálního minima. Tato nechtost je způsobena tzv. Hammingovou bariérou – dvě blízká přirozená čísla nemusí být vzájemně blízká v binární reprezentaci. Z tohoto důvodu jsme přijali nový přístup, který zachází se všemi proměnnými jako s vektory v  $n$ -rozměrném prostoru. Tento princip byl v poslední době užit v několika stochastických optimalizačních algoritmech – jako příklad uveďme metodu *diferenciální evoluce* [12], která využívá pouze reálného kódování. Z důvodu změny kódování je také nutno odpovídajícím způsobem přeformulovat klasické genetické operátory. Za prvé, operátor křížení generuje ze tří celočíselných jedinců  $X_1$ ,  $X_2$  a  $X_3$  opět celočíselného potomka  $Y$  pomocí vztahu

$$Y = X_1 + \lfloor U(-2,2)(X_2 - X_3) \rfloor, \quad (16)$$

kde  $U(-2,2)$  je reálná náhodná proměnná s rovnoměrným rozdělením na intervalu  $\langle -2;2 \rangle$ . Za druhé, operátor mutace je určen vztahem

$$Y = X + \lfloor N(0, \sigma) \rfloor, \quad (17)$$

kde  $X$  značí rodiče a  $Y$  potomka,  $N(0, \sigma)$  je náhodný vektor s Gaussovým rozdělením. Nejlepších výsledků bylo dosaženo pro hodnotu parametru  $\sigma = |X_2 - X_3|$ , kde  $X_2$  a  $X_3$  jsou další dva vektory náhodně vybrané z populace. Podrobnější popis spolu s testy výkonnosti prezentovaného algoritmu lze nalézt v [5].

## 5. Paralelní implementace

Pro paralelizaci prezentované úlohy bylo využito implicitního paralelismu genetického algoritmu [8]. Vlastní optimalizační algoritmus pak běží na kořenovém procesoru (root), zatímco jednotlivé podřízené procesory (slaves) vyhodnocují návrhy odpovídající proměnným zaslaným kořenovým procesorem (tento model paralelizace je také označován jako *globální* [1]). Pro úplnost uveďme, že v současné implementaci je použita *synchronní* verze, kdy kořenový procesor čeká na odpověď podřízených procesorů. Výhoda zvoleného paralelizačního schématu je jeho nezávislost na zvolené hardwarové platformě a fakt, že změny oproti sekvenční verzi jsou minimální a to včetně potenciálně časově náročného nastavení vnitřních parametrů algoritmu AUSA.

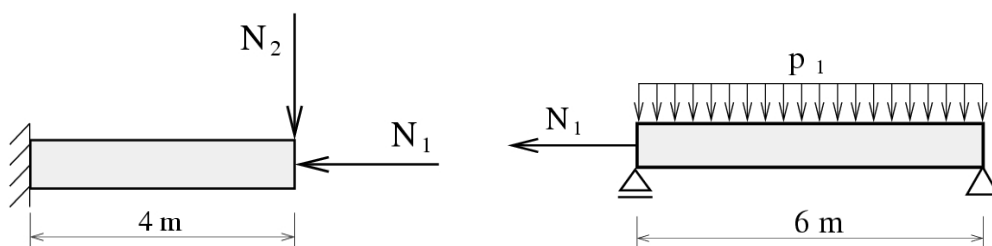
Vlastní paralelní implementace je založena na „master-slave“ paradigmatu. Rovnoměrné rozdělení úloh mezi jednotlivé procesory je zajištěno pomocí dynamického rozdělování zátěže (dynamic load balancing). Celkový výkon optimalizačního algoritmu je do značné míry závislý na rychlosti komunikace mezi jednotlivými procesory. Ta je odvozena jednak od čekací doby (latency, tj. čas potřebný k navázání komunikace mezi jednotlivými procesory) a jednak od šířky komunikace (bandwidth, tj. počet bytů, které mohou být přeneseny sítí během jedné sekundy). Jak bylo demonstrováno v [3], šířka komunikace je nejkritičtějším parametrem pro zvolenou metodu; v této práci jsou dále uvedeny doporučení pro zohlednění této skutečnosti pro daný optimalizační algoritmus.

## 6. Příklady

Pro ilustraci postupů diskutovaných v tomto příspěvku byly vybrány dvě úlohy řešené v práci [2] klasickými optimalizačními metodami. Jako první příklad je uvažován návrh konzoly délky  $l = 4$  m s průřezovými parametry  $h = 0,6$  m a  $b = 0,4$  m (viz Obr.5a); třída betonu je C 12/15, zatímco výztužné profily jsou uvažovány z oceli V 10 425



(charakteristická hodnota meze kluzu 410 MPa). Konzola byla zatížena dvěma zatěžovacími stavy ( ${}^1N_1 = 1800$  kN,  ${}^1N_2 = 100$  kN) a ( ${}^2N_1 = 300$  kN,  ${}^2N_2 = 100$  kN). V prvním kroku se ukázalo, že pro dané geometrické parametry neexistuje optimální návrh; při respektování omezení vyplývajících z konstrukčních zásad totiž nelze umístit do průřezu dostatečné množství výztužných prvků. Po změně šířky průřezu na  $b=0,5$  m bylo nalezeno optimální řešení odpovídající profilu podélných výztužných prvků 22 mm a profilu 6 mm smykové výztuže. Výsledné plochy výztuže  $A_{S1} = 2\,281$  mm<sup>2</sup> a  $A_{S2} = 1\,140$  mm<sup>2</sup> se příliš (v součtu) neliší od hodnot  $A_{S1} = 1\,790$  mm<sup>2</sup> a  $A_{S2} = 1\,518$  mm<sup>2</sup>, které byly prezentovány v [2]. Uvedený příklad ukazuje důležitost respektování konstrukčních zásad při návrhu výztužných prvků; je poměrně komplikované toto omezení zahrnout do klasických optimalizačních algoritmů, zatímco v prezentované metodice tato skutečnost nepředstavuje podstatné omezení.



Obrázek 5: Příklady (a) konzola, (b) prostý nosník

Jako druhý příklad byl zvolen prostý nosník, Obr. 5b. Rozpětí nosníku bylo uvažováno  $l=6$  m, šířka průřezu byla 0,3 m, zatímco výška byla uvažována 0,6 m. Třída betonu a oceli byla stejná jako v předchozím případě. Nosník byl zatížen třemi zatěžovacími stavy: ( ${}^1p_1 = 62.5$  kN/m,  ${}^1N_1 = -240$  kN), ( ${}^2p_1 = 62.5$  kN/m,  ${}^2N_1 = -1440$  kN) a ( ${}^3p_1 = 62.5$  kN/m,  ${}^3N_1 = 480$  kN). Obdobně jako v předchozím případě, rozměry průřezu byly nedostatečné vzhledem k množství požadované výztuže. První přípustná šířka nosníku v tomto případě je  $b = 0,4$  m a průměr výztužných prvků je roven 32 mm. Optimální řešení je pak dáno parametry  $b = 0,5$  m a průměrem podélné výztuže 18 mm, zatímco parametry smykové výztuže jsou stejné jako v předchozím příkladu. Optimální plochy výztuže ( $A_{S1} = 2\,545$  mm<sup>2</sup>,  $A_{S2} = 0$  mm<sup>2</sup>) se opět v součtu příliš neliší od hodnot  $A_{S1} = 2\,078$  mm<sup>2</sup>,  $A_{S2} = 989$  mm<sup>2</sup> uvedených v [2], nicméně opět zahrnují konstrukční doporučení zvolené normy a jako takové jsou tedy pro praktický návrh vhodnější.

## 7. Shrnutí

Tento příspěvek představuje automatizovanou návrhovou metodu, jejímž cílem je nabídnout projektantovi užitečný nástroj pro efektivní a úsporný návrh železobetonových rámových konstrukcí. Navržený optimalizační přístup se ukázal být vhodným pro daný problém; jeho výhody jsou patrné již pro velmi jednoduché konstrukční systémy. Nevýhodou algoritmu AUSA je jednak nutnost nastavení rozličných vnitřních parametrů metody, což může vést ke zdoluhavému ladění, a jednak velký počet ocenění cílové funkce. Tento problém je vyřešen paralelizací úlohy, jež je v současné době připravována pro komplikovanější cílovou funkci zahrnující zpřesněnou analýzu vnitřních sil a posunů metodou konečných prvků. V blízké budoucnosti je rovněž uvažováno propojení optimalizační procedury s nejrozšířenějším projekčním programem AutoCAD.

## 8. Poděkování

Tato práce byla podporována Ministerstvem školství, mládeže a tělovýchovy, výzkumným záměrem MSM 210000003. Autoři by dále chtěli poděkovat Ing. Richardovi Vondráčkovi za četné konzultace a softwarovou podporu při řešení dané problematiky.

## 8. Literatura

- [1] E. Cantú-Paz. A survey of parallel genetic algorithms (Illi-GAL Report No. 97003). Technical report, Urbana, IL: University of Illinois at Urbana-Champaign, 1997.
- [2] I. Laníková. Optimalizovaný návrh výztuže rámových konstrukcí [An optimum design of reinforcement within frame structures], In *Concrete Days '99*, pages 272-279. The Czech Concrete Society, 1999.
- [3] M. Lepš and Z. Bittnar. Reinforced concrete beam design optimization by parallel genetic algorithm based strategy. In H.A. Mang, F.G. Rammerstorfer, and J. Eberhardsteiner, editors, *Proceedings of The Fifth World Congress on Computational Mechanics (WCCM V)*. Vienna University of Technology, Austria, 2002. <http://wccm.tuwien.ac.at>
- [4] CEN, Brussels. *Eurocode 2 Part 1.1, Design of Concrete Structures*, ENV 1992 1-1, 1991.
- [5] O. Hrstka, A. Kučerová, M. Lepš, and J. Zeman. A competitive comparison of different types of evolutionary algorithms. In B.V.H. Topping and B. Kumar, editors, *Proceedings of The Sixth International Conference on the Applications of Artificial Intelligence to Civil and Structural Engineering*, pages 87-88. Civil-Comp Press, 2001.
- [6] M. Lepš, K. Matouš, and Z. Bittnar. Genetic algorithm in optimization of reinforced concrete beam. *Acta Polytechnica*, 39(2):145-155, 1999.
- [7] M. Lepš and M. Šejnoha. New approach to optimization of reinforced concrete beams. In *Computational Concrete Structures Technology*, pages 143-151. Civil-Comp Press, 2000.
- [8] S.W. Mahfoud and D. E. Goldberg. Parallel recombinative simulated annealing – A genetic algorithm. *Parallel Computing*, 21(1):1-28, 1995.
- [9] K. Matouš, M. Lepš, J. Zeman, and M. Šejnoha. Applying genetic algorithms to selected topics commonly encountered in engineering practice. *Computer Methods in Applied Mechanics and Engineering*, 19(13-14):1629-1650, 2000.
- [10] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag, third edition, 1996.
- [11] J. Procházka, J. Bradáč, J. Krátký, J. Filipová, and H. Hanzalová. Betonové konstrukce. Příklady navrhování podle Eurocode 2 [Concrete structures. Examples of design according to EC2]. Lecture notes, Procon, 1997.
- [12] R. Storn. On the usage of differential evolution for function optimization. In *NAPHIS 1996*, pages 519-523. Berkeley, 1996.
- [13] R. Vondráček. Area Integral of a Stress Function over a Beam Cross-Section, *Proceedings of The Sixth International Conference on Computational Structures Technology*, Civil-Comp Press, (2002).