

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV MATEMATIKY
FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF MATHEMATICS

MODELY STOCHASTICKÉHO PROGRAMOVÁNÍ V INŽENÝRSKÉM NÁVRHU

THE SELECTED STOCHASTIC PROGRAMS IN ENGINEERING DESIGN

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. MICHAL ČAJÁNEK

VEDOUCÍ PRÁCE
SUPERVISOR

RNDr. PAVEL POPELA, Ph.D.

Abstrakt

V diplomové práci je rozebrána úloha dvojstupňového stochastického programování s omezením ve tvaru parciálních diferenciálních rovnic eliptického typu. Je navrženo výpočtové schéma, přičemž důraz je kladen na aproximační techniky. Zavádíme metodu aproximace náhodných proměnných stochastické úlohy a k diskretizaci omezení využíváme vhodné numerické metody, nejdříve metodu konečných diferencí, následně metodu konečných prvků. Dále jsou formulovány úlohy matematického programování popisující průhyb membrány s náhodným zatížením. Následuje určení vhodnosti použití stochastické optimalizace namísto příslušné deterministické úlohy a posouzení kvality aproximace založené na simulační metodě Monte Carlo a teorii intervalových odhadů. Výsledné matematické modely implementujeme a řešíme pomocí všeobecného algebraického modelovacího systému GAMS. Výsledky prezentujeme v numerické i grafické podobě.

Summary

Two-stage stochastic programming problem with PDE constraint, specially elliptic equation is formulated. The computational scheme is proposed, whereas the emphasis is put on approximation techniques. We introduce method of approximation of random variables of stochastic problem and utilize suitable numerical methods, finite difference method first, then finite element method. There is also formulated a mathematical programming problem describing a membrane deflection with random load. It is followed by determination of the acceptableness of using stochastic optimization rather than deterministic problem and assess the quality of approximations based on Monte Carlo simulation method and the theory of interval estimates. The resulting mathematical models are implemented and solved in the general algebraic modeling system GAMS. Graphical and numerical results are presented.

Klíčová slova

Dvojstupňová stochastická optimalizace, omezení ve tvaru PDR, Poissonova rovnice, metoda konečných diferencí, metoda konečných prvků

Keywords

Two-stage stochastic programming problem, PDE constraint, Poisson equation, finite difference method, finite element method

ČAJÁNEK, M. *Modely stochastického programování v inženýrském návrhu*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2009. 44 s. Vedoucí RNDr. Pavel Popela, Ph.D.

Prohlašuji, že jsem diplomovou práci *Modely stochastického programování v inženýrském návrhu* vypracoval samostatně pod vedením RNDr. Pavla Popely, Ph.D., s použitím materiálů uvedených v seznamu literatury.

Bc. Michal Čajánek

Děkuji vedoucímu diplomové práce RNDr. Pavlu Popelovi, Ph.D. za vedení mé diplomové práce a Ing. Evě Žampachové za cenné rady, které mi pomohly při řešení úlohy.

Bc. Michal Čajánek

Obsah

1	Úvod	3
2	Nelineární programování	4
2.1	Formulace úlohy nelineárního programování	4
2.2	Konvexní funkce	5
2.3	Volné extrémy	9
2.4	Úloha kvadratického programování	10
3	Stochastické programování	11
3.1	Formulace úlohy stochastického programování	11
3.2	WS a HN rozhodnutí	11
3.3	Deterministické přístupy	12
4	Úloha stochastické optimalizace	13
4.1	Formulace úlohy	13
4.2	Scénářový přístup	15
4.3	Aproximace úlohy: metoda sítí	15
4.3.1	Diskretizace prostorových proměnných	15
4.3.2	Aproximace účelové funkce	17
4.3.3	Úloha matematického programování	17
4.3.4	Výsledky	18
4.3.5	Hladkost řízení	25
4.3.6	Výsledky	26
4.4	Odhad kvality řešení	30
4.4.1	Intervalový odhad chyby aproximace	30
4.4.2	Výsledky	31
4.5	Aproximace úlohy: metoda konečných prvků	31
4.5.1	Klasická formulace	31
4.5.2	Slabá formulace	32
4.5.3	Konečně prvkový model	33
4.5.4	Bázové funkce	34
4.5.5	Elementární soustava rovnic	36
4.5.6	Soustava rovnic	38
4.5.7	Úloha matematického programování	39
4.5.8	Výsledky	39
5	Závěr	41
	Literatura	41
	Seznam zkratk a symbolů	43
	Přílohy	44

OBSAH

1. Úvod

V inženýrských oborech je třeba řešit optimalizační problémy s omezeními ve tvaru parciálních diferenciálních rovnic. Významnou inženýrskou oblastí zabývající se těmito problémy je navrhování konstrukce, kde úlohy dělíme na stacionární a nestacionární. Nestacionární problémy popisují parciální diferenciální rovnice parabolického a hyperbolického typu, například se řeší problémy kmitání nosníku nebo vlnění desky. Naopak stacionární problémy jsou popsány eliptickými parciálními diferenciálními rovnicemi. Tyto úlohy popisují ustálené stavy nestacionárních úloh. Příkladem jsou například ohyb nosníku, deformace desky nebo deformace (průhyb) membrány. V této práci se zaměříme na sestavení matematických modelů popisujících právě průhyb membrány.

Čím dál více se v modelování uvažují náhodné elementy. V problematice návrhu konstrukce náhodnost nejčastěji představuje náhodné zatížení. Tím se dostáváme od deterministické úlohy k úloze spojitě stochastické optimalizace. Proto odvodíme aproximace náhodných proměnných [5] a sestavíme postup aproximace omezení metodou konečných diferencí [8] a následně jako druhý přístup metodou konečných prvků [1, 4]. Dostaneme se tak k úlohám kvadratického programování [6], které dokážeme řešit pomocí optimalizačního software. Na závěr bude posouzena vhodnost použití stochastické optimalizace oproti příslušné deterministické optimalizaci a následně pro získané řešení bude zaveden jeho odhad kvality pomocí simulační metody [7].

Diplomová práce je zahrnuta do řešení úkolů projektu MŠMT České republiky čís. 1M06047, projektu GA ČR reg. čís. 103/08/1658 a výzkumného záměru MŠMT České republiky čís. MSM0021630519.

2. Nelineární programování

Budeme se zabývat optimalizačními úlohami, které obsahují nelineární výrazy v účelové funkci a v omezeních [6].

2.1. Formulace úlohy nelineárního programování

Obecná úloha nelineárního programování má tvar

$$\min \{f(\mathbf{x}) \mid \mathbf{g}(\mathbf{x}) \circ \mathbf{0}, \mathbf{x} \in X\} \quad (2.1)$$

Proměnné značíme $\mathbf{x} = (x_1, \dots, x_n)^T$ a nabývají hodnot ze základní množiny $X \subset \mathbb{R}^n$, popisující například nezápornost proměnných. Hledáme přípustné řešení \mathbf{x}_{\min} , které minimalizuje účelovou (kriteriální) funkci $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Body $\mathbf{x} \in X$ považujeme za přípustné, pokud splňují omezení ve tvaru rovnic a nerovnic. Nebude-li zřejmé, co je proměnná a co parametr, uvedeme označení proměnné pod symbolem min. Symbol $\mathbf{0}$ značí nulový vektor, zde dimenze m , \circ označuje sloupcový vektor symbolů $\leq, =$, a omezení jsou určena vektorovou funkcí $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^m$. Množinu přípustných řešení $C = \{\mathbf{x} \in X \mid \mathbf{g}(\mathbf{x}) \circ \mathbf{0}\}$ lze zapsat ve tvaru

$$\begin{aligned} C &= \bigcap_{i=1}^m C_i = \bigcap_{i=1}^m \{\mathbf{x} \in X \mid g(\mathbf{x}) \circ 0\} = \\ &= \{\mathbf{x} \in X \mid g(\mathbf{x}) \leq 0, 1 \leq i \leq l; g(\mathbf{x}) = 0, l+1 \leq i \leq m\}. \end{aligned}$$

DEFINICE 2.1.1 (EXTRÉMY FUNKCÍ) Pro funkci $f : C \rightarrow \mathbb{R}$ definujeme, že rozhodnutí $\mathbf{x}_{\min} \in C$ je bodem

$$\left\{ \begin{array}{l} \text{lokálního} \\ \text{globálního} \end{array} \right\} \times \left\{ \begin{array}{l} \text{ostrého} \\ \text{neostrého} \end{array} \right\} \text{ minima}$$

$$\iff$$

$$\left\{ \begin{array}{l} \exists O(\mathbf{x}_{\min}) : \forall \mathbf{x} \in C \cap O(\mathbf{x}_{\min}) \setminus \{\mathbf{x}_{\min}\} \\ \forall \mathbf{x} \in C \setminus \{\mathbf{x}_{\min}\} \end{array} \right\} \times f(\mathbf{x}_{\min}) \left\{ \begin{array}{l} < \\ \leq \end{array} \right\} f(\mathbf{x}).$$

Uvedené schéma můžeme použít k obvyklé definici. Rozhodnutí \mathbf{x}_{\min} je bodem lokálního neostrého minima funkce $f : C \rightarrow \mathbb{R}$ právě tehdy, když existuje okolí $O(\mathbf{x}_{\min})$ takové, že pro všechna $\mathbf{x} \in C \cap O(\mathbf{x}_{\min}) \setminus \{\mathbf{x}_{\min}\}$ platí $f(\mathbf{x}_{\min}) \leq f(\mathbf{x})$. Obdobné schéma můžeme sestavit pro \mathbf{x}_{\max} .

Vidíme, že funkční hodnota lokálního ostrého minima musí být menší než funkční hodnota v jiných bodech některého jeho okolí. Protože existují „patologické funkce“, které mají v okolí některého bodu nekonečně mnoho extrémů, hovoříme o izolovaném extrému, pokud je tento extrém jediným na některém svém okolí. Původní zápis úlohy (2.1) můžeme považovat za označení optimální hodnoty účelové funkce $\min \{f(\mathbf{x}) | \mathbf{x} \in C\}$ respektive $\max \{f(\mathbf{x}) | \mathbf{x} \in C\}$ a z kontextu pak rozlišujeme, zda se zabýváme lokálními nebo globálními extrémy. Podobně značíme pro minimum

$$\mathbf{x}_{\min} \in \operatorname{argmin} \{f(\mathbf{x}) | \mathbf{x} \in C\} = \{\mathbf{x}_0 \in C | f(\mathbf{x}_0) = \min \{f(\mathbf{x}) | \mathbf{x} \in C\}\}$$

a pro maximum

$$\mathbf{x}_{\max} \in \operatorname{argmax} \{f(\mathbf{x}) | \mathbf{x} \in C\} = \{\mathbf{x}_0 \in C | f(\mathbf{x}_0) = \max \{f(\mathbf{x}) | \mathbf{x} \in C\}\}.$$

Pojmy zavedené pro množinu C z (2.1) zobecníme pro libovolnou konvexní množinu $S \subset \mathbb{R}^n$, viz Definice 2.2.1, pouhou záměnou symbolů C a S ve formulacích.

2.2. Konvexní funkce

Nyní se věnujme vlastnostem úlohy (2.1). Klíčovým pojmem je konvexnost množin a funkcí.

DEFINICE 2.2.1 (KONVEXNÍ MNOŽINA) *Množinu $S \subset \mathbb{R}^n$ nazveme konvexní množinou, jestliže pro libovolné dva body $\mathbf{x}_1, \mathbf{x}_2 \in S$ a pro libovolné $\alpha \in (0; 1)$ platí*

$$\alpha \mathbf{x}_1 + (1 - \alpha) \mathbf{x}_2 \in S.$$

Konvexní množina tedy s každými dvěma svými body obsahuje i celou úsečku, která je spojuje.

DEFINICE 2.2.2 (KONVEXNÍ FUNKCE) *Mějme reálnou funkci $f : S \rightarrow \mathbb{R}$, kde $S \subset \mathbb{R}^n$ je neprázdná konvexní množina. Řekneme, že f je konvexní funkcí na S právě tehdy, když pro každé dva body $\mathbf{x}_1, \mathbf{x}_2$ z množiny S a pro libovolné $\lambda \in (0; 1)$ platí*

$$f(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2) \leq \lambda f(\mathbf{x}_1) + (1 - \lambda) f(\mathbf{x}_2).$$

Pokud platí ostrá nerovnost pro každé body $\mathbf{x}_1, \mathbf{x}_2$, které jsou různé, hovoříme o ryze konvexní funkci. Pokud platí opačná nerovnost \geq respektive $>$, hovoříme o konkávní, případně ryze konkávní funkci.

Z definice vidíme, že konvexní i konkávní funkce jsou definovány pouze na konvexních množinách. Hlavní význam konvexních funkcí pro optimalizaci vymezuje následující tvrzení.

2.2. KONVEXNÍ FUNKCE

VĚTA 2.2.3 (O MINIMU KONVEXNÍ FUNKCE) *Nechť $S \subset \mathbb{R}^n$ je neprázdná konvexní množina a $f : S \rightarrow \mathbb{R}$ je konvexní funkce na S . Je-li \mathbf{x}_{\min} bodem lokálního minima funkce f , potom je také bodem globálního minima f . Je-li f ryze konvexní, je to minimum izolované a jediné.*

V případě konvexní účelové funkce f a konvexní množiny S hovoříme o úloze konvexního programování a věta říká, že pak nemusíme rozlišovat mezi lokálními a globálními extrémy.

Konvexnost množiny přípustných řešení.

Označme $S_\alpha = \{\mathbf{x} \in S \mid f(\mathbf{x}) \leq \alpha\}$. Je-li $f : S \rightarrow \mathbb{R}$ konvexní funkce, potom je S_α konvexní množina pro každé $\alpha \in \mathbb{R}$. Tento poznatek můžeme využít pro nelineární programování následovně. Předpokládejme, že množina X je konvexní. Jsou-li omezení úlohy (2.1) tvaru $g_i(\mathbf{x}) \leq 0$ pro $i \in I = \{1, \dots, m\}$ a funkce g_i jsou konvexní, potom jsou konvexní množiny $C_i = \{\mathbf{x} \in X \mid g_i(\mathbf{x}) \leq 0\}$. Víme, že průnik konvexních množin je konvexní množinou, a proto je konvexní i množina přípustných řešení $C = \bigcap_{i \in I} C_i$.

Spojitosť.

Konvexní funkce f je spojitá ve všech vnitřních bodech svého definičního oboru. Klasická Weierstrassova věta říká, že spojitá funkce f na kompaktní množině S nabývá svého globálního minima a maxima. Kompaktní množina je množina, která je uzavřená a omezená.

Derivace.

Existence parciálních derivací konvexní funkce f , a tím i gradientu ∇f , není obecně zaručena, ale vždy existují směrové derivace

$$f'_d(\mathbf{x}) = \lim_{\lambda \rightarrow 0^+} \frac{f(\mathbf{x} + \lambda \mathbf{d}) - f(\mathbf{x})}{\lambda}$$

pro všechny body $\mathbf{x} \in S$, a smysluplné směry $\mathbf{d} \in \mathbb{R}^n$, tj. takové, že existuje $\lambda > 0$, splňující $\mathbf{x} + \lambda \mathbf{d} \in S$

Subgradient.

Konvexní funkce $f : S \rightarrow \mathbb{R}$ je taková funkce, jejíž nadgraf

$$\text{epi} - f = \{(\mathbf{x}; y) \mid \mathbf{x} \in S; y \geq f(\mathbf{x})\},$$

tj. množina bodů ležících nad grafem funkce, je konvexní množina. Pro každý vnitřní bod \mathbf{x}_0 množiny S existuje vektor \mathbf{u} takový, že nadrovina $H = \{(\mathbf{x}; y) \mid y = f(\mathbf{x}_0) + \mathbf{u}^T(\mathbf{x} - \mathbf{x}_0)\}$ je opěrnou nadrovinou množiny $\text{epi} - f$ v bodě $(\mathbf{x}_0; f(\mathbf{x}_0))$ a zejména platí

$$f(\mathbf{x}) \geq f(\mathbf{x}_0) + \mathbf{u}^T(\mathbf{x} - \mathbf{x}_0).$$

Vektor u se pak nazývá subgradient a na rozdíl od gradientu existuje ve všech vnitřních bodech S , i když nemusí být definován jednoznačně. Množina subgradientů v daném bodě

je konvexní. Existence subgradientů ve všech vnitřních bodech S pak zaručuje konvexnost funkce uvnitř S .

Gradient.

Zabývejme se diferencovatelnými konvexními funkcemi. Je-li konvexní funkce f diferencovatelná v bodě \mathbf{x}_0 , pak existuje jediný subgradient, který je zároveň gradientem $\nabla f(\mathbf{x}_0)$. Navíc platí, že funkce f , diferencovatelná v každém bodě \mathbf{x}_0 otevřené konvexní množiny S , je konvexní právě tehdy, když je splněno $f(\mathbf{x}) \geq f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0)$ pro libovolné $\mathbf{x} \in S$. Proto se také říká, že graf diferencovatelné konvexní funkce leží nad tečnou.

Hessova matice.

Jestliže existují spojitě parciální derivace druhého řádu, Taylorův polynom druhého stupně v bodě \mathbf{x}_0 pro funkci $f : S \rightarrow \mathbb{R}$ má tvar

$$T_2(\mathbf{x}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T \mathbf{H}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0),$$

kde $\mathbf{H}(\mathbf{x}_0)$ je symetrická Hessova matice druhých parciálních derivací funkce $f(\mathbf{x})$ v bodě \mathbf{x}_0 . Funkce f je konvexní právě tehdy, když pro libovolný bod $\mathbf{x}_0 \in S$ platí, že $\mathbf{H}(\mathbf{x}_0)$ je pozitivně semidefinitní. Z pozitivní definitnosti $\mathbf{H}(\mathbf{x}_0)$, $\forall(\mathbf{x}_0) \in S$, pak vyplývá ryzí konvexnost f , ale z ryzí konvexnosti plyne jen pozitivní semidefinitnost $\mathbf{H}(\mathbf{x}_0)$, pro $\forall(\mathbf{x}_0) \in S$. Konvexnost funkce můžeme tedy testovat pomocí pozitivní semidefinitnosti matic. V případě kvadratické účelové funkce se ověření zjednoduší, protože $\forall(\mathbf{x}_0) \in S : \mathbf{H}(\mathbf{x}_0) = \mathbf{D}$. Pro funkci jedné proměnné pouze stačí zkoumat znaménko druhé derivace. Konvexnost lze také ověřovat z definice a další rozšířenou metodou zjišťování konvexnosti je vyčíslení hlavních minorů matice $\mathbf{H}(\mathbf{x}_0)$

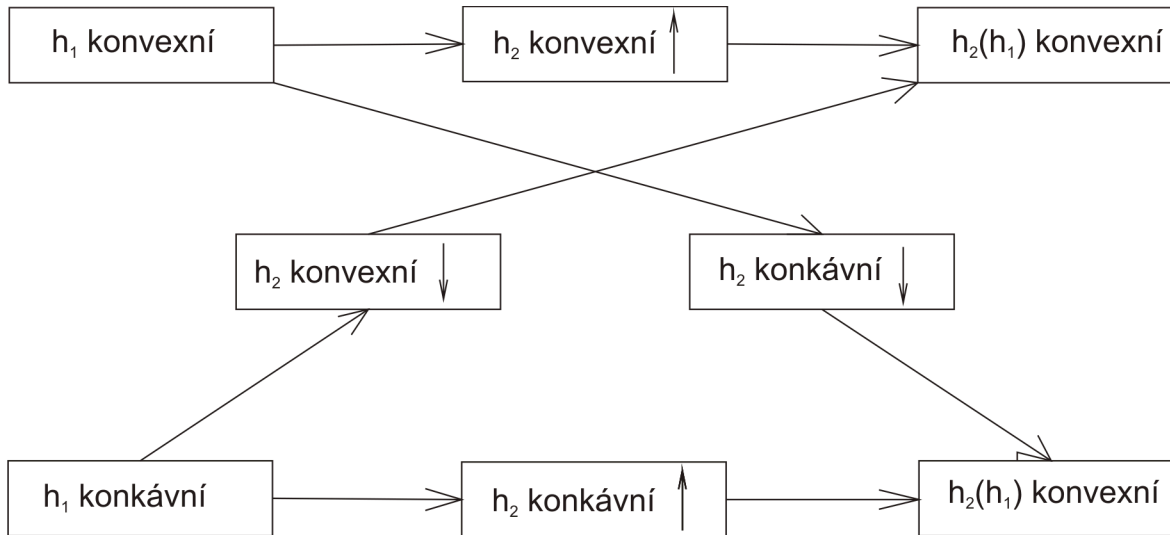
Konvexnost složených funkcí.

Vítanou pomůckou pro zkoumání konvexnosti složených funkcí $f(\mathbf{x})$ a $\mathbf{g}(\mathbf{x})$ v (2.1) může být schéma 2.1, které shrnuje poznatky pro reálné funkce h_1, h_2 a pro složenou funkci $h_2(h_1)$. Symbol \uparrow znamená rostoucí funkci a druhý symbol \downarrow značí klesající funkci. Z obrázku můžeme vyvodit například následující pravidlo: Je-li h_1 konvexní a h_2 konvexní a rostoucí, pak složená funkce $h_2(h_1)$ je konvexní. Obecněji, pro funkce více proměnných platí, že nezáporná lineární kombinace

$$f(\mathbf{x}) = \sum_{j=1}^k \alpha_j f_j(\mathbf{x}); \quad \alpha_j > 0; \quad j = 1, \dots, k$$

konvexních funkcí $f_j(\mathbf{x})$ je rovněž konvexní funkce. Dále součet konvexních funkcí je opět konvexní funkce, ale součin konvexních funkcí nemusí být konvexní. Rovněž funkce $h(\mathbf{x}) = \max_j \{f_j(\mathbf{x})\}$ je konvexní pro konvexní funkce $f_j(\mathbf{x}), j = 1, \dots, k$.

2.2. KONVEXNÍ FUNKCE



Obrázek 2.1: Konvexnost a konkávnost složených funkcí

Extrémy konvexních funkcí.

Doplňme naše poznatky o extrémech konvexních funkcí. Rozhodnutí \mathbf{x}_0 je bodem minima konvexní funkce f na S právě tehdy, když v tomto bodě existuje subgradient \mathbf{u} , který splňuje $\mathbf{u}^T(\mathbf{x} - \mathbf{x}_0) \geq 0$ pro všechny body $\mathbf{x} \in S$. Vektor $-\mathbf{u}$ je možným směrem poklesu $f(\mathbf{x})$ a uvedená nerovnice říká, že tento směr svírá s přípustným směrem $\mathbf{x} - \mathbf{x}_0$ tupý úhel, a tedy neexistuje přípustný směr poklesu, a proto hodnotu funkce nelze dále zlepšit. Je-li navíc S otevřená množina (například \mathbb{R}^n), potom vektor $\mathbf{0}$ je prvkem množiny všech subgradientů v bodě \mathbf{x}_0 . Je-li dále f v bodě \mathbf{x}_0 diferencovatelná, potom neexistuje jediný subgradient $\mathbf{u} = \nabla f(\mathbf{x}_0)$, a nutnou a postačující podmínkou existence minima v \mathbf{x}_0 je, že

$$\nabla f(\mathbf{x}_0) = \mathbf{0}.$$

Zobecnění konvexností funkcí.

Pojem konvexnosti zobecníme, a tak zeslabením podmínek rozšíříme skupinu efektivně řešitelných optimalizačních úloh. Řekneme, že f je kvazikonvexní funkce na konvexní množině S právě tehdy, když pro každé $\mathbf{x}_1, \mathbf{x}_2 \in S$ a libovolné $\lambda \in (0; 1)$ platí

$$f(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2) \leq \max\{f(\mathbf{x}_1); f(\mathbf{x}_2)\}. \quad (2.2)$$

Pro kvazikonvexní funkci f platí, že dříve definovaná množina S_α je vždy konvexní. Je-li f kvazikonvexní spojitá funkce na polyedru S , potom nabývá svého maxima v některém krajním bodě. Poznamenejme, že při řešení se pak lze inspirovat algoritmem simplexové metody. Příkladem kvazikonvexní funkce je $y = x^3$, která není konvexní. Součet kvazikonvexních funkcí nemusí být kvazikonvexní funkcí.

Ryzí kvazikonvexnost funkce f , která vyžaduje splnění ostré nerovnosti v (2.2) pro všechna $\mathbf{x}_1 \neq \mathbf{x}_2$ splňující $f(\mathbf{x}_1) \neq f(\mathbf{x}_2)$, zaručuje, že lokální minimum je zároveň globálním minimem. Dále se zavádí silná kvazikonvexnost (unimodalita), která zaručuje jedinečnost existence tohoto minima.

Protože pro diferencovatelné kvaziokonvexní funkce nemusí platit, že $\nabla f(\mathbf{x}_0) = 0$ zaručuje, že \mathbf{x}_0 je bodem minima, zavádí se pseudokonvexnost f . Pro pseudokonvexní funkci bude platit, že nulovost gradientu v bodě zaručí existenci minima v tomto bodě. Řekneme, že diferencovatelná funkce f na konvexní množině S je pseudokonvexní právě tehdy, když pro všechna $\mathbf{x}_1, \mathbf{x}_2 \in S$ platí,

$$\text{že z } \nabla f(\mathbf{x}_1)^T(\mathbf{x}_2 - \mathbf{x}_1) \geq 0 \text{ plyne } f(\mathbf{x}_2) \geq f(\mathbf{x}_1).$$

Shrňme vztahy mezi jednotlivými typy konvexních funkcí. Ryze konvexní funkce je rovněž konvexní. Diferencovatelná konvexní funkce je pseudokonvexní a ta je kvazikonvexní. Poznamenejme, že ryze kvazikonvexní funkce nemusí být kvazikonvexní.

Všechny uvedené typy konvexnosti funkcí se vztahovaly k celé množině S , ale konvexnost funkce může být rovněž definována lokálně vzhledem k jednomu pevnému bodu \mathbf{x}_0 . Vlastnosti takové funkce platí lokálně a zejména již neplatí tvrzené o globálních extrémech.

2.3. Volné extrémy

Zabývejme se dále volnými extrémy úlohy $\min\{f(\mathbf{x})|\mathbf{x} \in \mathbb{R}^n\}$. Všimněme si, že je-li f diferencovatelná v \mathbf{x}_0 a existuje-li vektor \mathbf{d} takový, že $\nabla f(\mathbf{x}_0)^T \mathbf{d} < 0$, potom musí existovat $\delta > 0$ tak, že platí

$$f(\mathbf{x}_0 + \lambda \mathbf{d}) < f(\mathbf{x}_0)$$

pro všechna $\lambda \in (0; \delta)$, což znamená, že \mathbf{d} je směrem poklesu hodnot f v \mathbf{x}_0 . Podmínky, které vylučují existenci směru poklesu v bodě \mathbf{x}_0 a tím zaručují existenci minima v tomto bodě, uvádí následující věta.

VĚTA 2.3.1 (NUTNÉ A POSTAČUJÍCÍ PODMÍNKY) *Uvažujme optimalizační úlohu na volný extrém. Potom platí:*

Nutná podmínka 1. řádu: *Jestliže funkce f má v bodě \mathbf{x}_0 spojitě parciální derivace 1. řádu a má tam lokální minimum, potom $\nabla f(\mathbf{x}_0) = \mathbf{0}$.*

Nutná podmínka 2. řádu: *Jestliže funkce f má v bodě \mathbf{x}_0 spojitě parciální derivace 2. řádu a má tam lokální minimum, potom $\nabla f(\mathbf{x}_0) = \mathbf{0}$ a dále $\mathbf{H}(\mathbf{x}_0)$ je pozitivně semidefinitní.*

Postačující podmínka 1. řádu: *Má-li funkce f v bodě \mathbf{x}_0 spojitě parciální derivace 2. řádu, $f(\mathbf{x}_0) = \mathbf{0}$ a $\mathbf{H}(\mathbf{x}_0)$ je pozitivně semidefinitní, potom \mathbf{x}_0 je bodem ryzího lokálního minima funkce f .*

Nutná a postačující podmínka 1. řádu: *Je-li funkce f pseudokonvexní (platí i pro f konvexní a diferencovatelnou) v \mathbf{x}_0 , pak \mathbf{x}_0 je globální minimum právě tehdy, když $\nabla f(\mathbf{x}_0) = \mathbf{0}$.*

2.4. ÚLOHA KVADRATICKÉHO PROGRAMOVÁNÍ

2.4. Úloha kvadratického programování

VĚTA 2.4.1 (KARUSH–KUHN–TUCKER) *Nechť funkce f a \mathbf{g} jsou diferencovatelné v bodě \mathbf{x}_0 , který je bodem lokálního minima úlohy $\min\{f(\mathbf{x})|\mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$. Nechť dále platí, že sloupce matice $\nabla\mathbf{g}(\mathbf{x}_0)^T$, odpovídající gradientům aktivních omezení jsou lineárně nezávislé (podmínka regularity). potom existují koeficienty \mathbf{u} tak, že platí*

$$\nabla f(\mathbf{x}_0) + \nabla\mathbf{g}(\mathbf{x}_0)^T \mathbf{u} = \mathbf{0}, \quad \mathbf{u}^T \mathbf{g}(\mathbf{x}_0) = 0, \quad \mathbf{u} \geq \mathbf{0}.$$

Věta uvádí KKT podmínky nutné pro existenci minima v \mathbf{x}_0 . Pro bod \mathbf{x}_0 podezřelý z existence extrému platí, že vektor největšího spádu $-\nabla f(\mathbf{x}_0)$ je možné vyjádřit jako nezápornou lineární kombinaci gradientů $\nabla g_i(\mathbf{x}_0)$, určených aktivními omezeními.

Uvažujeme úlohu s kvadratickou účelovou funkcí, se symetrickou maticí \mathbf{H} a s lineárními omezeními:

$$\min\{\mathbf{c}^T \mathbf{x} + \frac{1}{2} \mathbf{x}^T \mathbf{H} \mathbf{x} | \mathbf{A} \mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}. \quad (2.3)$$

Tuto úlohu je možné řešit obecnými algoritmy nelineárního programování. Navíc KKT podmínky můžeme vyjádřit ve tvaru úlohy lineární komplementarity, viz (2.4)–(2.7) ve které hledáme $\mathbf{w} \geq \mathbf{0}$ a $\mathbf{z} \geq \mathbf{0}$ tak, aby platilo

$$\mathbf{w} - \mathbf{M} \mathbf{z} = \mathbf{q}$$

$$\mathbf{w}^T \mathbf{z} = 0.$$

Pro řešení této úlohy použijeme Lemkeho nebo Cottleho–Dantzigův algoritmus nebo můžeme při řešení úlohy využít simplexovou metodu lineárního programování.

Lagrangeova funkce je

$$L(\mathbf{x}, \mathbf{u}, \mathbf{v}) = \mathbf{c}^T \mathbf{x} + \frac{1}{2} \mathbf{x}^T \mathbf{H} \mathbf{x} + (\mathbf{A} \mathbf{x} - \mathbf{b})^T \mathbf{u} - \mathbf{x}^T \mathbf{v}.$$

KKT podmínky jsou pak tvaru $\nabla_x L = \mathbf{0}$, $\nabla_u L = \mathbf{0}$, $\nabla_v L \leq \mathbf{0}$ a $\mathbf{v}^T \nabla_v L = 0$ pro $\mathbf{v} \geq \mathbf{0}$. Po dosazení

$$\mathbf{A} \mathbf{x} = \mathbf{b} \quad (2.4)$$

$$\mathbf{H} \mathbf{x} + \mathbf{A}^T \mathbf{u} - \mathbf{v} = -\mathbf{c} \quad (2.5)$$

$$\mathbf{x}^T \mathbf{v} = 0 \quad (2.6)$$

$$\mathbf{x}, \mathbf{v} \geq \mathbf{0} \quad (2.7)$$

Je-li \mathbf{H} pozitivně semidefinitní, je úloha (2.3) konvexní a její optimální řešení je řešením soustavy $m+n$ rovnic o $m+2n$ neznámých (2.4)–(2.5) za podmínek (2.6)–(2.7), které určují, že nejvýše $m+n$ proměných může být nenulových.

3. Stochastické programování

V této kapitole rozebereme formulaci stochastického programování [5, 9].

3.1. Formulace úlohy stochastického programování

Úlohu stochastického programování lze obecně formulovat ve tvaru

$$\min \{f(\mathbf{x}(\xi)) | \mathbf{g}(\mathbf{x}, \xi) \circ \mathbf{0}, \mathbf{x} \in C(\xi)\}. \quad (3.1)$$

Neznámou této úlohy je tedy rozhodnutí $\mathbf{x} \in X \subset \mathbb{R}^n$, splňující podmínku přípustnosti $\mathbf{x} \in C(\xi)$ a kritérium f . Stejně jako v kapitole 2 \circ reprezentuje sloupcový vektor relací $\leq, <$. Množinu přípustných řešení lze explicitně vyjádřit ve tvaru

$$\begin{aligned} C(\xi) &= \bigcap_{i=1}^m C_i(\xi) = \bigcap_{i=1}^m \{\mathbf{x} \in X | g(\mathbf{x}, \xi) \circ 0\} = \\ &= \{\mathbf{x} \in X | g(\mathbf{x}, \xi) \leq 0, 1 \leq i \leq l; g(\mathbf{x}, \xi) = 0, l+1 \leq i \leq m\}, \end{aligned}$$

kde ξ značí náhodný vektor definovaný na pravděpodobnostním prostoru (Ξ, Σ, P) a jeho rozdělení pravděpodobnosti nezávisí na rozhodnutí \mathbf{x} . Reálná funkce $f : \mathbb{R}^n \times \Xi \rightarrow \mathbb{R}$ a vektorová funkce $\mathbf{g} : \mathbb{R}^n \times \Xi \rightarrow \mathbb{R}^m$ jsou měřitelné vzhledem ke ξ pro každé $\mathbf{x} \in \mathbb{R}^n$. Při tom předpoklad měřitelnosti není z hlediska aplikací omezením, avšak je nutný z důvodu pravděpodobnostního přístupu.

Úlohu (3.1) nazýváme původní úlohou matematického programování. Z hlediska řešení není tento zápis zcela korektní, neboť nevíme, jak chápat symbol \min reprezentující operaci minimalizace účelové funkce $f(\mathbf{x}, \xi)$ před realizací náhodného vektoru ξ . Tuto nejasnost odstraníme zavedením tzv. deterministického přepisu, jehož řešení považujeme za řešení původní úlohy.

3.2. WS a HN rozhodnutí

Deterministický přepis je konstruován tak, aby byla vhodně odstraněna náhodnost z úlohy (3.1). Vlastní konstrukci předchází zodpovězení otázek týkajících se rozhodnutí \mathbf{x} a rozsahu znalostí o rozdělení ξ .

Podle toho, kdy je známa realizace náhodné veličiny, rozlišujeme

- WS (wait–and–see) rozhodnutí
- HN (here–and–now) rozhodnutí

O WS rozhodnutí hovoříme v případě, že rozhodnutí je vyžadováno až po realizaci náhodné veličiny. Rozhodnutí \mathbf{x} tak reaguje na pozorování ξ a je tedy měřitelnou funkcí $\mathbf{x}(\xi)$ náhodného vektoru ξ . Pak také hodnoty $f(\mathbf{x}(\xi), \xi)$ jsou definované avšak definované pro všechna ξ . Tento přístup je zejména používán při dlouhodobém plánování a v matematické statistice.

3.3. DETERMINISTICKÉ PŘÍSTUPY

Stochastické programování se zabývá především HN rozhodováním, kdy pozorovatel musí přijmout rozhodnutí dříve než zná pozorování, což pro praktické aplikace typické. V tomto případě tedy rozhodnutí matematicky modelujeme pomocí vektoru \mathbf{x} , který musí být stejný pro všechny budoucí realizace ξ .

3.3. Deterministické přístupy

V této kapitole se seznámíme s deterministickými přepisy účelové funkce [5, 9]. Již dříve jsme zmínili, že je nutné tyto deterministické přepisy zavést, aby optimalizační problém měl smysl a nezávisel na realizaci náhodné veličiny ξ .

DEFINICE 3.3.1 (UNDERLYING PROGRAM – UP) *Definujeme základní (stochastickou) úlohu jako*

$$? \in \operatorname{argmin}_x \{F(\mathbf{x}, \xi) \mid \mathbf{x} \in C(\xi)\}, \quad (3.2)$$

kde $\xi : \Xi \rightarrow \mathbb{R}^N$ je náhodný vektor a (Ξ, Σ, P) je daný pravděpodobnostní prostor.

DEFINICE 3.3.2 (EO DETERMINISTICKÝ PŘEPIS) *Mějme základní (stochastickou) úlohu, viz definice 3.3.1. Definujeme její here–and–now (HN) deterministický přepis pomocí střední hodnoty účelové funkce (Expected objective – EO)*

$$? \in \operatorname{argmin}_x \{E(F(\mathbf{x}, \xi)) \mid \mathbf{x} \in C(\xi) \text{ skoro jistě}\}, \quad (3.3)$$

DEFINICE 3.3.3 (EV DETERMINISTICKÝ PŘEPIS) *Mějme základní (stochastickou) úlohu, viz definice 3.3.1. Definujeme její HN deterministický přepis pomocí střední hodnoty náhodné veličiny (Expected value – EV)*

$$? \in \operatorname{argmin}_x \{F(\mathbf{x}, E(\xi)) \mid \mathbf{x} \in C(E(\xi))\}, \quad (3.4)$$

DEFINICE 3.3.4 (EEV DETERMINISTICKÝ PŘEPIS) *Pro EV deterministicky ekvivalentní úlohu, viz definice 3.3.3, definujeme EEV úlohu jako*

$$EEV = E_\xi(F(\mathbf{x}_{\min}^{EV}, \xi)), \quad (3.5)$$

EEV znamená střední hodnotu účelové funkce pro optimální řešení EV úlohy. Pokud $\mathbf{x}_{\min}^{EV} \notin C(\xi)$ skoro jistě, tj. je nepřipustné, pak je EEV plus nekonečno.

Nadefinovali jsme deterministické přepisy EO, EV a EEV. Kladem EV úlohy jsou výrazně jednodušší výpočty, zatímco kladem EO úlohy jsou věrohodnější výsledky. Vhodnost použití stochastického programování lze posoudit na základě hodnoty VSS.

DEFINICE 3.3.5 (VSS) *Definujeme hodnotu stochastického řešení (Value of Stochastic Solution – VSS) jako*

$$VSS = EEV - z_{\min}^{EO}. \quad (3.6)$$

VSS je důležitá charakteristika, která říká, jak vhodné je použití HN aproximace oproti EV aproximace. Malé hodnoty VSS znamenají, že použití EV programování k aproximaci stochastické úlohy je dobré.

4. Úloha stochastické optimalizace

Ve fyzikálních problémech, například jako vedení tepla, kmitání, či vlnění, lze hledat ustálený (tj. nezávislý na čase) stav. V případě rovnice vedení tepla tedy uvažujeme situaci, kdy se teplota v čase nemění. Podobně v případě vlnové rovnice se kmitání struny nebo membrány přechází na průhyb (deformaci).

Vlnová rovnice i rovnice vedení tepla nabývají v stacionárním případě stejného tvaru

$$-\Delta v = f, \quad (4.1)$$

kde $\Delta = \sum_{i=1}^n \frac{\partial^2}{\partial x_i^2}$ je Laplaceův operátor a f je dané funkce proměnných x_1, \dots, x_n . Rovnice (4.1) se nazývá Poissonova. Speciálním případem Poissonovy rovnice je Laplaceova rovnice

$$\Delta v = 0.$$

Pro rovinný případ, tj. $n = 2$, označíme $x_1 = x$ a $x_2 = y$. Pak je rovnice (4.1) tvaru

$$\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} = -f(x, y).$$

Pro tyto rovnice nemají počáteční podmínky smysl, uvažujeme pouze podmínky okrajové, např. Dirichletova okrajová podmínka

$$v = v_0 \text{ na } \Gamma,$$

předepisující hodnotu řešení na hranici Γ oblasti Ω , v níž rovnici uvažujeme.

Uvedené rovnice mají řadu dalších fyzikálních interpretací. Poissonova rovnice např. popisuje elektrický potenciál, šíření příměsí difuzí a proudění tekutiny tenkou vrstvou proměnné šířky.

4.1. Formulace úlohy

V inženýrských oborech lze některé problémy popsat jako optimalizační úlohu s omezením ve tvaru parciálních diferenciálních rovnic [10]. Navíc řada úloh nemusí být jen deterministického charakteru, ale může se v nich vyskytovat náhodnost, problémy nazýváme stochastické problémy.

V následující části se budeme zabývat stochastickými problémy popisující stacionární děje, přesněji průhybem (deformací) membrány

$$\frac{\partial^2 v}{\partial x^2}(x, y) + \frac{\partial^2 v}{\partial y^2}(x, y) = -f(x, y) \text{ v } \Omega = \{(x, y) : x \in \langle 0, a \rangle \text{ } y \in \langle 0, b \rangle\} \quad (4.2)$$

s okrajovou Dirichletovou podmínkou

$$v(x, y) = v_0 \text{ na } \Gamma, \quad (4.3)$$

kde $v(x, y)$ popisuje průhyb membrány v bodě o souřadnicích $[x, y]$, $f(x, y)$ je daná hustota vnějších sil působících na membránu a $a, b \in \mathbb{R}$. V celé této práci budeme uvažovat, že hodnota řešení na hranici Γ je rovna nule ($v_0 = 0$).

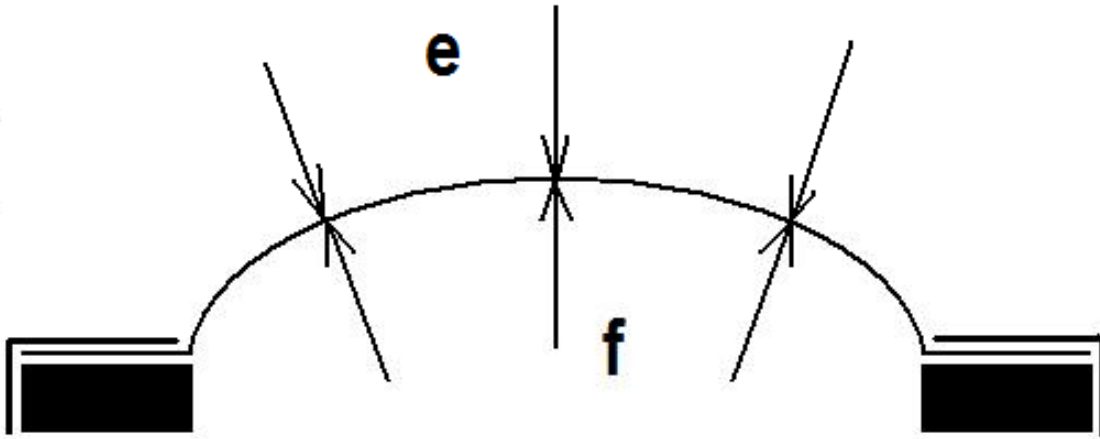
4.1. FORMULACE ÚLOHY

Deterministický problém (4.2)–(4.3) převedeme na stochastický problém zavedením náhodnosti. Náhodnost v problému realizuje náhodná veličina

$$\xi : \Xi \rightarrow \mathbb{R},$$

kde (Ξ, Σ, P) je pravděpodobnostní prostor.

Nyní předpokládejme, že průhyb membrány můžeme regulovat silou, která nepřímo působí na membránu, viz Obr. 4.1. Dále uvažujme, že chceme docílit minimální odchylky průhybu membrány od předem daného průhybu.



Obrázek 4.1: Průhyb membrány

Dostáváme se k úloze spojitě stochastické optimalizace. Jelikož lze provést rozhodnutí před i po realizaci náhodné veličiny ξ , je naším cílem optimalizovat následující úlohu.

$$z_{\min}^{\text{EO}} = \min_{e, v(\xi)} E(F(\xi, e, v(\xi))) = \min_e E \int_{y_0}^{y_1} \int_{x_0}^{x_1} (v(\xi, x, y) - u(x, y))^2 dx dy \quad (4.4)$$

$$\frac{\partial^2 v}{\partial x^2}(\xi, x, y) + \frac{\partial^2 v}{\partial y^2}(\xi, x, y) = -f(\xi, x, y) - \mathbf{T}(\xi, x, y)e(x, y) \quad (4.5)$$

$$x \in \langle 0, a \rangle, y \in \langle 0, b \rangle, \xi \in \Xi, \quad (4.6)$$

kde $v(\xi, x, y)$ je výsledný průhyb, $f(\xi, x, y)$ náhodné vnější zatížení, $e(x, y)$ řízení prvního stupně, které nepřímo působí na membránu, což zprostředkovává matice $\mathbf{T}(\xi, x, y)$.

Aby optimalizační úloha měla smysl a nezávisela na realizaci náhodné veličiny ξ , zvolíme vhodný deterministický přepis. Použili jsme střední hodnotu E účelové funkce F (tzv. EO (expected objective) úloha)). Dalším možným deterministickým přepisem je tzv. EV (expected value) úloha, kde náhodnou veličinu ξ nahradíme její střední hodnotou $E(\xi)$. Rozbor těchto dvou přístupů je proveden v odstavci 3.3.

4.2. Scénářový přístup

Aproximaci úlohy spojitě stochastické optimalizace (4.4)–(4.6) provedeme ve dvou krocích. Nejdříve využijeme scénářového přístupu k diskretizaci náhodnosti [5].

Předpokládáme, že náhodná veličina ξ má diskrétní rozdělení pravděpodobnosti s konečným počtem scénářů ξ_s , kde $s = 1, \dots, R$. Jednotlivé scénáře mají pravděpodobnost $p_s = P(\xi = \xi_s) = \frac{1}{R}$.

4.3. Aproximace úlohy: metoda sítí

V druhém kroku aproximuje PDR omezení využitím numerických metod. Nejjednodušší je tzv. metoda konečných diferencí (metoda sítí) [2, 8], kdy na oblast Ω oblasti definuje dělení a řešení problému hledáme v uzlových bodech. Parciální derivace a integrace se nahradí příslušnými numerickými formulami pro numerickou derivaci a numerickou integraci [3]. Tyto úpravy Poissonovy rovnice vedou na soustavu lineárních rovnic. Další numerickou metodou je metoda konečných prvků, ta je rozebrána v kapitole 4.5.

4.3.1. Diskretizace prostorových proměnných

Laplaceův operátor vyjádříme v diskrétní formě, která je vhodná pro numerický výpočet. Využijeme formule pro výpočet druhé derivace $f''(x)$

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2), \quad (4.7)$$

kde $O(h^2)$ je diskretizační chyba. Formule bývá označována jako druhá centrální diference. Vzorec (4.7) aplikujeme na funkci $v(x, y)$, tím získáme aproximace druhých parciálních derivací $v''_{xx}(x, y)$ a $v''_{yy}(x, y)$

$$\Delta v(x, y) = \frac{v(x+h, y) + v(x-h, y) + v(x, y+h) + v(x, y-h) - 4v(x, y)}{h^2} + O(h^2). \quad (4.8)$$

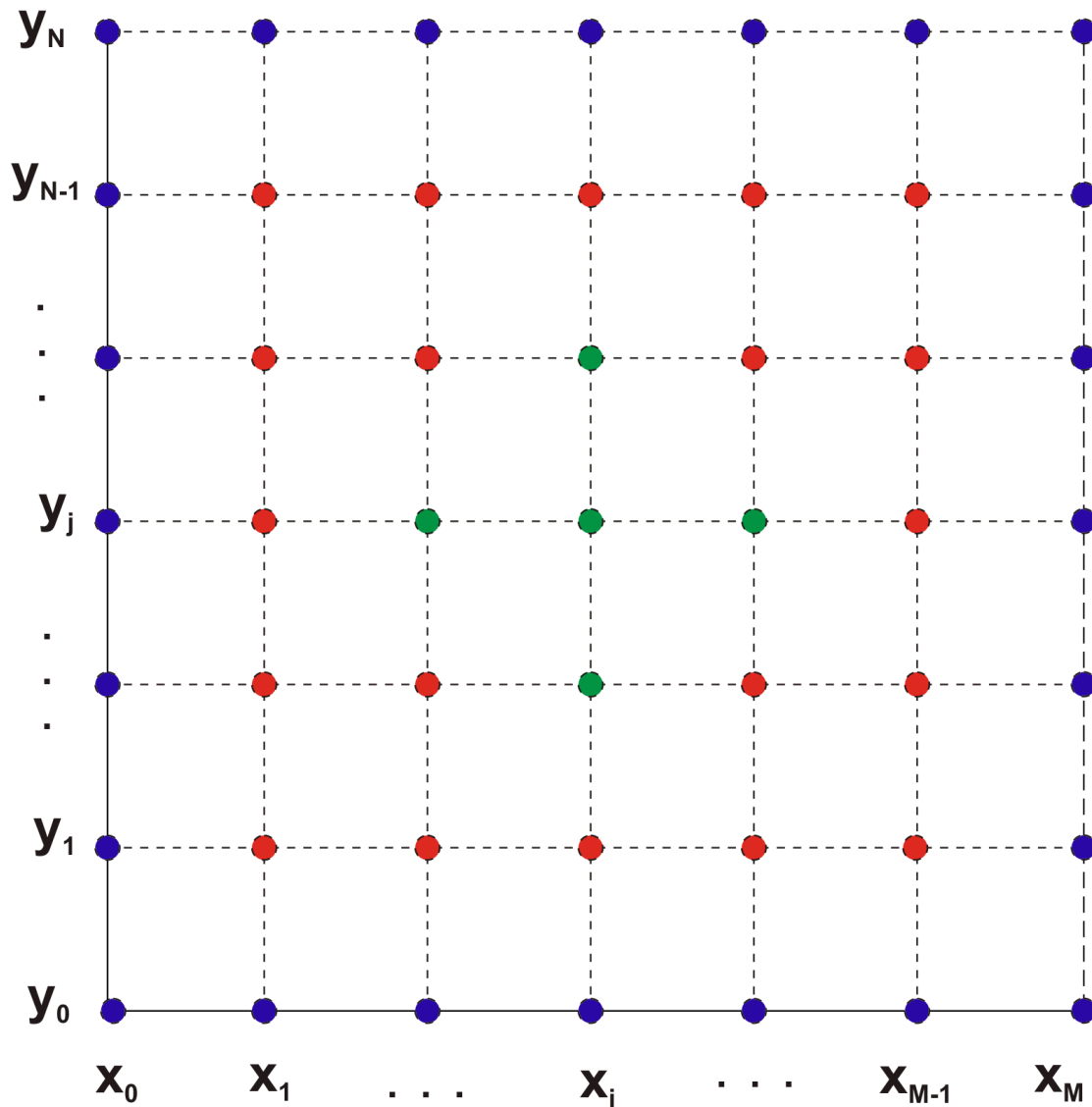
Předpokládáme, že oblast Ω je obdélník $\Omega = \{(x, y) : 0 \leq x \leq a, 0 \leq y \leq b\}$. Na obdélníku uvažujeme rovnoměrné dělení $x_i = ih$, $i = 0, \dots, M$ a $y_j = jh$, $j = 0, \dots, N$, $h = \frac{a}{M} = \frac{b}{N}$, kde h je krok dělení na osách x a y , viz obr. 4.2.

K řešení Poissonovy rovnice (4.2) využijeme aproximaci

$$\Delta v(x, y) \approx \frac{v(x+h, y) + v(x-h, y) + v(x, y+h) + v(x, y-h) - 4v(x, y)}{h^2} = f(x, y), \quad (4.9)$$

která má řád přesnosti $O(h^2)$ v každém vnitřním bodě sítě $(x, y) = (x_i, y_j)$ pro $i = 1, \dots, M-1$ a $j = 1, \dots, N-1$.

4.3. APROXIMACE ÚLOHY: METODA SÍTÍ

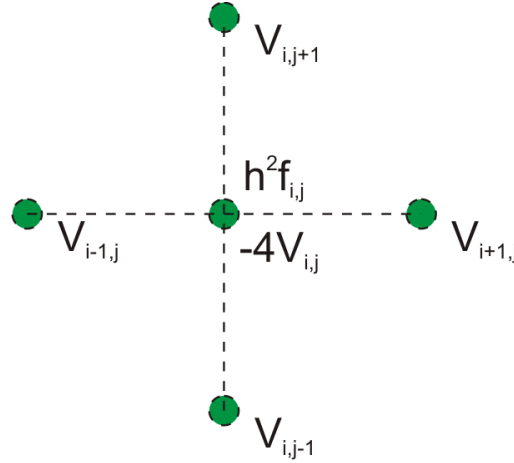


Obrázek 4.2: Síť pro diferenční rovnice

Označíme-li $v_{i,j} = v(x_i, y_j)$ a $f_{i,j} = f(x_i, y_j)$, pak rovnici (4.9) zapíšeme ve tvaru

$$\Delta v_{i,j} \approx \frac{v_{i+1,j} + v_{i-1,j} + v_{i,j+1} + v_{i,j-1} - 4v_{i,j}}{h^2} = f_{i,j}, \quad (4.10)$$

zvanému pětibodová diferenční formule pro Poissonovu rovnici. Tato formule dává do souvislosti funkční hodnotu $v_{i,j}$ se sousedními hodnotami $v_{i+1,j}$, $v_{i-1,j}$, $v_{i,j+1}$, $v_{i,j-1}$ a hodnotou $f_{i,j}$, viz obr. 4.3.



Obrázek 4.3: Poissonova šablona

Okrajová podmínka se vyjádří ve tvaru

$$\begin{aligned}
 v(x_0, y_j) = v_{0,j} = 0 & \quad \text{pro } 1 \leq j \leq N - 1 \quad (\text{levý okraj}) \\
 v(x_i, y_0) = v_{0,j} = 0 & \quad \text{pro } 1 \leq i \leq M - 1 \quad (\text{dolní okraj}) \\
 v(x_M, y_j) = v_{0,j} = 0 & \quad \text{pro } 1 \leq j \leq N - 1 \quad (\text{pravý okraj}) \\
 v(x_i, y_N) = v_{0,j} = 0 & \quad \text{pro } 1 \leq i \leq M - 1 \quad (\text{horní okraj})
 \end{aligned} \tag{4.11}$$

4.3.2. Aproximace účelové funkce

K diskretizaci účelové funkce využijeme složené Simpsonovy formule pro numerický výpočet integrálu

$$I_s^n(f(x)) \approx \frac{h}{3} (f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \cdots + 2f(x_{n-2}) + 4f(x_{n-1}) + 2f(x_n)), \tag{4.12}$$

4.3.3. Úloha matematického programování

V odstavci 4.2 jsme rozebrali aproximaci náhodné veličiny ξ pomocí scénářového přístupu, v odstavci 4.3.2 jsme odvodili aproximaci úlohy (4.4)–(4.6) pomocí metody sítí. Dále jsme nadefinovali deterministické přepisy, viz 3.3. Pomocí těchto znalostí převedeme úlohu spojité stochastické optimalizace na následující úlohy matematického programování.

EO úloha matematického programování:

$$z_{\min}^{\text{EO}} = \min \sum_{s=1}^R \sum_{j=0}^N \sum_{i=0}^M p_s \frac{h^2}{9} a_i b_j (V_{s,i,j} - U_{i,j})^2 \tag{4.13}$$

$$\begin{aligned}
 V_{s,i,j} &= \frac{1}{4} (V_{s,i+1,j} + V_{s,i-1,j} + V_{s,i,j+1} + V_{s,i,j-1}) + h^2 (f_{s,i,j} + \mathbf{T}_{s,i,j} e_{i,j}) \\
 i &= 1 \dots N - 1, \quad j = 1 \dots M - 1
 \end{aligned} \tag{4.14}$$

$$V_{s,0,j} = V_{s,M,j} = 0, \quad j = 0 \dots N \tag{4.15}$$

$$V_{s,i,0} = V_{s,i,N} = 0, \quad i = 0 \dots M \tag{4.16}$$

4.3. APROXIMACE ÚLOHY: METODA SÍTÍ

EV úloha matematického programování:

$$z_{\min}^{\text{EV}} = \min \sum_{j=0}^N \sum_{i=0}^M \frac{h^2}{9} a_i b_j (V_{i,j} - U_{i,j})^2 \quad (4.17)$$

$$V_{i,j} = \frac{1}{4} (V_{i+1,j} + V_{i-1,j} + V_{i,j+1} + V_{i,j-1}) + h^2 \left(\sum_{s=1}^R f_{s,i,j} p_s + \left(\sum_{s=1}^R \mathbf{T}_{s,i,j} p_s \right) e_{i,j} \right) \quad (4.18)$$

$$i = 1 \dots N - 1, \quad j = 1 \dots M - 1$$

$$V_{0,j} = V_{M,j} = 0, \quad j = 0 \dots N \quad (4.19)$$

$$V_{i,0} = V_{i,N} = 0, \quad i = 0 \dots M \quad (4.20)$$

EEV úloha matematického programování:

$$z_{\min}^{\text{EEV}} = \min \sum_{s=1}^R \sum_{j=0}^N \sum_{i=0}^M p_s \frac{h^2}{9} a_i b_j (V_{s,i,j} - U_{i,j})^2 \quad (4.21)$$

$$V_{s,i,j} = \frac{1}{4} (V_{s,i+1,j} + V_{s,i-1,j} + V_{s,i,j+1} + V_{s,i,j-1}) + h^2 \left(f_{s,i,j} + \mathbf{T}_{s,i,j} e_{i,j}^{\text{EV}} \right) \quad (4.22)$$

$$i = 1 \dots N - 1, \quad j = 1 \dots M - 1$$

$$V_{s,0,j} = V_{s,M,j} = 0, \quad j = 0 \dots N \quad (4.23)$$

$$V_{s,i,0} = V_{s,i,N} = 0, \quad i = 0 \dots M \quad (4.24)$$

Jedná se o úlohy kvadratického programování, tj. účelová funkce je kvadratická a omezení jsou lineární, proto můžeme říct, že se jedná o konvexní úlohy, viz odstavec 2.2.

4.3.4. Výsledky

Aproximované úlohy matematického programování (4.13), (4.17) a (4.21) byly implementovány a řešeny pomocí všeobecného algebraického modelovacího systému GAMS řešičem nelineárních úloh CONOPT. Získána výsledná data byly vykresleny pomocí MATLABu.

Uvažujeme membránu čtvercové plochy o rozměrech $a = 1$ a $b = 1$ jednotkové délky s rovnoměrným dělením na obou stranách s krokem dělení $h = 0,05$, tudíž počty diskretizačních kroků jsou $M = N = 20$. Požadovaná výchylka je $u(x, y) = 0$ a hodnota řešení na okraji (okrajová podmínka) je $v(x, y) = 0$ na Γ .

4. ÚLOHA STOCHASTICKÉ OPTIMALIZACE

Dále uvažujeme, že výsledné náhodné zatížení $f(\xi, x, y)$ je dáno kombinací pěti elementárních zatížení konstantního, lineárního, kvadratického, sinusového a exponenciálního průběhu.

$$\begin{aligned} f_1(x, y) &= 1 \\ f_2(x, y) &= \frac{x}{4} + \frac{y}{3} + \frac{1}{2} \\ f_3(x, y) &= -3(x - 0,4)^2 - 2(y - 0,5)^2 \\ f_4(x, y) &= \sin\left(\frac{10x + 5y}{2}\right) + \frac{1}{2} \\ f_5(x, y) &= \exp(x, y) \end{aligned}$$

Výsledné náhodné zatížení je definované vztahem

$$f(\xi, x, y) = \sum_{k=1}^5 A_k(\xi) f_k(x, y),$$

kde koeficienty A_k jsou dány jako výběr z rovnoměrného rozdělení

$$A_k(\xi) \sim U(0, 1)$$

a platí

$$\sum_{k=1}^5 A_k(\xi) = 1 \text{ pro } \forall \xi.$$

Matice \mathbf{T} popisující nepřímé působení řízení $e(x, y)$ je volena ve tvaru (viz [5])

$$\mathbf{T}(\xi, x, y) = \mathbf{T}_1(x, y)\xi,$$

kde matice $\mathbf{T}_1(x, y)$ byl generován z rovnoměrného rozdělení $U(0, 9; 1, 1)$ a $\xi \sim U(0, 10)$.

Výpočty v GAMSu byly provedeny pro 100 scénářů.

Pro problém optimalizace průhybu membrány popsany v odstavci 4.3.3 jsme získali následující hodnoty zvolené účelové funkce

$$z_{\min}^{\text{EO}} = 2.071542 \cdot 10^{-4} \quad (4.25)$$

$$z_{\min}^{\text{EV}} = 8.578203 \cdot 10^{-7} \quad (4.26)$$

$$z_{\min}^{\text{EEV}} = 2.281571 \cdot 10^{-4} \quad (4.27)$$

Z výsledků vidíme že hodnota z_{\min}^{EO} je malá, což znamená, že rozdíl mezi průhybem membrány a požadovaným průhybem je malý.

4.3. APROXIMACE ÚLOHY: METODA SÍTÍ

Pro posouzení vhodnosti použití stochastické optimalizace namísto příslušné deterministické úlohy vypočteme veličinu VSS, viz 3.3. V našem případě jsme dosáhli hodnoty

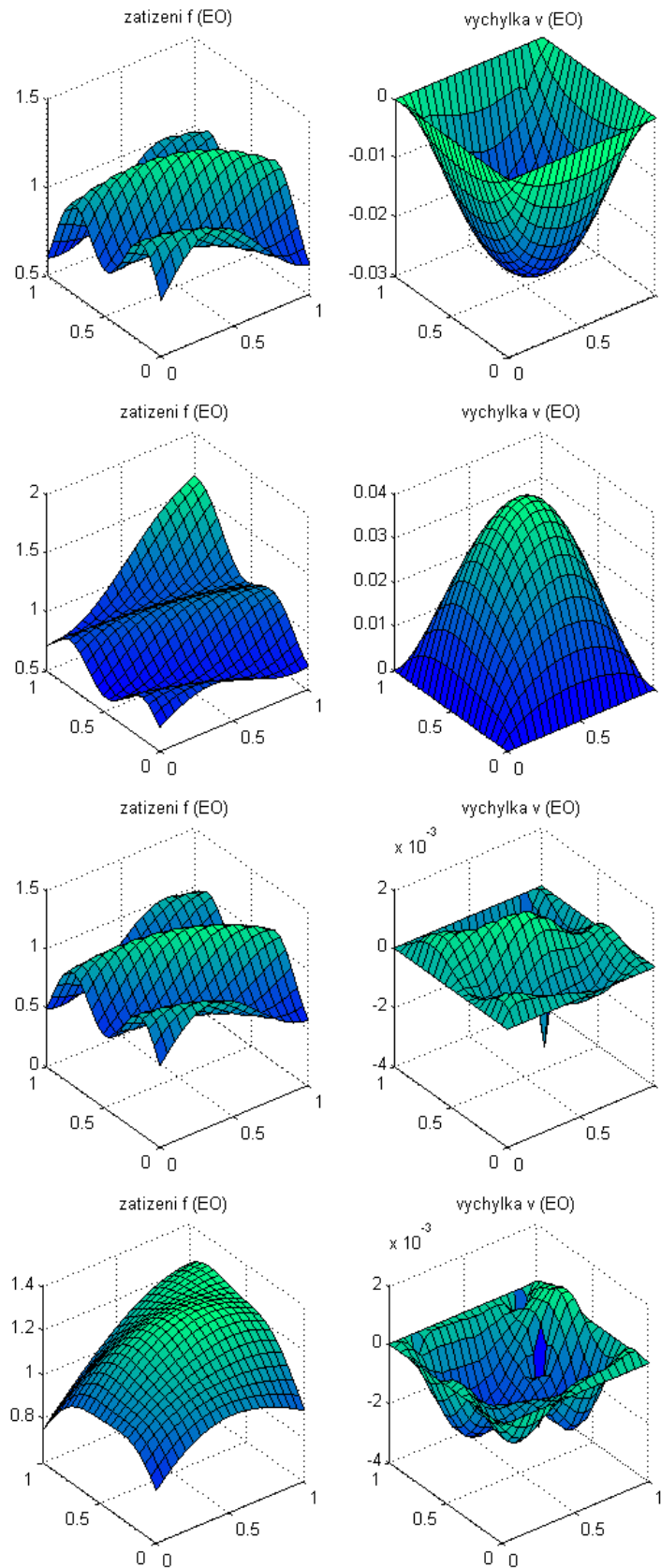
$$\text{VSS} = 2.100286 \cdot 10^{-5}.$$

Hodnota je kladná, což znamená, že použití stochastické optimalizace zlepší hodnotu účelové funkce oproti deterministické optimalizace.

Pro přehlednost jsou znázorněny průběhy zatížení, řízení i průhybu membrány pouze pro osm scénářů EO úlohy, dále pro porovnání jsou zobrazeny i výsledky EV úlohy.

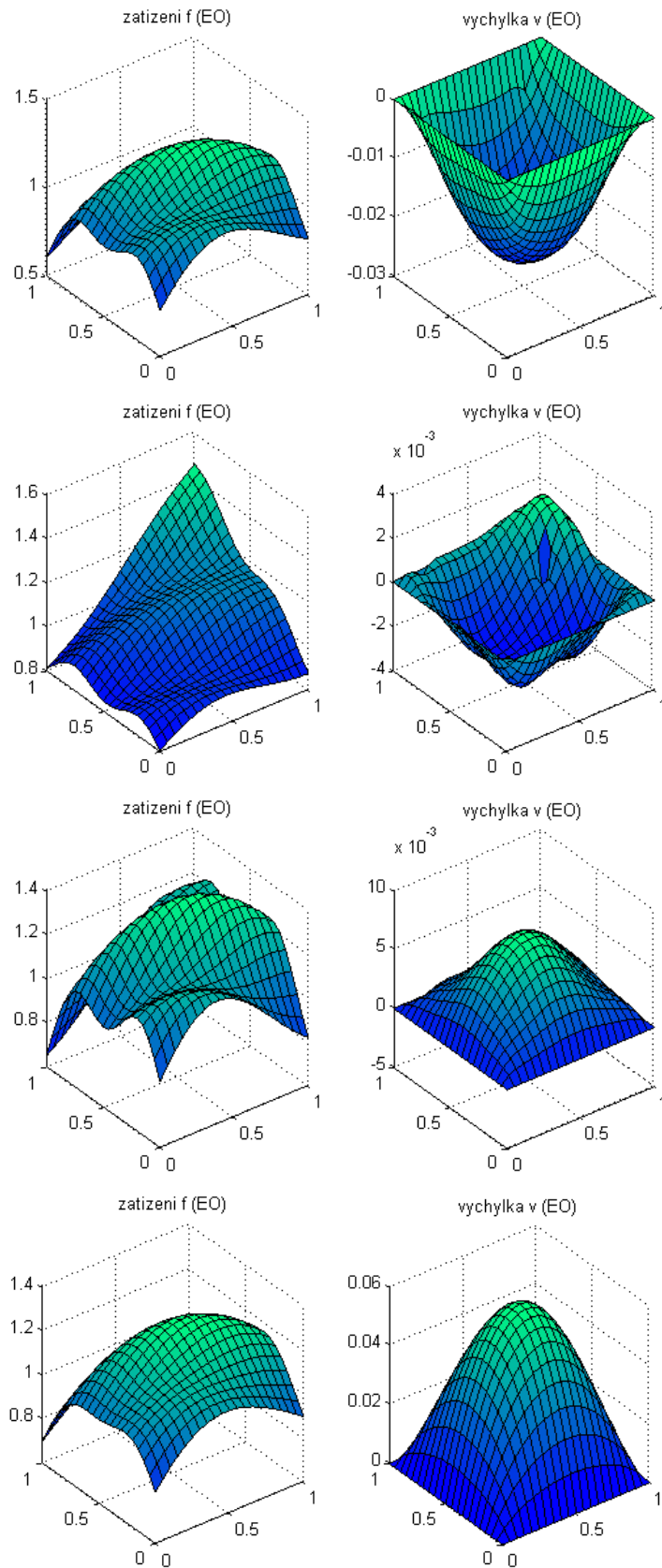
Na obrázku 4.3.4 lze vidět že řízení není dostatečně hladké, což se projeví i na výsledném průhybu membrány.

4. ÚLOHA STOCHASTICKÉ OPTIMALIZACE



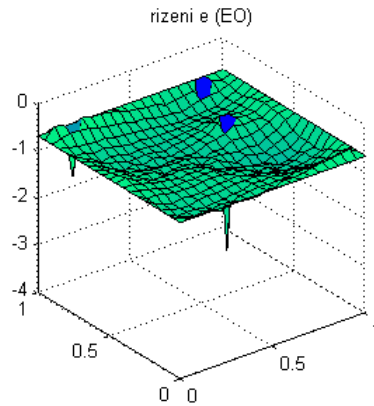
Obrázek 4.4: Zátížení a průhyb membrány pro různé scénáře

4.3. APROXIMACE ÚLOHY: METODA SÍTÍ

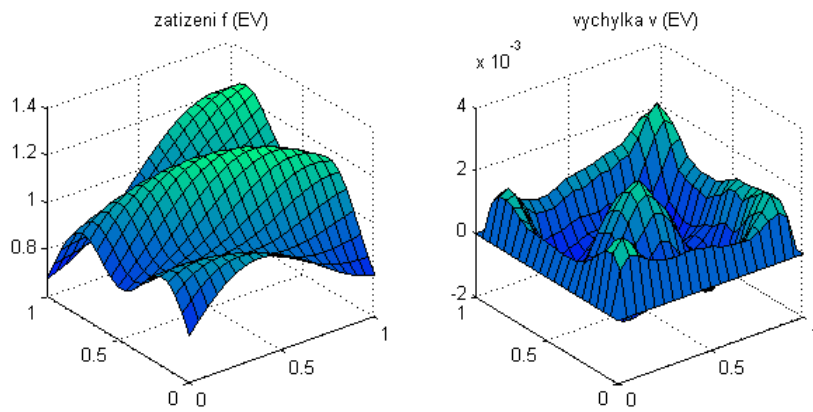


Obrázek 4.5: Zatížení a průhyb membrány pro různé scénáře v EO úloze

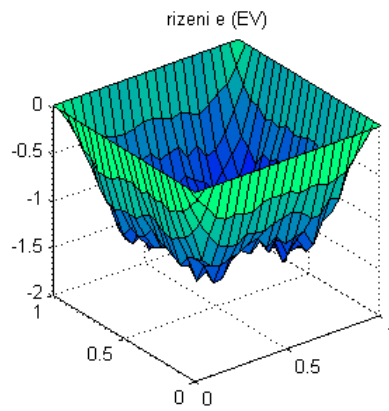
4. ÚLOHA STOCHASTICKÉ OPTIMALIZACE



Obrázek 4.6: Konstantní řízení v EO úloze



Obrázek 4.7: Zatížení a průhyb membrány v EV úloze



Obrázek 4.8: Konstantní řízení v EV úloze

4.3. APROXIMACE ÚLOHY: METODA SÍTÍ

```
.. vEV(i,j) =e= 1/4*(vEV(i-1,j)+vEV(i+1,j)+vEV(i,j-1)+vEV(i,j+1)+
    sqr(h)*(Ef(i,j)+ET(i,j)*eEV(i,j)));
expected_value_f(i,j) .. Ef(i,j)=e= sum(s,f(s,i,j)*prob(s));
expected_value_T(i,j) .. ET(i,j) =e= sum(s,T(s,i,j)*prob(s));
OP1EV(j) .. vEV('0',j) =e= 0;
OP2EV(i,j)$ (ord(i) eq card(i)) .. vEV(i,j) =e= 0;
OP3EV(i) .. vEV(i,'0') =e= 0;
OP4EV(i,j)$ (ord(j) eq card(j)) .. vEV(i,j) =e= 0;

ufEEV .. zEEV =e= sum((s,i,j),prob(s)*sqr(h)/9*a(i)*b(j)*sqr(vEVs(s,i,j)-u(i,j)))
pruhybEEV(s,i,j)$ ((ord(i) ne 1)and(ord(i) ne card(i))and(ord(j) ne 1)and
    (ord(j) ne card(j)))
    .. vEVs(s,i,j) =e= 1/4*(vEVs(s,i-1,j)+vEVs(s,i+1,j)+vEVs(s,i,j-1)+
        vEVs(s,i,j+1)+sqr(h)*(f(s,i,j)+T(s,i,j)*eEVs(i,j)));
OP1EEV(s,j) .. vEVs(s,'0',j) =e= 0;
OP2EEV(s,i,j)$ (ord(i) eq card(i)) .. vEVs(s,i,j) =e= 0;
OP3EEV(s,i) .. vEVs(s,i,'0') =e= 0;
OP4EEV(s,i,j)$ (ord(j) eq card(j)) .. vEVs(s,i,j) =e= 0;

model model_pruhybEO /ufEO,pruhyb,OP1,OP2,OP3,OP4/;
model model_pruhybEV /ufEV,pruhybEV,OP1EV,OP2EV,OP3EV,OP4EV,expected_value_f,
    expected_value_T/;
model model_pruhybEEV /ufEEV,pruhybEEV,OP1EEV,OP2EEV,OP3EEV,OP4EEV/;

e.l(i,j)=-0.7;

solve model_pruhybEO using nlp minimizing zEO;
solve model_pruhybEV using nlp minimizing zEV;
eEVs.up(i,j)=eEV.l(i,j);
eEVs.lo(i,j)=eEV.l(i,j);
solve model_pruhybEEV using nlp minimizing zEEV;
VSS=zEEV.l-zEO.l;
zlepseni=zEEV.l/zEO.l;
```

Obrázek 4.9: Ukázka prostředí GAMSu

4.3.5. Hladkost řízení

Nyní chceme dosáhnout, aby funkce $e(x, y)$ byla hladká. K vyšetření průběhu funkce v matematické analýze používáme derivace funkce. K úlohám matematického programování, viz 4.3.3 přidáme další omezení popisující průběh řízení $e(x, y)$. Předpokládejme, že musí být splněna následující podmínky

$$\alpha_1 \leq \frac{e_{i+1,j} - e_{i,j}}{h} - \frac{e_{i,j} - e_{i-1,j}}{h} \leq \beta_1 \quad i = 1, \dots, M-1, j = 0, \dots, N \quad (4.28)$$

$$\alpha_2 \leq \frac{e_{i,j+1} - e_{i,j}}{h} - \frac{e_{i,j} - e_{i,j-1}}{h} \leq \beta_2 \quad i = 1, \dots, N-1, i = 0, \dots, M. \quad (4.29)$$

Podmínky omezují hodnoty rozdílů derivací ve dvou sousedních uzlových bodech, kde $\alpha_1, \alpha_2, \beta_1, \beta_2$ jsou předem dané konstanty. Využili jsme formule pro výpočet první zpětné difference s diskretizační chybou řádu $O(h)$.

Úlohy matematického programování 4.3.3 popisující náš problém nyní budou ve tvaru

EO úloha matematického programování:

$$z_{\min}^{\text{EO}} = \min \sum_{s=1}^R \sum_{j=0}^N \sum_{i=0}^M p_s \frac{h^2}{9} a_i b_j (V_{s,i,j} - U_{i,j})^2 \quad (4.30)$$

$$V_{s,i,j} = \frac{1}{4} (V_{s,i+1,j} + V_{s,i-1,j} + V_{s,i,j+1} + V_{s,i,j-1}) + h^2 (f_{s,i,j} + \mathbf{T}_{s,i,j} e_{i,j}) \quad (4.31)$$

$$i = 1 \dots N-1, \quad j = 1 \dots M-1$$

$$V_{s,0,j} = V_{s,M,j} = 0, \quad j = 0 \dots N \quad (4.32)$$

$$V_{s,i,0} = V_{s,i,N} = 0, \quad i = 0 \dots M \quad (4.33)$$

$$\alpha_1 \leq \frac{e_{i-1,j} - 2e_{i,j} - e_{i+1,j}}{h} \leq \beta_1 \quad i = 1, \dots, M-1, j = 0, \dots, N \quad (4.34)$$

$$\alpha_2 \leq \frac{e_{i,j-1} - 2e_{i,j} - e_{i,j+1}}{h} \leq \beta_2 \quad i = 1, \dots, N-1, i = 0, \dots, M \quad (4.35)$$

EV úloha matematického programování:

$$z_{\min}^{\text{EV}} = \min \sum_{j=0}^N \sum_{i=0}^M \frac{h^2}{9} a_i b_j (V_{i,j} - U_{i,j})^2 \quad (4.36)$$

$$V_{i,j} = \frac{1}{4} (V_{i+1,j} + V_{i-1,j} + V_{i,j+1} + V_{i,j-1}) + h^2 \left(\sum_{s=1}^R f_{s,i,j} p_s + \left(\sum_{s=1}^R \mathbf{T}_{s,i,j} p_s \right) e_{i,j} \right) \quad (4.37)$$

$$i = 1 \dots N-1, \quad j = 1 \dots M-1$$

$$V_{0,j} = V_{M,j} = 0, \quad j = 0 \dots N \quad (4.38)$$

$$V_{i,0} = V_{i,N} = 0, \quad i = 0 \dots M \quad (4.39)$$

$$\alpha_1 \leq \frac{e_{i-1,j} - 2e_{i,j} - e_{i+1,j}}{h} \leq \beta_1 \quad i = 1, \dots, M-1, j = 0, \dots, N \quad (4.40)$$

$$\alpha_2 \leq \frac{e_{i,j-1} - 2e_{i,j} - e_{i,j+1}}{h} \leq \beta_2 \quad i = 1, \dots, N-1, i = 0, \dots, M \quad (4.41)$$

4.3. APROXIMACE ÚLOHY: METODA SÍTÍ

EEV úloha matematického programování:

$$z_{\min}^{\text{EEV}} = \min \sum_{s=1}^R \sum_{j=0}^N \sum_{i=0}^M p_s \frac{h^2}{9} a_i b_j (V_{s,i,j} - U_{i,j})^2 \quad (4.42)$$

$$V_{s,i,j} = \frac{1}{4} (V_{s,i+1,j} + V_{s,i-1,j} + V_{s,i,j+1} + V_{s,i,j-1}) + h^2 (f_{s,i,j} + \mathbf{T}_{s,i,j} e_{i,j}^{\text{EV}}) \quad (4.43)$$

$$i = 1 \dots N-1, \quad j = 1 \dots M-1$$

$$V_{s,0,j} = V_{s,M,j} = 0, \quad j = 0 \dots N \quad (4.44)$$

$$V_{s,i,0} = V_{s,i,N} = 0, \quad i = 0 \dots M \quad (4.45)$$

$$\alpha_1 \leq \frac{e_{i-1,j} - 2e_{i,j} - e_{i+1,j}}{h} \leq \beta_1 \quad i = 1, \dots, M-1, j = 0, \dots, N \quad (4.46)$$

$$\alpha_2 \leq \frac{e_{i,j-1} - 2e_{i,j} - e_{i,j+1}}{h} \leq \beta_2 \quad i = 1, \dots, N-1, i = 0, \dots, M \quad (4.47)$$

4.3.6. Výsledky

Úlohy v odstavci 4.3.5 opět řešíme v GAMSu pro stejné zadání jako v odstavci 4.3.4 s rozdílem, že volíme hodnoty omezení

$$\alpha_1 = \beta_1 = -1$$

$$\alpha_2 = \beta_2 = 1$$

Pro tuto úlohu jsme docílili výsledků

$$z_{\min}^{\text{EO}} = 4.688023 \cdot 10^{-4} \quad (4.48)$$

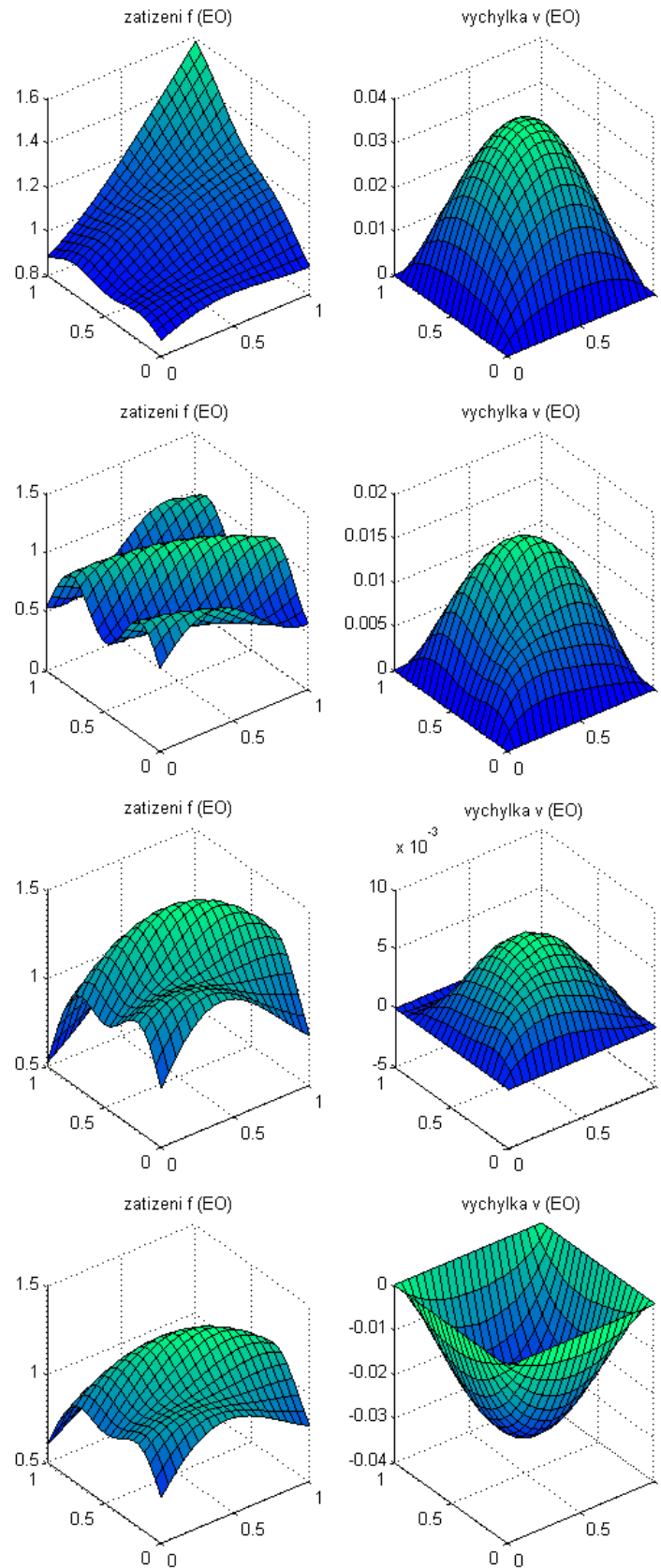
$$z_{\min}^{\text{EV}} = 1.392415 \cdot 10^{-7} \quad (4.49)$$

$$z_{\min}^{\text{EEV}} = 6.067492 \cdot 10^{-4} \quad (4.50)$$

$$\text{VSS} = 1.379469 \cdot 10^{-4}. \quad (4.51)$$

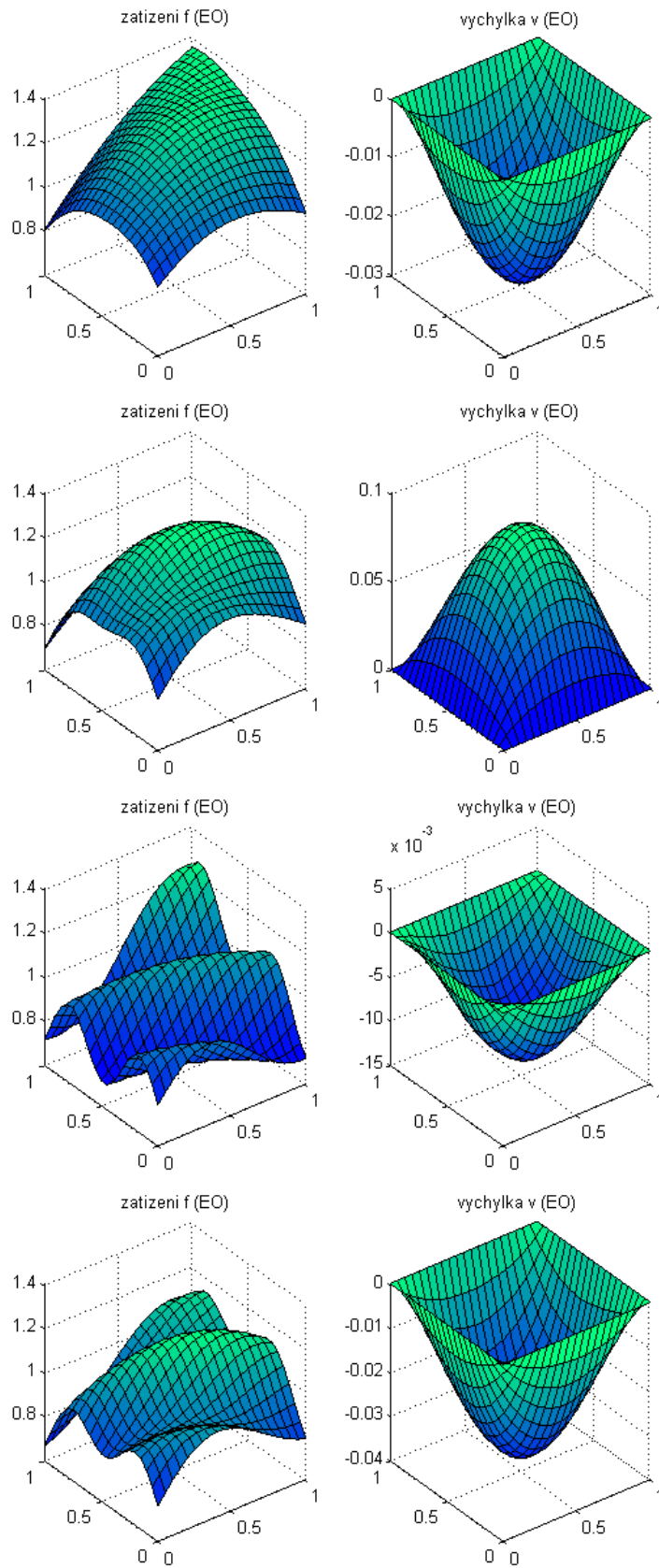
Jak je vidět na obrázcích docílili jsme hladkého průběhu řízení $e(x, y)$. Nepatrně se zhoršila hodnota účelová funkce oproti předchozí úloze, ale dosažená hodnota je postačující. Veličina VSS opět říká, že použitím stochastické optimalizace dosáhneme lepších výsledků.

4. ÚLOHA STOCHASTICKÉ OPTIMALIZACE



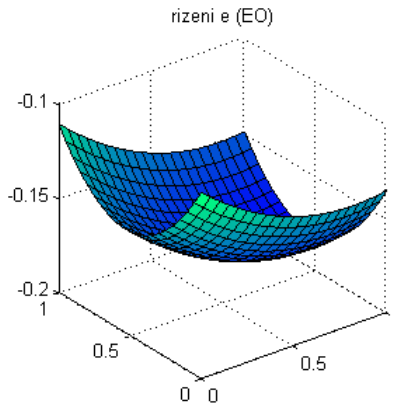
Obrázek 4.10: Zátížení a průhyb membrány pro různé scénáře

4.3. APROXIMACE ÚLOHY: METODA SÍTÍ

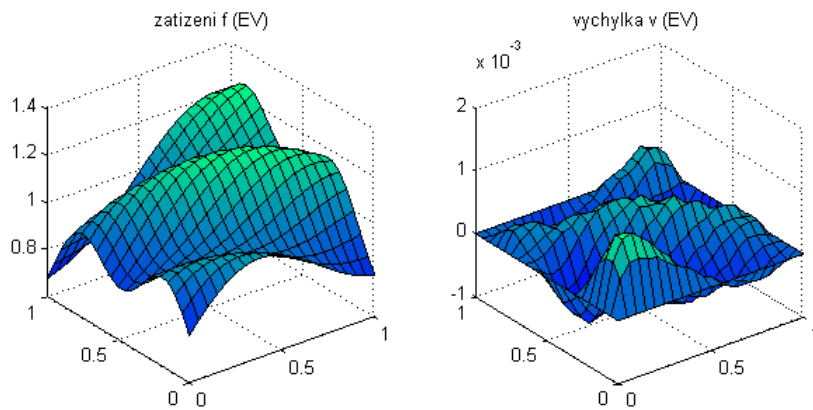


Obrázek 4.11: Zatížení a průhyb membrány pro různé scénáře v EO úloze

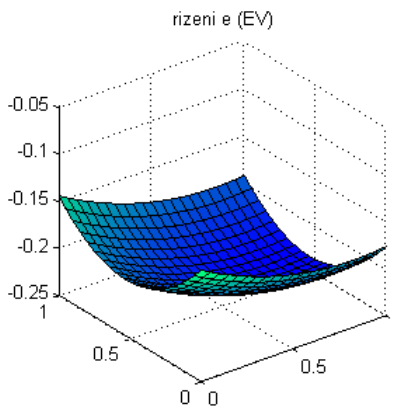
4. ÚLOHA STOCHASTICKÉ OPTIMALIZACE



Obrázek 4.12: Konstantní řízení v EO úloze



Obrázek 4.13: Zatížení a průhyb membrány v EV úloze



Obrázek 4.14: Konstantní řízení v EV úloze

4.4. Odhad kvality řešení

V této kapitole se budeme zabývat odhadem kvality kandidátního řízení $\hat{e}(x, y)$ získaného z úlohy stochastické optimalizace průhybu membrány pomocí metody sítí, viz kapitola 4.3. Určení kvality kandidátního řešení využijeme pravděpodobnostní dolní mez příslušné úlohy stochastického programování. Konstruujeme interval spolehlivosti chyby aproximace (optimality gap) [7, 10], což je rozdíl v hodnotách účelových funkcí s kandidátním řešením a optimálním řešením. Metodika mezí je založena na řešení aproximovaného problému generovaného výběrem náhodné veličiny ξ pomocí metody Monte Carlo.

4.4.1. Intervalový odhad chyby aproximace

Uvažovanou úlohu spojitě stochastické optimalizace (4.4) symbolicky zapíšeme ve tvaru

$$z^* = \min_{e, v(\xi)} E(F(\xi, e, v(\xi))). \quad (4.52)$$

Vygenerujeme náhodný výběr z ξ s n scénáři $\xi^1, \xi^2, \dots, \xi^n$. Pak je tato úloha aproximována SAA (sample average approximation) úlohou (4.30)–(4.35)

$$z_n^* = \min_{e, v(\xi)} \frac{1}{n} \sum_{i=1}^n F(\xi^i, e, v(\xi)), \quad (4.53)$$

kteřou řešíme vhodným deterministickým algoritmem.

Definujeme chybu aproximace jako

$$G(\hat{e}) = \min_{v(\xi)} E(F(\xi, \hat{e}, v(\xi))) - z^*. \quad (4.54)$$

K odhadu kvality kandidátního řešení \hat{e} použijeme intervalový odhad chyby aproximace. Pro odhad chyby aproximace vygenerujeme n_g výběrů o rozsahu n , tj.

$$\xi^{ij}, \quad i = 1, 2, \dots, n, j = 1, 2, \dots, n_g$$

a použijeme průměrování. Pak bodový odhad $G(\hat{e})$ získáme jako

$$\begin{aligned} \bar{G}_{n, n_g}(\hat{e}) &= \frac{1}{n_g} \sum_{j=1}^{n_g} G_n^j(\hat{e}) = \\ &= \frac{1}{n_g} \sum_{j=1}^{n_g} \left[\min_{v(\xi)} \frac{1}{n} \sum_{i=1}^n F(\xi^{ij}, \hat{e}, g(\xi^{ij})) - \min_{e, v(\xi)} \frac{1}{n} \sum_{i=1}^n F(\xi^{ij}, e, g(\xi^{ij})) \right]. \end{aligned} \quad (4.55)$$

Intervalový odhad se spolehlivostí $1 - \alpha$ získáme na základě centrální limitní věty

$$G(\hat{e}) \in \left[0, \bar{G}_{n, n_g}(\hat{e}) + \epsilon \right], \quad (4.56)$$

ϵ je odhad chyby vyjádřený jako

$$\epsilon = \frac{t_{1-\alpha}(n_g - 1) s_{n_g}(\hat{e})}{\sqrt{n_g}}, \quad (4.57)$$

kde $t_{1-\alpha}(n_g - 1)$ je $(1 - \alpha)$ -kvantil Studentova rozdělení s $(n_g - 1)$ stupni volnosti a $s_{n_g}(\hat{e}) = \sqrt{\frac{1}{n_g - 1} \sum_{j=1}^{n_g} \left(\bar{G}_n^j(\hat{e}) - \bar{G}_{n, n_g}(\hat{e}) \right)^2}$ je výběrová směrodatná odchylka.

4.4.2. Výsledky

Byla posuzována kvalita výsledků pro řešení \hat{e} úlohy (4.30)–(4.35). Kandidátní řešení bylo vypočteno pomocí 100 scénářů. Pro metodu Monte Carlo bylo generováno $n_g = 30$ realizací náhodné veličiny ξ . Dosáhli jsme následujících výsledků se spolehlivostí 95%

$$\text{Bodový odhad } \bar{G}_{n,n_g} = 0,5 \cdot 10^{-4}$$

$$\text{Odhad chyby } \epsilon = 0,2 \cdot 10^{-4}$$

$$\text{Intervalový odhad } G_{n,n_g} = \langle 0; 0,7 \cdot 10^{-4} \rangle$$

Bodový odhad hodnoty účelové funkce byl vypočten řešením pro 100 scénářů, získali jsme hodnotu

$$z_{100}^* = 4,69 \cdot 10^{-4}.$$

Šířka intervalového odhadu chyby aproximace tvoří 14% z bodového odhadu hodnoty účelové funkce poukazuje na dobrou kvalitu posuzovaného řešení.

4.5. Aproximace úlohy: metoda konečných prvků

V této části zavedeme základní pojmy metody konečných prvků [1, 4] aplikované na stacionární problém průhybu membrány na čtvercové oblasti s Dirichletovou okrajovou podmínkou. Řešení problému spočívá v zavedení slabé formulace Poissonovy rovnice (4.2). Následně popíšeme konečně prvkový model pro libovolný n –vrcholový element z hlediska interpolační funkce. Této funkci se říká bázová a je Lagrangeova typu, tzn. že mohou být konstruovány z Lagrangeových interpolačních polynomů. Následně sestavím interpolační funkce pro 4–uzlový obdélníkový element a zkompletujeme elementární rovnice v soustavu lineárních rovnic popisující celý problém.

Odvozený postup aproximace metodou konečných prvků je navíc proti zadání diplomové práce a je vyzkoušen jako první krok výzkumu, který bude dále rozvíjen.

4.5.1. Klasická formulace

Naším cílem je určit funkci popisující průhyb membrány

$$v(x, y) \in C^2(\Omega) \cap C(\bar{\Omega}) \quad (4.58)$$

vyhovující diferenciální rovnici

$$\frac{\partial^2 v}{\partial x^2}(x, y) + \frac{\partial^2 v}{\partial y^2}(x, y) = -f(x, y) \quad \text{v } \Omega \quad (4.59)$$

a splňující okrajovou podmínku

$$v(x, y) = 0 \quad \text{na } \Gamma. \quad (4.60)$$

Předpokládejme $\mathbf{n} = (n_x, n_y)^T \equiv (n_1, n_2)$ je jednotkový vektor vnější normály hranice a

$$\frac{\partial v}{\partial \mathbf{n}} = n_x \frac{\partial v}{\partial x} + n_y \frac{\partial v}{\partial y} = q_n \quad (4.61)$$

je derivace ve směru normály. Pro existenci klasického řešení je třeba předpokládat hladkost dat

$$f \in C(\bar{\Omega})$$

a že Ω je regulární oblast s hladkou hranicí Γ .

4.5. APROXIMACE ÚLOHY: METODA KONEČNÝCH PRVKŮ

4.5.2. Slabá formulace

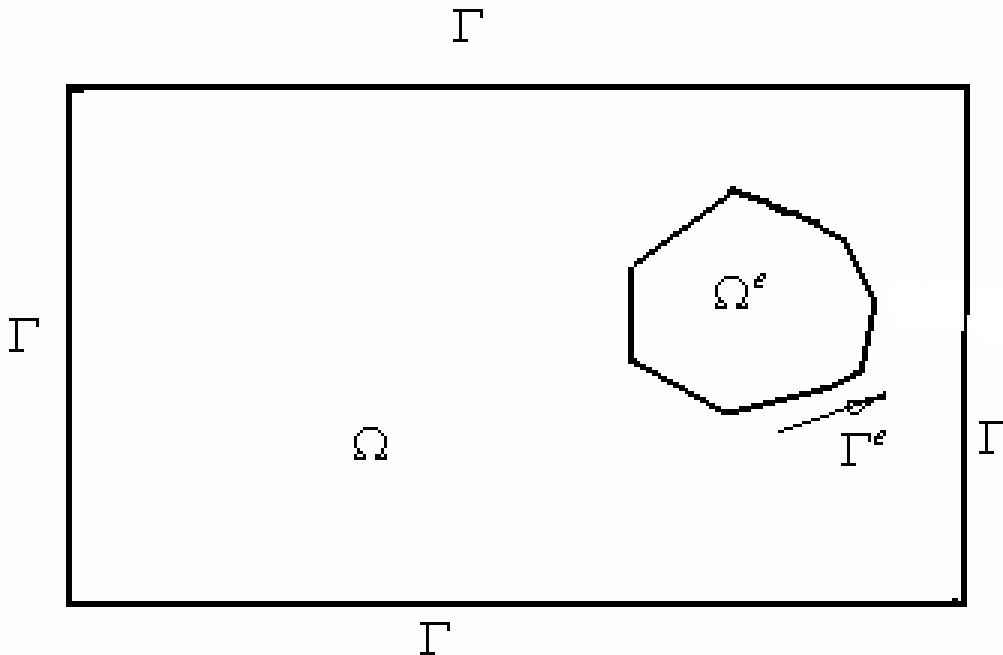
Nejprve zavedeme Greenovu formuli.

VĚTA 4.5.1 (GREENOVA FORMULE) *Nechť $f(x_1, x_2), g(x_1, x_2) \in H^1(\Omega)$. Potom platí*

$$\int_{\Omega} \frac{\partial f}{\partial x_i} dx = \int_{\Gamma} f g n_i ds - \int_{\Gamma} f \frac{\partial g}{\partial x_i} dx, \quad (4.62)$$

kde $dx = dx_1 dx_2$ a ds je diferenciál oblouku Γ .

K slabé formulaci se dostaneme vynásobením rovnice (4.59) testovací funkcí w , která splňuje podmínku $w = 0$ na Γ , následnou integrací přes libovolnou podmnožinu Ω^e s hranicí Γ^e a použitím Greenovy formule (4.62). Libovolnou podmnožinu může representovat n -vrcholový element uvnitř oblasti Ω , viz Obr. 4.5.2.



Obrázek 4.15: Elementární oblast

Dostáváme rovnici

$$\int_{\Omega^e} w(x, y) \left(\frac{\partial^2 v}{\partial x^2}(x, y) + \frac{\partial^2 v}{\partial y^2}(x, y) + f(x, y) \right) dx dy = 0. \quad (4.63)$$

Využitím vztahů $\frac{\partial}{\partial x} \left(w \frac{\partial v}{\partial x} \right) = \frac{\partial w}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial^2 v}{\partial x^2} w$ a $\frac{\partial}{\partial y} \left(w \frac{\partial v}{\partial y} \right) = \frac{\partial w}{\partial y} \frac{\partial v}{\partial y} + \frac{\partial^2 v}{\partial y^2} w$ dostaneme

$$\int_{\Omega^e} \left[\frac{\partial}{\partial x} \left(w \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left(w \frac{\partial v}{\partial y} \right) - \frac{\partial w}{\partial x} \frac{\partial v}{\partial x} - \frac{\partial w}{\partial y} \frac{\partial v}{\partial y} + w f \right] dx dy = 0. \quad (4.64)$$

Nyní využitím Greenovy formule dostáváme

$$\int_{\Omega^e} \frac{\partial}{\partial x} \left(w \frac{\partial v}{\partial x} \right) dx dy = \int_{\Gamma^e} w \frac{\partial v}{\partial x} n_x ds \quad (4.65)$$

$$\int_{\Omega^e} \frac{\partial}{\partial y} \left(w \frac{\partial v}{\partial y} \right) dx dy = \int_{\Gamma^e} w \frac{\partial v}{\partial y} n_y ds. \quad (4.66)$$

4. ÚLOHA STOCHASTICKÉ OPTIMALIZACE

S využitím derivace ve směru vnější normály (4.61) zapíšeme rovnici (4.64) ve tvaru

$$\int_{\Omega^e} \left[\frac{\partial w}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial w}{\partial y} \frac{\partial v}{\partial y} \right] dx dy = - \int_{\Omega^e} w f dx dy + \int_{\Gamma^e} w q_n ds. \quad (4.67)$$

Levá strana se rovnice se označuje $a(v, w)$ a pravá $L(w)$. $a(v, w)$ je symetrická bilineární forma a $L(w)$ je lineární funkcionál. Slabou formulaci Poissonovy rovnice (4.59) rozumíme úlohu

$$\text{najít } v \in W \text{ splňující } a(v, w) = L(w) \quad \forall w \in V,$$

kde W je množina přípustných řešení a $V = \{w \in H^1(\Omega) | w = 0 \text{ na } \Gamma\}$ je prostor testovacích funkcí.

4.5.3. Konečně prvkový model

V tomto odstavci popíšeme vytvoření konečně prvkového modelu na základě předpokládaného přibližného řešení. Řešení problému (4.67) pro libovolný n -vrcholový element je definováno vztahem

$$v^e(x, y) = \sum_{j=1}^n v^e(x, y) \psi_j^e(x, y), \quad (4.68)$$

kde

$v^e(x, y)$ je uzlová hodnota řešení $v(x, y)$ v j -tém uzlu elementu

$\psi_j^e(x, y)$ je bázová funkce pro $v(x, y)$ v j -tém uzlu elementu.

Bázová funkce musí splňovat následující podmínky

$$\psi_j^e(x_k, y_k) = \delta_{jk} \text{ pro všechny uzly } (j, k) = 1, 2, \dots, n, \quad (4.69)$$

$$\sum_{j=1}^n \psi_j^e(x, y) = 1 \text{ v oblasti } \Omega^e, \quad (4.70)$$

kde δ_{jk} je Kroneckerovo delta definované

$$\delta_{jk} = 1 \text{ pro } j = k$$

$$\delta_{jk} = 0 \text{ pro } j \neq k$$

Substitucí rovnice (4.68) do slabé formulace (4.67) dostaneme

$$\int_{\Omega^e} \left[\frac{\partial w}{\partial x} \sum_{j=1}^n v_j^e \frac{\partial \psi_j^e}{\partial x} + \sum_{j=1}^n v_j^e \frac{\partial \psi_j^e}{\partial y} \frac{\partial v}{\partial y} \right] dx dy = - \int_{\Omega^e} w f dx dy + \int_{\Gamma^e} w q_n ds. \quad (4.71)$$

4.5. APROXIMACE ÚLOHY: METODA KONEČNÝCH PRVKŮ

Protože $w_i = \psi_j^e$ pro $i = 1, 2, \dots, n$ a po záměně pořadí integrace a sumace v (4.71), řešení rovnice vede na problém řešení soustavy lineárních rovnic. Dostáváme tak koněčně prvkový model pro n -vrcholový element.

$$\sum_{j=1}^n K_{ij}^e V_j^e = F_i^e + Q_i^e \text{ pro } i = 1, 2, \dots, n, \quad (4.72)$$

kde

$$K_{ij}^e = \int_{\Omega^e} \left[\frac{\partial \psi_i^e}{\partial x} \frac{\partial \psi_j^e}{\partial x} + \frac{\partial \psi_i^e}{\partial y} \frac{\partial \psi_j^e}{\partial y} \right] dx dy \quad (4.73)$$

$$F_i^e = - \int_{\Omega^e} [\psi_i^e f] dx dy \quad (4.74)$$

$$Q_i^e = \int_{\Gamma^e} \psi_i^e q_n ds. \quad (4.75)$$

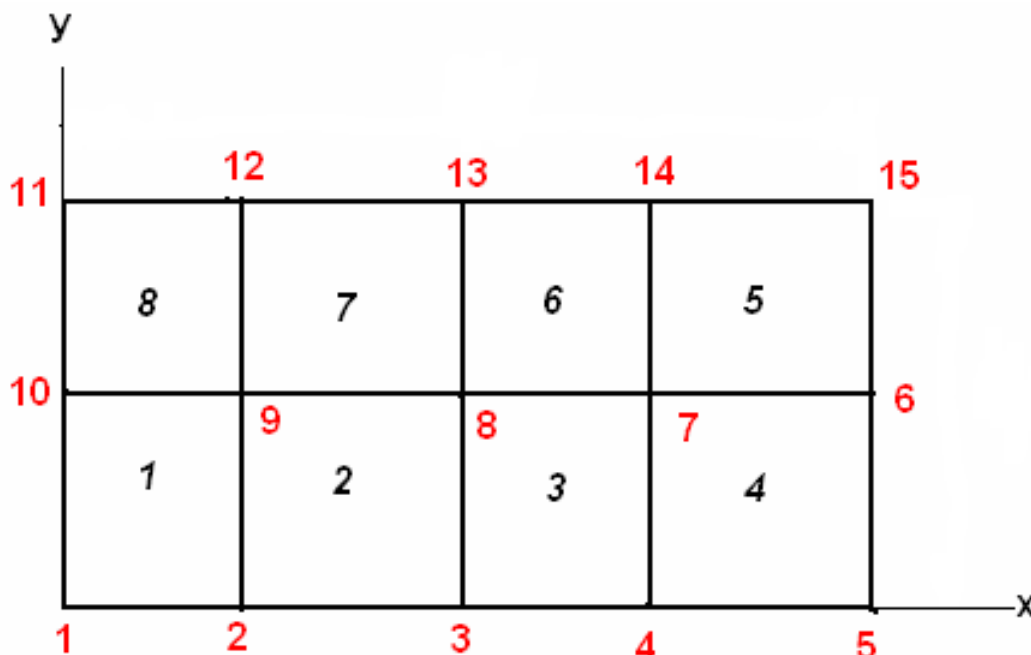
K_{ij} se nazývá elementární matice tuhosti a F_i je elementární vektor zatížení.

4.5.4. Bázové funkce

Předpokládejme, že oblast Ω je obdélníková, kterou pokryjeme obdélníkovou sítí S skládající se z obdélníkových elementů E takových, že uzávěry každých dvou různých obdélníku jsou buďto disjunktní nebo mají společnou stranu nebo vrchol a že platí

$$\bar{\Omega} = \bigcup_{E \in S} \bar{E}.$$

Elementární oblasti se také nazývají prvky a jejich vrcholy uzly. Síť charakterizuje počet prvků sítě (M), počet všech uzlů (N) a počet uzlů ležících na Γ (N^*), viz obr.4.16.



Obrázek 4.16: Obdélníková síť

4. ÚLOHA STOCHASTICKÉ OPTIMALIZACE

Lineární bázové funkce pro obdélníkovou oblast získáme pomocí součinu jednodimenzionálních Lagrangeových interpolačních polynomů. Lagrangeův interpolační polynom k -tého řádu pro n bodů $(x_i, i = 1, 2, \dots, n)$ je stejný jako jednodimenzionální interpolační funkce

$$L_k(x) = \prod_{i=1, i \neq k}^n \frac{x - x_i}{x_k - x_i} = \psi_k^e(x). \quad (4.76)$$

Ta splňuje podmínky

$$L_k(x_j) = \delta_{jk} \quad (4.77)$$

$$\sum_{k=1}^n L_k(x) = 1. \quad (4.78)$$

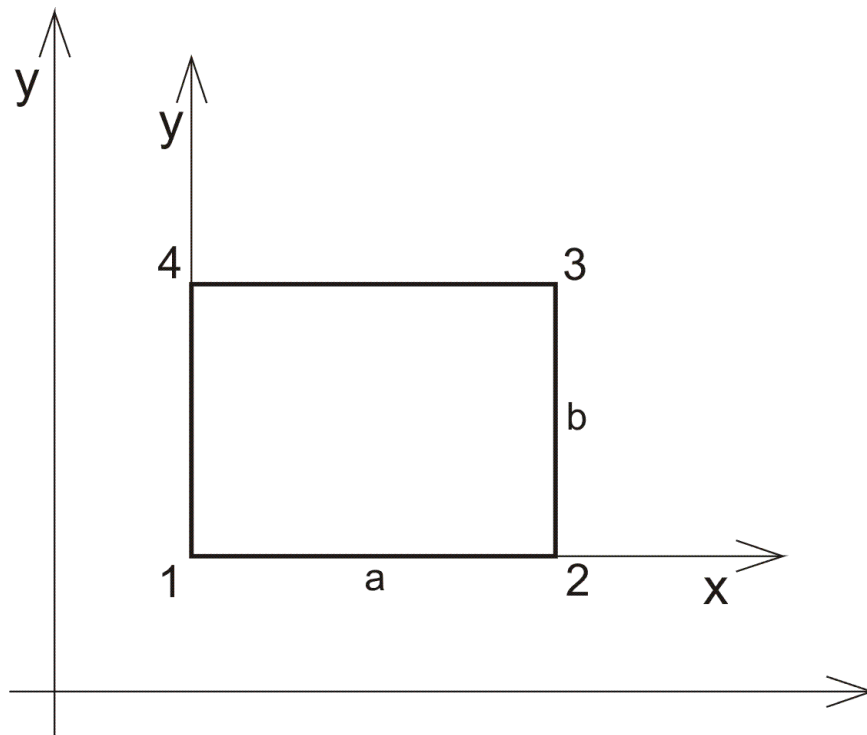
Pro čtyř vrcholový obdélníkový element, viz obr. 4.17, jsou bázové funkce tvaru

$$\psi_1^e(x, y) = \left(1 - \frac{x}{a}\right) \left(1 - \frac{y}{b}\right)$$

$$\psi_2^e(x, y) = \frac{x}{a} \left(1 - \frac{y}{b}\right)$$

$$\psi_3^e(x, y) = \frac{x}{a} \frac{y}{b}$$

$$\psi_4^e(x, y) = \left(1 - \frac{x}{a}\right) \frac{y}{b}$$



Obrázek 4.17: Elementární oblast

Bázové funkce jsou definovány ve smyslu lokálních souřadnic x, y .

4.5.5. Elementární soustava rovnic

Koeficienty elementární matice tuhosti se počítají ze vztahu

$$K_{ij}^e = \int_{\Omega^e} \left[\frac{\partial \psi_i^e}{\partial x} \frac{\partial \psi_j^e}{\partial x} + \frac{\partial \psi_i^e}{\partial y} \frac{\partial \psi_j^e}{\partial y} \right] dx dy. \quad (4.79)$$

Pro numerický výpočet elementární matice tuhosti se většinou používá Gauss–Legendreova kvadratura, ale v našem případě lze provést přímý výpočet integrálu. Nejdříve vypočteme parciální derivace bázových funkcí

$$\begin{aligned} \frac{\partial \psi_1^e(x, y)}{\partial x} &= \frac{1}{ab} \frac{\partial(a-x)(b-y)}{\partial x} = \frac{y-b}{ab} \\ \frac{\partial \psi_1^e(x, y)}{\partial y} &= \frac{1}{ab} \frac{\partial(a-x)(b-y)}{\partial y} = \frac{x-a}{ab} \\ \frac{\partial \psi_2^e(x, y)}{\partial x} &= \frac{1}{ab} \frac{\partial x(b-y)}{\partial x} = \frac{y-b}{ab} \\ \frac{\partial \psi_2^e(x, y)}{\partial y} &= \frac{1}{ab} \frac{\partial x(b-y)}{\partial y} = -\frac{x}{ab} \\ \frac{\partial \psi_3^e(x, y)}{\partial x} &= \frac{1}{ab} \frac{\partial xy}{\partial x} = \frac{y}{ab} \\ \frac{\partial \psi_3^e(x, y)}{\partial y} &= \frac{1}{ab} \frac{\partial xy}{\partial y} = \frac{x}{ab} \\ \frac{\partial \psi_4^e(x, y)}{\partial x} &= \frac{1}{ab} \frac{\partial(a-x)y}{\partial x} = -\frac{y}{ab} \\ \frac{\partial \psi_4^e(x, y)}{\partial y} &= \frac{1}{ab} \frac{\partial(a-x)y}{\partial y} = \frac{x-a}{ab}. \end{aligned}$$

Dosazením do rovnice (4.79) a integrací dostaneme

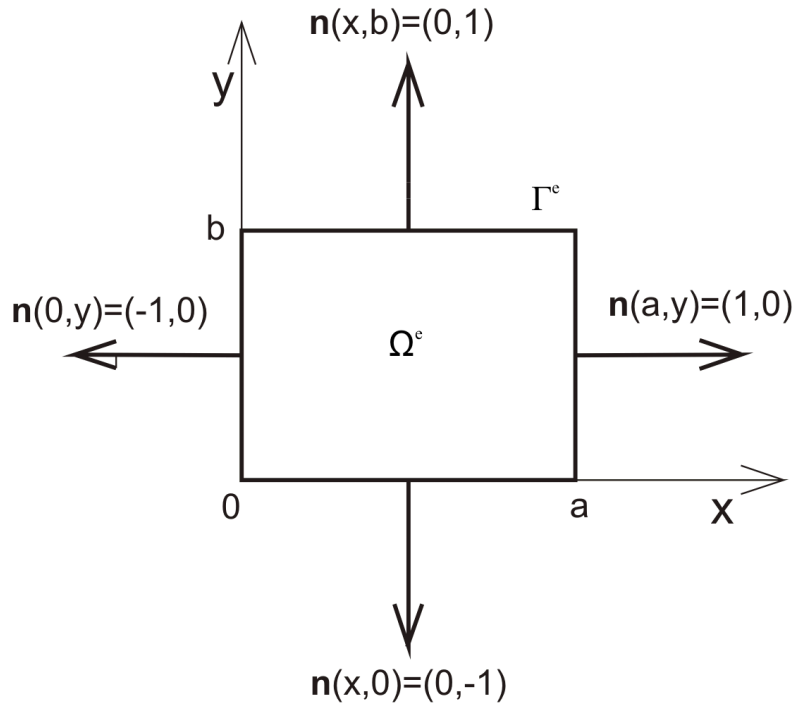
$$\begin{aligned} \mathbf{K}_{11}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a ((x-a)^2 + (y-b)^2) dx dy = \frac{a^2 + b^2}{3ab} \\ \mathbf{K}_{12}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a (-(x-a)x - (y-b)^2) dx dy = \frac{a^2 - 2b^2}{6ab} \\ \mathbf{K}_{13}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a ((x-a)x + (y-b)y) dx dy = -\frac{a^2 + b^2}{6ab} \\ \mathbf{K}_{14}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a (-(x-a)^2 - (y-b)y) dx dy = \frac{b^2 - 2a^2}{6ab} \\ \mathbf{K}_{21}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a (-(x-a)x - (y-b)^2) dx dy = \frac{a^2 - 2b^2}{6ab} \\ \mathbf{K}_{22}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a (x^2 + (y-b)^2) dx dy = \frac{a^2 + b^2}{3ab} \\ \mathbf{K}_{23}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a (-x^2 - (y-b)y) dx dy = \frac{b^2 - 2a^2}{6ab} \\ \mathbf{K}_{24}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a ((x-a)x + (y-b)y) dx dy = -\frac{a^2 + b^2}{6ab} \\ \mathbf{K}_{31}^e &= \frac{1}{a^2 b^2} \int_0^b \int_0^a ((x-a)x + (y-b)y) dx dy = -\frac{a^2 + b^2}{6ab} \end{aligned}$$

4. ÚLOHA STOCHASTICKÉ OPTIMALIZACE

$$\begin{aligned}
 \mathbf{K}_{32}^e &= \frac{1}{a^2b^2} \int_0^b \int_0^a (-x^2 - (y-b)y) dx dy = \frac{b^2 - 2a^2}{6ab} \\
 \mathbf{K}_{33}^e &= \frac{1}{a^2b^2} \int_0^b \int_0^a (x^2 + y^2) dx dy = \frac{a^2 + b^2}{3ab} \\
 \mathbf{K}_{34}^e &= \frac{1}{a^2b^2} \int_0^b \int_0^a (-(x-a)x - y^2) dx dy = \frac{a^2 - 2b^2}{6ab} \\
 \mathbf{K}_{41}^e &= \frac{1}{a^2b^2} \int_0^b \int_0^a (-(x-a)^2 - (y-b)y) dx dy = \frac{b^2 - 2a^2}{6ab} \\
 \mathbf{K}_{42}^e &= \frac{1}{a^2b^2} \int_0^b \int_0^a ((x-a)x + (y-b)y) dx dy = -\frac{a^2 + b^2}{6ab} \\
 \mathbf{K}_{43}^e &= \frac{1}{a^2b^2} \int_0^b \int_0^a (-(x-a)x - y^2) dx dy = \frac{a^2 - 2b^2}{6ab} \\
 \mathbf{K}_{44}^e &= \frac{1}{a^2b^2} \int_0^b \int_0^a ((x-a)^2 + y^2) dx dy = \frac{a^2 + b^2}{3ab}.
 \end{aligned}$$

Elementární matice tuhosti je typu 4×4 a je symetrická:

$$\mathbf{K}^e = \begin{pmatrix} \frac{a^2+b^2}{3ab} & \frac{a^2-2b^2}{6ab} & -\frac{a^2+b^2}{6ab} & \frac{b^2-2a^2}{6ab} \\ \frac{a^2-2b^2}{6ab} & \frac{a^2+b^2}{3ab} & \frac{b^2-2a^2}{6ab} & -\frac{a^2+b^2}{6ab} \\ -\frac{a^2+b^2}{6ab} & \frac{b^2-2a^2}{6ab} & \frac{a^2+b^2}{3ab} & \frac{a^2-2b^2}{6ab} \\ \frac{b^2-2a^2}{6ab} & -\frac{a^2+b^2}{6ab} & \frac{a^2-2b^2}{6ab} & \frac{a^2+b^2}{3ab} \end{pmatrix} \quad (4.80)$$



Obrázek 4.18: Vnější normály

4.5. APROXIMACE ÚLOHY: METODA KONEČNÝCH PRVKŮ

Uvažujeme—li rovnici (4.5), pak dle vzorce (4.74) složky vektoru elementárního náhodného zatížení vyjádříme ve tvaru

$$F_i^e = - \int_{\Omega^e} [\psi_i^e(f + T(\xi, x, y)e(x, y))] dx dy. \quad (4.81)$$

Hodnota integrálu (4.75) pro obdélníkovou oblast je

$$Q_i^e = 0,$$

proto již v další části práce nepočítáme s hodnotou Q_i^e .

Tedy řešení V_j ve vrcholech elementární oblasti Ω^e získáme vyřešením soustavy rovnic

$$\sum_{j=1}^n K_{ij}^e V_j^e = F_i^e \text{ pro } i = 1, 2, \dots, n. \quad (4.82)$$

4.5.6. Soustava rovnic

Obdélníkovou síť popisuje matice \mathbf{B} , která nám dává čísla (označení) uzlů jednotlivých prvků sítě. Například pro síť na obr. 4.16 je

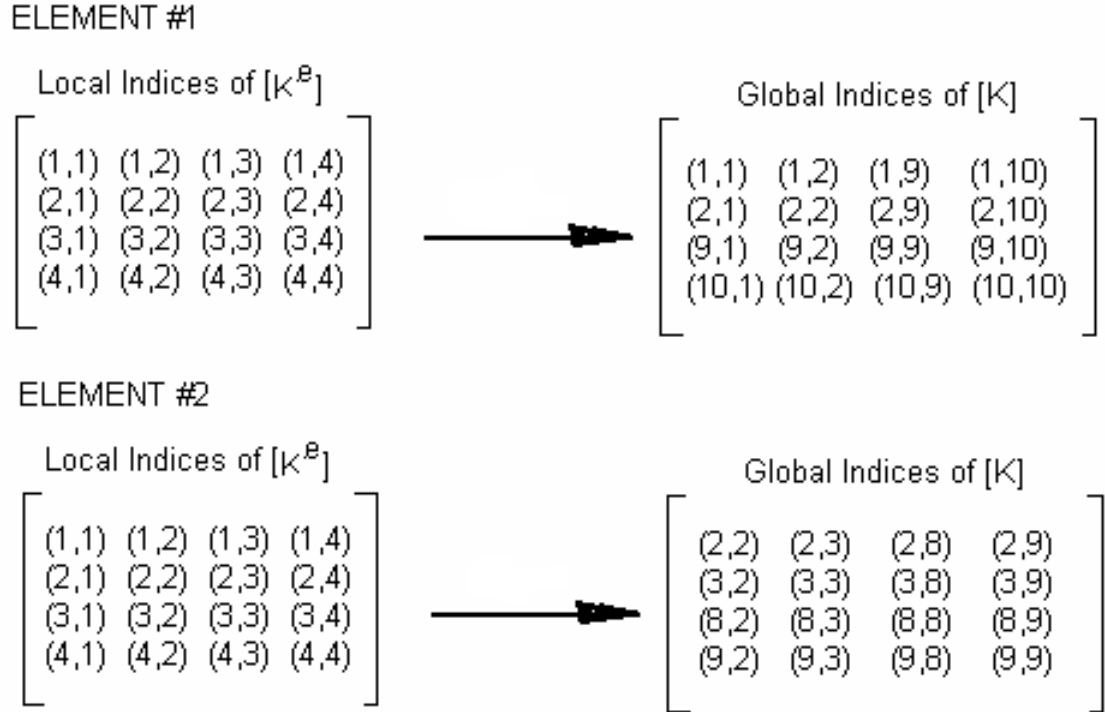
$$\mathbf{B} = \begin{pmatrix} 1 & 2 & 9 & 10 \\ 2 & 3 & 8 & 9 \\ 3 & 4 & 7 & 8 \\ 4 & 5 & 6 & 7 \\ 7 & 6 & 15 & 14 \\ 8 & 7 & 14 & 13 \\ 9 & 8 & 13 & 12 \\ 10 & 9 & 12 & 11 \end{pmatrix}$$

Pomocí matice \mathbf{B} se prvky elementární matice tuhosti a elementárního vektoru náhodného zatížení zobrazí na prvky matice tuhosti a vektoru náhodného zatížení. Například pro první dva prvky sítě se elementární matice tuhosti zobrazí na prvky matice tuhosti, viz obr. 4.19.

Matice tuhosti je složena z jednotlivých elementárních matic a je typu $N \times N$ (N je počet uzlů). Tím jsme aproximovali Poissonovu rovnici na soustavu rovnic

$$KV = F. \quad (4.83)$$

Úloha musí splňovat Dirichletovu okrajovou podmínku, proto při řešení soustavy rovnic (4.83) vynecháme ty sloupce a řádky, které odpovídají uzlům s předepsanou podmínkou.



Obrázek 4.19: Přejchod z K^e na K

4.5.7. Úloha matematického programování

Úlohu (4.5) řešenou pomocí metody konečných prvku je poměrně obtížné implementovat v GAMSu, proto se omezíme pouze na výpočet EV problémů, který je pro výpočet jednodušší. Dále pro jednoduchost uvažujeme následující účelovou funkci

$$z^{\text{EV}} = \min_e (F(E(\xi), e)) = \min_e \sum_{i=1}^N (V_i - U_i)^2, \quad (4.84)$$

kde V_i (respektive U_i) je průhyb membrány (respektive požadovaný průhyb) v i -tém uzlu sítě. Postupem uvedeným v kapitole 4.5 aproximujeme Poissonovu rovnici (4.5). Dostaneme tak následující úlohu matematického programování.

EV úloha matematického programování:

$$z^{\text{EV}} = \min_e (F(E(\xi), e)) = \min_e \sum_{i=1}^N (V_i - U_i)^2 \quad (4.85)$$

$$\mathbf{KV} = \mathbf{F} \quad (4.86)$$

$$V_i = 0, \quad \forall V_i \in \Gamma \quad (4.87)$$

4.5.8. Výsledky

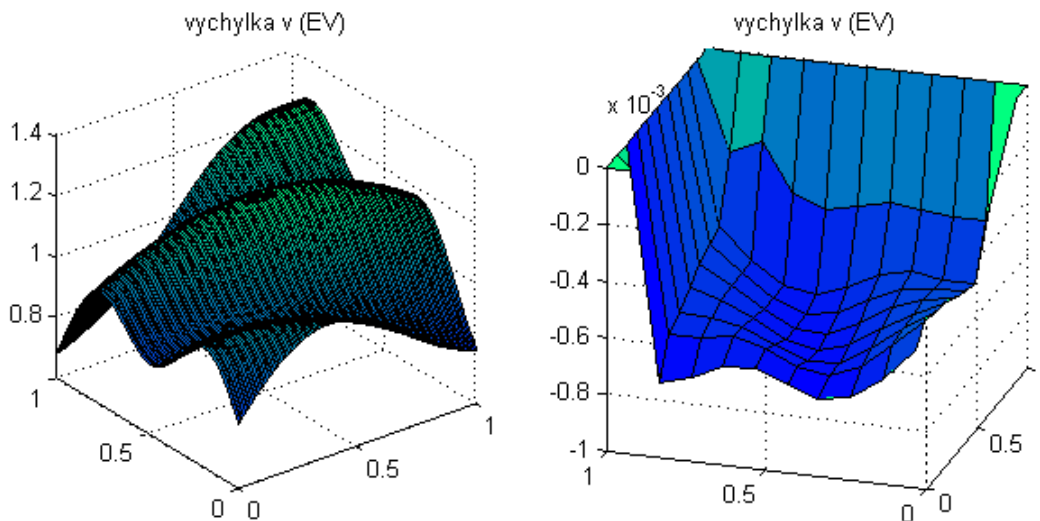
EV úloha (4.85)–(4.87) byla implementována do GAMSu se stejným zadáním jako v případě metody sítí, viz 4.3.4. Čtvercová oblast byla rozdělena na 100 čtverců, což znamená hledat řešení V_i v 121 uzlových bodů. Na jednotlivých elementárních oblastech bylo definované rovnoměrné dělení s krokem $h = 0,01$ potřebné pro numerický výpočet prvků vektoru vnějšího zatížení s využitím složené Simpsonovy formule (4.12).

4.5. APROXIMACE ÚLOHY: METODA KONEČNÝCH PRVKŮ

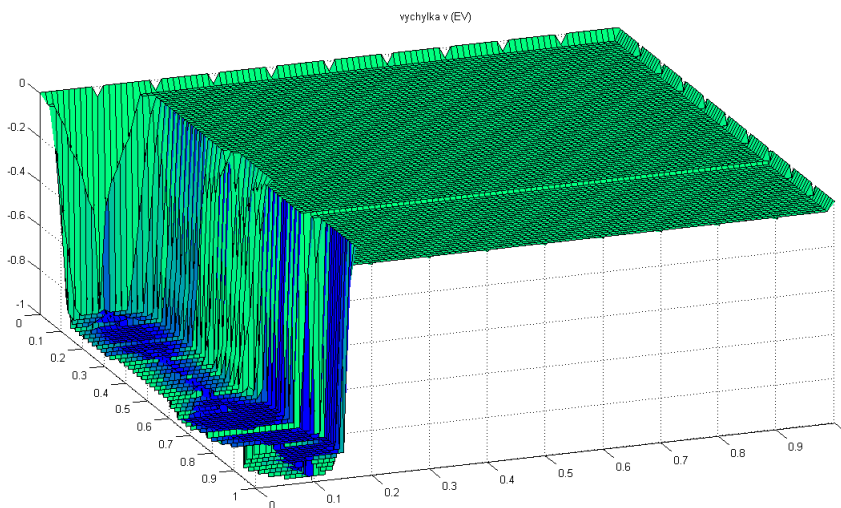
Bylo dosaženo hodnoty účelové funkce

$$z^{EV} = 3.185014 \cdot 10^{-5}.$$

Pro přesnější řešení problému optimalizace průhybu membrány je zapotřebí zavést komplexnější aproximaci. Jelikož se jedná a o prvotní implementaci metody konečných prvků do GAMSu dosažené výsledky nejsou věrohodné, viz následující obrázky průhybu membrány 4.20 a řízení membrány 4.21.



Obrázek 4.20: Průhyb a zatížení membrány



Obrázek 4.21: Řízení membrány

5. Závěr

V diplomové práci byla zformulována úloha spojité stochastické optimalizace s omezením ve tvaru parciálních diferenciálních rovnic popisující průhyb membrány. Náhodnost v problému byla reprezentována náhodným zatížením. Byly odvozeny matematické modely pomocí numerických metod, nejdříve metodou konečných diferencí (metoda sítí), poté metodou konečných prvků.

V případě metody sítí jsme formulovali EO, EV, EEV úlohy. Výsledky ukázaly nehladký průběh řízení, proto tyto úlohy byly doplněny navíc o omezení zohledňující průběh řízení. Bylo dosaženo přijatelných výsledků, které jsme posoudili pomocí hodnoty VSS určující vhodnost použití stochastické optimalizace a určením kvality řešení pomocí intervalového odhadu chyby aproximace. Stochastickým programováním jsme docílili lepších výsledků než deterministickým přístupem a dosáhli jsme dobré kvality řešení.

Dále jsme odvodili základní aproximaci omezení pomocí metody konečných prvků. Z hlediska náročnosti implementace a výpočtové náročnosti jsme odvodili pouze EV úlohu matematického programování. Prvotní implementací problému bylo dosaženo nevěrohodných výsledků. K dosažení lepších výsledků je potřeba složitějšího rozebrání aproximace metodou konečných prvků. Odvozený postup je nad rámec zadání diplomové práce a je vyzkoušen jako první krok výzkumu, který bude dál rozvíjen.

Literatura

- [1] Agbezuge L. Finite element solution of the poisson equation with dirichlet boundary conditions in a rectangular domain. p. 1–7
- [2] Ames, W. F. *Numerical Methods for Partial Diferential Equations (Computer Science and Scientific Computing)*. 2nd ed. San Francisco : Academic Press, 1977. 205. p
- [3] Čermák, L. *Numerické metody*. 1. vyd. Brno: CERM, 2005. 110 s. ISBN 80–214–3071–0
- [4] Čermák, L. *Algoritmy metody konečných prvků* . 1. vyd. Brno: PC–DIR, 2000. 106 s. ISBN 80–214–1697–1
- [5] Kall, P., Wallace, S.W. *Stochastic Programming*. 1st ed. New York: Wiley and Sons, 1994. 305 p.
- [6] Klapka, J., Dvořák, J., Popela, P. *Metody operačního výzkumu*. 2. vyd. Brno: VITIUM, 2001. 165 s. ISBN 80–214–1839–7
- [7] Mak, W.–K., Morton, D.P., Wood, R.K. Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters*, 1999, vol. 24, pp. 47–56.
- [8] Mathews, J. H., Fink, K. D. *Numerical Methods Using Matlab*. 3th ed. Upper Saddle River: Pearson Prentice Hall, 2004. 680 p. ISBN 0–13–065248–2
- [9] Popela, P. Stochastic programming. 2004, 72 p.
- [10] Žampachová, E. Aproximativní řešení úloh stochastické optimalizace s omezeními ve tvaru parciálních diferenciálních. *FSI Junior konference 2008*, 2008, s. 1–8

Seznam zkratek a symbolů

ξ	Náhodná veličina
ξ_s	Scénáře
(Ξ, Σ, P)	Pravděpodobnostní prostor
$v(x, y)$	Průhyb membrány
$u(x, y)$	Požadovaný průhyb membrány
$f(\xi, x, y)$	Náhodné vnější zatížení
$e(x, y)$	Regulační síla (konstantní řízení prvního stupně)
$\mathbf{T}(\xi, x, y)$	Matice popisující nepřímé působení $e(x, y)$ na membránu
\mathbf{K}^e	Elementární matice tuhosti
\mathbf{V}^e	Elementární vektor neznámých
\mathbf{F}^e	Elementární vektor zatížení
\mathbf{K}	Matice tuhosti
\mathbf{V}	Vektor neznámých
\mathbf{F}	Vektor zatížení

Přílohy

Obsah příloženého CD

Součástí diplomové práce je CD obsahující zdrojové kódy řešených úloh a získaná výsledná data. Podrobný přehled příložených souborů viz tabulka níže.

Obsah souboru, popř adresářů	Umístění, popř. název souboru
Text diplomové práce ve formátu pdf	text\diplomova prace.pdf
Zdrojový kód MKD	zdrojove kody-GAMS\MKD_nehladke.gms
Zdrojový kód MKD 2	zdrojove kody-GAMS\MKD_hladke.gms
Zdrojový kód MKP	zdrojove kody-GAMS\MKP_EV.gms
Zdrojový kód Monte Carlo	zdrojove kody-GAMS\MonteCarlo.gms
Data k MKD	vysledna data\metoda siti-nehladke\
Data k MKD 2	vysledna data\metoda siti-hladke\
Data k MKP	vysledna data\mkp\
Grafy výsledků MKD	vysledna data\metoda siti-nehladke\obr\
Grafy výsledků MKD 2	vysledna data\metoda siti-hladke\obr\
Grafy výsledků MKP	vysledna data\mkp\obr\