

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE  
FAKULTA CHEMICKÉJ A POTRAVINÁRSKEJ TECHNOLOGIE  
ÚSTAV INFORMATIZÁCIE, AUTOMATIZÁCIE A MATEMATIKY

**Software na automatickú viacnásobnú linearizáciu**

Bc. Alexander SZŮCS

DIPLOMOVÁ PRÁCA

2010

**SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE**  
**FAKULTA CHEMICKÉJ A POTRAVINÁRSKEJ TECHNOLOGIE**  
**ÚSTAV INFORMATIZÁCIE, AUTOMATIZÁCIE A MATEMATIKY**

**SOFTWARE NA AUTOMATICKÚ VIACNÁSOBNÚ  
LINEARIZÁCIU**

**Diplomová práca**

FCHPT-5414-26488

Študijný program: automatizácia a informatizácia v chémii a potravinárstve

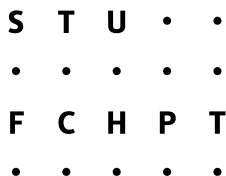
Číslo a názov študijného odboru: 5.2.14 automatizácia

Pracovisko (katedra/ústav): Ústav informatizácie, automatizácie a matematiky

Vedúci záverečnej práce/školiťel': Ing. Michal Kvasnica, Phd

**Bratislava 2010**

**Bc. Alexander Szűcs**



## ZADANIE DIPLOMOVEJ PRÁCE

Študent: **Bc. Alexander Szücs**  
ID študenta: 26488  
Študijný program: automatizácia a informatizácia v chémii a potravinárstve  
Študijný odbor: 5.2.14 automatizácia  
Vedúci práce: Ing. Michal Kvasnica, PhD.  
Miesto vypracovania: FCHPT STU v Bratislave

Názov práce: **Software na automatickú viacnásobnú linearizáciu**

Špecifikácia zadania:

Hybridné systémy predstavujú pomerne nový prístup k riadeniu fyzikálnych systémov pomocou digitálnych algoritmov. Ich princíp spočíva v spojení dvoch prístupov k modelovaniu: fyzikálna stránka procesu je opísaná diferenciálnymi alebo diferenčnými rovnicami, zatiaľ čo kód riadiaceho algoritmu využíva prepínacie pravidlá v tvare IF-THEN-ELSE podmienok. Hybridné systémy našli široké uplatnenie všade tam, kde sa kladie dôraz na úplnosť a správnosť návrhu riadiacich algoritmov, ako je napr. riadenie leteckej premávky, riadenie automobilov prostredníctvom adaptívnych tempomatov, regulácia napäťových prenosových sústav, či návrh komplexných riadiacich systémov pre systémy chemického a petrochemického priemyslu. Práca sa zameria na tvorbu matematických modelov hybridných systémov pomocou techniky viacnásobnej linearizácie. Cieľom práce je vytvorenie kompletného softwarového balíka, ktorý prevedie zadaný nelineárny matematický opis na hybridný systém tak, že sa nelinearity aproximujú pomocou viacerých, vhodne zvolených, lokálnych linearizácií.

Rozsah práce: 40

Zoznam odbornej literatúry:

1. WILLIAMS, P. *Model building in mathematical programming*. New York: John Wiley & Sons, 1999. 354 s. ISBN 0-471-99788-9.
2. TORRISI, F. HYSDEL. [online]. 2006. URL: <http://control.ee.ethz.ch/~hybrid/hysdel/>.

Riešenie zadania práce od: 15. 02. 2010

Dátum odovzdania práce: 22. 05. 2010

**L. S.**

**Bc. Alexander Szücs**

Študent

**prof. Ing. Miroslav Fikar, DrSc.**

Vedúci pracoviska

**prof. Ing. Miroslav Fikar, DrSc.**

Garant študijného programu

## **Pod'akovanie**

Rád by som sa poďakoval Ing. Michalovi Kvasnicovi, Phd za odborné vedenie a rady pri realizácii diplomovej práce a v neposlednom rade za ochotu pri riešení akejkoľvek problematiky nad rámec tejto práce. Taktiež by som sa chcel poďakovať rodine za morálnu i hmotnú podporu počas štúdia.

---

## **Abstrakt**

Cieľom práce bola tvorba softwarového balíka slúžiaceho na aproximáciu  $n$  – rozmerných funkčných závislostí. Balík by mal byť vhodným nástrojom pre analýzu dynamických vlastností hybridných systémov. Hybridné modely predstavujú takú formu matematického opisu, kde namiesto jedného operačného bodu sa uvažuje viacero aproximačných bodov, čím sa značne zvýši efektivita linearizácie. Takáto forma modelovania našlo uplatnenie všade tam, kde sa kladie dôraz na presnosť matematických modelov opisujúcich dynamické vlastnosti reálnych zariadení. Výstupom softwaru je hybridný model ľubovlného procesu (napr. prietokový chemický reaktor, autonómny robotický systém) v exportovateľnej forme do MPT toolboxu (Multi parametric toolbox). Tieto modely budú slúžiť na syntézu prediktívnych regulátorov založených na nových optimalizačných prístupoch.

---

## **Abstrakt**

The goal of the diploma thesis was to create a software package for approximation of  $n$ -th dimensional functional dependences. This package should be an efficient tool for analysis the dynamical behavior of hybrid systems. Hybrid models represent such a kind of mathematical description, where instead of one operation point are considered more approximation points, thus is significantly increased the efficiency of the linearization. Such a form of modeling finds application everywhere, where emphasis is placed on exactness of mathematical models, describing dynamical properties of real equipments. Output from the software is a hybrid model of an arbitrary process (e.g. continuous stirred – tank reactor or autonomic robotic system) in exportable form to MPT toolbox. (Multi parametric toolbox) These models will serve for synthesis of predictive controllers based on new optimizations approaches.

---

# OBSAH

<b>ZOZNAM SYMBOLOV A SKRATIEK.....</b>	<b>7</b>
<b>ÚVOD.....</b>	<b>8</b>
<b>1. TAYLOROV ROZVOJ.....</b>	<b>9</b>
1.1 APROXIMÁCIA NELINEÁRNYCH FUNKCIÍ .....	10
1.2 DAVIDONOVA METÓDA KUBICKEJ INTERPOLÁCIE .....	12
<b>2. CHARAKTERISTIKA POUŽÍVANÝCH FUNKCIÍ.....</b>	<b>13</b>
2.1 UKÁŽKA PRE JEDNOROZMERNÚ FUNKCIU .....	19
2.2 PRÍKLADY 1D FUNKCIÍ .....	20
2.3 UKÁŽKA PRE DVOJROZMERNÚ FUNKCIU .....	23
2.4 PRÍKLADY 2D FUNKCIÍ .....	25
2.5 UKÁŽKA PRE N – ROZMERNÚ FUNKCIU .....	37
2.6 PRÍKLAD N – ROZMERNEJ FUNKCIE .....	38
<b>3. ZÁVER.....</b>	<b>45</b>
<b>ZOZNAM POUŽITEJ LITERATÚRY .....</b>	<b>46</b>



## ZOZNAM SYMBOLOV A SKRATIEK

$f(x)$  – funkcia 1 reálnej premennej

$\frac{df}{dx}$  – derivácia funkcie podľa premennej  $x$

$f(x_1, x_2)$  – funkcia dvoch reálnych premenných

$f'(b)$  – funkčná hodnota derivácie v bode  $b$

$f'(a)$  – funkčná hodnota derivácie v bode  $a$

$x_{\text{zac}}$  – dolná hranica intervalu

$x_{\text{kon}}$  – horná hranica intervalu

$a$  – začiatok intervalu

$b$  – koniec intervalu

$f(x_1, x_2, \dots, x_n)$  – funkcia  $n$  – reálnych premenných

$x$  – nový odhad extrému funkcie

$f'(x)$  – funkčná hodnota v bode  $x$

$f(b)$  – funkčná hodnota v bode  $b$

$f(a)$  – funkčná hodnota v bode  $a$

$x_{\text{min}}$  – dolná hranica intervalu aproximovanej funkcie

$x_{\text{max}}$  – horná hranica intervalu aproximovanej funkcie

1D – jednorozmerná funkcia

2D – dvojrozmerná funkcia

$D(f)$  – definičný obor funkcie

## Úvod

Väčšina chemicko-technologických procesov má nelineárny charakter, čo z matematického hľadiska znamená, že ich dynamické vlastnosti sú opísané pomocou nelineárnych diferenciálnych rovníc. Keďže rozmanitosť nelineárnych funkcií je obrovská, doteraz odborníci zaoberajúci sa teóriou riadenia nenašli všeobecný opis resp. obecné riešenie na danú problematiku. Preto tie nelineárne diferenciálne rovnice sa snažíme pretransformovať na lineárne pomocou odchýlkových veličín. Voľbou vhodného operačného bodu, ktorú v drvivej väčšine predstavuje rovnovážna hodnota stavovej (riadenej) veličiny je možné vyriešiť veľkú škálu nelineárnych problémov. Matematický nástroj slúžiaci na transformáciu je práve linearizácia – Taylorovým rozvojom funkcie do nekonečného radu, kde sa uvažuje len lineárny člen a zvyšok radu sa zanedbáva. Problém tohto prístupu je v tom, že so zväčšujúcou sa vzdialenosťou od operačného bodu sa odchýlky medzi trajektóriami nelineárneho a linearizovaného modelu sa zväčšujú. Tento nedostatok sa odstraňuje metódou hybridného modelovania, ktorej princíp spočíva vo voľbe viacerých pracovných bodov, čím sa dosiahnu minimálne odchýlky medzi výstupnými trajektóriami. Toolbox by mal byť vhodnou pomôckou pri voľbe týchto linearizačných bodov, kde sa tieto body určia podľa vopred stanovených pravidiel a nie heuristicky.

# 1. TAYLOROV ROZVOJ

Taylorov rozvoj[1] ľubovolnej funkcie do nekonečného funkcionálneho radu so zanedbaním vyšších členov, okrem lineárneho, je veľmi využívaným matematickým nástrojom v oblasti teórie riadenia, kde slúži na aproximáciu nelineárnych funkcií s lineárnymi. Ak máme všeobecnú funkciu  $n$  – premenných, tak hovoríme, že funkcia je lineárna vzhľadom na niektorú zo svojich premenných, keď po derivácii podľa nej dostaneme iba konštantu. Napríklad  $f(x) = 4x - 7$  je lineárna, lebo po vykonaní derivácie získame len konštantu. Takisto funkcia  $f(x, y) = 2x + 6y^2$  je lineárna vzhľadom na  $x$ , ale je nelineárna vzhľadom na  $y$ . Ako príklady na ďalšie nelineárne funkcie sa dajú uviesť logaritmické, goniometrické alebo po častiach lineárne funkčné závislosti.

Všeobecný tvar rozvoja pre funkciu jednej reálnej premennej  $f(x)$  má nasledovný tvar:

$$f(x) \Big|_{x=x^s} \approx f(x^s) + \frac{df}{dx}(x - x^s)$$

a pre funkciu viac premenných  $f(x, y)$ ,

$$f(x, y) \Big|_{(x=x^s, y=y^s)} \approx f(x^s, y^s) + \frac{\partial f(x, y)}{\partial x}(x - x^s) + \frac{\partial f(x, y)}{\partial y}(y - y^s)$$

kde  $x^s$  a  $y^s$  predstavujú rovnovážne hodnoty stavových veličín, keď sa jedná o tvorbu linearizovaného matematického modelu procesu. V mojom prípade hore uvedené premenné predstavujú linearizačné body, ktoré sú určené podľa určitých pravidiel s rôznymi aproximačnými technikami. Tieto body slúžia na zostrojenie priamok pomocou Taylorovho rozvoja, ktorých priesečníky vymedzujú jednotlivé regióny, pričom každá z nich je spojená vlastným zákonom dynamiky. Keď by som predošlé tvrdenia chcel aplikovať na chemicko-technologický proces, tak čo sa dosiahne práve technikou viacnásobnej linearizácie je to, že získame  $n$  lineárnych dynamík a v závislosti od implementačného prostredia (Hysdel, Matlab) sa dokážeme medzi dynamikami prepínať napr. pomocou logických premenných a získať presnejšie priebehy výstupných veličín.

## 1.1 APROXIMÁCIA NELINEÁRNYCH FUNKCIÍ

Toolbox slúži na linearizáciu  $n$  – rozmerných funkcií  $f(x_1, x_2, \dots, x_n)$ , pričom musia byť rešpektované určité podmienky. Funkcia musí byť spojitá na celom definičnom obore, čo matematicky znamená,  $D(f) = R$  (Definičný obor funkcie tvorí množina všetkých reálnych čísiel). Je teda dovoľená linearizácia polynomických, niektorých goniometrických alebo exponenciálnych funkčných závislostí. Na druhej strane nie je, alebo je len čiastočne možná aproximácia hyperbolických, lineárne lomených funkčných predpisov, kde intervaly sú obmedzené bodmi nespojitosti alebo rozpätím definičného oboru. To v praxi znamená, že aproximácia funkcií typu  $(\arccos(x), \arcsin(x))$  je možná len na intervale  $\langle -1; 1 \rangle$  alebo racionálne lomené funkcie resp. goniometrické funkcie typu  $(tg(x), \cot g(x))$  sú aproximovateľné len na intervaloch, kde funkcia je spojitá.

V toolboxe používaný postup pri linearizácii 1D funkcií je nasledovný:

- Hľadanie pozície linearizačných bodov
- Nájdenie priesečníkov priamok, získaných Taylorovým rozvojom
- Vyhodnotenie linearizácií
- Výpočet sumy štvorcov odchýlok

Zoznam funkcií slúžiacich na aproximáciu jednorozmerných funkčných závislostí:

- `urc_pozicie2`
- `najdi_linearizacie`
- `vyhodnot_linearizacie`
- `vyhodnot_stvorce`
- `greedy_linearization`
- `eval_linearization`
- `obtain_the_intervals`
- `cubic_interpolation_min`
- `cubic_interpolation_max`

Existujú 2 prístupy aproximácie 2D funkcií. Prvá tzv. priama metóda využíva sekvenciu nasledovných bodov:

- Analýza stacionárnych bodov a inflexií
- Taylorov rozvoj nelineárnej funkcie v okolí aproximačného bodu
- Hľadanie priesečníkov dotykových plôch

Z týchto podmienok Taylorov rozvoj je realizovateľný, ale čo sa týka rozmiestnenia linearizačných bodov a hľadania regiónov problém je už obtiažnejší, totiž priesečníky dotykových rovín sú priamky a pokiaľ oni nie sú rovnobežné neexistuje exaktný výraz na výpočet ich vzdialenosti. Oproti predošlej metóde druhá, nepriama metóda transformuje aproximáciu 2D funkcie na linearizáciu dvoch kvadratických (parabolických) funkčných závislostí a tým odpadnú všetky ťažkosti vyplývajúce z aproximácie 2D funkcií[3].

Príklad: Uvažujme funkciu v tvare  $f(x_1, x_2) = x_1 x_2$ , túto funkciu dokážeme vyjadriť v tvare:

$$x_1 x_2 \approx \frac{1}{4} u_1^2 - \frac{1}{4} u_2^2, \text{ kde } u_1 = x_1 + x_2 \quad \text{a} \quad u_2 = x_1 - x_2.$$

Pomocou dvoch pomocných premenných sme pôvodne veľmi zložitý problém pretransformovali na dve separátne kvadratické funkcie. Po dosadení obidvoch premenných do pôvodného vzorca a po roznásobení člena nachádzajúceho sa na pravej strane výrazu pôvodného vzorca dostaneme výraz, ktorý sa nachádza na ľavej strane.

$$\frac{1}{4}(x_1 + x_2)^2 - \frac{1}{4}(x_1 - x_2)^2 = \frac{1}{4}x_1^2 + \frac{1}{2}x_1 x_2 + \frac{1}{4}x_2^2 - \frac{1}{4}x_1^2 + \frac{1}{2}x_1 x_2 - \frac{1}{4}x_2^2 = \frac{1}{2}x_1 x_2 + \frac{1}{2}x_1 x_2 = x_1 x_2$$

Funkcia slúžiaca na aproximáciu dvojrozmerných funkčných závislostí:

- `aproximuj_x1x2b`

Pri aproximácii funkcií  $\frac{1}{4}u_1^2$  a  $\frac{1}{4}u_2^2$  sa vykoná rovnaká procedúra, ktorá je uvedená pri postupe linearizácii 1D funkcií.

Pri aproximácii  $n$  – rozmerných funkcií napr.  $f(x_1, x_2, x_3) = x_1 \sin(x_2) \exp(x_3)$  sa postupuje tak, že najprv sa aproximuje prvá časť funkcie  $x_1 \sin(x_2)$  a potom druhá časť funkcie  $\exp(x_3)$ , kde  $\exp$  už predstavuje aproximáciu predošlého súčinu. Takýmto spôsobom sa dajú linearizovať funkcie  $n$  – premenných pomocou nepriamej metódy používanej pri 2D funkciách.

Zoznam funkcií slúžiacich na aproximáciu jednorozmerných funkčných závislostí:

- main\_aproximation
- mpt\_allcombs
- combinatoric\_number
- aproximuj\_x1x2graphs

Ako pomocný nástroj sa používa symbolický toolbox Matlab-u, ktorý umožňuje operácie matematickej analýzy v symbolickom tvare bez použitia zložitých algoritmov zaužívaných v numerickej matematike.

## 1.2 DAVIDONOVA METÓDA KUBICKEJ INTERPOLÁCIE

Na intervale  $\langle a, b \rangle$  nahradíme funkciu  $f(x)$  jej kubickou interpoláciou.

Interpolačný polynóm má tvar:

$$F(x) = A(x-c)^3 + B(x-c)^2 + C(x-c) + D, \text{ kde } c = \frac{(a+b)}{2}$$

Rovnica predstavuje interpolačný polynóm 3.stupňa pomocou stredu intervalu, ktorým je bod  $c$ . V rovnici sú štyri neznáme:  $A, B, C, D$ , ktoré určíme zo znalosti hodnôt funkcie a jej derivácie v hraničných bodoch intervalu  $\langle a, b \rangle$ .

Výsledné vzťahy pre Davidonovu metódu[2]:

pomocné premenné:

$$z = f'(b) + f'(a) - 3[f(b) - f(a)]/(b-a)$$

$$w = \sqrt{z^2 - f'(b)f'(a)}$$

Výsledný vzťah pre aproximáciu extrému na intervale:

$$x = \frac{a + z - f'(a) \pm w}{f'(b) - f'(a) \pm 2w} (b-a) \quad + \rightarrow \min \quad - \rightarrow \max$$

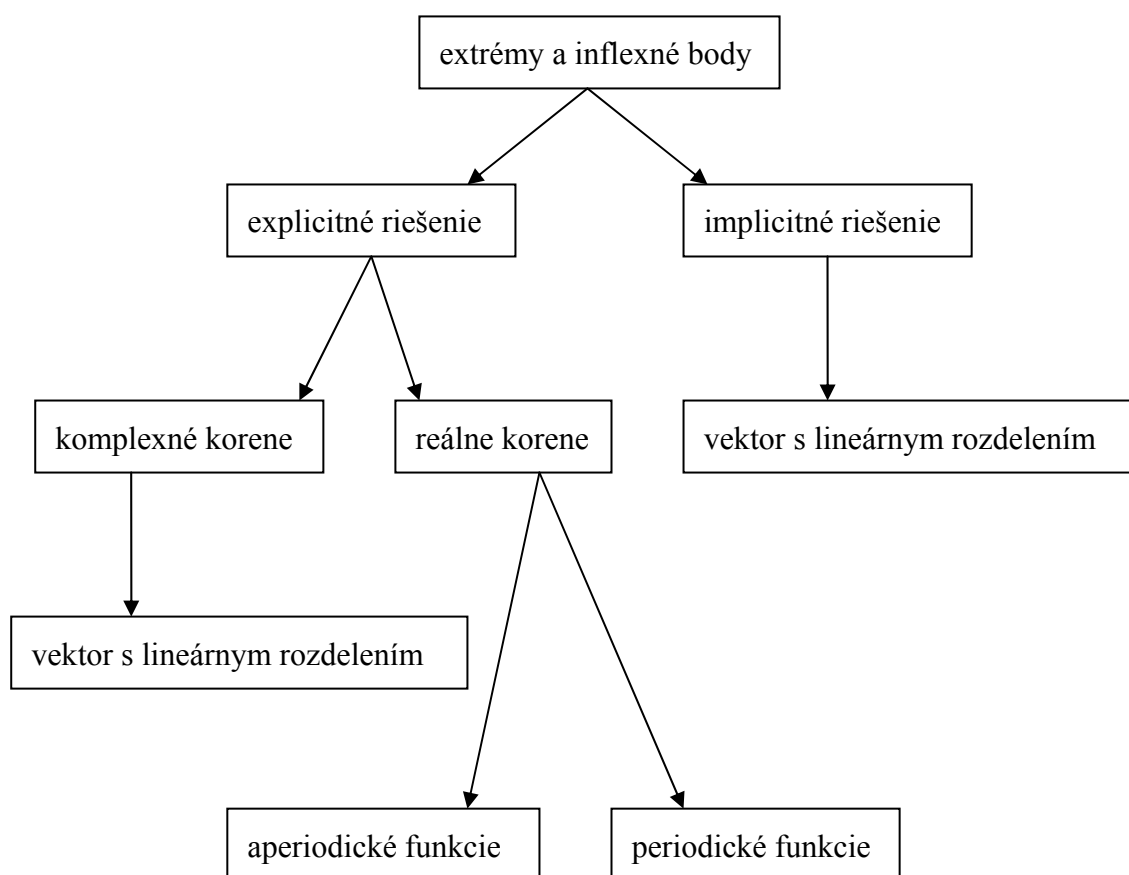
Hľadanie ďalších bodov sa uskutočňuje pomocou hore uvedených vzorcov, len na zúženom intervale.

$$\langle a, x \rangle \quad \text{ak } f'(x) > 0$$

$$\langle x, b \rangle \quad \text{ak } f'(x) < 0$$

## 2. CHARAKTERISTIKA POUŽÍVANÝCH FUNKCIÍ

1.URC\_POZICIE2: táto časť programu vráti pozície linearizačných bodov. Stratégia hľadania spočíva v nájdení prvej a druhej derivácie funkcie, a podľa ich existencie sa vyvíja stromová štruktúra hľadania pozícií ďalších bodov. Pokiaľ neexistuje priame (explicitné) riešenie tak rozmiestnenie aproximačných bodov bude lineárne, teda sa vytvorí vektor s lineárnym rozdelením. V prípade jestvujúcej (jestvujúcich) derivácií, teda stacionárnych a inflexných bodov problém pokračuje ošetrovaním typov týchto bodov. Keď sa jedná o komplexné korene, tak ako v predošlom prípade sa vytvorí vektor s lineárnym rozdelením v opačnom prípade sa vyberie stredný prvok vektora a v závislosti od krajných bodov daného intervalu, na ktorom pracujeme, pole aproximačných bodov sa doplní na príslušný počet od seba rovnako vzdialených bodov. Všetky hore uvedené fakty sa týkali aperiodických funkcií. Pri periodických funkciách pozície lin.bodov sa určujú pomocou Davidonovej metódy kubickej interpolácie.



Stromová štruktúra určenia linearizačných bodov

2.OBTAIN\_THE\_INTERVALS: funkcia slúži na vytvorenie sekvencie konvexných a konkávných funkcií. Táto požiadavka sa zabezpečí sledovaním znamienok funkčných hodnôt derivácie aproximovanej funkcie. Základná požiadavka teda je, aby funkcia na danom intervale bola unimodálna, čo znamená, že na danom intervale sa nachádza iba jediný extrém. Po určení hraníc intervalov, kde funkcie sú konvexné alebo konkávne, sa vykoná jednorozmerná optimalizácia metódou kubickej interpolácie, kde optimum (extrém) sa nájde vždy s nejakou požadovanou presnosťou.

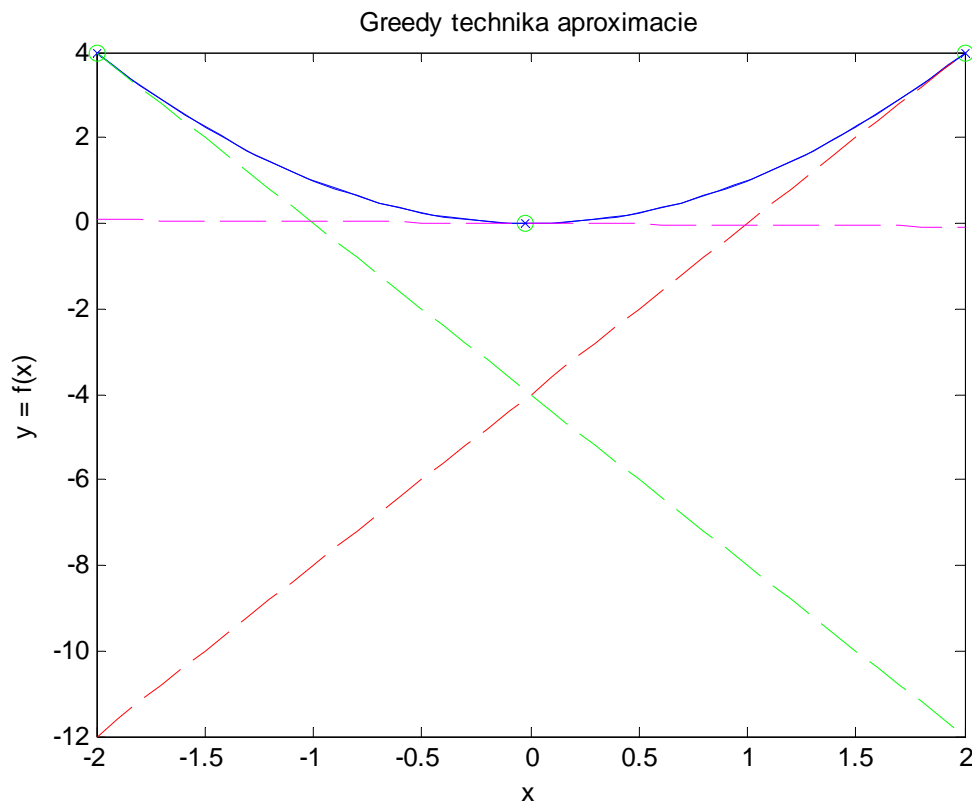
3.CUBIC\_INTERPOLATION\_MAX a CUBIC\_INTERPOLATION\_MIN: funkcie slúžia na nájdenie extrémov resp. inflexných bodov linearizovanej funkcie, metódou Davidonovej kubickej interpolácie. Používa sa hlavne kvôli rýchlej konvergencii k extrému, rýchlosť je porovnateľná s Newtonovou metódou dotyčníc. Postup pri nájdení extrémov je opísaný v sekcii: Davidonova metóda kubickej interpolácie.

4.GREEDY\_LINEARIZATION a EVAL\_LINEARIZATION: predstavuje druhú linearizačnú techniku, pomocou ktorej sa dá určiť rozmiestnenie aproximačných bodov. Volá sa greedy alebo “pažravá” technika, lebo napodobňuje techniku rozmiestnenia robenú človekom. Celá filozofia aproximácie spočíva v tom, že interval sa rozdelí pomocou 100 bodov na subintervaly vytvárajúc akúsi mriežku a jednotlivé body budú predstavovať „kandidátov“ pre ďalší linearizačný bod. Vo všetkých bodoch sa vyhodnotí rozdiel medzi pôvodnou a aproximovanou funkciou, a do výsledného vektora chýb sa uložia minimálne vzdialenosti, z ktorých sa znovu vyberie tá maximálna, lebo pomocou tohto bodu sa zmenší celková chyba aproximácie v najväčšej miere .



Grafické znázornenie “greedy“ techniky linearizácie:

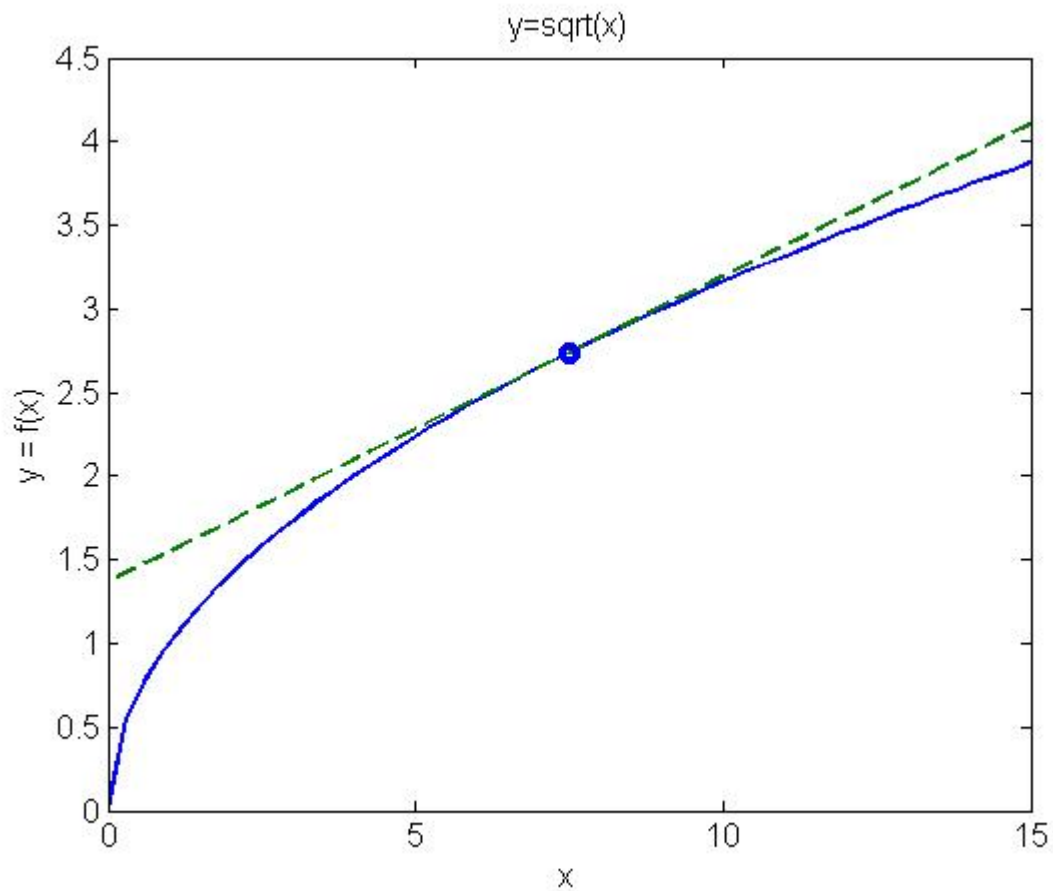
Príklad: uvažujme funkciu v tvare  $f(x) = x^2$  na intervale  $[-2, 2]$ ; Predpokladajme, že už poznáme pozíciu dvoch linearizačných bodov, a máme aj dotyčnice vedené cez nich (červená a zelená priamka).



Po získaní funkčných hodnôt v bodoch danej mriežky pôvodnej funkcie a medzi jednotlivými aproximáciami je evidentné, že v maximálnej miere znížime chybu aproximácie vtedy, keď ďalší bod umiestnime približne v strede intervalu (ružová priamka).

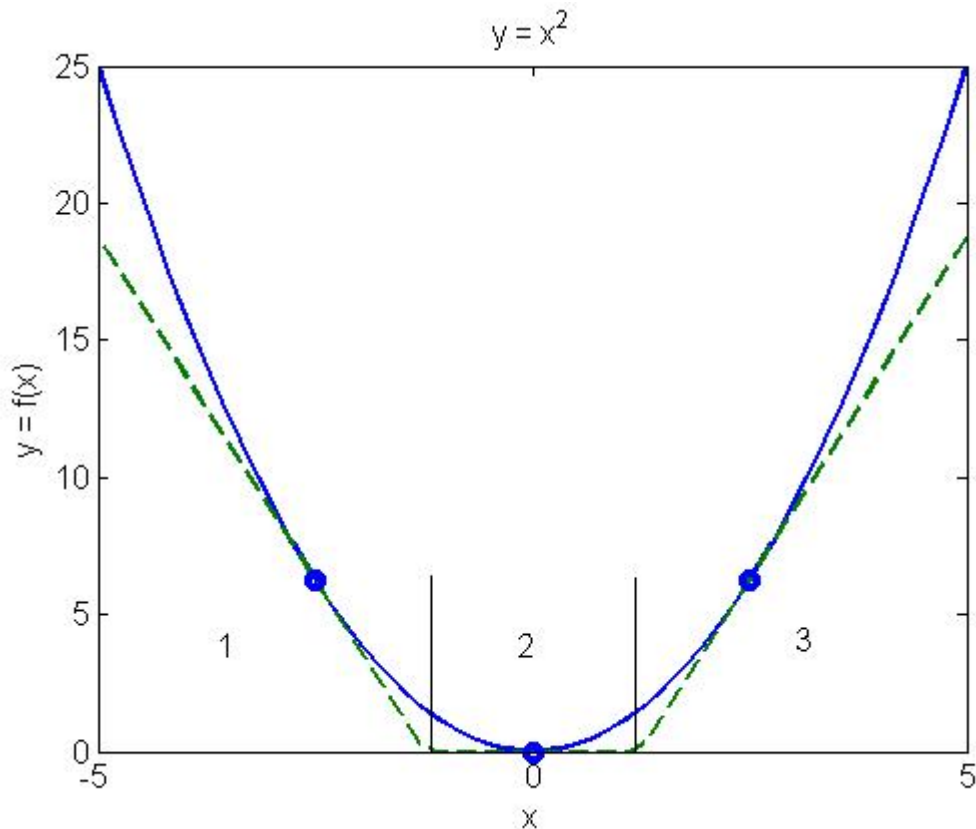
5.NAJDI\_LINEARIZÁCIE: užívateľ po zadaní želanom počte bodov a hraníc, ktoré vymedzujú oblasť linearizácie, funkcia sa aproximuje Taylorovým rozvojom. Je to aproximácia pri ktorej sa všetky ostatné členy (zvyšok nekonečného radu) zanedbajú a teda uvažuje sa lineárna závislosť. Dá sa ukázať, že ľubovoľná polynomiálna aproximácia rovnakého rádu dá horšie výsledky ako spomínaný Taylorov rozvoj.

Grafická reprezentácia aproximácie:



Podľa toho, koľko bodov zadá užívateľ, presne toľko aproximácií vykoná táto funkcia a nájde aj ich priesečníky, čo si môžeme predstaviť ako rovnosť dvoch priamok v smernicovom tvare:  $y_1 = a_1x + b_1$   $y_2 = a_2x + b_2$  kde  $a_1$  a  $a_2$  sú smernicami jednotlivých dotyčníc a  $b_1$  a  $b_2$  sú tzv. afinné členy. Afinné členy hovoria iba o tom, že o koľko je posunutá funkčná hodnota na y-ovej osi, keď argument sa rovná nule. Teda hľadáme taký bod  $x$  v ktorom aj prvá aj druhá funkcia nadobúda rovnakú funkčnú hodnotu, musí platiť:  $y_1 = y_2$ . Práve tieto spoločné body vymedzujú jednotlivé regióny.

Príklad : Keď užívateľ zadá 3 lin.body, zostroja sa tri dotyčnice, 3 regióny a v prípade piatich bodov to bude 5 dotyčníc, 5 regiónov. Prípad 3 lin.body môžeme vidieť na obrázku č.2



Obr. 2 Aproximácia paraboly v troch bodoch

6.VYHODNOT\_STVORCE : táto funkcia vypočíta, resp. vyhodnotí sumu štvorcov odchýlok medzi pôvodnou a linearizovanou funkciou. Má úlohu verifikácie, teda nám poskytne informáciu o tom, že do akej miery je naša aproximácia presná. Treba si však dať pozor, lebo jedna číselná (štatistická) charakteristika môže byť pre dve rôzne funkcie aj vyhovujúca ale aj zlá. To je z toho dôvodu, že pri verifikácii sa počítajú kvadráty odchýlok a aj veľmi malá odchýlka medzi funkciami môže zvýšiť hodnotu ukazovateľa, hlavne keď sa jedná o exponenciálnu alebo o polynomiálnu funkčnú závislosť s vysokým mocninovým koeficientom. Funkcia: VYHODNOT\_LINEARIZACIE preto nemá samostatný opis, lebo je volaná z funkcie: VYHODNOT\_ŠTVORCE. Získanie počtu a takisto nájdenie príslušného regiónu sa uskutočňuje v tejto funkcii.

7.APROXIMUJ\_X1X2: slúži na aproximáciu dvojrozmerných funkčných závislostí v tvare súčinu  $f(x_1)f(x_2)$ , pričom ten súčin sa aproximuje v tvare dvoch separátnych kvadratických funkčných závislostí.

8.MAIN\_APPROXIMATION: táto funkcia separuje vstupnú funkciu na subfunkcie podľa znamienok + a – a samozrejme podľa rozmeru. Keď funkcia je lineárna alebo konštantná, aproximácia sa nevykoná.

9.MPT\_ALLCOMBS: má podobnú funkciu ako príkaz meshgrid pri tvorbe argumentov pre dvojrozmernú funkciu. Program vypočíta argumenty pre n – rozmernú funkciu pre zadaný počet bodov v závislosti od rozmeru danej funkčnej závislosti. Maximálny možný rozmer je 10, v prípade potreby sa dá hornú hranicu rozmeru funkcie rozšíriť.

10.APROXIMUJ\_X1X2graphs a COMBINATORIC\_NUMBER: Keď rozmer funkcie nadobúda hodnotu väčšiu ako 2, danú funkciu nedokážeme vykresliť kvôli neexistujúcemu matematickému modelu, preto prvá funkcia vykreslí príslušný počet grafov, ktoré predstavujú rezy viacrozmerných funkcií, teda vždy niektoré premenné sú ponechané na nulovej hodnote. Užívateľ tým pádom má aj možnosť vizualizácie okrem číselných charakteristík (suma štvorcov odchýlok, maximálna chyba aproximácie). Druhá funkcia slúži na získanie počtu funkcií. Obecné povedané my potrebujeme určiť, že z n – rozmernej funkcie koľko 2D funkcií dokážeme zostrojiť. Matematicky to znamená, že z n prvkovej množiny potrebujeme vytvoriť k – prvkové množiny. Počet možných rezov sa dá získať pomocou kombinačného čísla:  $\frac{n}{k} = \frac{n!}{(n-k)!k!}$

Príklad: Nech je funkcia  $f(x_1, x_2, x_3, x_4) = x_1 \sin(x_2) \cos(x_3) x_4 \rightarrow n = 4$  a  $k = 2$

Po dosadení do vyššie uvedeného vzorca dostaneme, že počet všetkých možných rezov je 6.

## 2.1 UKÁŽKA PRE JEDNOROZMERNÚ FUNKCIU

Celá sekvencia hore popísaných funkcií je uložená do jedného skriptu s názvom „linearizuj“.

Pred volaním funkcie je nutné si zadefinovať symbolickú premennú príkazom *syms x*

V Matlabe sa teda zavolá funkcia príkazom :

```
>> [V,S] = linearizuj(f,x,xzac,xkon,ns)
```

, kde

$f, x, x_{zac}, x_{kon}, ns$  sú vstupné argumenty:

- $f$  : nelineárna funkcia
- $x_{zac}$  : začiatok intervalu na ktorom sa uvažuje aproximácia
- $x_{kon}$  : koniec intervalu na ktorom sa uvažuje aproximácia
- $ns$  : počet linearizačných bodov

$V, S$  sú návratovými hodnotami funkcie *linearizuj* :

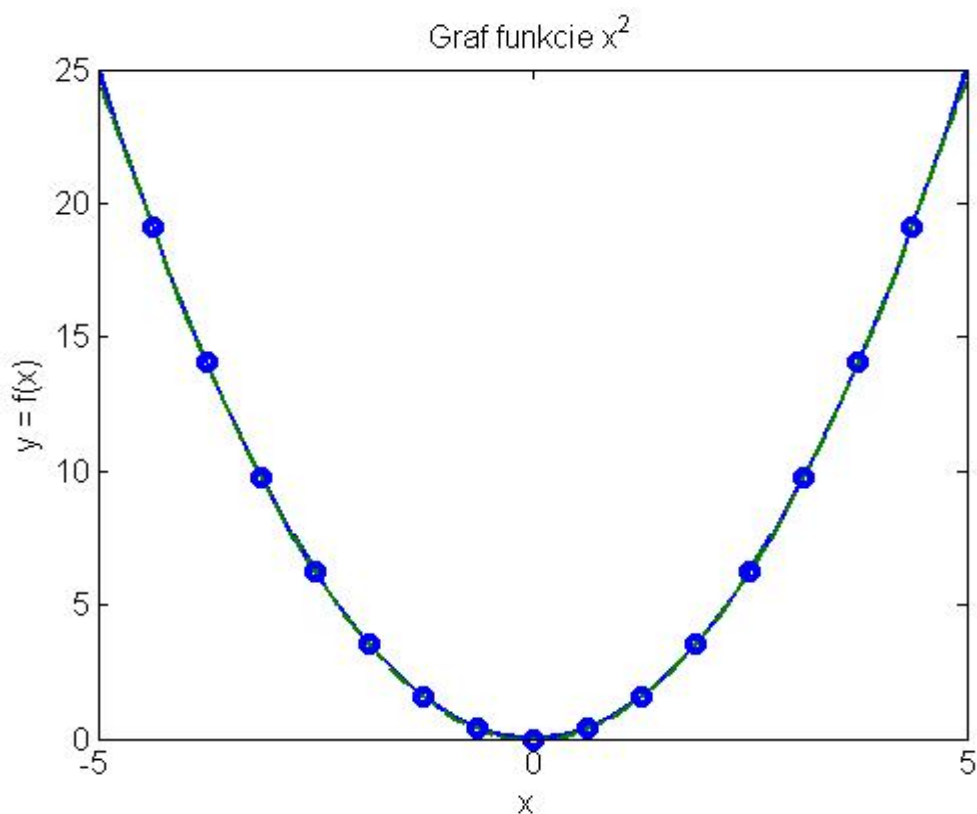
- $V$  je štruktúra so štyrmi položkami
  - 1.položka :  $V.xs$  – vektor linearizačných bodov
  - 2.položka :  $V.hranice\_regionov$  – pole obsahujúce hranice jednotlivých aproximovaných sekcií
  - 3.položka :  $V.a$  – vektor, ktorý obsahuje smernice dotyčníc, získaných z Taylorovho rozvoja
  - 4.položka :  $V.b$  – afinné členy v predošlom bode spomenutých priamok
- Premenná  $S$  je číselnou charakteristikou našej aproximácie, ktorá nám zároveň povie o úspešnosti linearizácie. Aproximácia je vyhodnotená na základe sumy štvorcov odchýlok.

## 2.2 PRÍKLADY 1D FUNKCIÍ

>>  $[V, S] = \text{linearizuj}(f, x, x_{\text{zac}}, x_{\text{kon}}, ns)$  ,kde

- $f = x^2$
- $x_{\text{zac}} = -5$
- $x_{\text{kon}} = 5$
- $ns = 15$

Po zadenovaní symbolickej premennej  $x$  a po zavolaní funkcie `linearizuj` by sme mali vidieť nasledovný graf s príslušnými číselnými hodnotami.



$V =$

$xs: [1 \times 15 \text{ double}]$

$hranice\_regionov: [1 \times 16 \text{ double}]$

$a: [1 \times 15 \text{ double}]$

$b: [1 \times 15 \text{ double}]$

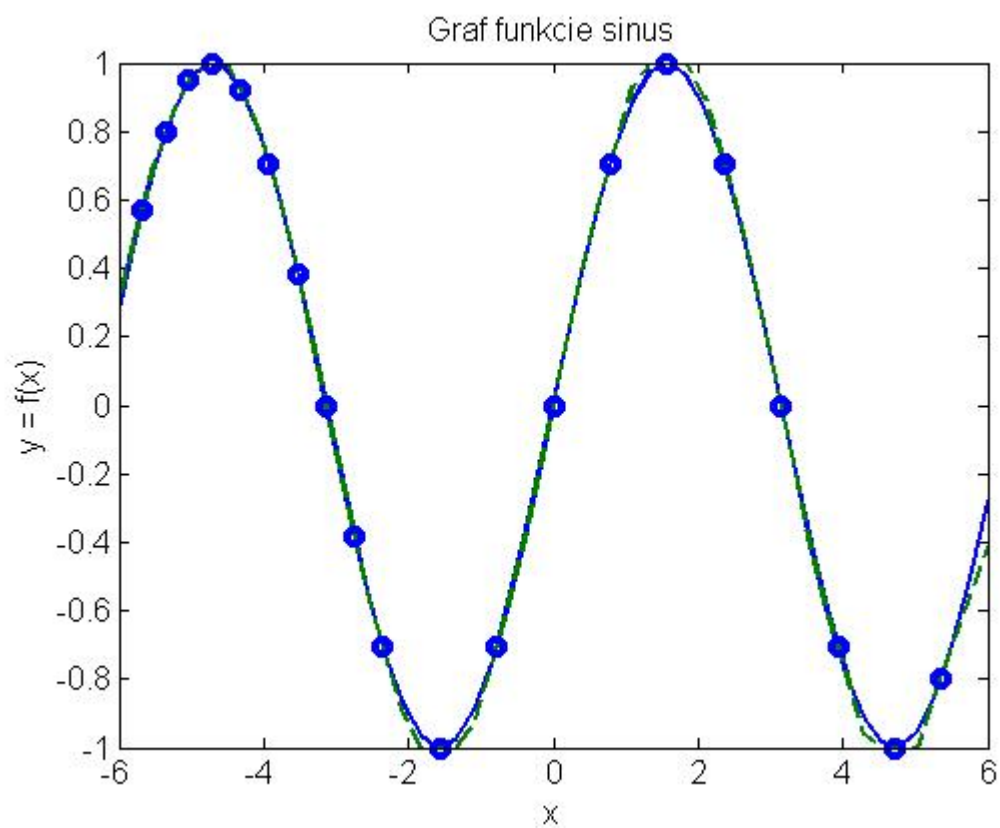
$S =$  0.4578

K ľubovolnej položke štruktúry  $V$  môžeme pristúpiť cez bodkový operátor, napr:  $V.a$

>> [V,S] = linearizuj(f,x,xzac,xkon,ns) ,kde

- $f = \sin(x)$
- $x_{zac} = -6$
- $x_{kon} = 6$
- $ns = 20$

Po tej istej procedúre popísanej v predošlom príklade by sme mali dostať takýto výsledok.



V =

xs: [1x20 double]

hranice\_regionov: [1x21 double]

a: [1x20 double]

b: [1x20 double]

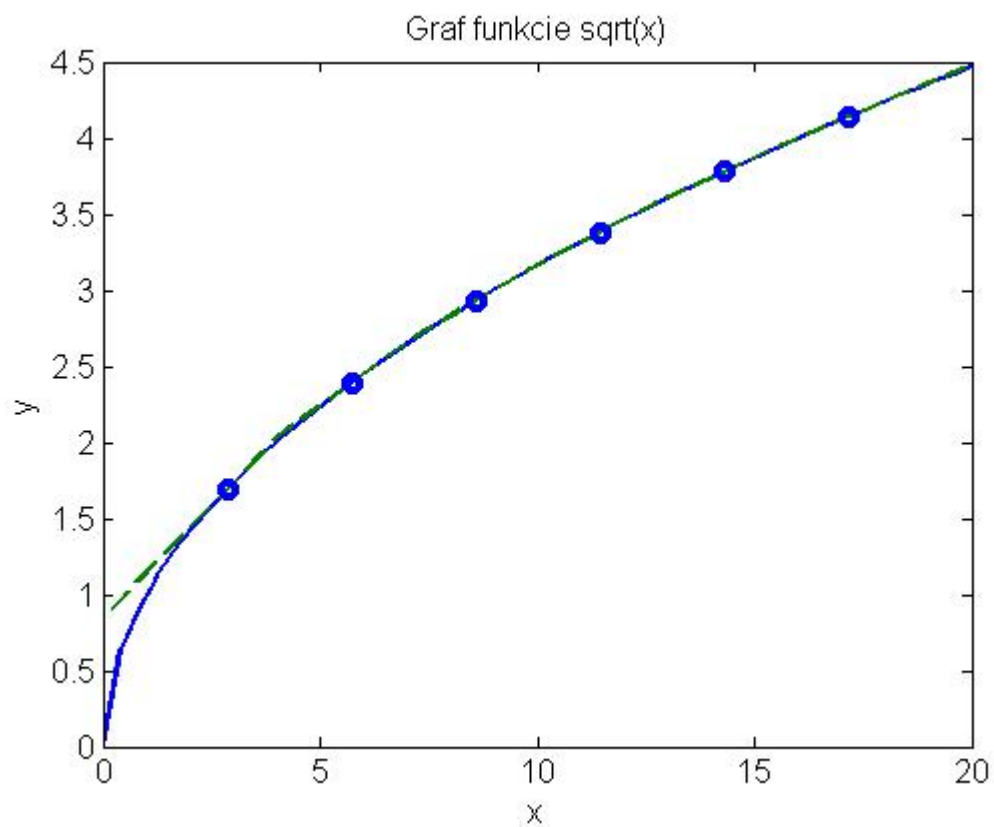
S =

0.0365

K ľubovolnej položke štruktúry V môžeme prístupit' cez bodkový operátor, napr: V.xs

>> [V,S] = linearizuj(f,x,xzac,xkon,ns) ,kde

- $f = \text{sqrt}(x)$
- $x_{\text{zac}} = 0$
- $x_{\text{kon}} = 20$
- $ns = 6$



V =

xs: [2.8571 5.7143 8.5714 11.4286 14.2857 17.1429]

hranice\_regionov: [0 4.0406 6.9985 9.8974 12.7775 15.6492 20]

a: [0.2958 0.2092 0.1708 0.1479 0.1323 0.1208]

b: [0.8452 1.1952 1.4639 1.6903 1.8898 2.0702]

S = 0.8700

Na grafe modrou čiarou je vykreslená pôvodná nelineárna funkcia a zelenou aproximovaná a jednotlivé lin. body sú vyznačené modrým krúžkom.



## 2.3 UKÁŽKA PRE DVOJROZMERNÚ FUNKCIU

Pokiaľ chceme aproximovať dvojrozmernú funkciu, potrebujeme na to funkciu `aproximuj_x1x2`, ktorá ju aproximuje v tvare  $af(x_1)f(x_2)$ , kde  $a$  je ľubovoľné reálne číslo.

Pred volaním funkcie je nutné si zadať symbolické premenné príkazom `syms x1 x2`

Syntax v MATLABE vyzerá nasledovne:

```
>> aproximuj_x1x2(f1, f2, x1, x2, x1min, x1max, x2min, x2max, u1nlin, u2nlin, a)
```

, kde

`f1, f2, x1, x2, x1min, x1max, x2min, x2max, u1nlin, u2nlin, a` sú vstupné argumenty:

- `f(x1), f(x2)` : aproximované funkcie
- `x1, x2` : symbolické premenné
- `x1min, x1max` – hranice funkcie `f(x1)`
- `x2min, x2max` – hranice funkcie `f(x2)`
- `u1nlin` – počet linearizačných bodov pre funkciu  $u_1^2$
- `u2nlin` – počet linearizačných bodov pre funkciu  $u_2^2$
- `a` – ľubovoľný násobok pôvodnej nelineárnej funkcie, nepovinný argument

`APROXIMUJ_X1X2` nemá žiadne návratové hodnoty, ale poskytne informáciu o sume štvorcov odchýlok a o maximálnej chybe aproximácie a samozrejme pomocou vykreslených grafov sa dá posúdiť úspešnosť aproximácie aj vizuálne.

Výstupom tejto funkcie sú údaje o aproximácii v príkazovom okne v podobe napr:

Suma štvorcov odchýlok : 0.000565

Maximálna chyba aproximácie : 0.000009

a tri grafy, na ktorých sú vykreslené nasledovné funkčné závislosti :

- pôvodná nelineárna funkcia
- aproximovaná funkcia
- chyba aproximácie

Pokiaľ sa medzi vstupnými argumentmi nevyskytuje parameter  $a$ , ktorý je nepovinný, tak má prednastavenú (defaultnú) hodnotu rovnú jednej. To znamená, že teraz sa aproximuje funkcia v tvare  $f(x_1)f(x_2)$ .

V príkazovom okne Matlabu syntax vyzerá nasledovne:

```
>> aproximuj_x1x2(f1, f2, x1, x2, x1min, x1max, x2min, x2max, u1nlin, u2nlin)
```

Aproximácia danej nelineárnej funkcie sa dá opísať nasledovným postupom:

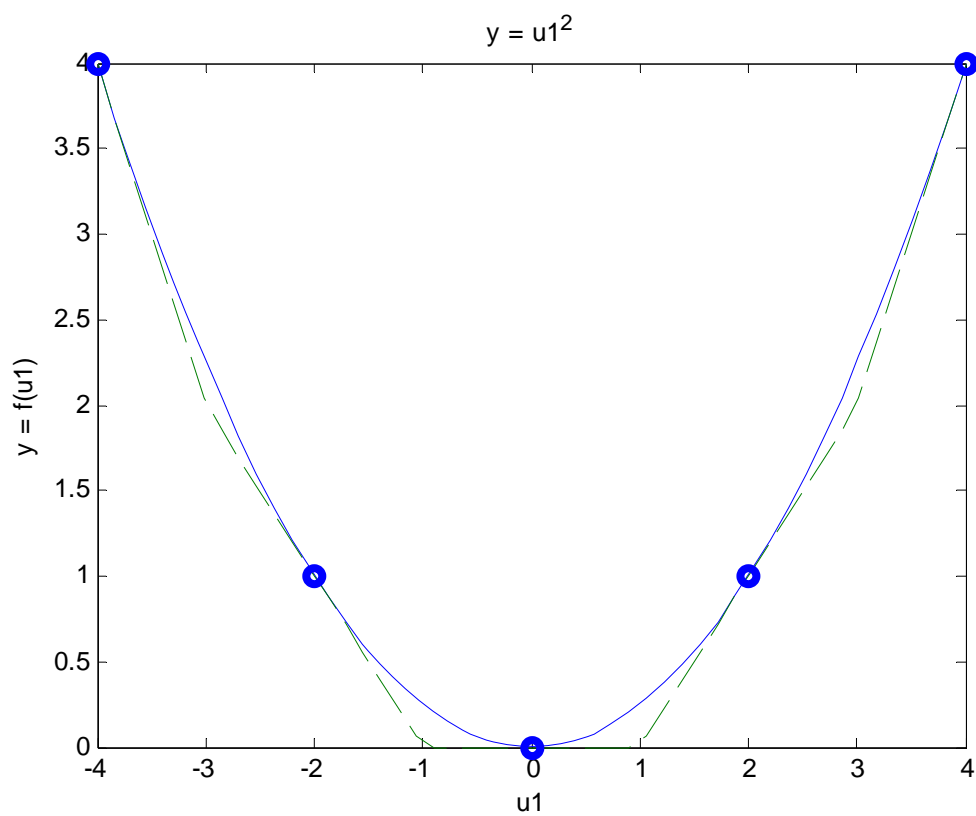
- určenie hraníc pre linearizáciu – výpočet maximálnych a minimálnych funkčných hodnôt
- zavolanie funkcie LINARIZUJ
- výpočet funkčných hodnôt pôvodnej funkcie
- vyhodnotenie aproximácie – výpočet funkčných hodnôt linearizovanej funkcie
- vyhodnotenie sumy štvorcov odchýlok

## 2.4 PŘÍKLADY 2D FUNKCÍ

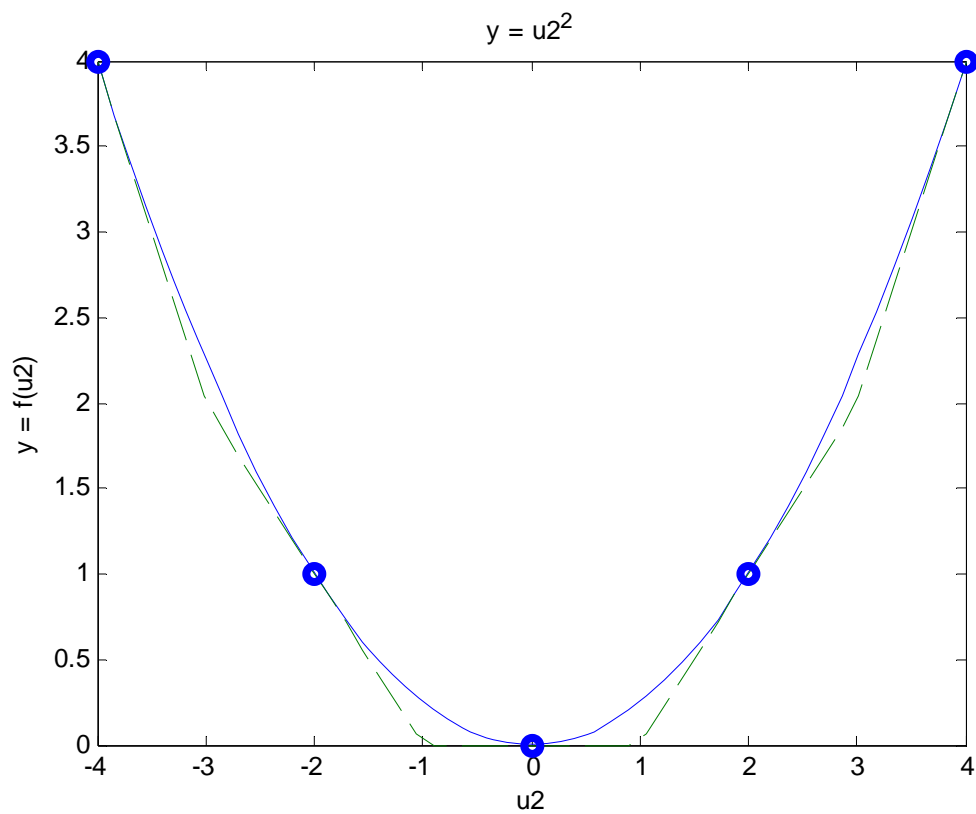
>> aproximuj\_x1x2(f1, f2, x1, x2, x1min, x1max, x2min, x2max, u1nlin, u2nlin)

- syms x1 x2
- f1 = x1
- f2 = x2
- x1min = -2
- x2min = 2
- x2mix = -2
- x2max = 2
- u1nlin = 5
- u2nlin = 5

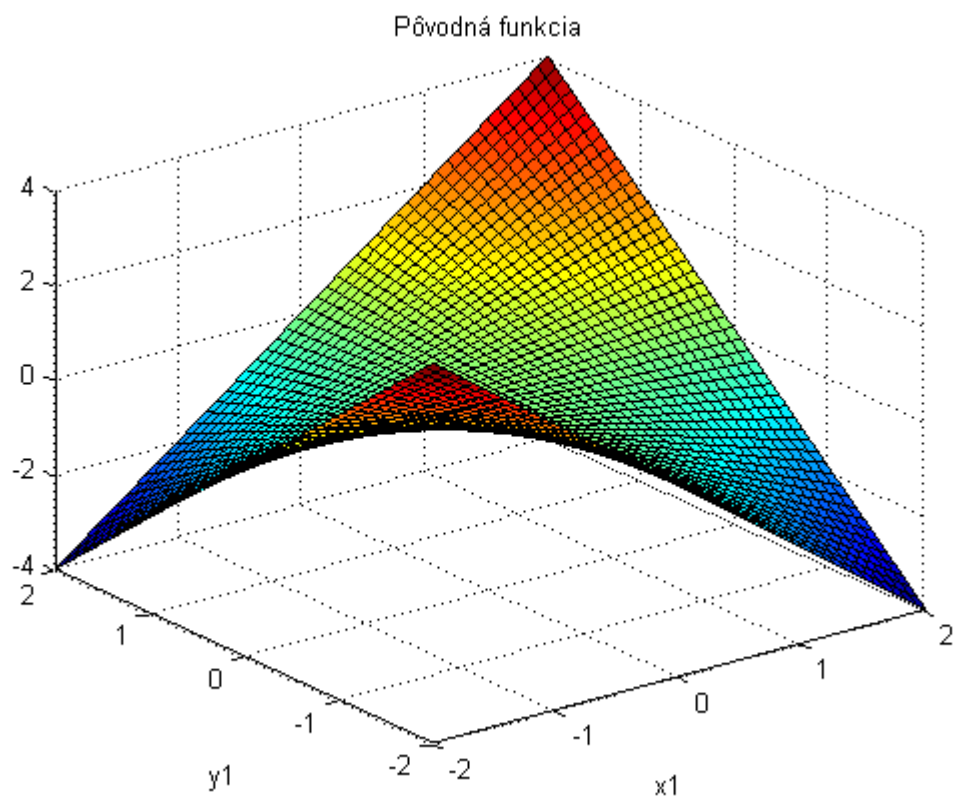
1. graf: Prvá linearizovaná parabola  $y = u_1^2$



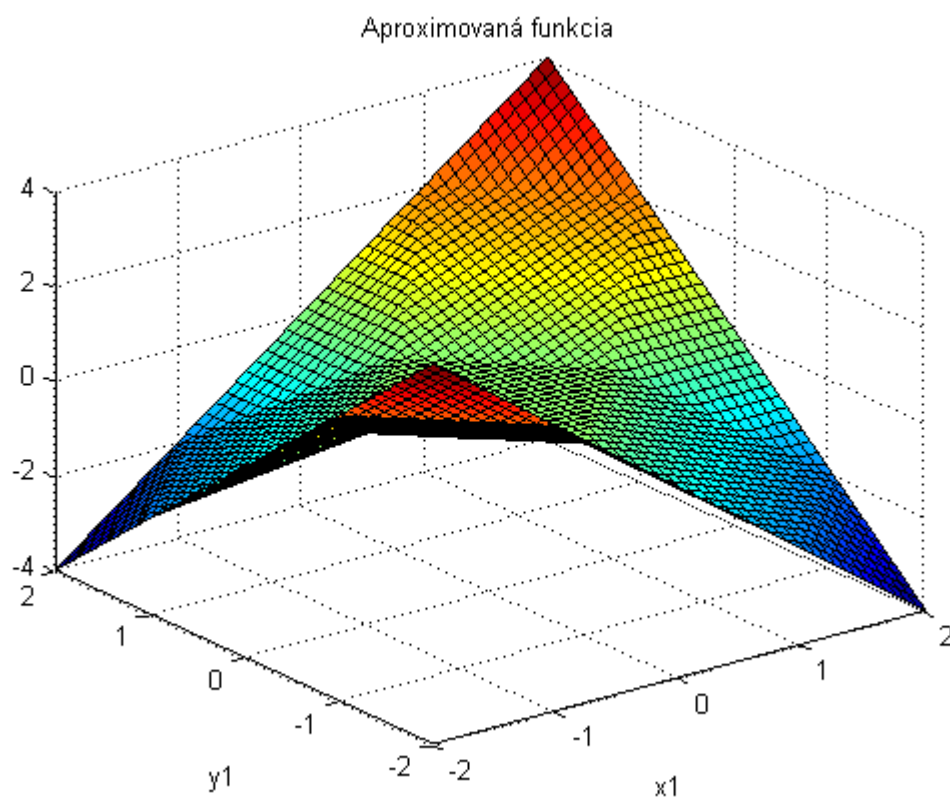
2. graf : Druhá aproximovaná kvadratická závislosť  $y = u_2^2$



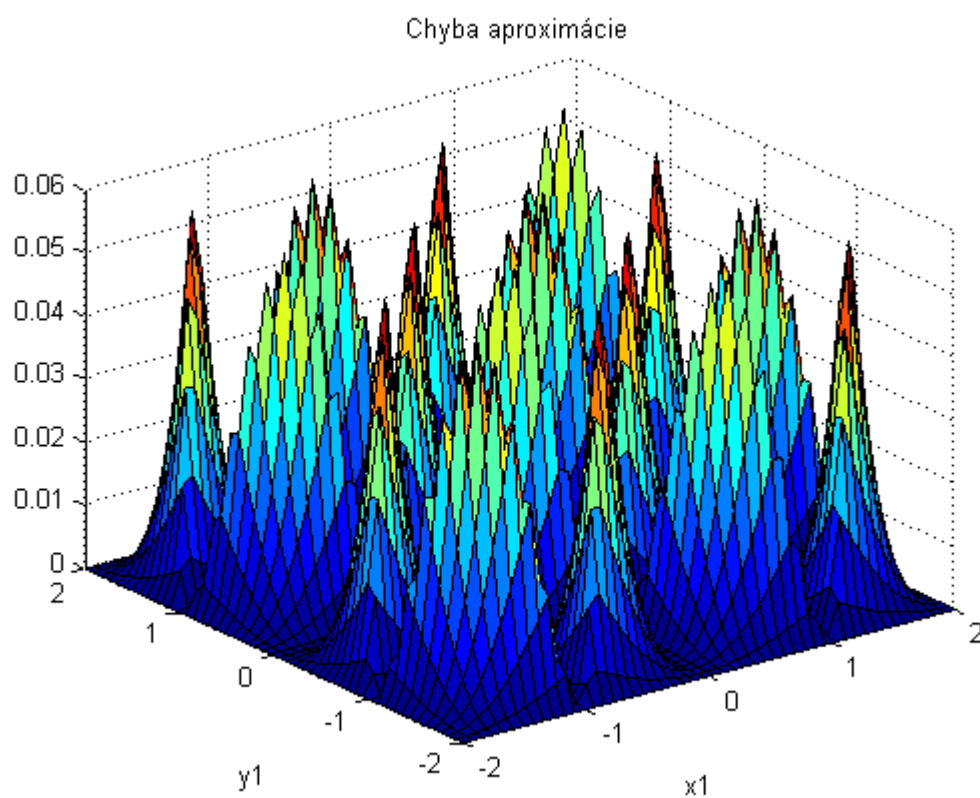
3. graf : Pôvodná nelineárna funkcia na užívateľom zadanom intervale



4. graf : Aproximovaná funkcia pre zadaný rozsah vstupných argumentov



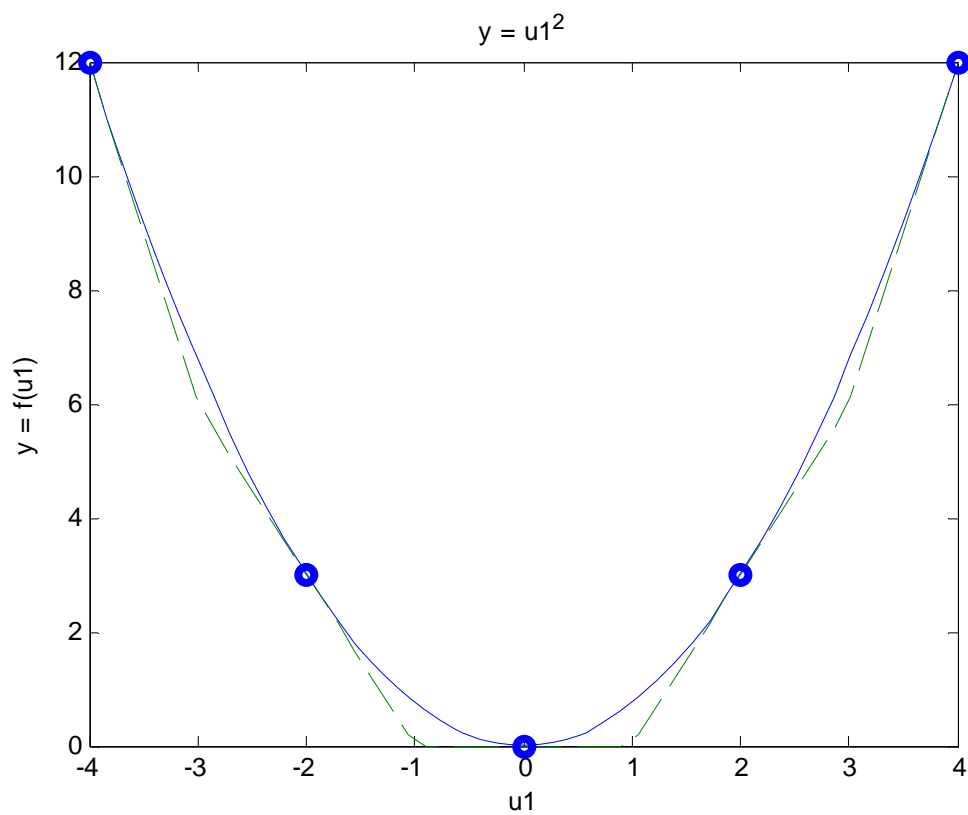
5. graf : Grafické znázornenie úspešnosti aproximácie



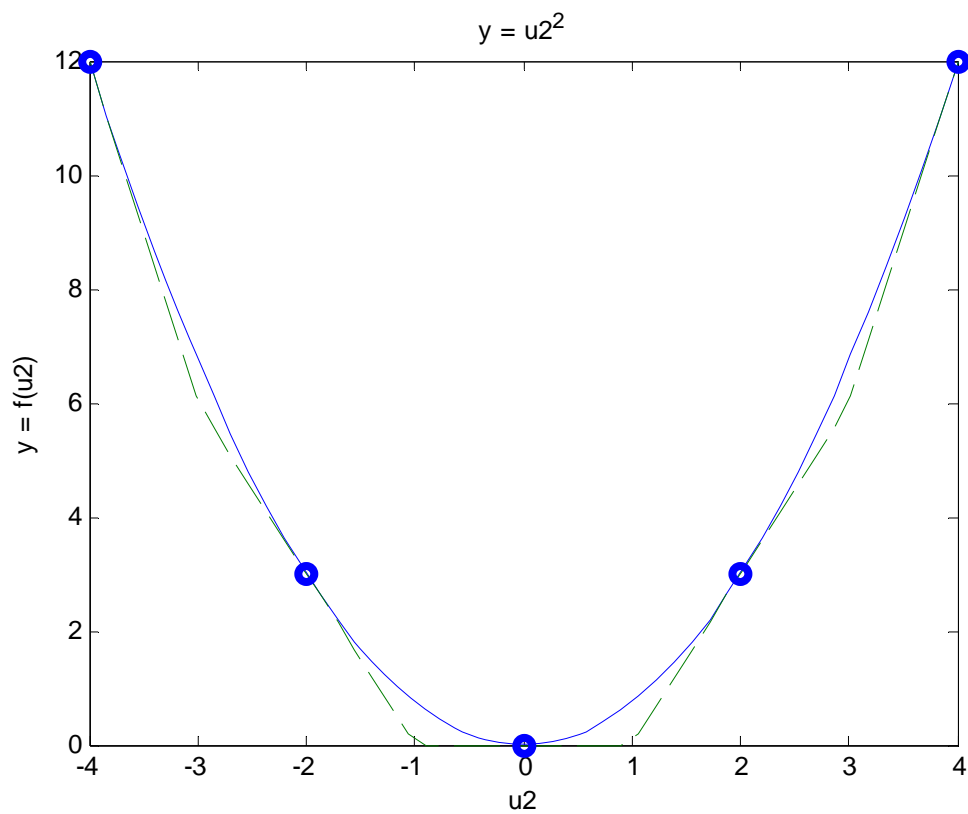
>> aproximuj\_x1x2(f1, f2, x1, x2, x1min, x1max, x2min, x2max, u1nlin, u2nlin,a)

- syms x1 x2
- f1 = x1
- f2 = x2
- x1min = -2
- x2min = 2
- x2mix = -2
- x2max = 2
- u1nlin = 5
- u2nlin = 5
- a = 3

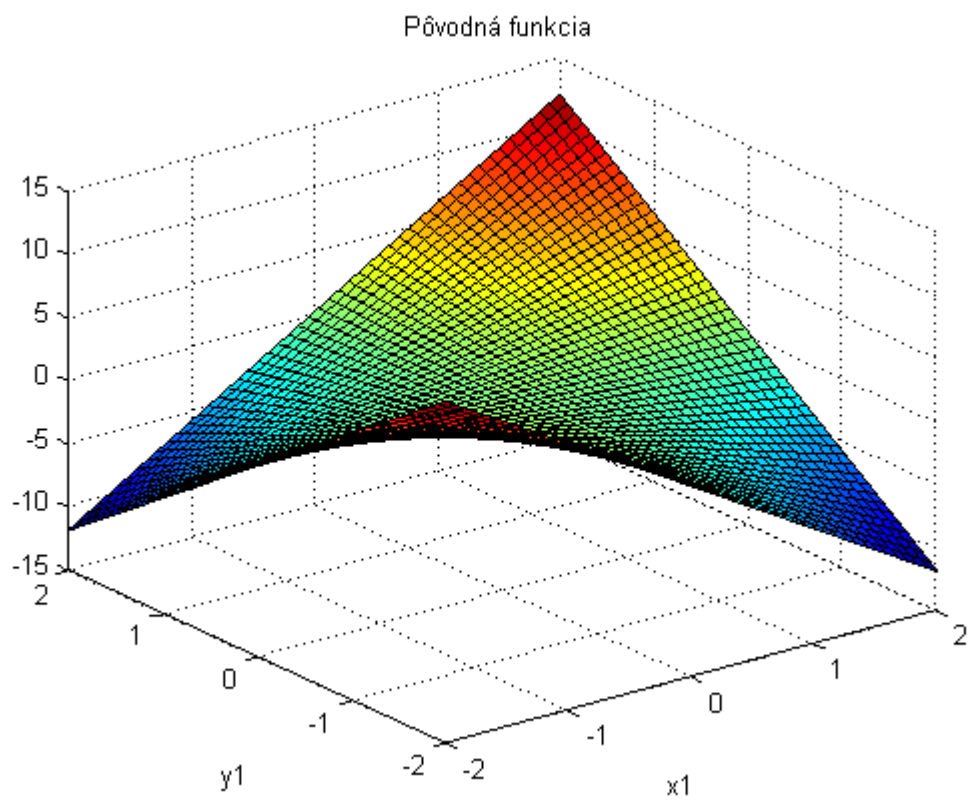
1. graf : Prvá linearizovaná parabola  $y = u_1^2$



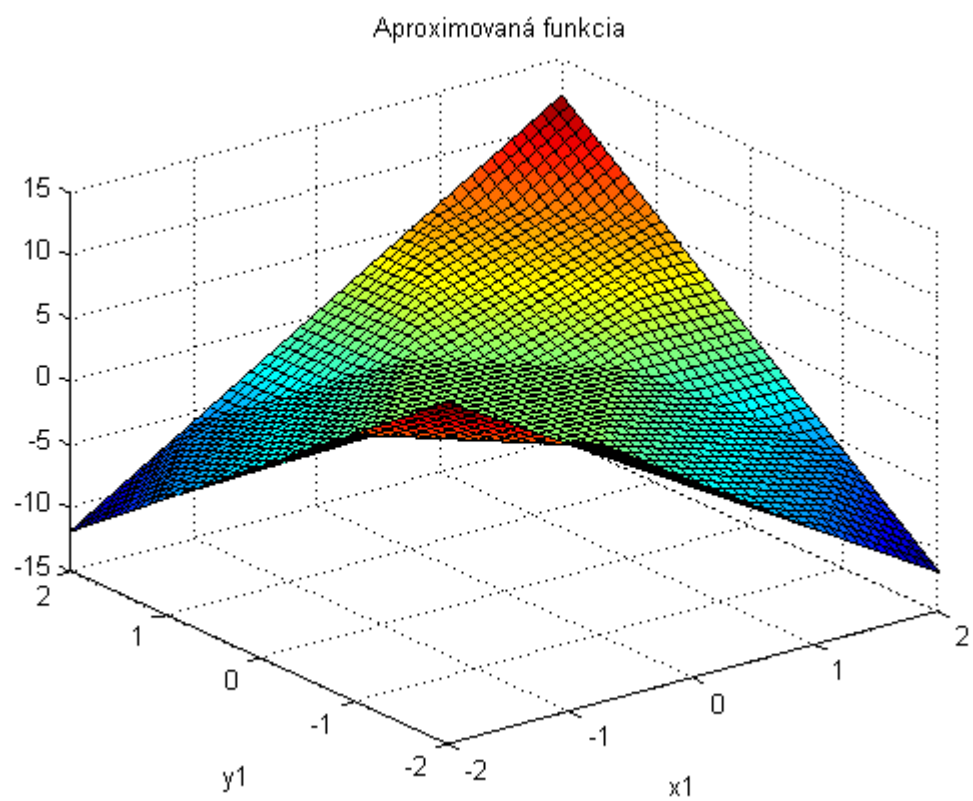
2. graf : Druhá aproximovaná kvadratická závislosť  $y = u_2^2$



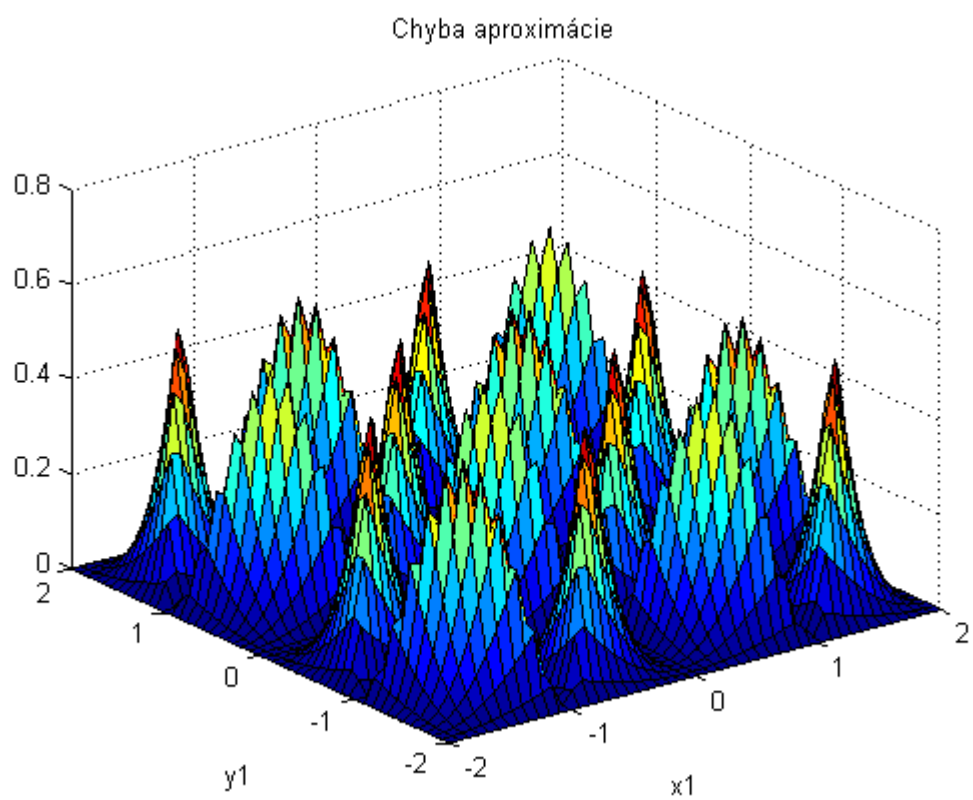
3. graf : Pôvodná nelineárna funkcia na užívateľom zadanom intervale



4. graf : Aproximovaná funkcia pre zadaný rozsah vstupných argumentov



5. graf : Grafické znázornenie úspešnosti aproximácie

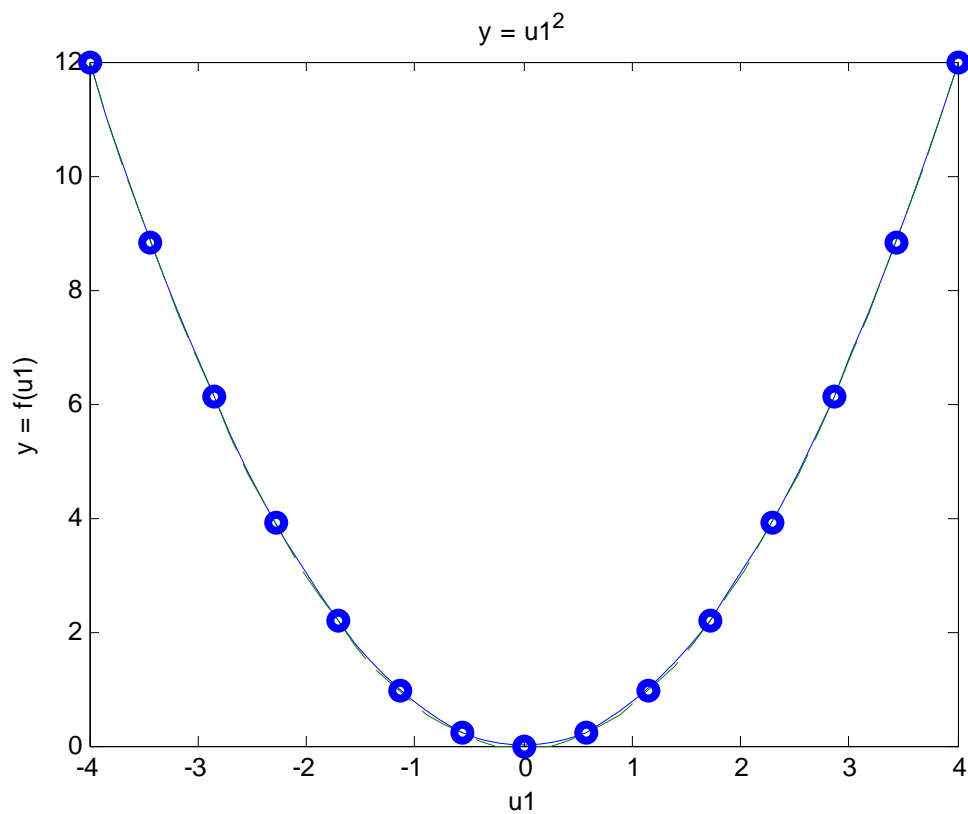




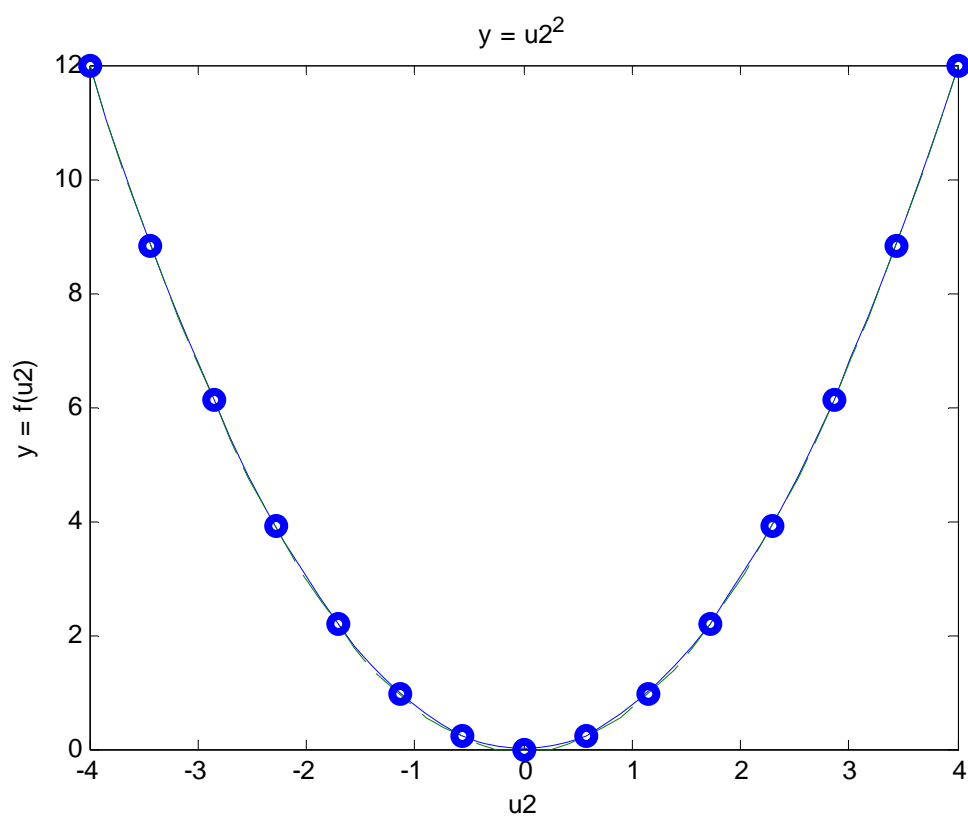
>> aproximuj\_x1x2(f1, f2, x1, x2, x1min, x1max, x2min, x2max, u1nlin, u2nlin,a)

- syms x1 x2
- f1 = x1
- f2 = x2
- x1min = -2
- x2min = 2
- x2mix = -2
- x2max = 2
- u1nlin = 15
- u2nlin = 15
- a = 3

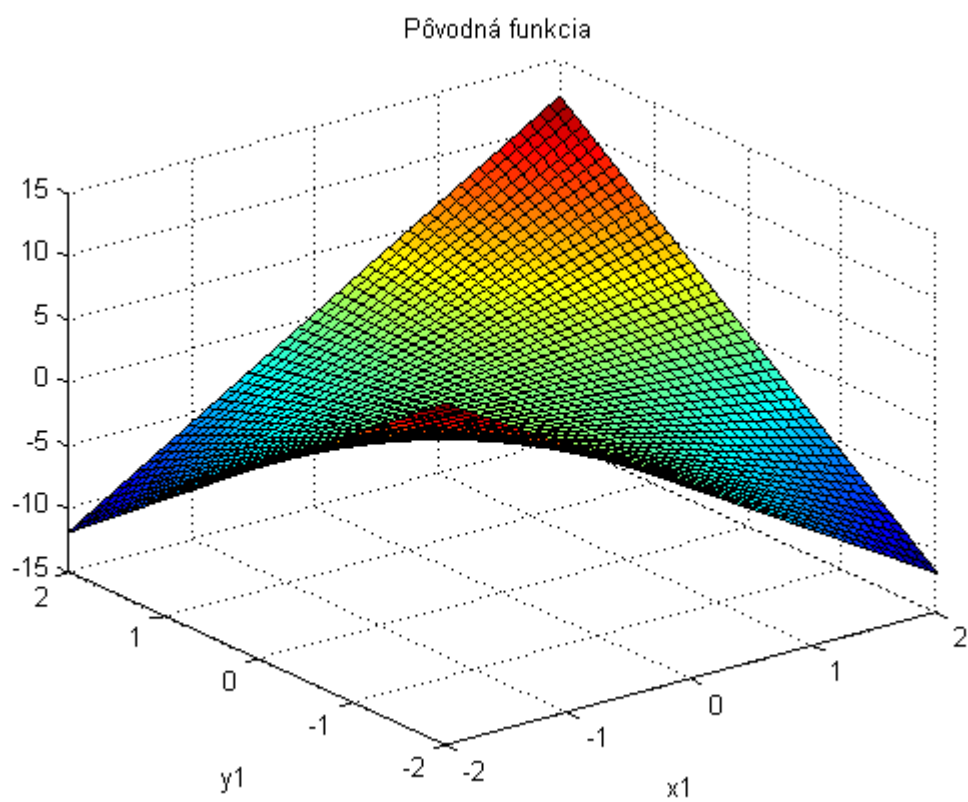
1. graf : Prvá linearizovaná parabola  $y = u_1^2$



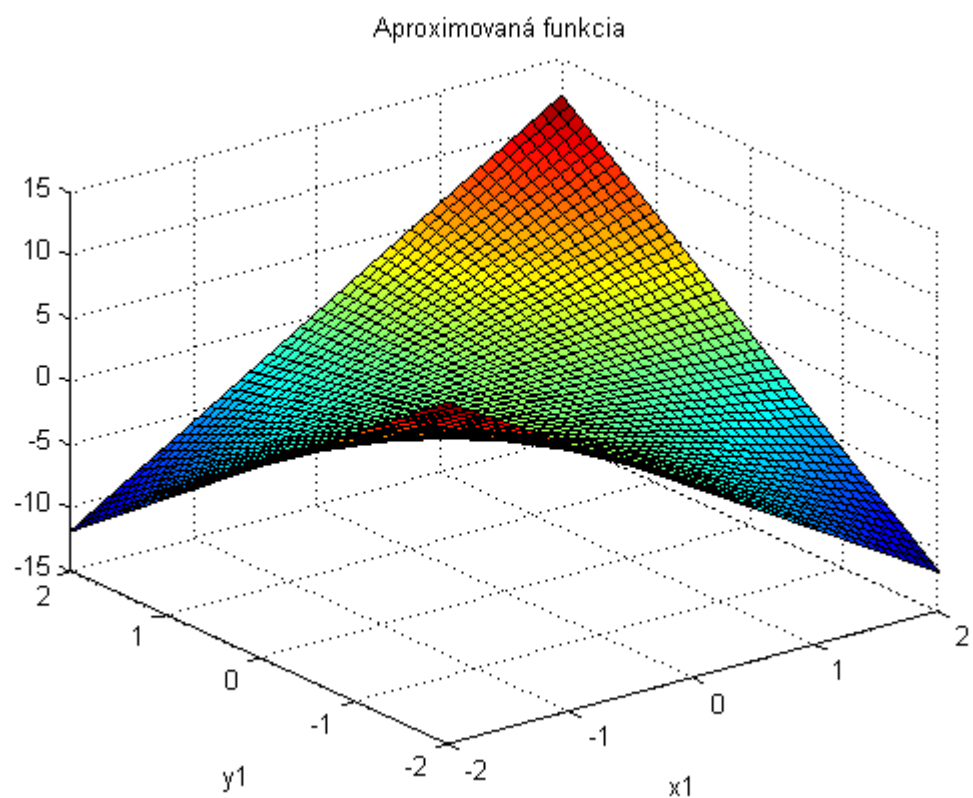
2. graf : Druhá aproximovaná kvadratická závislosť  $y = u_2^2$



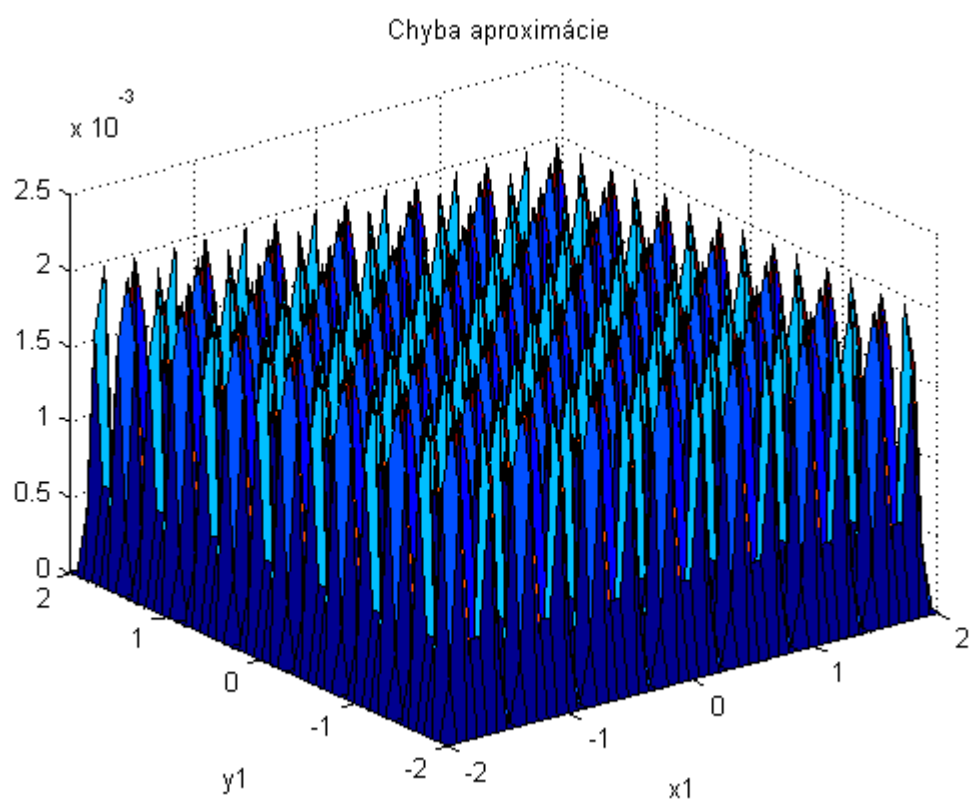
3. graf : Pôvodná nelineárna funkcia na užívateľom zadanom intervale



4. graf : Aproximovaná funkcia pre zadaný rozsah vstupných argumentov



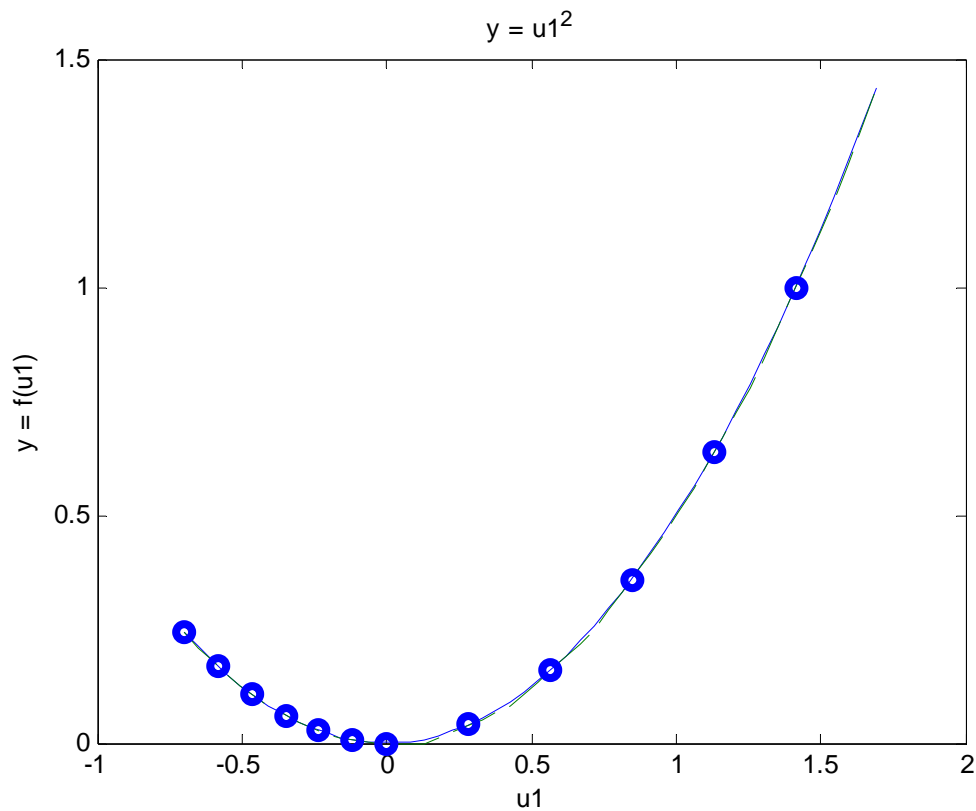
5. graf : Grafické znázornenie úspešnosti aproximácie



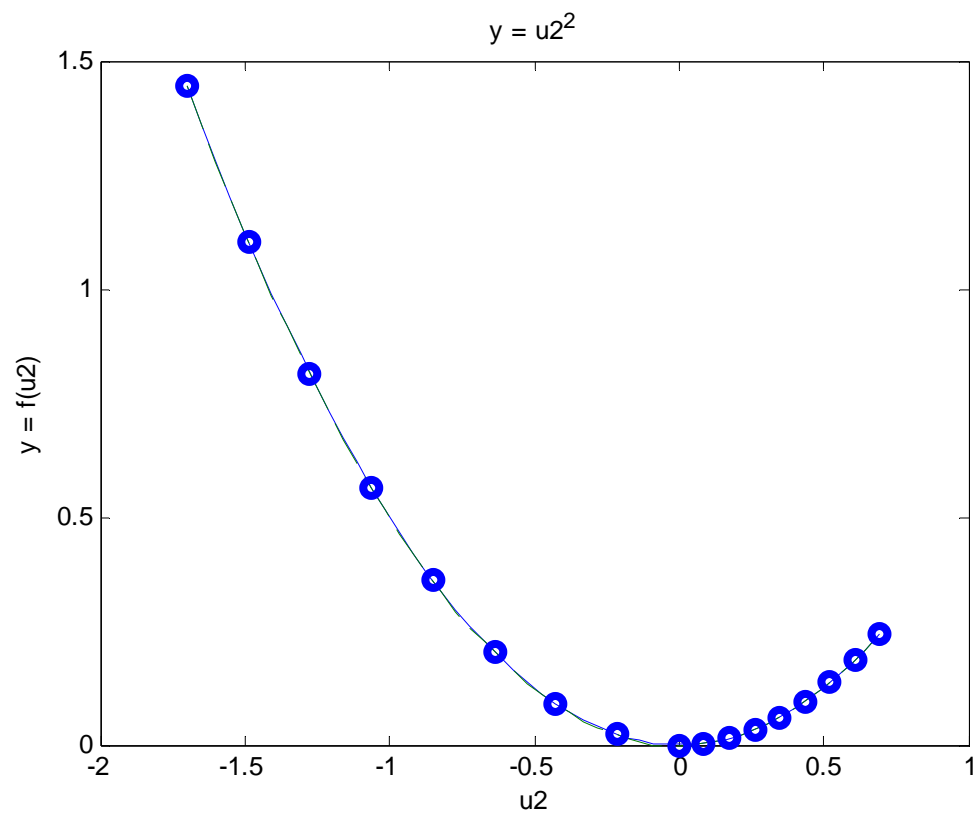
>> aproximuj\_x1x2(f1, f2, x1, x2, x1min, x1max, x2min, x2max, u1nlin, u2nlin)

- syms x1 x2
- f1 = cos(x1)
- f2 = log10(x2)
- x1min = -4
- x2min = 4
- x2mix = 2
- x2max = 5
- u1nlin = 12
- u2nlin = 17
- a = 2

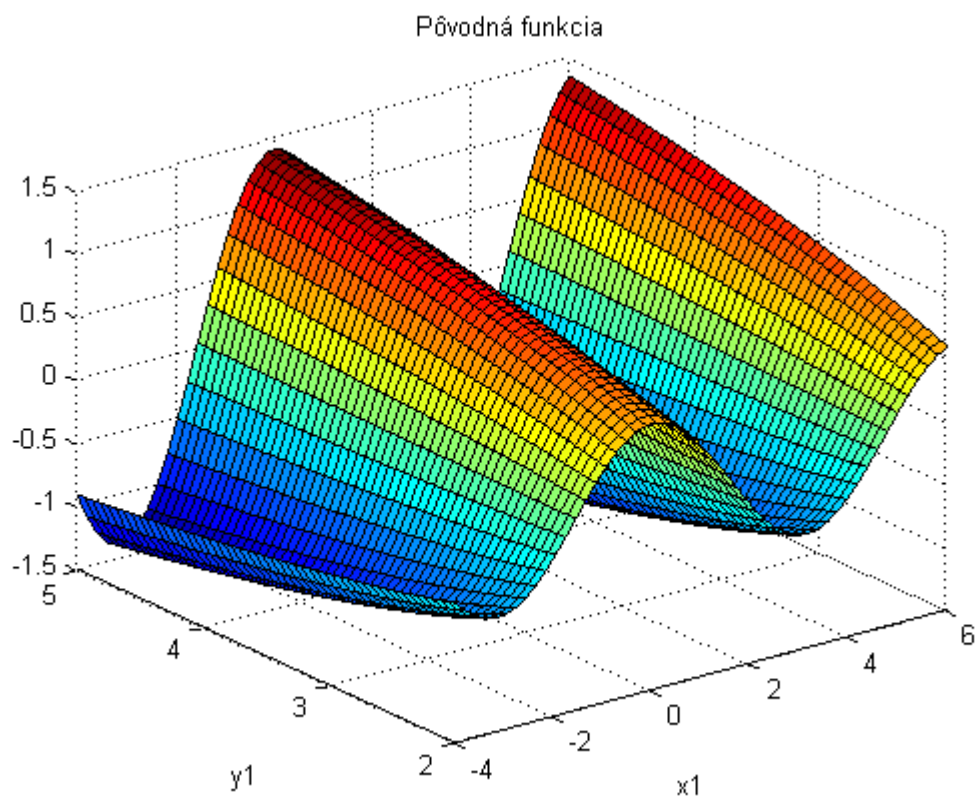
1. graf : Prvá linearizovaná parabola  $y = u_1^2$



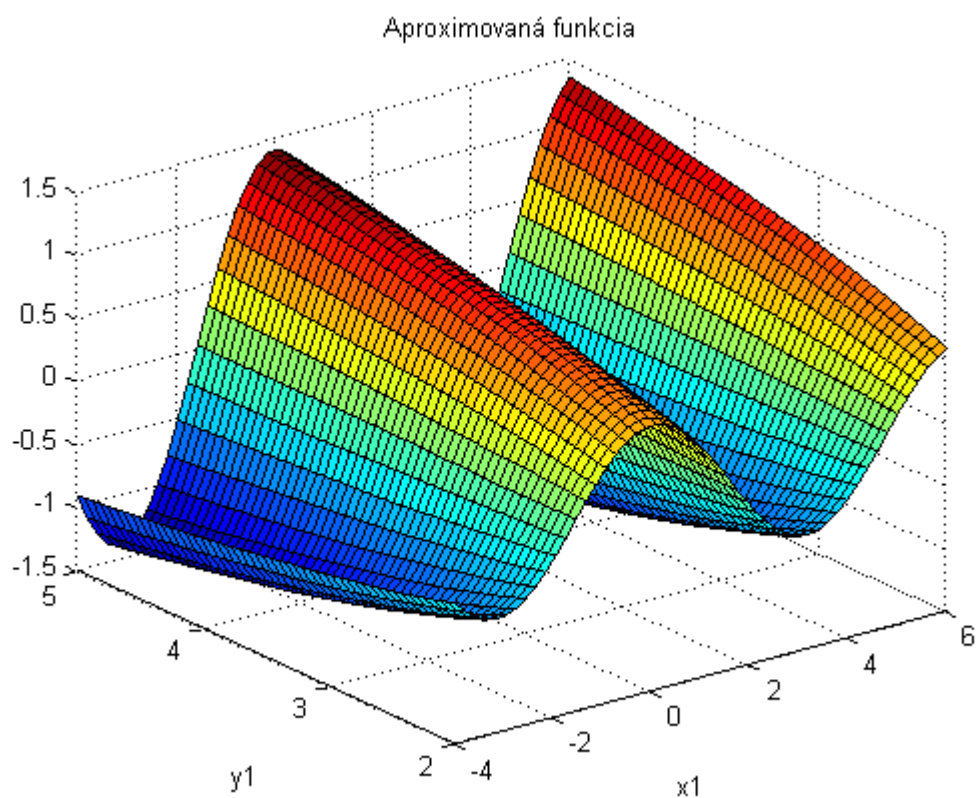
2. graf : Druhá aproximovaná kvadratická závislosť  $y = u_2^2$



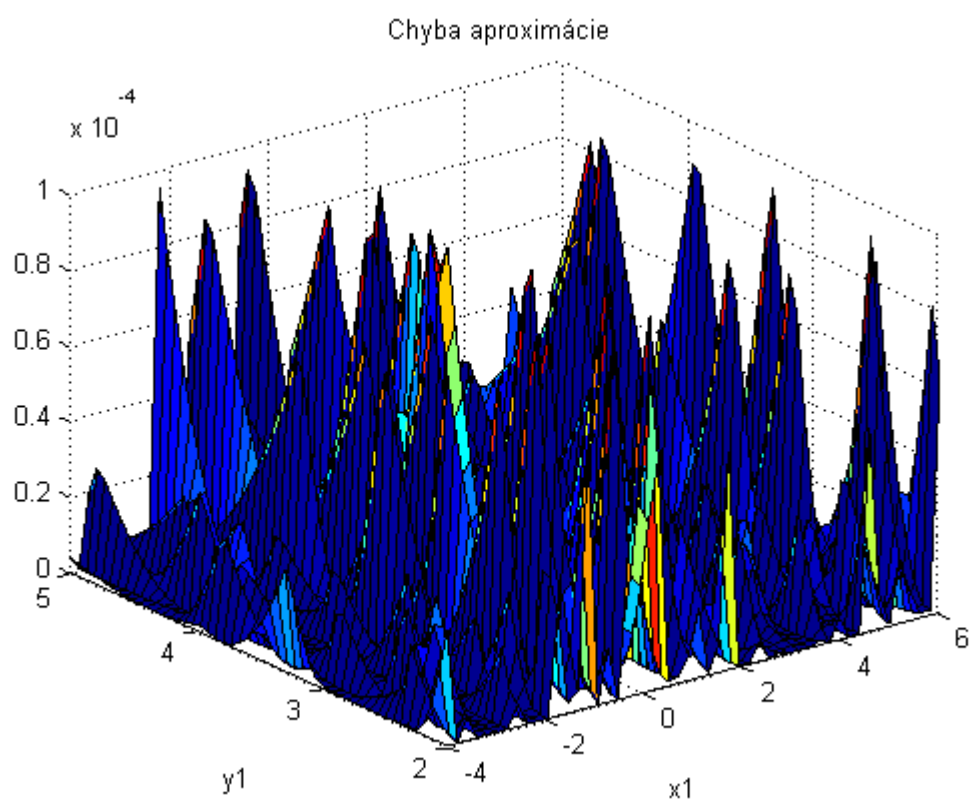
3. graf : Pôvodná nelineárna funkcia na užívateľom zadanom intervale



4. graf : Aproximovaná funkcia pre zadaný rozsah vstupných argumentov



5. graf : Grafické znázornenie úspešnosti aproximácie



## 2.5 UKÁŽKA PRE N – ROZMERNÚ FUNKCIU

Pokiaľ chceme aproximovať  $n$  – rozmernú funkciu, potrebujeme na to funkciu `main_aproximation`, ktorá slúži na linearizáciu ľubovoľnej funkcie.

Napr:  $f = x_3 \cos(x_4) x_1 \sin(x_2)$ . Pred volaním funkcie je nutné si zadať symbolické premenné príkazom `syms x1 x2 x3 .... xn`, kde maximálna hodnota  $n$  je 10.

Syntax v MATLABE vyzerá nasledovne:

```
>> main_aproximation(f, x, xmin, xmax, unlin)
```

- $f = x_1 \log(x_5) \cos(x_7) \sin(x_3)$  : aproximovaná funkcia
- $x = [x_1, x_2, x_3, x_4]$  : vektor symbolických premenných
- $xmin = [x_{1min} \ x_{2min} \ x_{3min} \ x_{4min}]$  – hranice funkcií pre jednotlivé premenné
- $xmax = [x_{1min} \ x_{2min} \ x_{3min} \ x_{4min}]$  – hranice funkcií pre jednotlivé premenné
- $unlin = [u_{1nlin} \ u_{2nlin} \ u_{3nlin} \ u_{4nlin}]$  – vektor aproximačných bodov

Výstupom tejto funkcie sú údaje o aproximácii v príkazovom okne v podobe napr:

Suma štvorcov odchýlok: 0.002030

Maximálna chyba aproximácie 0.000132

a tri grafy, na ktorých sú vykreslené nasledovné funkčné závislosti :

- pôvodná nelineárna funkcia
- aproximovaná funkcia
- chyba aproximácie

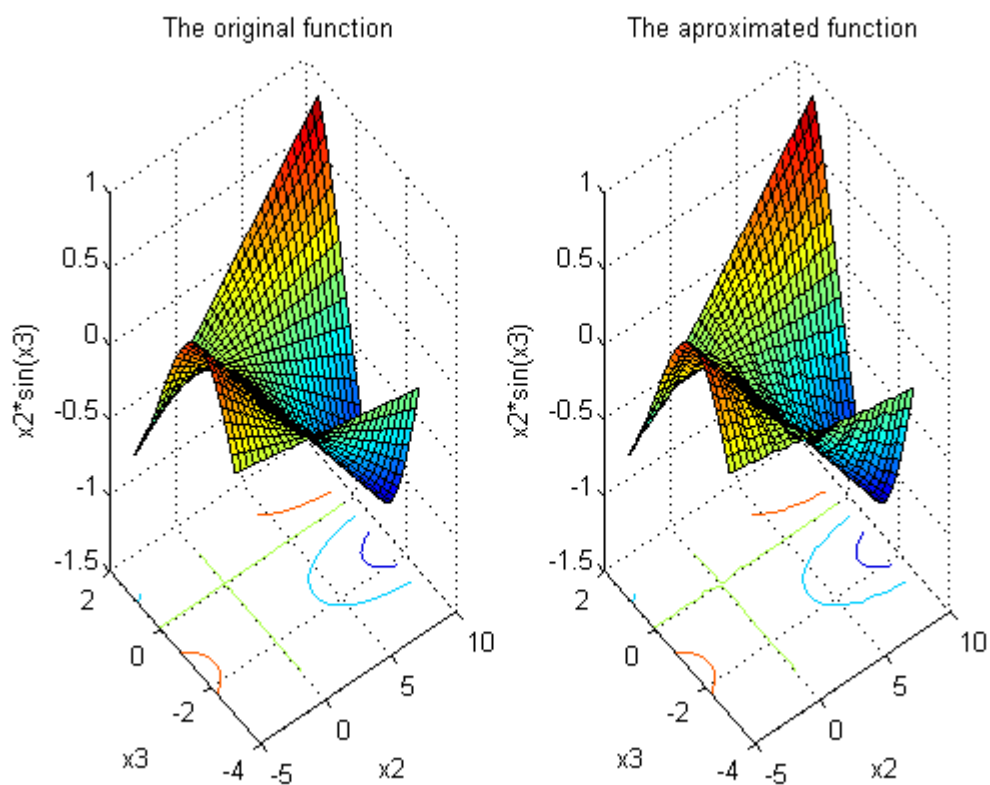
pričom pôvodná a aproximovaná funkcia je vykreslená v jednom grafe v rámci „subplot“ a grafické znázornenie sumy štvorcov odchýlok je na samostatnom grafe. Výsledky sú normalizované, teda pred výpočtom chýb, vektor funkčných hodnôt je rozdelený maximálnou funkčnou hodnotou daného vektora a až potom sa realizuje vyhodnotenie aproximácie.

## 2.6 PRÍKLAD N – ROZMERNEJ FUNKCIE

```
>> main_aproximation(f, x, xmin, xmax, unlin)
```

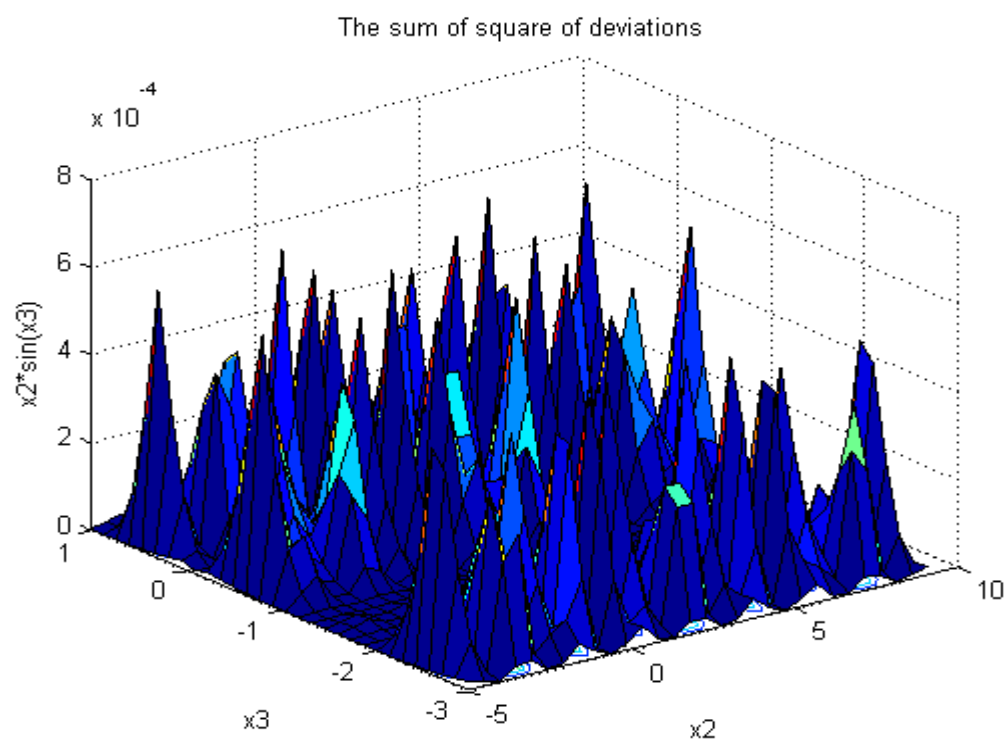
- $x = [x_2 \ x_3 \ x_4 \ x_5]$
- $xmin = [-2\pi, -5, -3, -1]$
- $xmax = [2\pi, 9, 1, 4]$
- $unlin = [10 \ 10 \ 10 \ 10]$
- $f = x_2 \sin(x_3) \cos(x_5) x_4$

1.graf : grafické znázornenie pôvodnej a aproximovanej funkcie

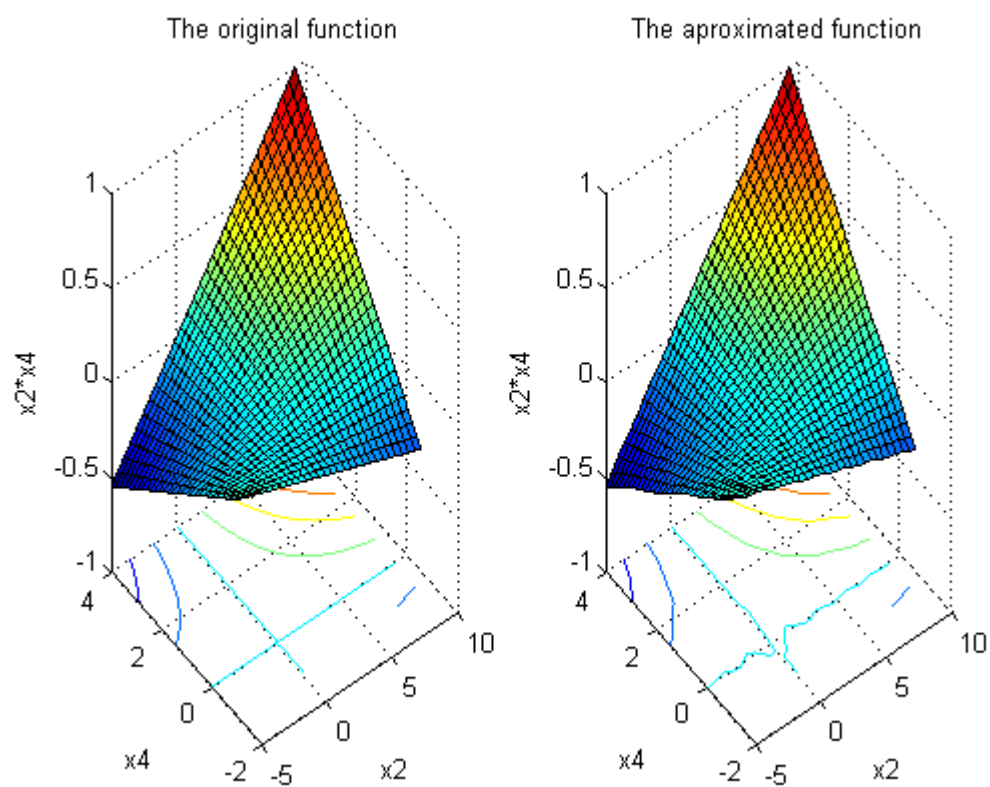




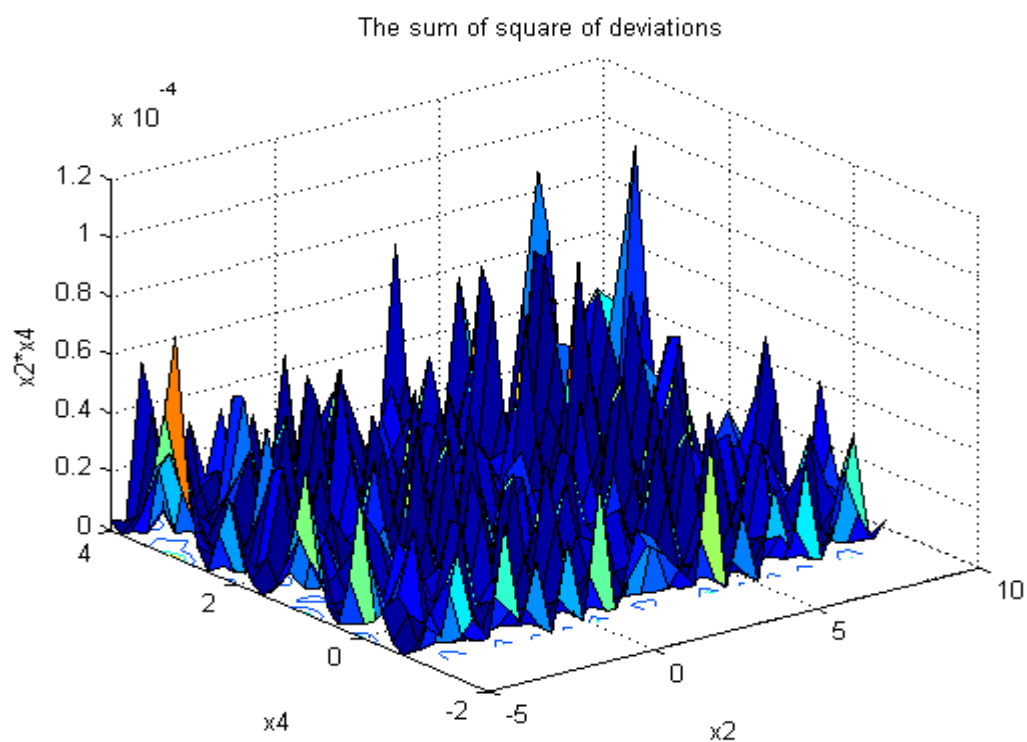
2.graf : suma štvorcov odchýlok pre 1.rez



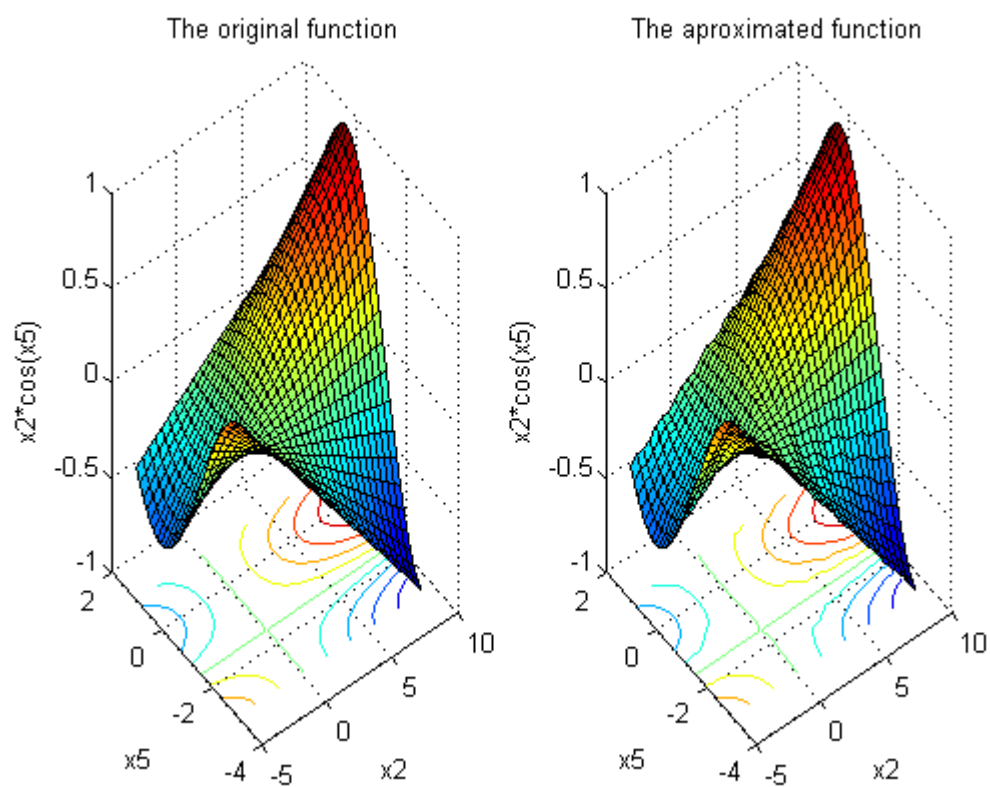
3.graf : Rez vytvorený pri konštantných rovinách  $x_3 = 0$  a  $x_5 = 0$



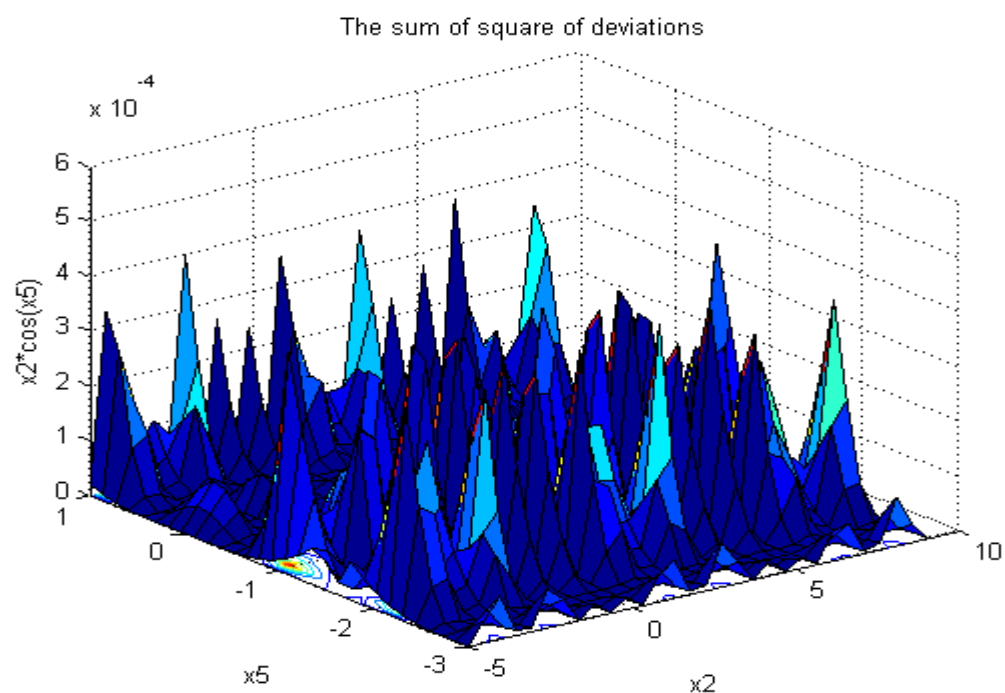
4.graf : chyba aproximácie pre 2.rez



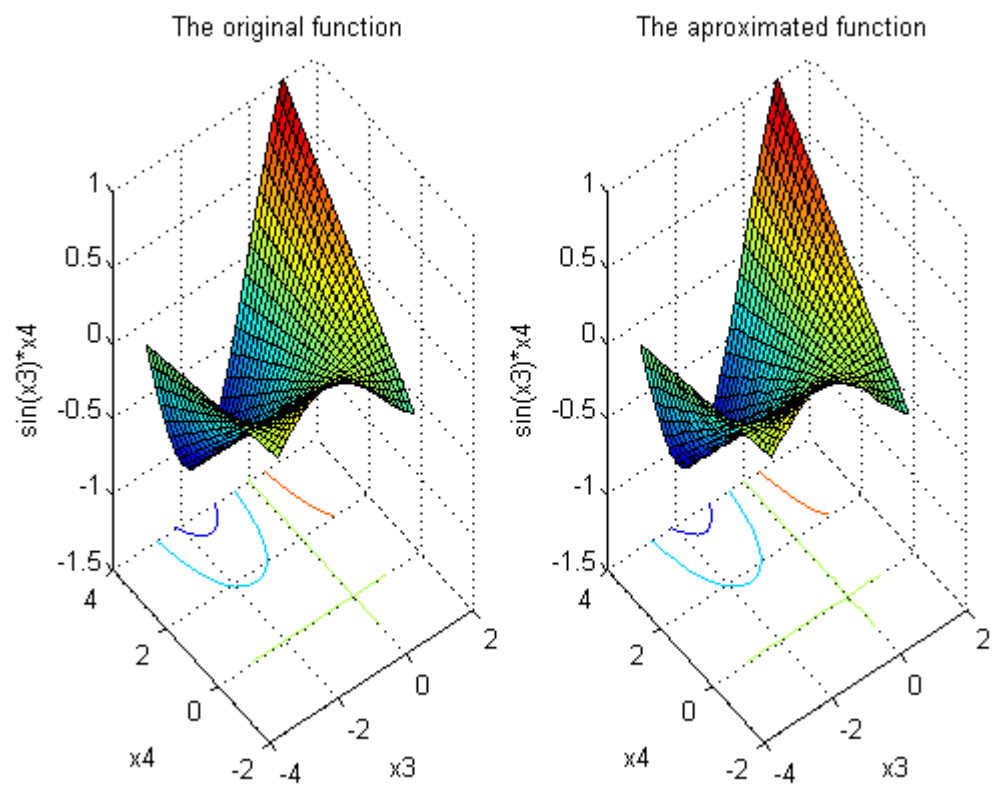
5.graf : Rez vytvorený pri konštantných rovinách  $x_3 = 0$  a  $x_4 = 0$



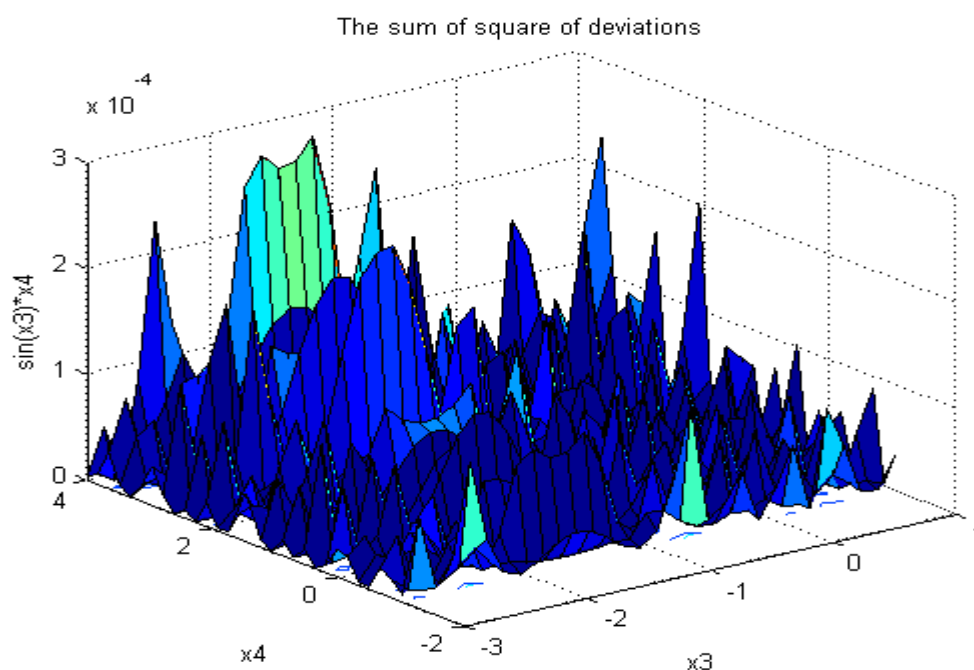
6.graf : Grafické znázornenie chyby linearizácie pre 3.rez



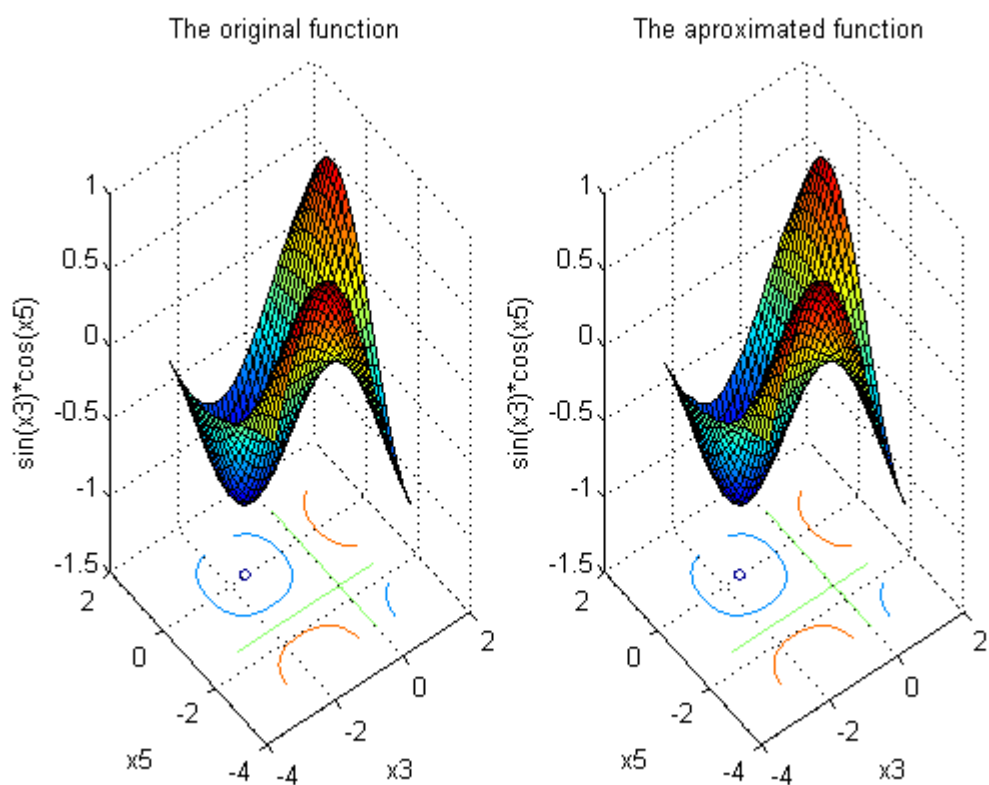
7.graf : 2D funkcia  $\sin(x_3)x_4$



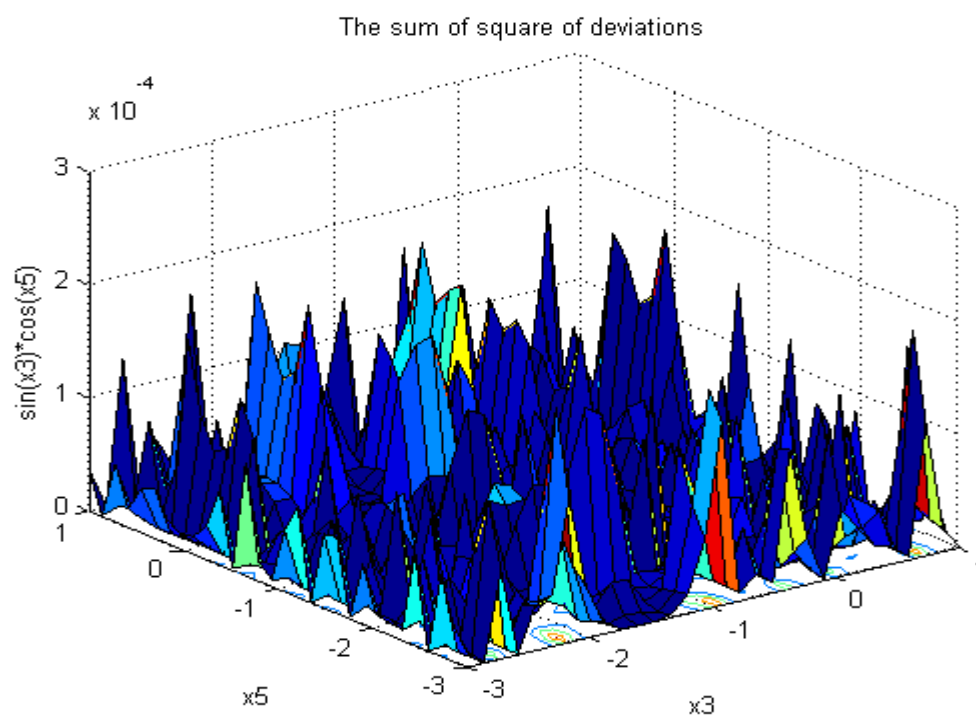
8.graf : Chyba linearizácie pre 4.rez



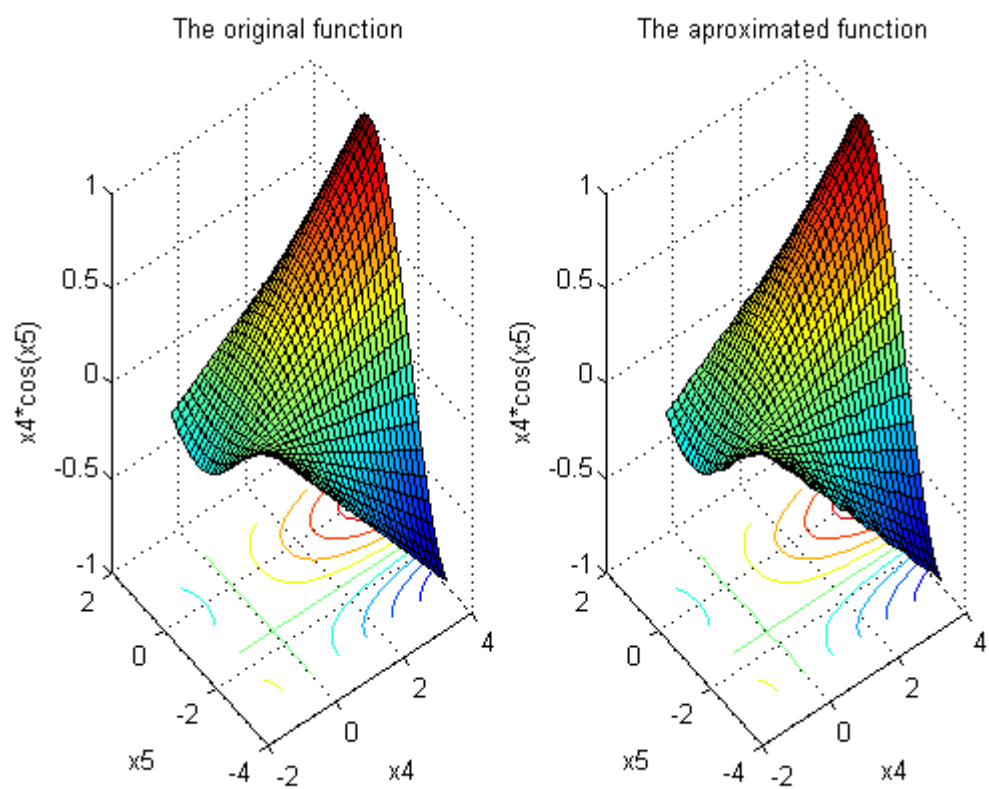
9.graf : Pôvodná a aproximovaná funkcia(5.rez)



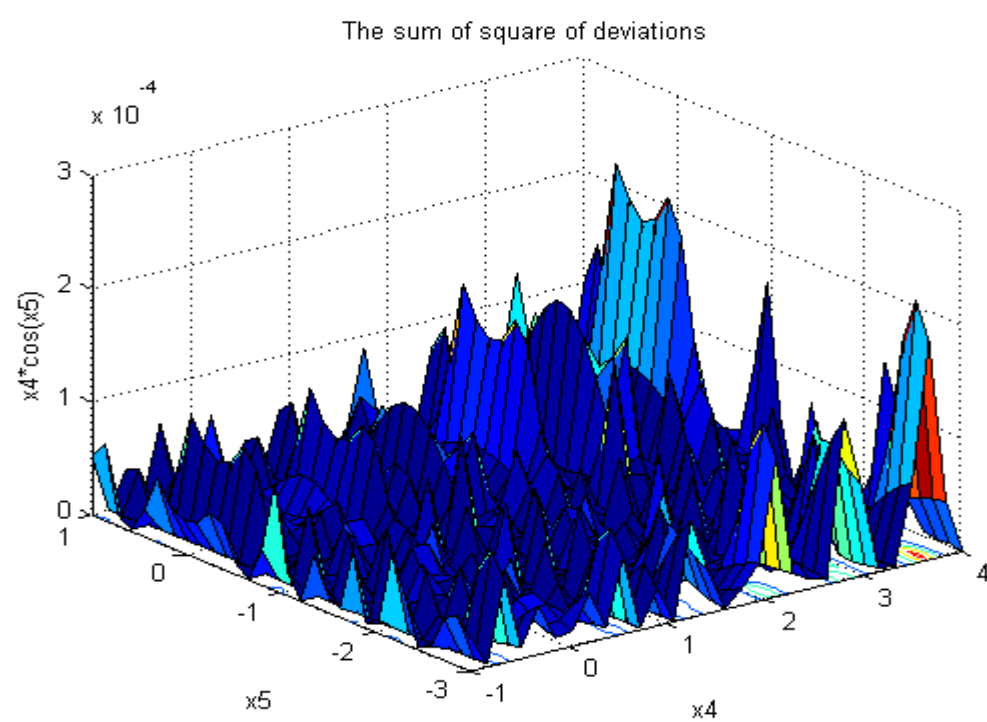
10.graf : Chyba aproximácie (5.rez)



11.graf : Graf pôvodnej a aproximovanej funkcie(6.rez)



12.graf : aproximačná chyba (6.rez)



### 3. ZÁVER

Cieľom práce bola tvorba toolboxu pre automatickú viacnásobnú linearizáciu  $n$  – rozmerných funkčných závislostí, ktorý by mal za úlohu prácu urobiť efektívnejšou a pohodlnejšou pri zistení pozície linearizačných bodov, ale aj pri určení hranice splatnosti regiónov. Software umožňuje linearizovať ľubovoľnú spojitú nelineárnu funkciu rôznymi aproximačnými technikami. Predpokladá sa, že bude vhodným nástrojom pre analýzu a syntézu prediktívnych regulátorov založených na nových optimalizačných prístupoch, ktoré v sebe zahŕňajú aj spojité aj logické (diskrétné) premenné. V budúcnosti bude cieľom práce návrh rôznych hybridných modelov chemicko - technologických procesov a export do balíka HYSDEL.

## Zoznam použitej literatúry

- [1] BAKOŠOVÁ, M., FIKAR, M.: *Riadenie procesov*. Bratislava : Vydavateľstvo STU, 2008. 193s
- [2] ROSINOVÁ, D., DÚBRAVSKÁ, M.: *Optimalizácia*. Bratislava : Vydavateľstvo STU, 2008. 195s
- [3] WILLIAMS, P. Model building in mathematical programming. New York : John Wiley & Sons, 1999. 354 s.