



FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI



ZÁPADOČESKÁ
UNIVERZITA
V PLZNI

Bakalářská práce

Adaptivní přímovazební kompenzace statických sil působících
na mechatronický systém

Západočeská Univerzita V Plzni
Katedra Kybernetiky
BPAŘR

Pavel Březina
letní semestr
16. května 2023

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta aplikovaných věd

Akademický rok: 2022/2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Pavel BŘEZINA**

Osobní číslo: **A20B0314P**

Studijní program: **B0714A150005 Kybernetika a řídící technika**

Specializace: **Automatické řízení a robotika**

Téma práce: **Adaptivní přímovazební kompenzace statických sil působících na mechatronický systém**

Zadávající katedra: **Katedra kybernetiky**

Zásady pro vypracování

1. Seznamte se s modelováním mechanických systémů pro návrh algoritmů automatického řízení.
2. Seznamte se s metodami přímovazební kompenzace statických sil působících na mechanický systém.
3. Navrhněte algoritmus přímovazební kompenzace statických sil pro vybranou třídu mechatronických systémů.
4. Navrhněte metodu adaptace řídícího algoritmu při perturbaci parametrů mechanického systému.
5. Simulačně otestujte navržené algoritmy.
6. V případě možnosti otestujte navržené algoritmy na vhodném laboratorním modelu mechatronického systému.

Rozsah bakalářské práce:

30-40 stránek A4

Rozsah grafických prací:

Forma zpracování bakalářské práce: **tištěná**

Seznam doporučené literatury:

Igor Evgenievich Irodov. Fundamental laws of mechanics. Mir, 1980.

David Morin. Introduction to classical mechanics: with problems and solutions. Cambridge University Press, 2008.

Ramon Vilanova and Antonio Vivaldi. PID control in the third millennium. Springer, 2012.

Dierk Schröder. Intelligent observer and control design for nonlinear systems. Springer Science & Business Media, 2000.

Vedoucí bakalářské práce:

Ing. Václav Helma

Výzkumný program 1

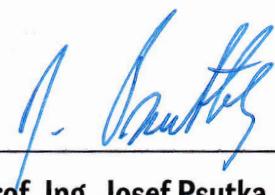
Datum zadání bakalářské práce:

17. října 2022

Termín odevzdání bakalářské práce: **22. května 2023**



Doc. Ing. Miloš Železný, Ph.D.
děkan



Prof. Ing. Josef Psutka, CSc.
vedoucí katedry

VYNECHAT PŘI TISKU

Prohlášení

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne 16. května 2023

.....

Poděkování

Rád bych poděkoval Ing. Václavu Helmovi, vedoucímu této bakalářské práce, za řádné vedení, přátelskou komunikaci a věnovaný čas pravidelným konzultacím, které značně pomohly směru vývoje této práce.

Abstrakt

Tato práce se zabývá automatickou kompenzací statických sil působících na mechatronický systém pomocí proudové kalibrační tabulky. Konkrétně se v práci prozkoumávají možnosti automatické aktualizace této tabulky pomocí NURBS interpolace a aproximace. Toto zahrnuje interpolaci a approximaci 2D křivek, 3D křivek, 3D povrchů a 4D nadpovrchů, včetně jejich ukázek na obecných a konkrétních datech týkajících se problému této práce. Výsledkem práce je autonomní aktualizace proudové kalibrační tabulky za využití approximace 4D nadpovrchu.

Klíčová slova

Abstract

This paper deals with the automatic compensation of static forces acting on a mechatronic system using a current calibration table. Specifically, the work explores the possibilities of automatically updating this table using NURBS interpolation and approximation. This includes the interpolation and approximation of 2D curves, 3D curves, 3D surfaces and 4D hypersurfaces, including their demonstration on general and specific data relevant to the problem of this thesis. The result of this work is an autonomous update of the current calibration table using the 4D hypersurface approximation.

Keywords

Obsah

1	Úvod	1
2	Řízený a řídící systém	3
2.1	Popis řízeného systému	3
2.2	Zjednodušený model systému	3
2.3	Struktura řídícího systému	5
2.3.1	Schéma regulační smyčky	6
2.4	Návrh řešení autonomní rekalibrace proudové tabulky	8
3	NURBS teorie	9
3.1	NURBS křivky	9
3.2	NURBS povrchy	10
3.3	NURBS interpolace křivky	11
3.3.1	Ukázka 2D interpolace	13
3.3.2	Ukázka 3D interpolace	13
3.4	NURBS interpolace povrchu	15
3.4.1	Interpolace 4D nadpovrchu	17
3.5	NURBS aproximace křivky	19
3.5.1	Cílená NURBS aproximace křivky	25
3.6	NURBS aproximace povrchu	25
3.6.1	NURBS aproximace povrchu s užitím Gaussovy funkce	31
3.6.2	NURBS aproximace 4D povrchu s užitím Gaussovy funkce	35
4	Návrh systému automatické kalibrace proudové tabulky	39
4.1	Aplikace NURBS teorie pro modelování kalibrační tabulky	39
4.1.1	1 DoF model	39
4.1.2	2D řez kalibrační tabulkou	39
4.1.3	Interpolace kalibračních tabulek	39
4.1.4	Interpolace napříč kalibračními tabulkami	40
4.2	Analýza měřených dat	46
4.2.1	Analýza zaznamenaného pohybu jednoho kloubu	46
4.2.2	Analýza testu odchylky proudu	51
4.3	Aktualizace kalibrační tabulky	57
5	Závěr	59

Seznam obrázků

1	Výpočetní tomograf s 8 stupni volnosti Phillips Azurion 7 C20	2
2	Schéma zjednodušeného ramene robota	4
3	Potřebný točivý moment motoru pro dosažení úhlu φ pro různé parametry 1 DoF modelu	5
4	Schéma regulační smyčky CT pro jednu osu — C_{Arc}	6
5	Vizualizace kompenzační tabulek pro osu C_{Arc}	7
6	Porovnání algoritmů interpolace ve 2D	14
7	Porovnání algoritmů interpolace ve 3D	14
8	Porovnání algoritmů interpolace ve 2D	16
9	Porovnání algoritmů interpolace ve 3D	16
10	Interpolační povrchy pro 4D interpolaci č. 1	18
11	Ukázka průběhu interpolace mezi povrhy přes souřadnici w č. 1	18
12	Aproximace křivky metodou nejmenších čtverců pro různé parametry	21
13	Aproximace křivky metodou nejmenších čtverců pro různé parametry	21
14	Aproximace křivky metodou nejmenších čtverců pro různé parametry	22
15	Aproximace křivky metodou nejmenších čtverců pro různé parametry	22
16	Aproximace křivky metodou nejmenších čtverců pro různé parametry	23
17	Aproximace křivky metodou nejmenších čtverců pro různé váhy bodů	23
18	Aproximace křivky metodou nejmenších čtverců pro různé váhy bodů	24
19	Aproximace křivky metodou nejmenších čtverců pro různé váhy bodů	24
20	Aproximace povrchu metodou nejmenších čtverců č. 1	28
21	Aproximace povrchu metodou nejmenších čtverců č. 2	28
22	Aproximace povrchu pro různé váhy č. 1	29
23	Aproximace povrchu pro různé váhy č. 2	29
24	Aproximace povrchu pro různé váhy č. 3	30
25	Aproximace povrchu pro různé váhy č. 4	30
26	Ukázka Gaussovo funkce $f(x, y)$	32
27	Ukázka aproximace povrchu č. 1 s užitím Gaussovo funkce s různými parametry	33
28	Ukázka aproximace povrchu č. 2 s užitím Gaussovo funkce s různými parametry	34
29	Ukázka průběhu Gaussovo funkce $f(x, y, w)$	36
30	4D aproximace pro různé parametry Gaussovo funkce č. 1 (Průhlednost approximačních bodů odpovídá jejich vzdálenosti v ose w)	37

31	4D aproximace pro různé parametry Gaussovo funkce č. 2 (Průhlednost approximačních bodů odpovídá jejich vzdálenosti v ose w)	38
32	Cílená aproximace křivky metodou nejmenších čtverců na 1 DoF modelu pro různé váhy nových bodů	40
33	Cílená aproximace křivky metodou nejmenších čtverců na reálných datech	41
34	Cílená aproximace křivky metodou nejmenších čtverců na reálných datech (přiblíženo)	41
35	Ukázka interpolace na reálných datech č. 1	42
36	Ukázka interpolace na reálných datech č. 2	42
37	Ukázka interpolace na reálných datech č. 3	43
38	Ukázka interpolace na reálných datech č. 4	43
39	Interpolační povrchy pro 4D interpolaci kompenzačních tabulek - dopředný pohyb	44
40	Ukázka průběhu interpolace mezi kompenzačními tabulkami přes souřadnici Ids — dopředný pohyb	44
41	Interpolační povrchy pro 4D interpolaci kompenzačních tabulek - zpětný pohyb	45
42	Ukázka průběhu interpolace mezi kompenzačními tabulkami přes souřadnici Ids — zpětný pohyb	45
43	Ukázka aplikace filtru konstantní rychlosti	48
44	Ukázka aplikace všech filtrů	49
45	Vložení vhodných bodů ze záznamu měření jednoho kloubu	50
46	Ukázka Gaussovo aproximace povrchu pro hodnoty z měření jednoho kloubu	50
47	Ukázka trajektorií kloubu C_{Arc} pro 4 proudové kalibrační tabulky . .	52
48	Vypočtený kompenzační proud pro testovací trajektorie	53
49	Zpracování zašuměných záznamů kompenzačního proudu pro testovací trajektorie	54
50	Proložení zpracovaných dat pomocí smooth NURBS interpolací	55
51	Proložení zpracovaných dat pomocí NURBS aproximace NURBS interpolací	56
52	Ukázka aktualizace CCT pro zpětný pohyb, $Ids = 1$ na základě měření získaného z analýzy testu odchylky proudu	58

1 Úvod

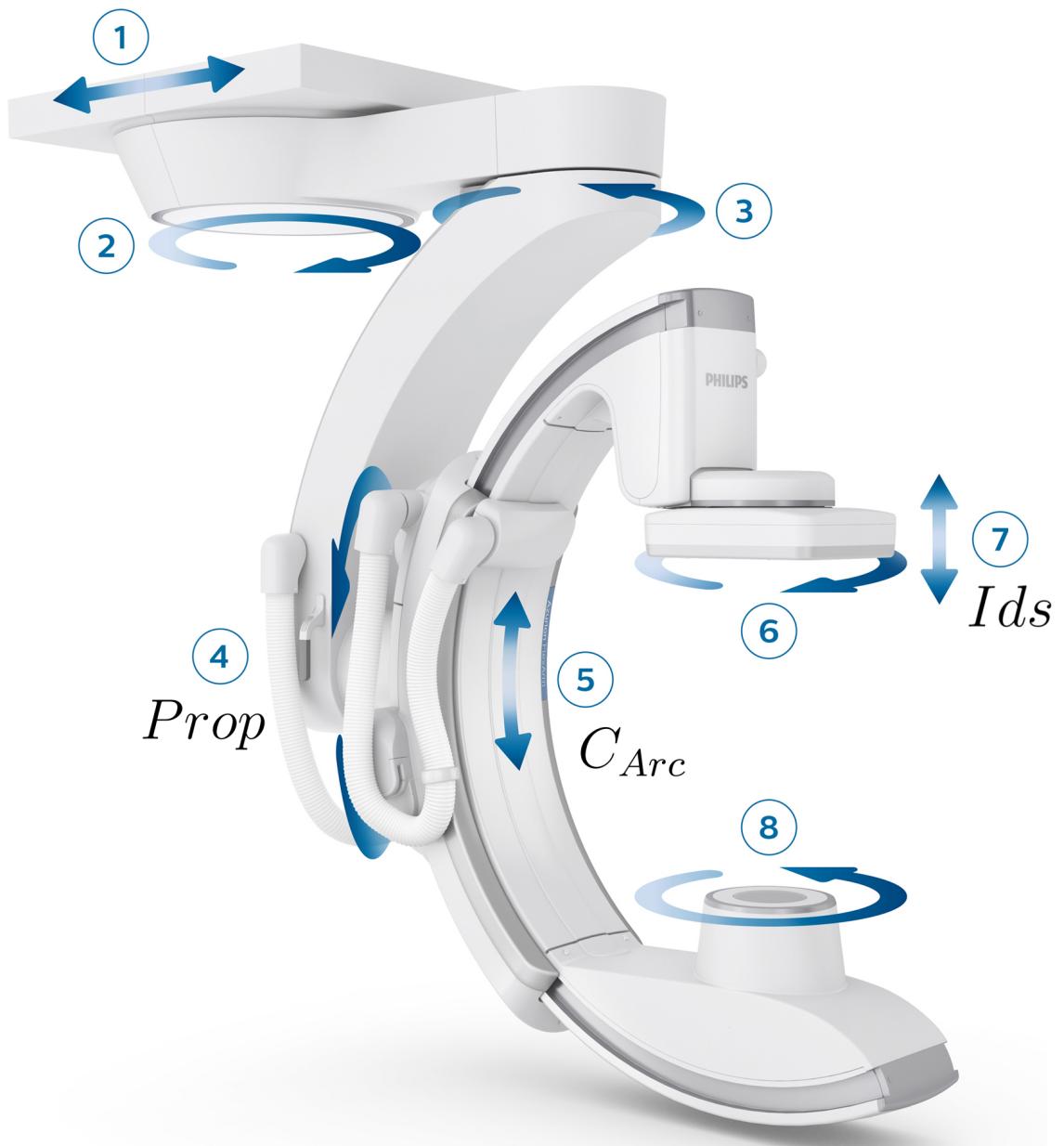
V práci je nejprve popsána struktura řídících smyček pohybu lékařského CT manipulátoru Phillips Azuron 7 C20 včetně proudové kalibrační tabulky (CCT). Kalibrační tabulka hraje zásadní roli pro dosažení přesných pohybů jednotlivých kloubů a zajištění přesného a konzistentního chování při lékařských zákrocích. Z důvodu přirozeného opotřebení stroje je však nutná pravidelná rekalibrace, která vede k nežádoucím odstávkám. Cílem této práce je proto navrhnout řešení, jak minimalizovat potřebu rekurentních manuálních rekalibrací stroje pomocí vhodného algoritmu aktualizace CCT, čímž by se zvýšila přesnost a spolehlivost manipulátoru a také snížil počet odstávek stroje.

Tato práce se zabývá řešením pomocí B-spline interpolace/aproximace. Interpolace a approximace jsou rozhodujícími technikami pro problém aktualizace CCT, které umožňují vyvodit správné hodnoty v bodech ležících mimo síť naměřených bodů v této tabulce.

Text je rozdělen na několik hlavních částí:

1. **Řízený a řídící systém** — část zaměřená na řízený a řídící systém, která obsahuje technické specifikace robota, popis problému a také zvolený přístup řešení
2. **NURBS teorie** — obsáhlá kapitola, ve které je rozebrána veškerá teorie k NURBS křivkám/povrchům včetně kompletně vlastní implementace v Matlabu dle knížky Prof. Dr. Les Piegl, *The NURBS Book*. Kapitola obsahuje ukázky výsledků jednotlivých algoritmů na obecných datech.
3. **Návrh systému automatické kalibrace proudové tabulky** — část, ve které jsou vizualizovány výsledky aplikace principů NURBS teorie na reálných datech. Dále v této kapitole byly také využity techniky z oblasti zpracování signálů, za účelem nalezení vhodných částí měření, které by se daly využít pro aktualizaci CCT.

V práci jsou uvedeny znormované hodnoty poloh kloubů z důvodu důvěryhodnosti dat. V případě ukázky animací, je zobrazeno pouze 6 snímků pro tištěnou verzi, plná verze animace je obsažena v příloze či v digitální verzi dokumentace. Celá tato práce včetně zdrojového kódu je veřejně dostupná na GitHubu — [link](#).



Obrázek 1: Výpočetní tomograf s 8 stupni volnosti Phillips Azurion 7 C20 — Phillips, *Azurion 7 C20*

2 Řízený a řídící systém

Tato kapitola je zaměřena na popis řízeného systému a jeho zjednodušeného modelu. Dále je popsána struktura řídícího systému včetně regulační smyčky a následně je uveden návrh řešení autonomní rekalibrace proudové tabulky.

2.1 Popis řízeného systému

Jedná se o sériový robotický manipulátor s 8 stupni volnosti — model Phillips Azurion 7 C20. Manipulátor je špičkový lékařský zobrazovací systém určený pro použití v zákokrové kardiologii a radiologii^[1]. Stroj lze vidět na obrázku č. 1.

Pro účely této práce jsou klíčové klouby č. 4, 5 a 7, které mají i své vlastní pojmenování os s příslušným rozsahem pohybu^[2]:

- Kloub č. 5 — $C_{Arc} \in [0, 1]$
- Kloub č. 4 — $Prop \in [0, 1]$
- Kloub č. 7 — $Ids \in [0, 1]$

Pro jednotlivé klouby byl navržen regulátor pohybu na základě fyzikálních parametrů stroje. Tyto parametry se mohou časem měnit, až se nakonec začnou projevovat na průběhu regulace. Toto může být například způsobeno nerovnoměrným opotřebením klíčových dílů zodpovědných za přesný pohyb robota a nebo pouhým postupným třením kabelů. Nejen tímto způsobené nepřesnosti lze kompenzovat přímovazební složkou doplněnou o proudovou kalibrační tabulkou.

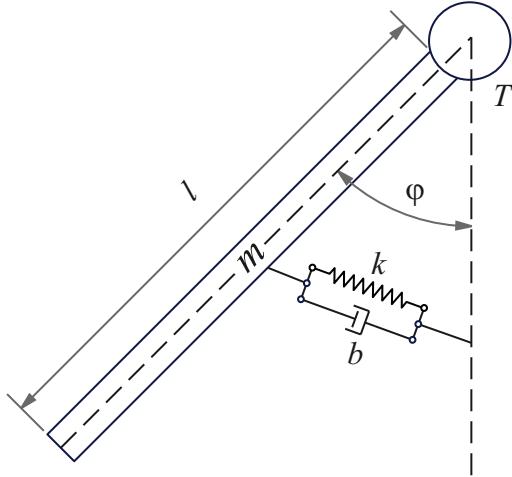
2.2 Zjednodušený model systému

Nyní vytvoříme zjednodušený 1 DoF model reálného systému (viz obrázek č. 2), který poslouží pro názornější vizualizaci výsledků dosažených navrženými algoritmy a jejich snazší otestování a také porovnání různých přístupů. Uvažujeme, rameno o délce l a hmotnosti m , které je na jednom konci ukotvené na hrídeli prostřednictvím rotačního kloubu s pohonem generující moment síly (točivý moment) T . Abychom mohli nějakým způsobem lépe simulovali změnu fyzikálních parametrů, zavedeme do

^[1]Phillips, *Azurion 7 C20 with FlexArm Image-guided therapy system*.

^[2]Přesné hodnoty byly znormovány z důvodu důvěryhodnosti dat

modelu také pružnost k a koeficient tlumení b pohybu ramene. Výsledné schéma je na obrázku č. 2.



Obrázek 2: Schéma zjednodušeného ramene robota

Budeme vycházet z Newtonovo druhého pohybového zákona pro rotační pohyb — konkrétně z rovnice pro moment síly \mathbf{M} , rovnice pro moment hybnosti \mathbf{L} a jejich vzájemného vztahu:

$$\mathbf{M} = \mathbf{r} \times \mathbf{F} \quad (1)$$

$$\mathbf{L} = \mathbf{r} \times \mathbf{p} = \mathbf{r} \times m \cdot \mathbf{v} \quad (2)$$

$$\mathbf{M} = \frac{d\mathbf{L}}{dt} \quad (3)$$

Dosazením parametrů našeho modelu dostáváme:

$$M = -mgl \cdot \sin(\varphi(t)) - k \cdot \varphi(t) - b \cdot \dot{\varphi}(t) + T(t) \quad (4)$$

$$L = ml^2 \cdot \dot{\varphi}(t) \quad (5)$$

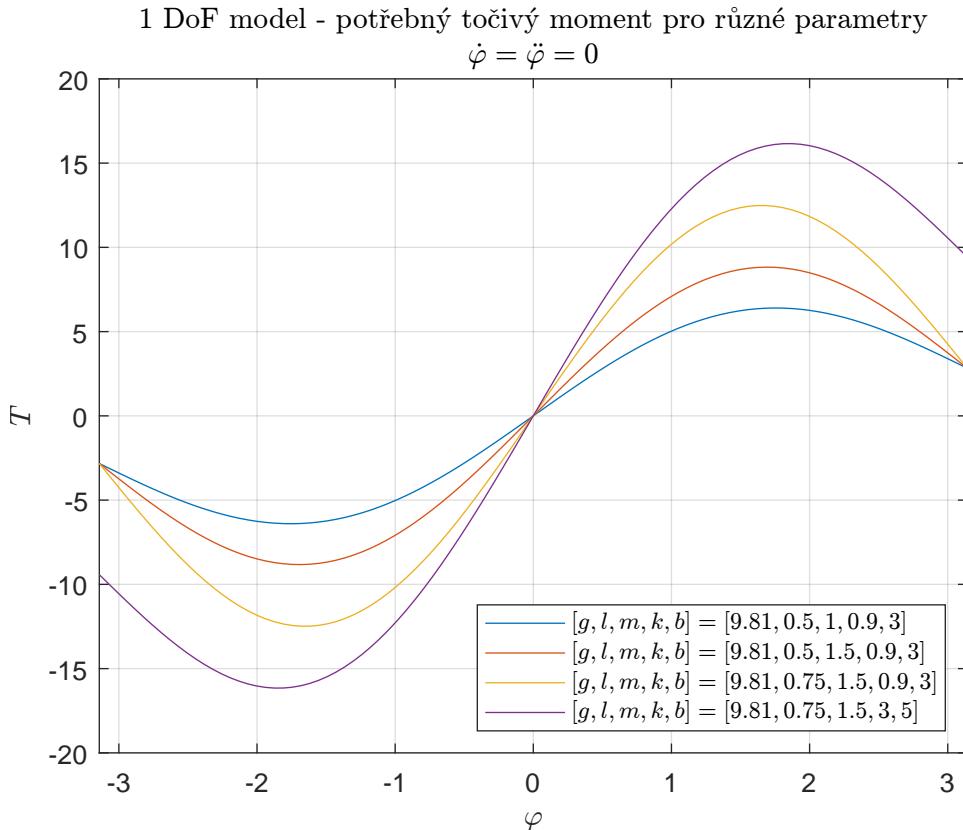
Výsledný model:

$$\dot{L} = M \quad (6)$$

$$ml^2 \cdot \ddot{\varphi}(t) = -mgl \cdot \sin(\varphi(t)) - k \cdot \varphi(t) - b \cdot \dot{\varphi}(t) + T(t) \quad (7)$$

$$\Rightarrow \ddot{\varphi}(t) = \frac{-mgl \cdot \sin(\varphi(t)) - k \cdot \varphi(t) - b \cdot \dot{\varphi}(t) + T(t)}{l^2 m} \quad (8)$$

Na tomto modelu můžeme například vykreslit potřebný točivý moment T v ustáleném stavu ($\ddot{\varphi} = \dot{\varphi} = 0$) pro daný konstantní úhel φ na základě různých fyzikálních parametrů — viz obrázek č. 3. Tímto můžeme simulovat změny fyzických parametrů, které se časem probíhají i na reálném stroji a také na výsledcích můžeme později odzkoušet interpolační/aproximační algoritmy.



Obrázek 3: Potřebný točivý moment motoru pro dosažení úhlu φ pro různé parametry 1 DoF modelu

2.3 Struktura řídícího systému

Kompenzace proudu pohonu pro osu C_{Arc} je závislá na polohách všech tří kloubů (C_{Arc} , $Prop$, Ids) a také na směru pohybu^[3].

^[3]Kompenzace proudu pohonu pro zbylé osy také pravděpodobně závisí na ostatních osách, nicméně jsou dostupná data pouze pro kompenzační tabulkou osy C_{Arc}

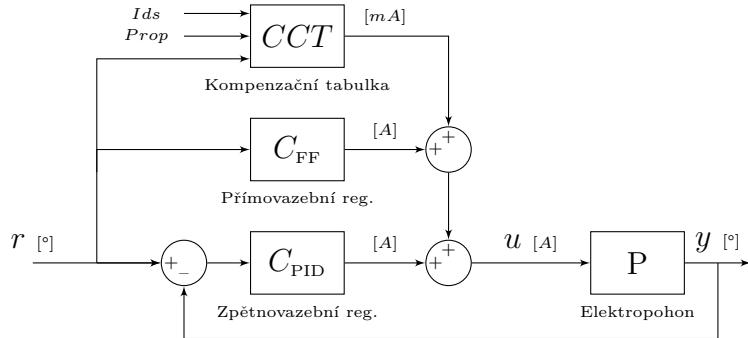
V současném řídícím algoritmu jsou uvažovány pouze dvě možné polohy kloubu Ids a dva možné směry pohybu (dopředný a zpětný), což nám umožní vizualizovat proudovou kalibrační tabulkou prostřednictvím čtyř 3D grafů povrchu. Vizualizovaná data z těchto 4 tabulek pro kloub C_{Arc} lze vidět na obrázcích č. 5a, 5b, 5c, 5d.

2.3.1 Schéma regulační smyčky

K zajištění přesného a plynulého pohybu regulační smyčka obsahuje 3 hlavní kompenzátory:

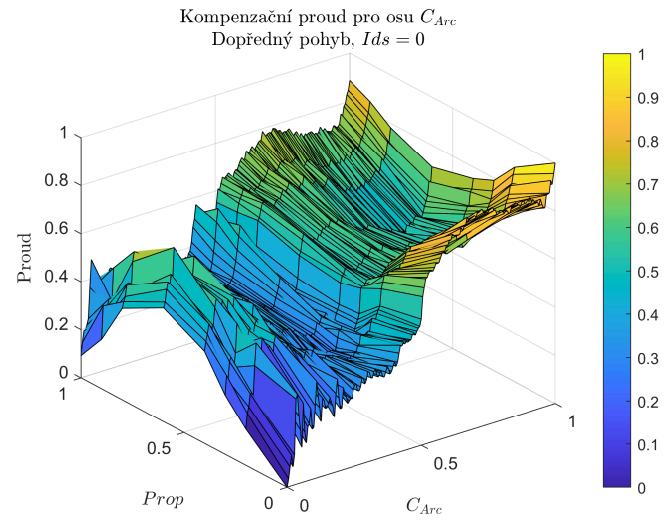
- C_{PID} — zpětnovazební PID regulátor **dohodit lepší popis**
- C_{FF} — přímovazební modelově orientovaný regulátor
- CCT — přímovazební datově orientovaný regulátor v podobě kompenzační tabulky

Schéma regulační smyčky pro polohu kloubu C_{Arc} vypadá přibližně takto:

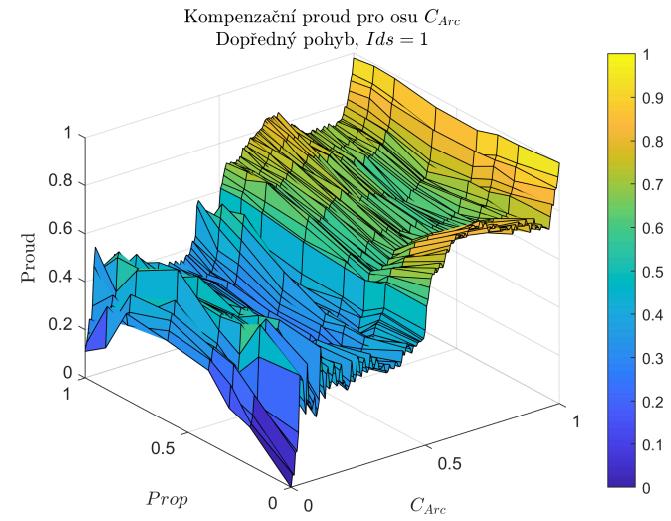


Obrázek 4: Schéma regulační smyčky CT pro jednu osu — C_{Arc}

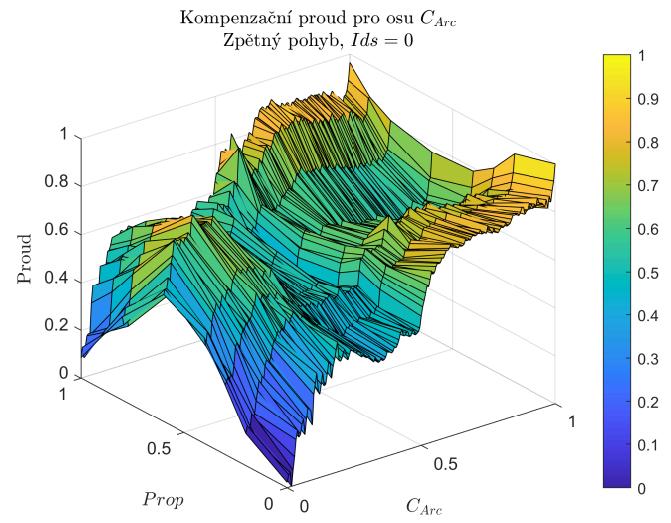
Oba regulátory běžně operují v řádu ampér, zatímco kompenzační tabulka v řádu stovek miliamper. To naznačuje, že tabulka má největší vliv při malých rychlostech (např. při precizních pohybech), kdy výstupy regulátorů jsou v podobném rozsahu jako tabulka.



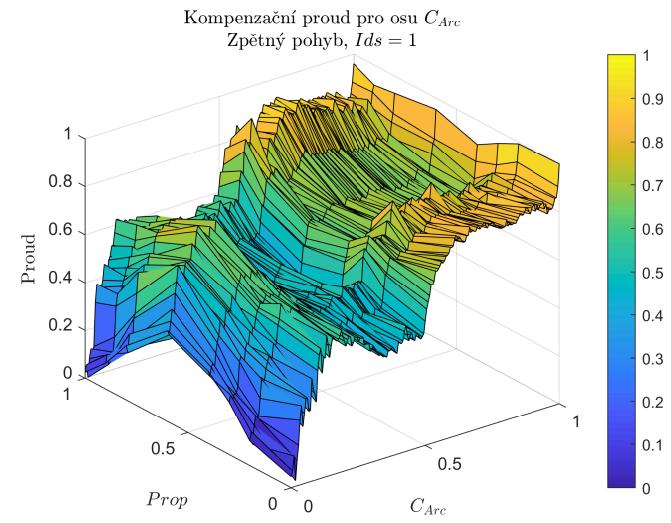
(a) Vizualizace kompenzační tabulky pro osu x — dopředný pohyb, $Ids = 0$



(b) Vizualizace kompenzační tabulky pro osu x — dopředný pohyb, $Ids = 1$



(c) Vizualizace kompenzační tabulky pro osu x — zpětný pohyb, $Ids = 0$



(d) Vizualizace kompenzační tabulek pro osu x — dopředný pohyb, $Ids = 1$

Obrázek 5: Vizualizace kompenzační tabulek pro osu C_{Arc}

2.4 Návrh řešení autonomní rekalibrace proudové tabulky

Při běžné operaci robota můžeme sledovat polohy a rychlosti jednotlivých kloubů pro které se pohyb vyhlažuje pomocí proudové kalibrační tabulky. Sběrem vhodných bodů pro aktualizaci CCT se zabývá sekce [Analýza zaznamenaného pohybu jednoho kloubu](#).

Kompenzační proud pohonu pro jeden směr pohybu je momentálně popsán dvěma kalibračními tabulkami (2 polohy kloubu Ids), kde každá tabulka popisuje 3D povrch. Nový způsob ukládání těchto závislostí je pomocí 4D NURBS nadpovrchu, což je čtyřrozměrný prostor, který již obsahuje kompletní informaci o kompenzačním proudu pro libovolné polohy všech tří kloubů $C_{Arc}, Prop, Ids$ v rámci jejich stanoveného rozsahu.

Uvažujme tedy kompenzační tabulku popsanou 4D nadpovrchem. Vhodně konstruovaný algoritmus aktivně sleduje pohyby manipulátoru a vyhovující hodnoty kompenzačního proudu uloží pro účely aktualizace CCT. Na základě uložených hodnot můžeme poupravit 4D nadpovrch tak, aby na těchto místech byl kompenzační proud popsán přesněji.

Kroky výsledného algoritmu robota pro jednu relaci by mohly vypadat nějak takto:

1. Načtení a vytvoření kalibrační tabulky ve formě NURBS 4D nadpovrchu — tj. provedení NURBS interpolace **rozvést proč interpolace**
2. V průběhu práce s robotem se za běhu
 - (a) z nadpovrchu vypočítá kompenzační proud na základě aktuálních poloh kloubů $C_{Arc}, Prop, Ids$.
 - (b) nasbírají vhodné body pro budoucí aktualizaci kompenzační tabulky.
3. Na konci relace na základě nasbíraných bodů proběhne přepočtení a následné uložení kompenzační tabulky — tj. provedení NURBS approximace **rozvést proč approximace viz článek**

V této práci postupně projdeme všechny tyto zmíněné kroky.

3 NURBS teorie

Tato kapitola je zaměřena na algoritmy pro práci s NURBS křivkami/povrchy, které jsou ručně implementovány v Matlabu podle knížky Prof. Dr. Les Piegl, *The NURBS Book*. V každé podkapitole je uveden odpovídající zdroj z této knížky. Implementace těchto algoritmů je nezbytná pro realizaci autonomní rekalibrace CCT dle návrhu uvedeného v předchozích kapitolách.

V kapitole se hojně užívá pojmu „interpolace“ a „aproximace“, popřípadě ještě „extrapolace“, zde je vysvětlení těchto pojmu:

- Interpolace — Hledání spojité funkce, jejichž funkční hodnoty odpovídají námi zadaným hodnotám na příslušných souřadnicích. Existuje mnoho interpolačních metod například: lineární interpolace, kosinové interpolace, kubická interpolace a polynomiální interpolace. Zejména polynomiální interpolace není pro naše účely vhodná, protože pro velké množství bodů nabývá vysokého řádu a má tendenci kmitat, toto obzvlášt platí pro ekvidistantně zadáné interpolační body.
- Aproximace — Hledání spojité funkce, která nějakým vhodným způsobem pro danou úlohu nejlépe popisuje approximační body. Pro naše účely budeme používat NURBS approximaci založenou na vážené metodě nejmenších čtverců. Tento přístup nám umožní volit stupeň polynomu bázových funkcí, míru redukce bodů (pomocí počtu řídících bodů) a také váhy jednotlivých bodů.
- Extrapolace — Využití interpolované/aproximované funkce mimo původní rozsah interpolačních/aproximačních bodů. Přesnost extrapolace závisí na charakteru zvolené interpolační/aproximační metody a zdrojových dat. V této práci se extrapolace nevyužívá, protože jí není potřeba, a také proto, že NURBS metodu nelze pro extrapolaci použít.

3.1 NURBS křivky^[4]

Neuniformní racionální B-spline (NURBS) křivku $\mathbf{C}(u)$ ^[5] definujeme následovně:

$$\mathbf{C}(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_i \quad a \leq u \leq b \quad (9)$$

^[4] Prof. Dr. Les Piegl, *The NURBS Book*, kapitola 3.2.

^[5] Jedná se tedy o vektorovou funkci skalární proměnné.

kde

- p značí stupeň křivky
- n značí počet řídících bodů
- \mathbf{P}_i jsou řídící body, $\dim \mathbf{P}_i \geq 2$
- parametry a a b lze znormovat bez ztráty obecnosti — nejčastěji se udávají hodnoty $a = 0, b = 1$, které budeme též používat
- $N_{i,p}$ jsou B-spline bázové funkce stupně p definované na neperiodickém neekvidistantním uzlovém vektoru \mathbf{U}
- $u \in \mathbf{U}$

Pro uzlový vektor \mathbf{U} platí:

$$\mathbf{U} = \left\{ \underbrace{a, \dots, a}_{p+1}, u_{p+1}, \dots, u_{m-p-1}, \underbrace{b, \dots, b}_{p+1} \right\}^{m+1} \quad (10)$$

kde $m = n + p + 1$.

Bázové funkce lze definovat rekurzivně (výhoda jednoduché implementace):

$$N_{i,0}(u) = \begin{cases} 1 & u_i \leq u < u_{i+1} \\ 0 & jinak \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (11)$$

3.2 NURBS povrchy^[6]

Pro sestrojení NURBS povrchu potřebujeme obousměrnou síť řídících bodů $\mathbf{P}_{i,j}$ a dva uzlové vektory \mathbf{U} a \mathbf{V} , poté je možné sestrojit povrch $\mathbf{S}(u, v)^{[7]}$:

$$\mathbf{S}(u, v) = \sum_{i=0}^n \sum_{j=0}^m N_{i,p}(u) N_{j,q}(v) \mathbf{P}_{i,j} \quad (12)$$

kde

^[6] Prof. Dr. Les Piegl, *The NURBS Book*, kapitola 3.4.

^[7] Jedná se tedy o vektorovou funkci dvou skalárních proměnných.

- p značí stupeň křivek ve směru u
- q značí stupeň křivek ve směru v
- n značí počet řídících bodů ve směru u
- m značí počet řídících bodů ve směru v
- $\mathbf{P}_{i,j}$ je síť řídících bodů, $\dim \mathbf{P}_{i,j} \geq 3$
- $u \in \mathbf{U}, v \in \mathbf{V}$
- $a \leq u \leq b, a \leq v \leq b$ — stejně jako u [NURBS křivky](#) budeme uvažovat $a = 0$, $b = 1$ bez ztráty obecnosti

Pro uzlové vektory \mathbf{U} a \mathbf{V} platí:

$$\mathbf{U} = \left\{ \underbrace{a, \dots, a}_{p+1}, u_{p+1}, \dots, u_{r-p-1}, \underbrace{b, \dots, b}_{p+1} \right\}^{r+1} \quad (13)$$

$$\mathbf{V} = \left\{ \underbrace{a, \dots, a}_{q+1}, u_{q+1}, \dots, u_{s-q-1}, \underbrace{b, \dots, b}_{q+1} \right\}^{s+1} \quad (14)$$

kde $r = n + p + 1, s = m + q + 1$.

Definice bázových funkcí zůstává stejná — viz (11).

3.3 NURBS interpolace křivky^[8]

Mějme množinu $n + 1$ bodů $\{\mathbf{Q}_k\} k = 0, \dots, n$, které chceme interpolovat NURBS křivkou stupně p . Pokud každému bodu \mathbf{Q}_k přiřadíme parametr \bar{u}_k , a vhodně sestrojíme uzlový vektor $\mathbf{U} = \{u_0, \dots, u_m\}$, můžeme sestavit soustavu lineárních rovnic o rozměru $(n + 1) \times (n + 1)$:

$$\mathbf{Q}_k = \mathbf{C}(\bar{u}_k) = \sum_{i=0}^n N_{i,p}(\bar{u}_k) \mathbf{P}_i \quad (15)$$

kde řídící body \mathbf{P}_i tvoří našich $n + 1$ neznámých.

Parametr \bar{u}_k lze zvolit více způsoby, z nichž jsou běžné například:

^[8] Prof. Dr. Les Piegl, *The NURBS Book*, kapitola 9.2.1.

- ekvidistantní (equally spaced):

$$\bar{u}_0 = 0 \quad \bar{u}_n = 1 \quad (16)$$

$$\bar{u}_k = \frac{k}{n} \quad k = 1, \dots, n-1 \quad (17)$$

Tato metoda je vhodná pro rovnoměrně rozprostřená data.

- délka tětivy (chord length) — nechť d je celková délka tětivy:

$$d = \sum_{k=1}^n \|\mathbf{Q}_k - \mathbf{Q}_{k-1}\| \quad (18)$$

potom

$$\bar{u}_0 = 0 \quad \bar{u}_n = 1 \quad (19)$$

$$\bar{u}_k = \bar{u}_{k-1} + \frac{\|\mathbf{Q}_k - \mathbf{Q}_{k-1}\|}{d} \quad k = 1, \dots, n-1 \quad (20)$$

Tato metoda je vhodná pro obecná data.

- dostředivá metoda (centripetal method) — nechť d :

$$d = \sum_{k=1}^n \sqrt{\|\mathbf{Q}_k - \mathbf{Q}_{k-1}\|} \quad (21)$$

potom

$$\bar{u}_0 = 0 \quad \bar{u}_n = 1 \quad (22)$$

$$\bar{u}_k = \bar{u}_{k-1} + \frac{\|\mathbf{Q}_k - \mathbf{Q}_{k-1}\|}{d} \quad k = 1, \dots, n-1 \quad (23)$$

Tato metoda je vhodná pro obecná data s náhlými změnami směru.

Pro tyto metody je doporučený způsob výpočtu \mathbf{U} metodou průměrování:

$$\begin{aligned} u_0 &= \dots = u_p = 0 & u_{m-p} &= \dots = u_m = 1 \\ u_{j+p} &= \frac{1}{p} \sum_{i=j}^{j+p-1} \bar{u}_i & j &= 1, \dots, n-p \end{aligned} \quad (24)$$

Nyní můžeme sestavit matici \mathbf{N} $(n+1) \times (n+1)$:

$$\mathbf{N} = \begin{bmatrix} N_{0,p}(\bar{u}_0) & \cdots & N_{n+1,p}(\bar{u}_0) \\ \vdots & \ddots & \vdots \\ N_{0,p}(\bar{u}_{n+1}) & \cdots & N_{n+1,p}(\bar{u}_{n+1}) \end{bmatrix} \quad (25)$$

Hledané řídící body \mathbf{P} již spočteme vyřešením soustavy linárních rovnic:

$$\mathbf{P} = \mathbf{N}^{-1} \mathbf{Q} \quad (26)$$

Takto zavedená interpolace funguje pro libovolnou dimenzi bodů \mathbf{Q}_k .

3.3.1 Ukázka 2D interpolace

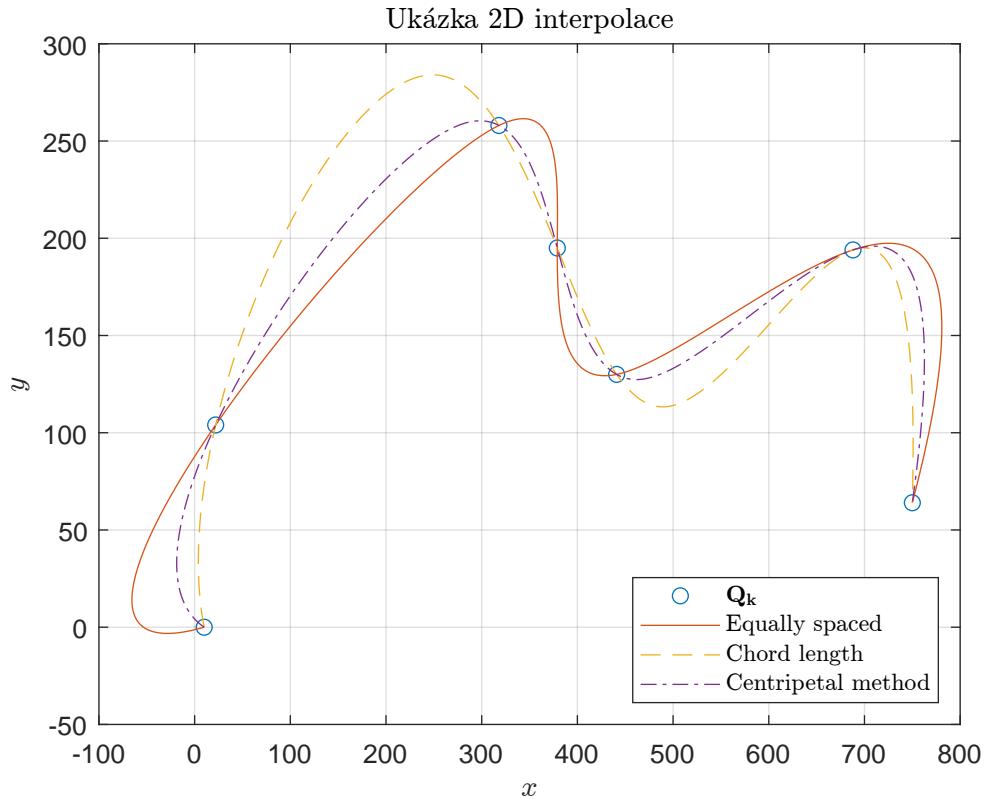
Na obrázku č. 6 jsou porovnány zmíněné algoritmy výpočtu parametru \bar{u}_k pro tyto body:

$$\mathbf{Q}_k = [\mathbf{x} \quad \mathbf{y}] = \begin{bmatrix} 10 & 0 \\ 22 & 104 \\ 318 & 258 \\ 379 & 195 \\ 441 & 130 \\ 688 & 194 \\ 750 & 64 \end{bmatrix} \quad (27)$$

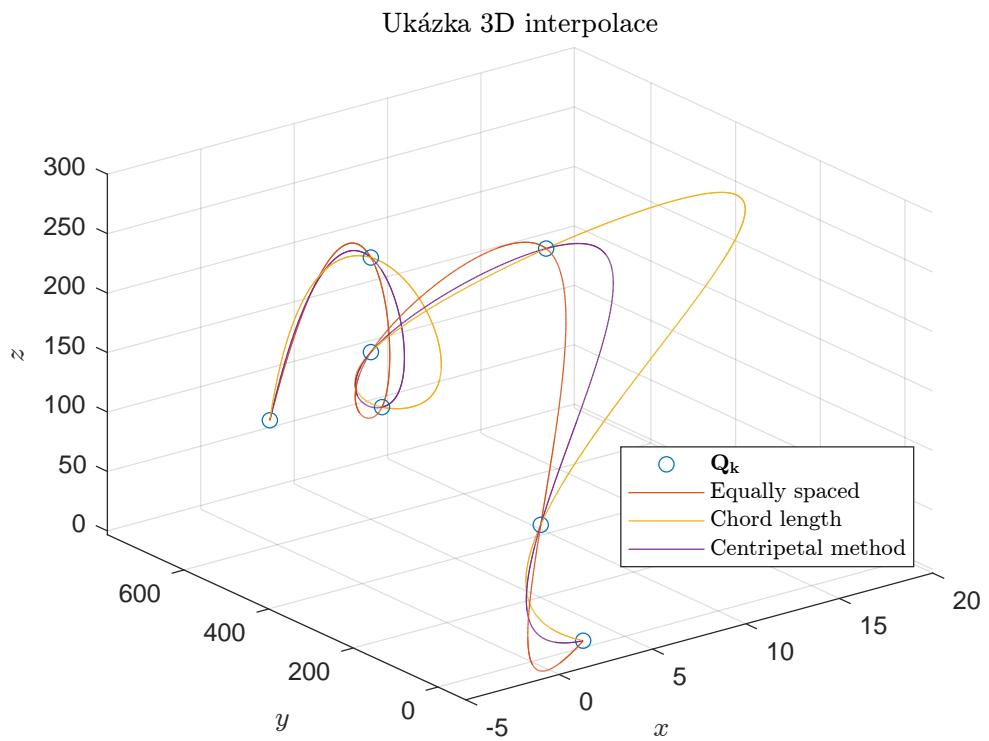
3.3.2 Ukázka 3D interpolace

Na obrázku č. 7 jsou porovnány zmíněné algoritmy výpočtu parametru \bar{u}_k pro tyto body:

$$\mathbf{Q}_k = [\mathbf{x} \quad \mathbf{y} \quad \mathbf{z}] = \begin{bmatrix} 3 & 10 & 0 \\ 1 & 22 & 104 \\ 8 & 318 & 258 \\ 0 & 379 & 195 \\ 2 & 441 & 130 \\ 7 & 688 & 194 \\ 3 & 750 & 64 \end{bmatrix} \quad (28)$$



Obrázek 6: Porovnání algoritmů interpolace ve 2D



Obrázek 7: Porovnání algoritmů interpolace ve 3D

3.4 NURBS interpolace povrchu^[9]

Interpolace povrchu je podobná interpolaci 3D křivky — máme množinu $(n+1) \times (m+1)$ bodů $\{\mathbf{Q}_{k,l}\}$, $k = 0, \dots, n$ a $l = 0, \dots, m$ ležících na mřížce, které chceme interpolovat NURBS povrchem stupně p a q , tzn.:

$$\mathbf{Q}_{k,l} = \mathbf{S}(\bar{u}_k, \bar{v}_l) = \sum_{i=0}^n \sum_{j=0}^m N_{i,p}(\bar{u}_k) N_{j,q}(\bar{v}_l) \mathbf{P}_{i,j} \quad (29)$$

Stejně jako u interpolace křivky musíme vhodně zvolit parametry \bar{u}_k a \bar{v}_l a uzlové vektory \mathbf{U} a \mathbf{V} . Užitím běžných metod (20) a (23) získáme vektory $\hat{\bar{u}}_k$ a $\hat{\bar{v}}_l$, které musíme poté zprůměrovat přes všechny hodnoty, tzn.:

$$\bar{u}_k = \frac{1}{m+1} \sum_{j=0}^m \hat{\bar{u}}_j \quad k = 0, \dots, n \quad (30)$$

$$\bar{v}_l = \frac{1}{n+1} \sum_{j=0}^n \hat{\bar{v}}_j \quad l = 0, \dots, m \quad (31)$$

Uzlové vektory \mathbf{U} a \mathbf{V} spočteme již stejně jako u interpolace křivky, viz 24.

Oproti křivce, v tomto případě $\mathbf{P}_{i,j}$ již není matice, ale tenzor. Tento problém můžeme zjednodušit na interpolaci křivek postupně v obou směrech zafixováním jedné z proměnných k nebo l , tj.:

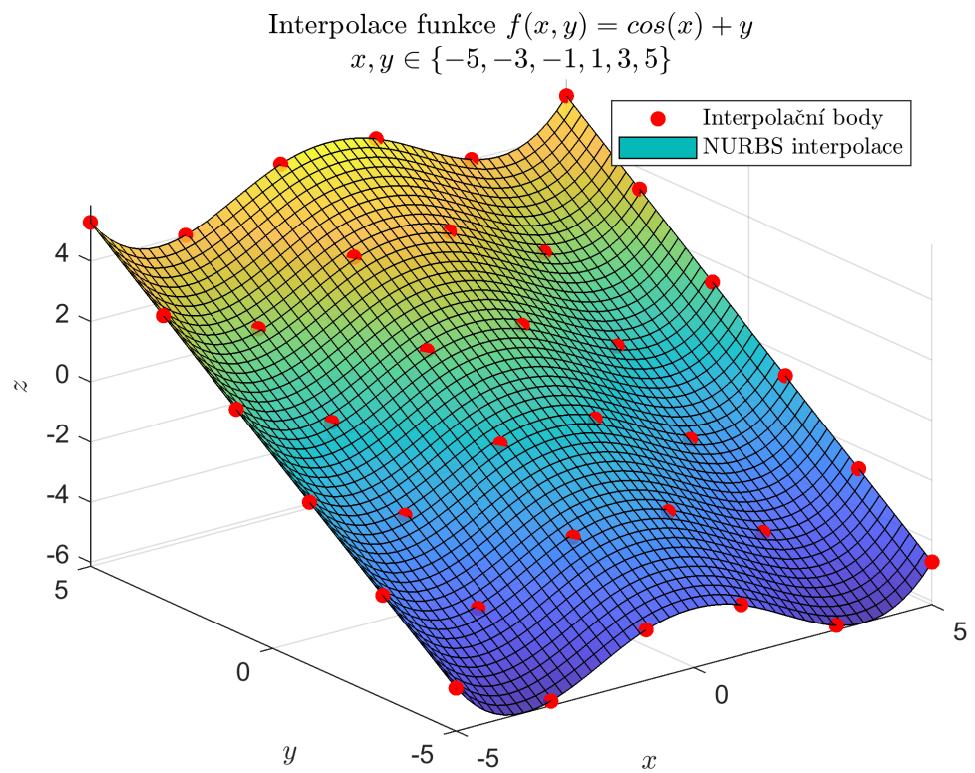
$$\mathbf{Q}_{k,l} = \sum_{i=0}^n N_{i,p}(\bar{u}_k) \left(\sum_{j=0}^m N_{j,q}(\bar{v}_l) \mathbf{P}_{i,j} \right) = \sum_{i=0}^n N_{i,p}(\bar{u}_k) \mathbf{R}_{i,l} \quad (32)$$

kde

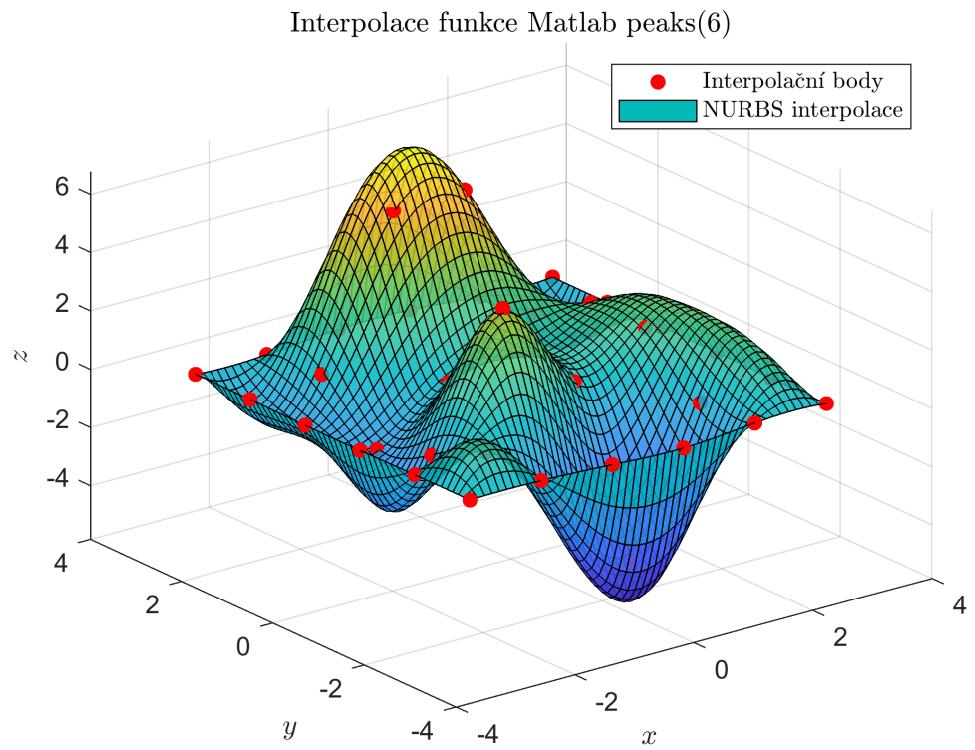
$$\mathbf{R}_{i,l} = \sum_{j=0}^m N_{j,q}(\bar{v}_l) \mathbf{P}_{i,j} \quad (33)$$

Tato metoda funguje pro libovolné pořadí směru interpolace křivek. Ukázka interpolace je na obrázcích č. 8, 9, 35b, 36b, 37b, 38b.

^[9]Prof. Dr. Les Piegl, *The NURBS Book*, kapitola 9.2.5.



Obrázek 8: Porovnání algoritmů interpolace ve 2D



Obrázek 9: Porovnání algoritmů interpolace ve 3D

3.4.1 Interpolace 4D nadpovrchu^[10]

Interpolací 4D nadpovrchu je zde myšleno vytvoření vektorové funkce tří skalárních proměnných např.: $\mathbf{S}(u, v, \psi)$. Uvažujme sadu 3D povrchů definovaných mřížkou bodů o stejné velikosti $i \times j$. Každý povrch uvažujeme na jiné souřadnici $w^{[11]}$. Přiřazením souřadnice w každému povrchu získají tuto souřadnici také všechny body těchto povrchů, tj. každý bod je tvořen souřadnicemi $[x, y, z, w]$.

Výpočet má následující kroky:

1. Pro konkrétní hodnoty i, j vybereme body $\mathbf{Q}_{i,j}$ ze všech rovin. Pro tuto sadu bodů provedeme interpolaci 4D křivky (viz [NURBS interpolace křivky](#)). Tímto získáme $i \times j$ 4D křivek $\mathbf{C}_{i,j}(u)$.
2. Pro libovolnou hodnotu w , $w_{min} \leq w \leq w_{max}$ najdeme body $\hat{\mathbf{Q}}_{i,j}$ na všech křivkách $\mathbf{C}_{i,j}(u)$ takové, že 4. souřadnice bodů $\hat{\mathbf{Q}}_{i,j}$ odpovídá zvolené hodnotě $w^{[12]}$.
3. Předchozím krokem jsme získali mřížku bodů $\hat{\mathbf{Q}}_{i,j}$ pro konkrétní hodnotu w . Mřížku stačí proložit povrchem pomocí [NURBS interpolace povrchu](#) a již dostáváme nadpovrch pro danou hodnotu souřadnice $w^{[13]}$.

Nadpovrh lze vizualizovat například pomocí animace jako průchod mezi klasickými povrhy prostřednictvím zbyvající proměnné w . Tento algoritmus je například vhodný v případě, že bychom chtěli vyplnit “prázdný” prostor souřadnice Ids mezi tabulkami [5a](#) a [5b](#).

Vizualizaci 4D povrchu, který je tvořen třemi 3D povrhy:

$$\mathbf{Q}_{i,j}(1) = [\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{W}] = [\mathbf{X}, \mathbf{Y}, \cos(\mathbf{X}) + \mathbf{Y}, 10] \quad (34)$$

$$\mathbf{Q}_{i,j}(2) = [\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{W}] = [\mathbf{X}, \mathbf{Y}, \cos(\mathbf{Y}) + \mathbf{X}, 20] \quad (35)$$

$$\mathbf{Q}_{i,j}(3) = [\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{W}] = [\mathbf{X}, \mathbf{Y}, \cos(2 \cdot \mathbf{Y}) + \cos(2 \cdot \mathbf{X}), 30] \quad (36)$$

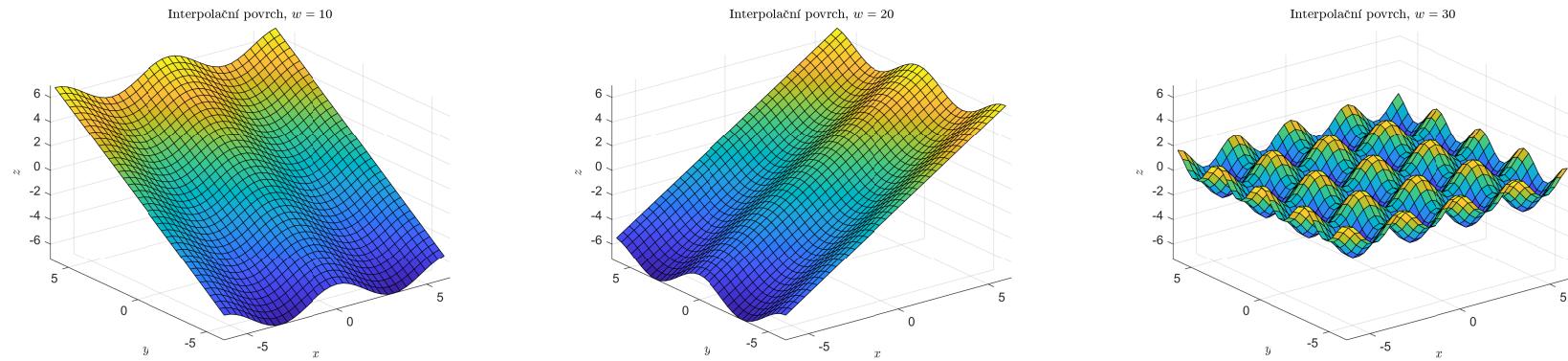
lze vidět na animaci na obrázku č. [11](#).

^[10]Téma této sekce již není diskuzí Prof. Dr. Les Piegl, *The NURBS Book*, tudíž řešení tohoto problému je již značně postaveno na mých vlastních nápadech realizace.

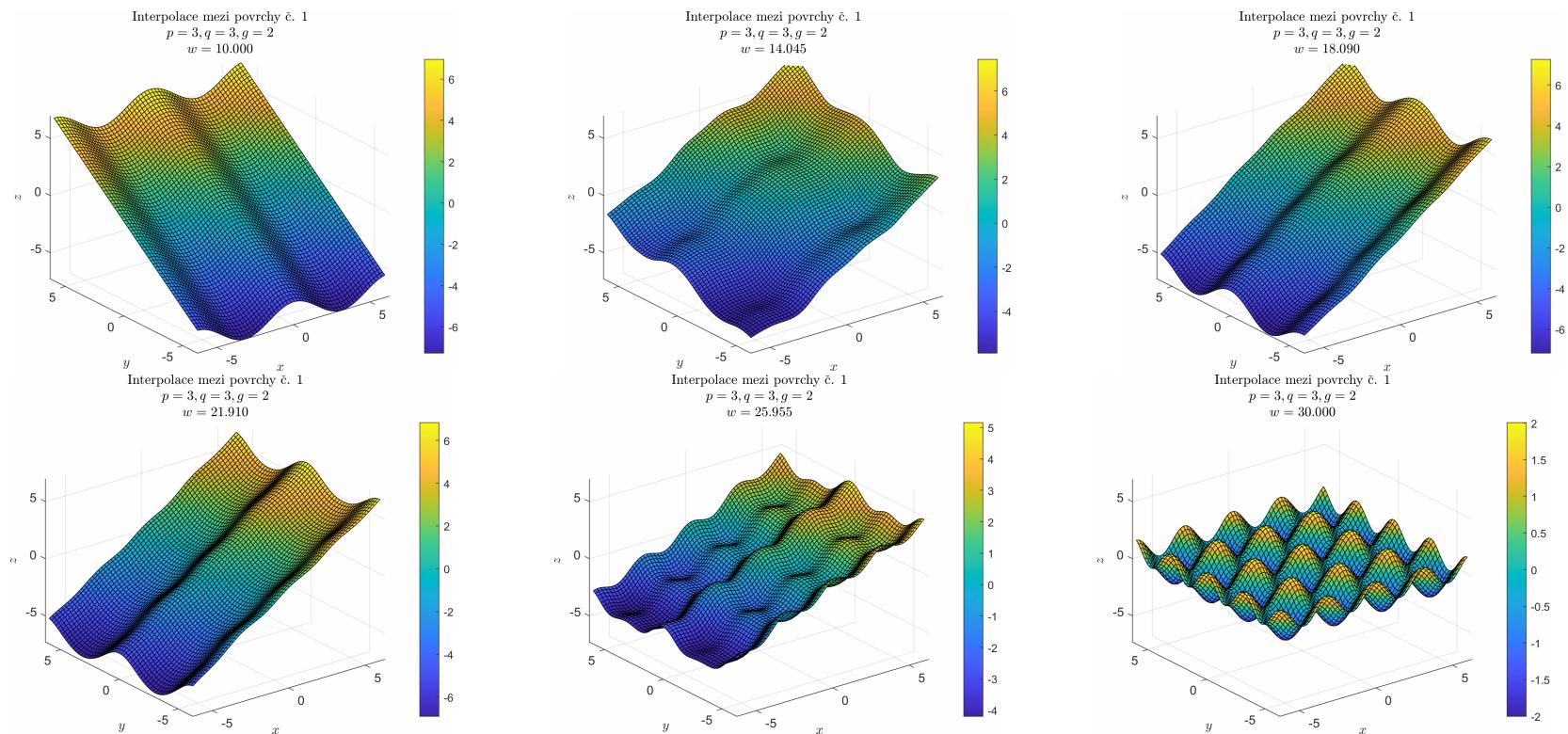
^[11]Souřadnici w si lze představit jako čas, tím pádem máme sadu 3D povrchů, kde se každý nachází v jakoby jiném časovém okamžiku.

^[12]Tento výpočet, pokud vím, nelze provést analyticky, existují pouze heuristické metody. Proto pro nalezení vhodné hodnoty s nějakou stanovenou přesností jsem použil metodu půlení intervalu, jelikož horní mez w_{max} a dolní mez w_{min} jsou známy.

^[13]Opakováním kroků [2](#) a [3](#) lze vypočítat zbylé potřebné povrhy s libovolným rozlišením souřadnice w .



Obrázek 10: Interpolaci povrchy pro 4D interpolaci č. 1



Obrázek 11: Ukázka průběhu interpolace mezi povrchy přes souřadnici w č. 1

3.5 NURBS aproximace křivky

Mějme množinu $m + 1$ bodů \mathbf{Q}_k $k = 0, \dots, m$ a hledáme křivku stupně p ve tvaru:

$$\mathbf{C}(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_i \quad u \in [0, 1], \quad n \geq p \geq 1 \quad (37)$$

pro kterou platí:

- krajní body interpoluje, tj.: $\mathbf{Q}_0 = \mathbf{C}(0)$, $\mathbf{Q}_m = \mathbf{C}(1)$
- zbytek bodů approximuje váženou metodou nejmenších čtverců:

$$\mathbf{P} = (\mathbf{N}^T \mathbf{W} \mathbf{N})^{-1} \mathbf{R} \quad (38)$$

kde

- \mathbf{N} je matice $(m - 1) \times (n - 1)$:

$$\mathbf{N} = \begin{bmatrix} N_{1,p}(\bar{u}_1) & \cdots & N_{n-1,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N_{1,p}(\bar{u}_{m-1}) & \cdots & N_{n-1,p}(\bar{u}_{m-1}) \end{bmatrix} \quad (39)$$

- \mathbf{R} je matice $(n - 1) \times (\dim \mathbf{Q})$:

$$\mathbf{R} = \begin{bmatrix} N_{1,p}(\bar{u}_1)\omega_1 \mathbf{R}_1 + \cdots + N_{1,p}(\bar{u}_{m-1})\omega_{m-1} \mathbf{R}_{m-1} \\ \vdots \\ N_{n-1,p}(\bar{u}_1)\omega_1 \mathbf{R}_1 + \cdots + N_{n-1,p}(\bar{u}_{m-1})\omega_{m-1} \mathbf{R}_{m-1} \end{bmatrix} \quad (40)$$

kde

$$\mathbf{R}_k = \mathbf{Q}_k - N_{0,p}(\bar{u}_k) \mathbf{Q}_0 - N_{n,p}(\bar{u}_k) \mathbf{Q}_m \quad k = 1, \dots, m - 1 \quad (41)$$

- \mathbf{W} je diagonální matice $(m - 1) \times (m - 1)$ ^[14] s váhami jednotlivých bodů:

$$\mathbf{W} = \begin{bmatrix} \omega_1 & 0 & \cdots & 0 \\ 0 & \omega_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \omega_{m-1} \end{bmatrix} \quad (42)$$

^[14]Váhy krajních bodů nehrají roli, protože tyto body jsou interpolovány

- \mathbf{P} je matice hledaných řídících bodů $(n+1) \times (\dim \mathbf{Q})$

Opět musíme vhodně zvolit parametry \bar{u}_k a uzlový vektor \mathbf{U} . Parametr \bar{u} můžeme vypočítat pomocí předpisu 20.

Prvky uzlového vektoru poté vypočteme následovně:

$$d = \frac{m+1}{n-p+1} \quad (43)$$

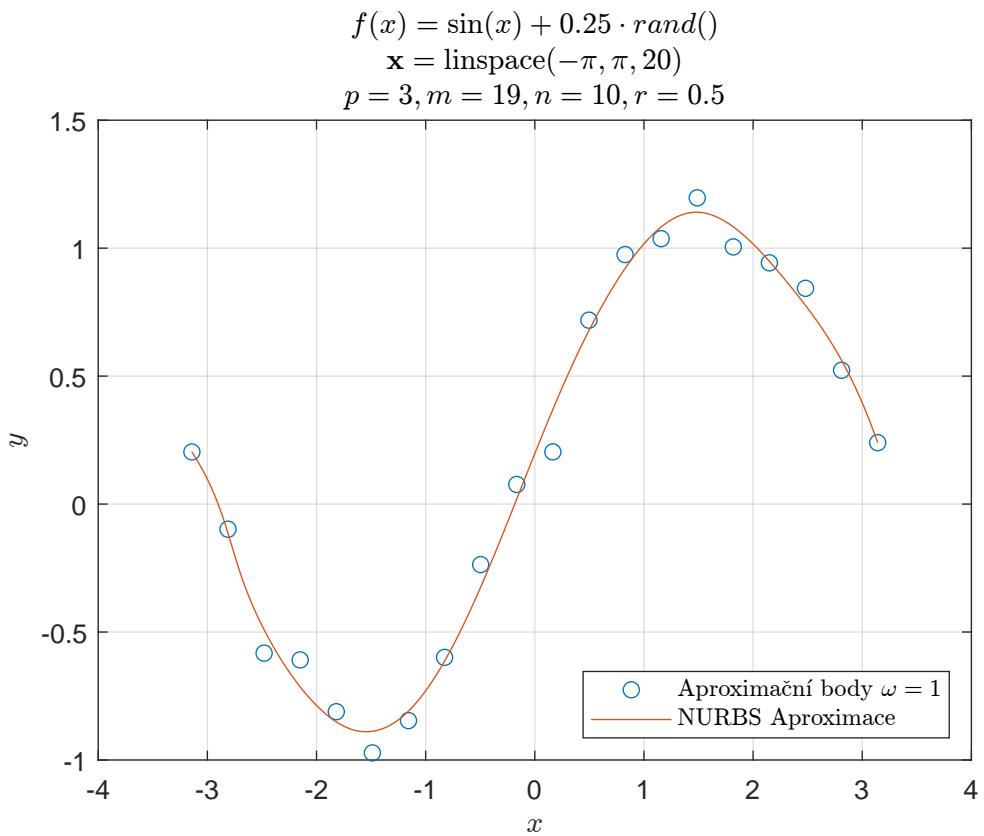
$$i = \lfloor j \cdot d \rfloor \quad \alpha = j \cdot d - i \quad (44)$$

$$u_{p+j} = (1-\alpha) \cdot \bar{u}_{i-1} + \alpha \cdot \bar{u}_i \quad j = 1, \dots, n-p \quad (45)$$

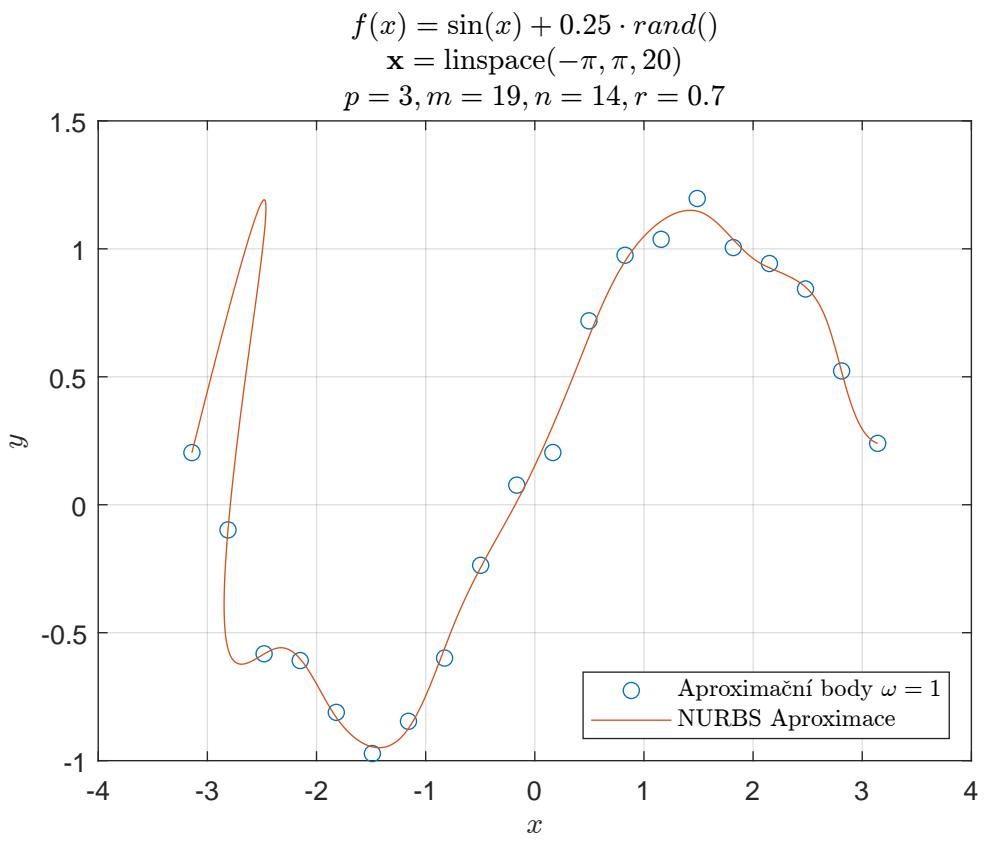
$$u_0 = \dots = u_p = 0 \quad u_{n+1} = \dots = u_{n+p+1} = 1 \quad (46)$$

Takto navržená approximace je citlivá na poměr počtu řídících bodů a počtu approximačních bodů tj.: $r = \frac{n}{m+1}$. Jak se r blíží k 1, tím je větší šance vzniku velkých kmitů, obzvlášt v případě zašuměných dat — viz obrázky č. 12, 13, 14, 15.

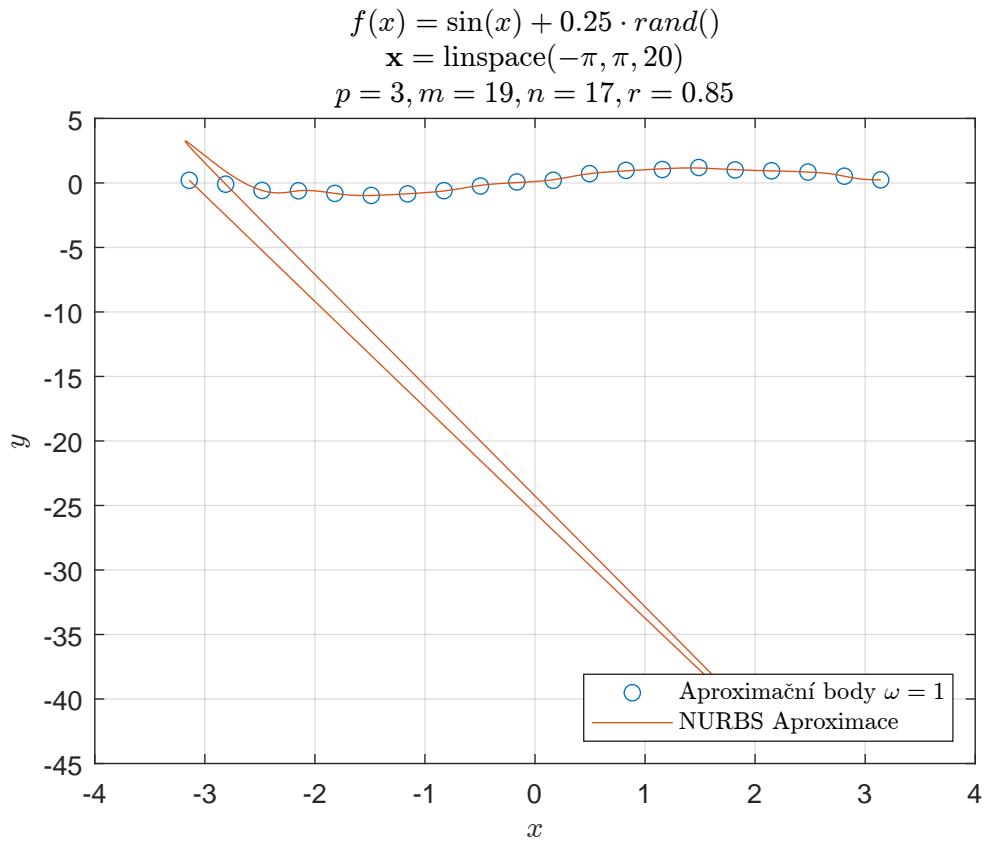
Ideálním případem pro tento algoritmus je když platí $m \gg n$ — viz obrázek č. 16. Obrázky č. 17, 18, 19 dále ukazují chování algoritmu pro různé váhy bodů.



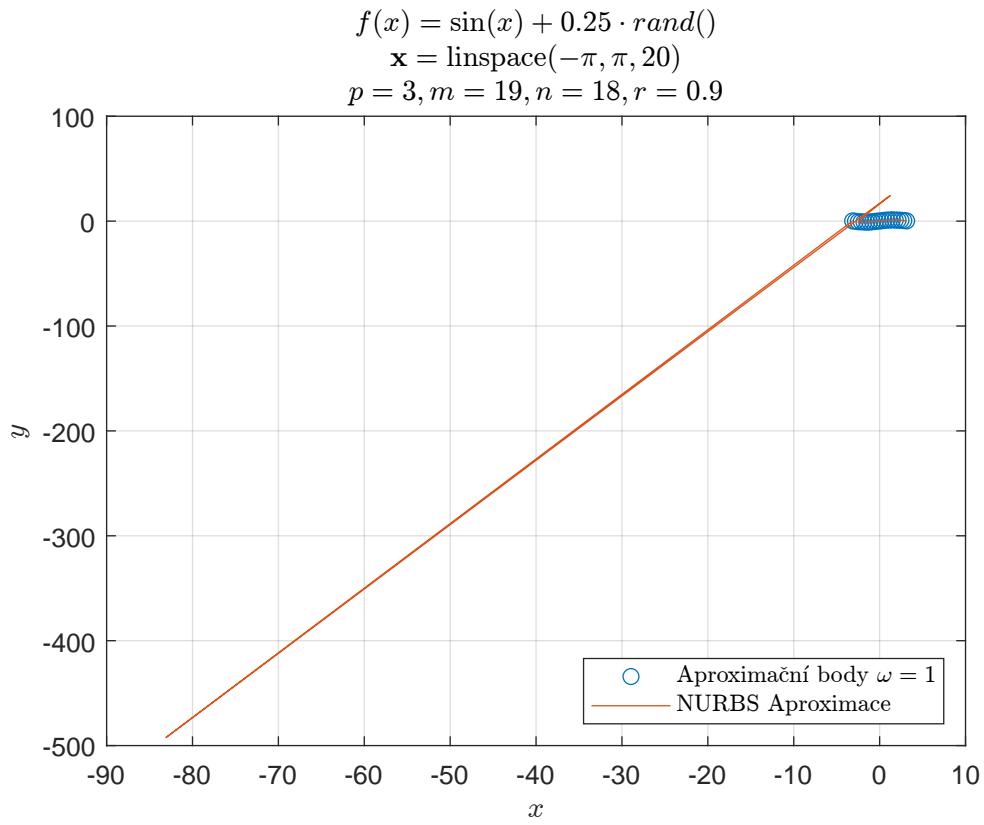
Obrázek 12: Aproximace křivky metodou nejmenších čtverců pro různé parametry



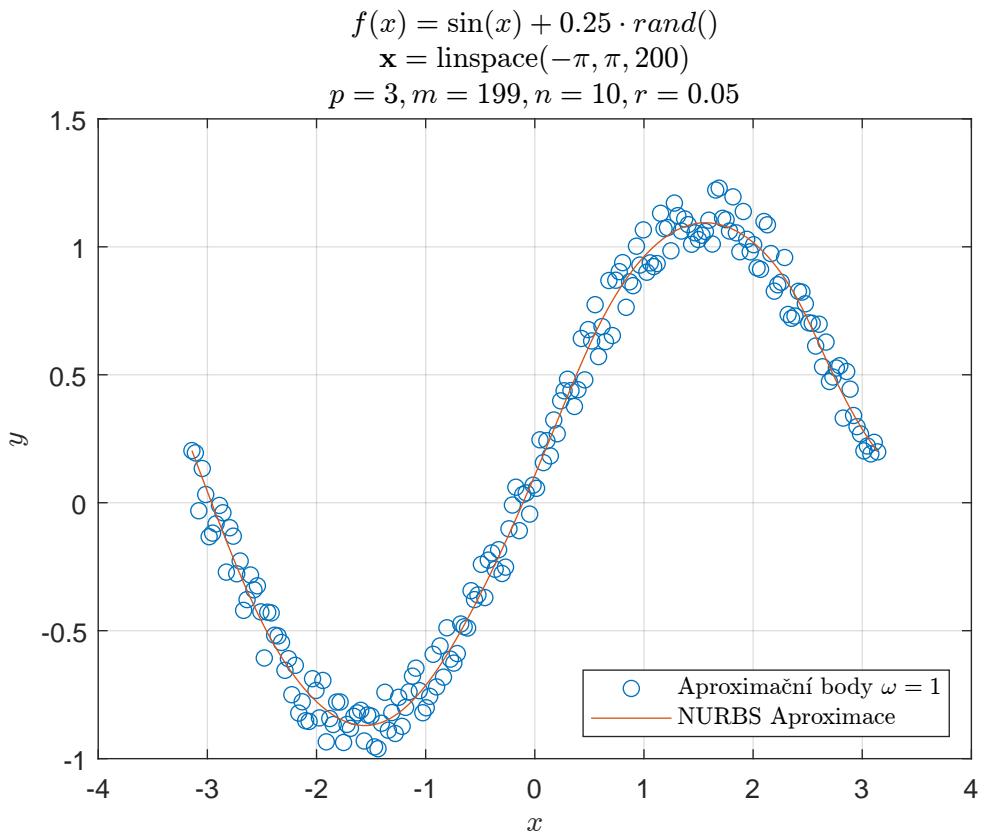
Obrázek 13: Aproximace křivky metodou nejmenších čtverců pro různé parametry



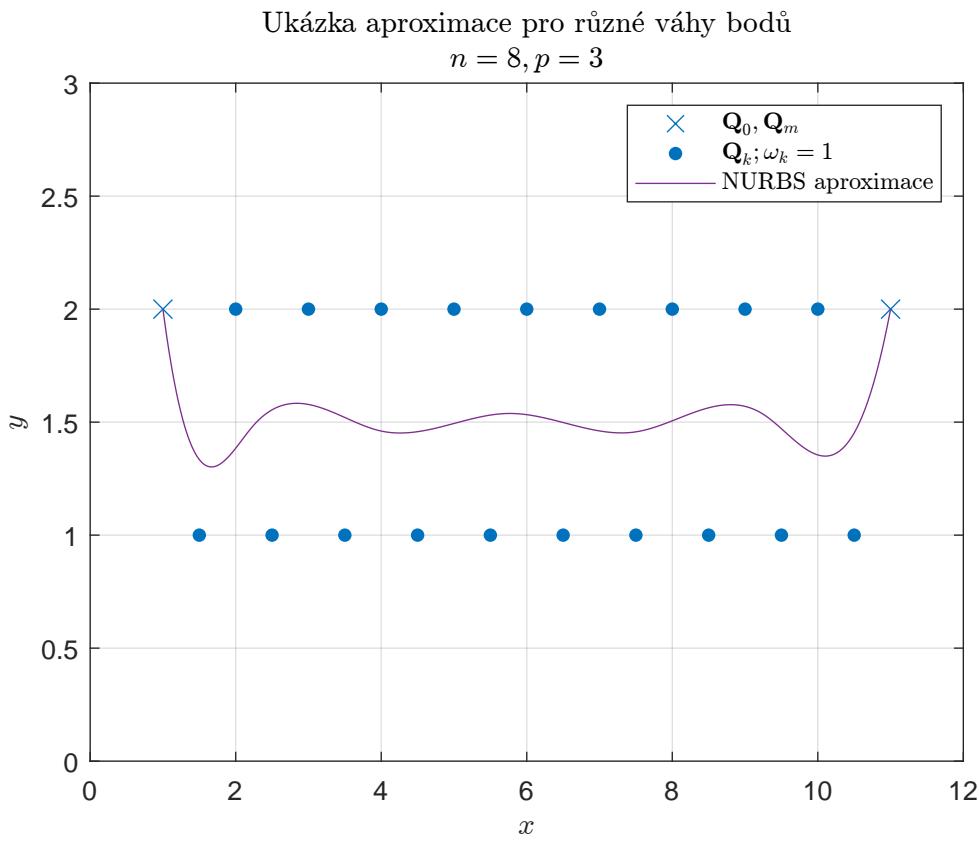
Obrázek 14: Aproximace křivky metodou nejmenších čtverců pro různé parametry



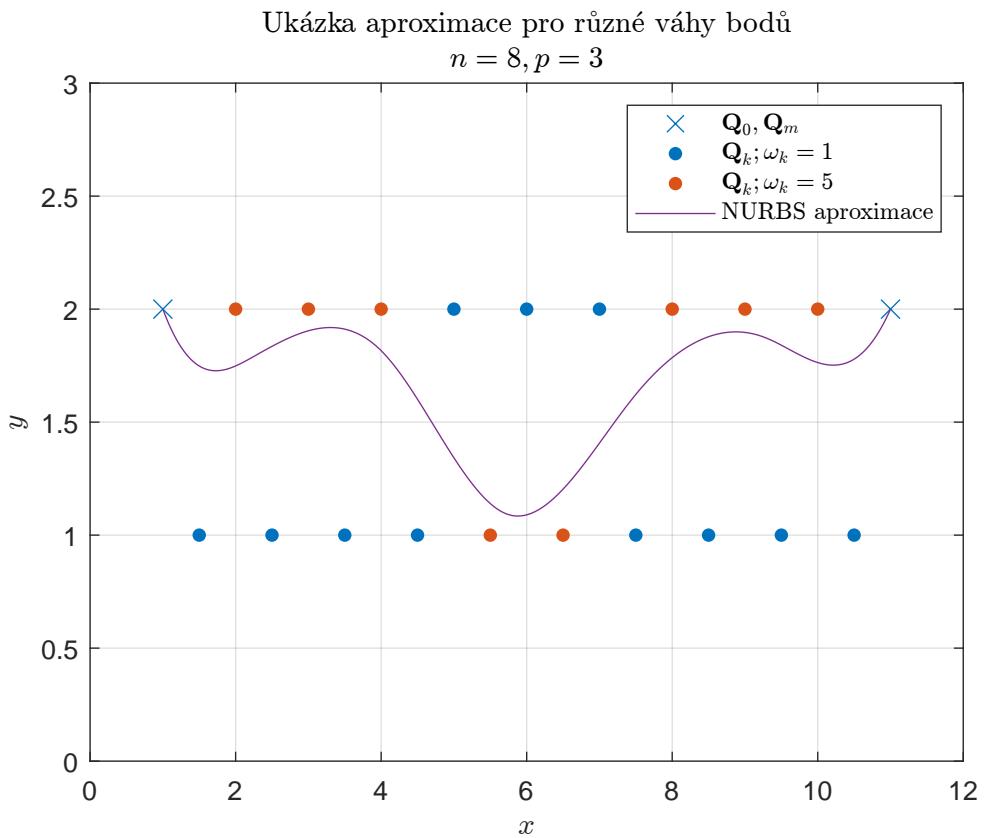
Obrázek 15: Aproximace křivky metodou nejmenších čtverců pro různé parametry



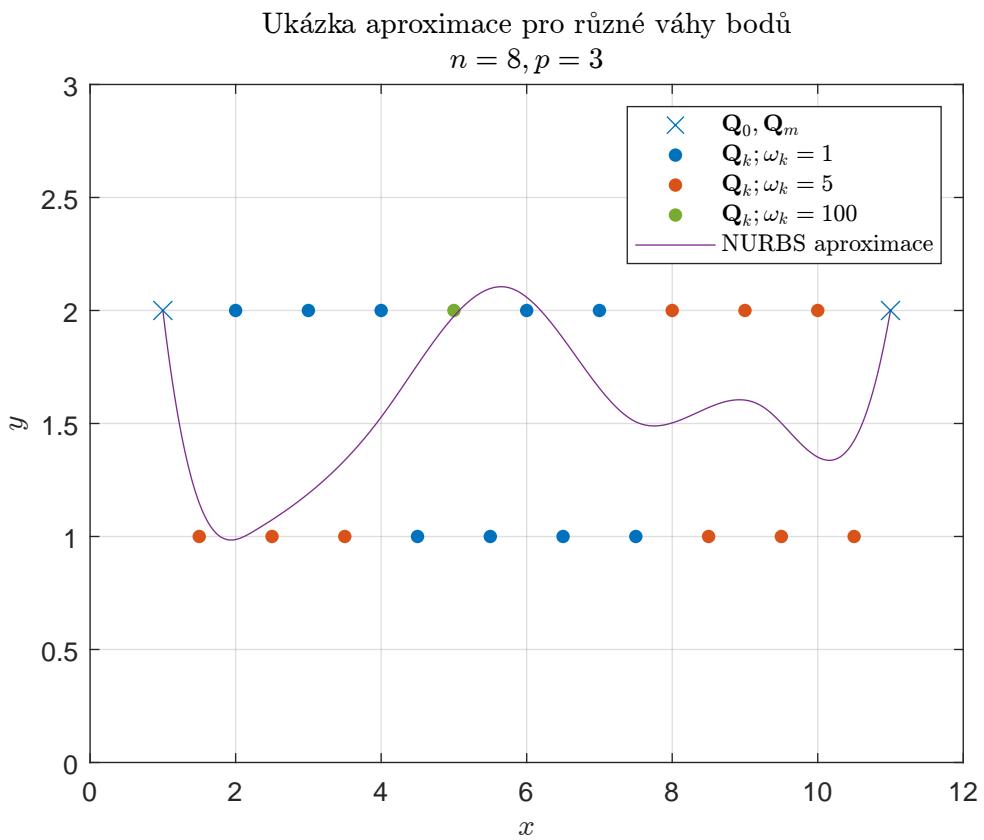
Obrázek 16: Aproximace křivky metodou nejmenších čtverců pro různé parametry



Obrázek 17: Aproximace křivky metodou nejmenších čtverců pro různé váhy bodů



Obrázek 18: Aproximace křivky metodou nejmenších čtverců pro různé váhy bodů



Obrázek 19: Aproximace křivky metodou nejmenších čtverců pro různé váhy bodů

3.5.1 Cílená NURBS approximace křivky

Pro naše využití bychom v mnoha případech chtěli approximovat pouze část křivky a zbytek interpolovat, proto zavedeme tzv. cílenou approximaci. Mějme množinu $m+1$ bodů \mathbf{Q}_k $k=0, \dots, m$ a approximační interval $\mathbf{I}_{ap} \subset \mathbf{Q}_k$ o délce l .

Na tomto intervalu \mathbf{I}_{ap} provedeme approximaci ze sekce 3.5 s n řídícími body, $n \leq l-2$. Tímto získáme křivku, ze které vybereme několik vhodných (např. $2n$ ekvidistantních) bodů \mathbf{Q}_{ap} . Interval \mathbf{I}_{ap} v \mathbf{Q}_k nahradíme body \mathbf{Q}_{ap} a provedeme interpolaci.

Tímto se i vhodně vyhneme problému, který u approximace nastává v případě $n \sim l$, protože si lze snadno ohlédat počet řídících bodů n k počtu approximačních bodů l .

3.6 NURBS approximace povrchu

Stejně jako u interpolace počítáme s množinou $(r+1) \times (s+1)$ bodů $\mathbf{Q}_{k,l}$, $k=0, \dots, r$ a $l=0, \dots, s$ ležících v mřížce, které chceme approximovat pomocí vážené metody nejmenších čtverců NURBS povrchem stupně p a q ve tvaru:

$$S(\bar{u}_k, \bar{v}_l) = \sum_{i=0}^r \sum_{j=0}^s N_{i,p}(\bar{u}_k) N_{j,q}(\bar{v}_l) \mathbf{P}_{i,j} \quad (47)$$

Tento algoritmus interpoluje všechny rohové body a zbytek approximuje, tj. approximovaná plocha je tvořena body:

$$\begin{array}{ccccccccc} \mathbf{Q}_{0,0} & \hat{\mathbf{Q}}_{0,1} & \cdots & \hat{\mathbf{Q}}_{0,s-1} & \mathbf{Q}_{0,s} & & & \\ \hat{\mathbf{Q}}_{1,0} & \ddots & & & \hat{\mathbf{Q}}_{1,s} & & & \\ \vdots & & \ddots & & \vdots & & & \\ & & & \ddots & & \hat{\mathbf{Q}}_{r-1,s} & & \\ \hat{\mathbf{Q}}_{r-1,0} & & & & & & & \\ \mathbf{Q}_{r,0} & \hat{\mathbf{Q}}_{r,1} & \cdots & \hat{\mathbf{Q}}_{r,s-1} & \mathbf{Q}_{r,s} & & & \end{array} \quad (48)$$

kde $\hat{\mathbf{Q}}_{i,j}$ je approximovaný body $\mathbf{Q}_{i,j}$.

Stejně jako u interpolace povrchu budeme nejprve approximovat křivky v jednom směru a poté ve druhém. Parametry \bar{u}_k a \bar{v}_l a uzlové vektory \mathbf{U} a \mathbf{V} získáme stejně jako u interpolace povrchu - viz 3.4. Nejprve approximujeme body ve směru u , tím získáme

tenzor řídících bodů aproximačních křivek \mathbf{T} o rozměrech $(n+1) \times (s+1) \times (\dim \mathbf{Q})$:

$$\mathbf{T}_{0,i} = \mathbf{Q}_{0,i} \quad i = 0, \dots, s \quad (49)$$

$$\mathbf{T}_{n,i} = \mathbf{Q}_{r,i} \quad i = 0, \dots, s \quad (50)$$

$$\mathbf{T}_{1\dots n-1,i} = (\mathbf{N}_u^T \mathbf{W}_u(i) \mathbf{N}_u)^{-1} \mathbf{R}_u(i) \quad i = 0, \dots, s \quad (51)$$

kde:

- \mathbf{N}_u je matice $(r-1) \times (n-1)$:

$$\mathbf{N}_u = \begin{bmatrix} N_{1,p}(\bar{u}_1) & \cdots & N_{n-1,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N_{1,p}(\bar{u}_{r-1}) & \cdots & N_{n-1,p}(\bar{u}_{r-1}) \end{bmatrix} \quad (52)$$

- $\mathbf{W}_u(i)$ je matice $(r+1) \times (s+1)$ obsahující váhy bodů $\mathbf{Q}_{0\dots r,0\dots,s}$:

$$\mathbf{W}_u(i) = \begin{bmatrix} \omega_0^u(i) & 0 & \cdots & 0 \\ 0 & \omega_1^u(i) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \omega_r^u(i) \end{bmatrix} = \text{diag}(\mathbf{W}_{0\dots r,i}) \quad (53)$$

kde matice \mathbf{W} obsahuje váhy jednotlivých bodů, tj. $\mathbf{W}_{i,j}$ odpovídá váze bodu $\mathbf{Q}_{i,j}$ (váhy rohových bodů nehrají roli).

- $\mathbf{R}_u(i)$ je matice $(r-1) \times (\dim \mathbf{Q})$:

$$\mathbf{R}_u(i) = \begin{bmatrix} N_{1,p}(\bar{u}_1)\omega_1^u(i)\mathbf{R}_1^u(i) + \cdots + N_{1,p}(\bar{u}_{r-1})\omega_{r-1}^u(i)\mathbf{R}_{r-1}^u(i) \\ \vdots \\ N_{n-1,p}(\bar{u}_1)\omega_1^u(i)\mathbf{R}_1^u(i) + \cdots + N_{n-1,p}(\bar{u}_{r-1})\omega_{r-1}^u(i)\mathbf{R}_{r-1}^u(i) \end{bmatrix} \quad (54)$$

kde

$$\mathbf{R}_k^u(i) = \mathbf{Q}_{k,i} - N_{0,p}(\bar{u}_k)\mathbf{Q}_{0,i} - N_{n,p}(\bar{u}_k)\mathbf{Q}_{r,i} \quad k = 1, \dots, r-1 \quad (55)$$

Nyní stačí vypočítat řídící body aproximovaného povrchu \mathbf{P} o stejných rozměrech jako je \mathbf{T} . Postup je analogický ke směru u , kde jako nové aproximační body použijeme \mathbf{T} , tj.:

$$\mathbf{P}_{i,0} = \mathbf{T}_{i,0} \quad i = 0, \dots, n \quad (56)$$

$$\mathbf{P}_{i,m} = \mathbf{T}_{i,s} \quad i = 0, \dots, n \quad (57)$$

$$\mathbf{P}_{i,1\dots m-1,i} = (\mathbf{N}_v^T \mathbf{W}_v(i) \mathbf{N}_v)^{-1} \mathbf{R}_v(i) \quad i = 0, \dots, n \quad (58)$$

kde:

- \mathbf{N}_v je matice $(s - 1) \times (m - 1)$:

$$\mathbf{N}_u = \begin{bmatrix} N_{1,q}(\bar{v}_1) & \cdots & N_{m-1,q}(\bar{v}_1) \\ \vdots & \ddots & \vdots \\ N_{1,q}(\bar{v}_{s-1}) & \cdots & N_{m-1,q}(\bar{v}_{s-1}) \end{bmatrix} \quad (59)$$

- $\mathbf{W}_v(i)$ je diagonální matice $(n + 1) \times (s + 1)$ obsahující váhy bodů $\mathbf{T}_{0\dots n, 0\dots s}$:

$$\mathbf{W}_v(i) = \begin{bmatrix} \omega_0^v(i) & 0 & \cdots & 0 \\ 0 & \omega_1^v(i) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \omega_s^v(i) \end{bmatrix} = \text{diag}(\mathbf{W}_{i,0\dots s}^V) \quad (60)$$

kde matice \mathbf{W}^V je přeskálovaná matice \mathbf{W} na velikost $(n + 1) \times (s - 1)$, stejným způsobem jako se škáluje obrázek, tj.:

$$\mathbf{W}^V = \text{imresize}(\mathbf{W}, [n + 1, s + 1]) \quad (61)$$

Po provedení approximace ve směru u máme pro approximaci ve směru v , pouze $(n + 1) \times (s + 1)$ bodů (namísto původních $(r + 1) \times (s + 1)$), proto je tato redukce nutná. Důležité je, že výsledné prvky matice v sobě nějakým způsobem nesou váhy pro matici původní velikosti. Funkce `imresize` může produkovat záporné hodnoty, ty ale stačí nahradit například výchozí hodnotou 1.

- $\mathbf{R}_v(i)$ je matice $(s - 1) \times (\dim \mathbf{Q})$:

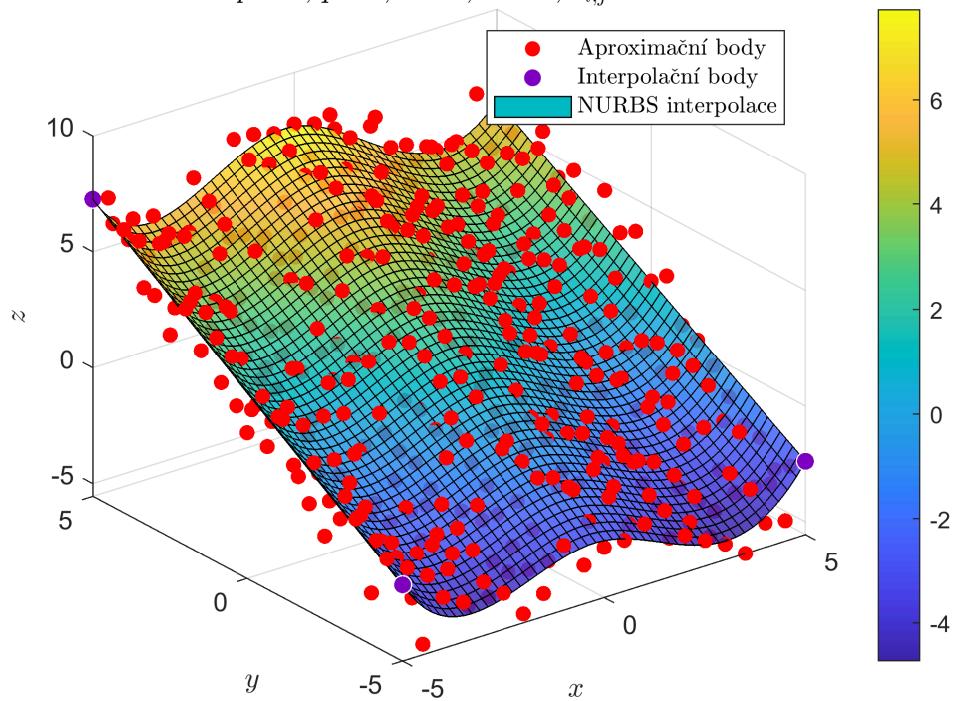
$$\mathbf{R}_v(i) = \begin{bmatrix} N_{1,q}(\bar{v}_1)\omega_1^v(i)\mathbf{R}_1^v(i) + \cdots + N_{1,q}(\bar{v}_{s-1})\omega_{s-1}^v\mathbf{R}_{s-1}^v(i) \\ \vdots \\ N_{m-1,q}(\bar{v}_1)\omega_1^v\mathbf{R}_1^v(i) + \cdots + N_{m-1,q}(\bar{v}_{s-1})\omega_{s-1}^v\mathbf{R}_{s-1}^v(i) \end{bmatrix} \quad (62)$$

kde

$$\mathbf{R}_l^v(i) = \mathbf{T}_{i,l} - N_{0,q}(\bar{v}_l)\mathbf{T}_{i,0} - N_{m,q}(\bar{v}_l)\mathbf{T}_{i,s} \quad l = 1, \dots, s - 1 \quad (63)$$

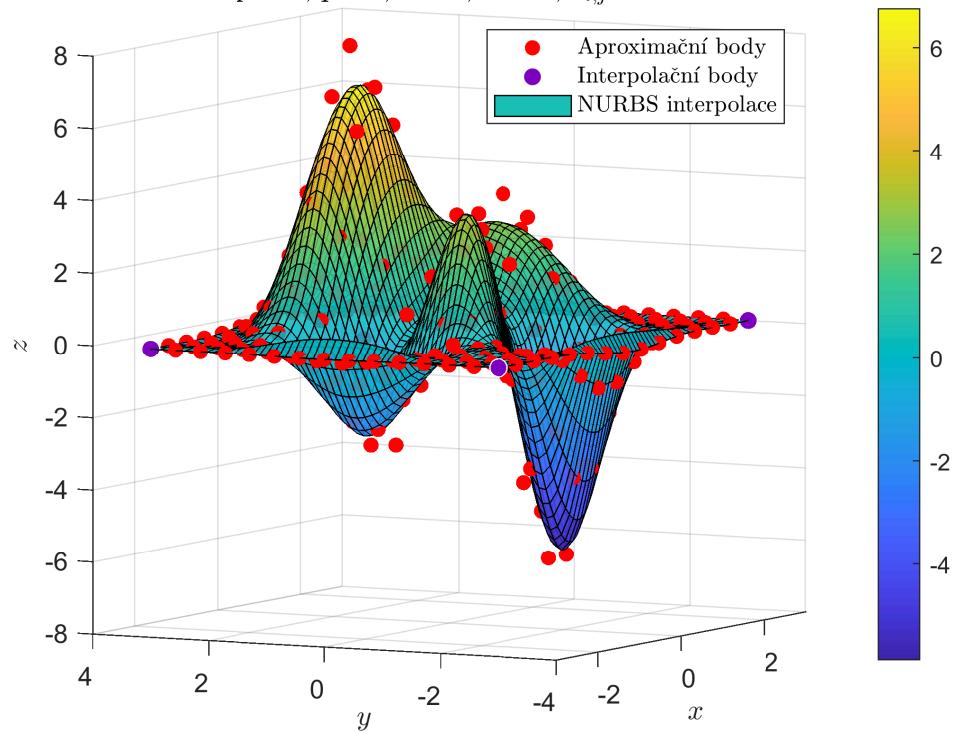
Ukázka různých approximací jsou na obrázcích č. 20, 21, 22, 23, 24 a 25.

Aproximace funkce $f(x, y) = \cos(x) + y + 3 \cdot \text{rand}()$
 $p = 1, q = 3, n = 1, m = 4, \omega_{i,j} = 1$



Obrázek 20: Aproximace povrchu metodou nejmenších čtverců č. 1

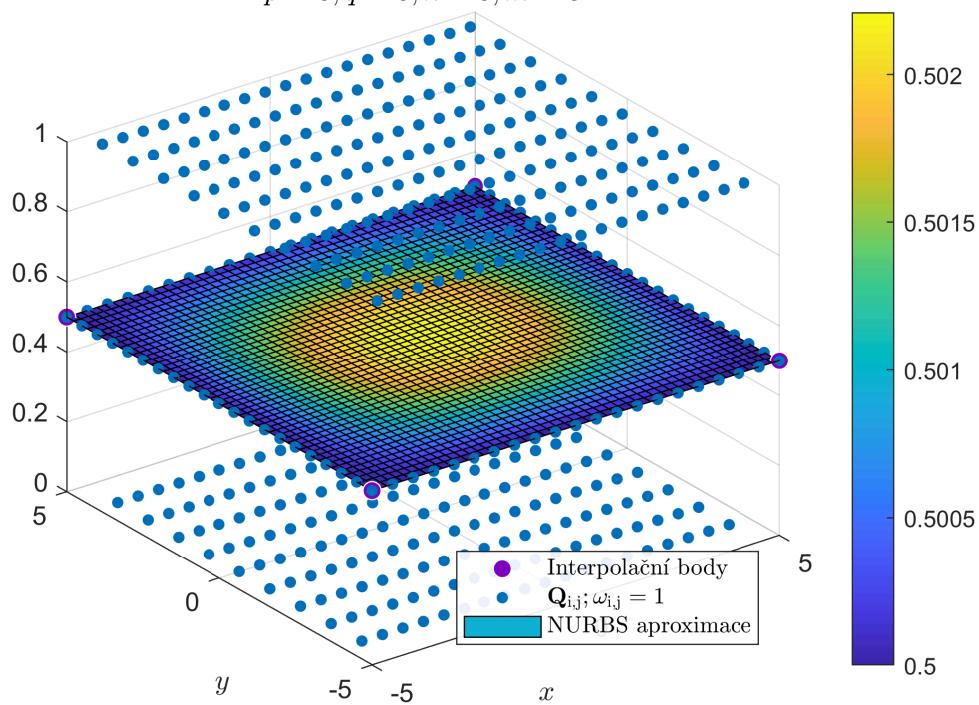
Aproximace funkce Matlab peaks(15)
 $p = 3, q = 3, n = 7, m = 7, \omega_{i,j} = 1$



Obrázek 21: Aproximace povrchu metodou nejmenších čtverců č. 2

Ukázka approximace povrchu pro různé váhy bodů

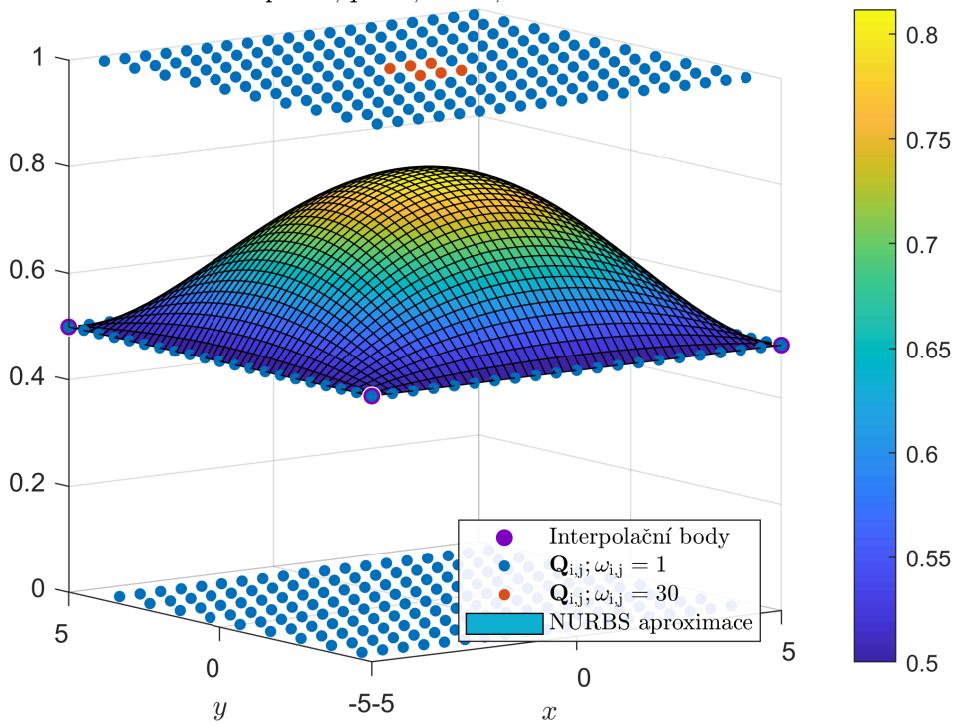
$$p = 3, q = 3, n = 3, m = 3$$



Obrázek 22: Aproximace povrchu pro různé váhy č. 1

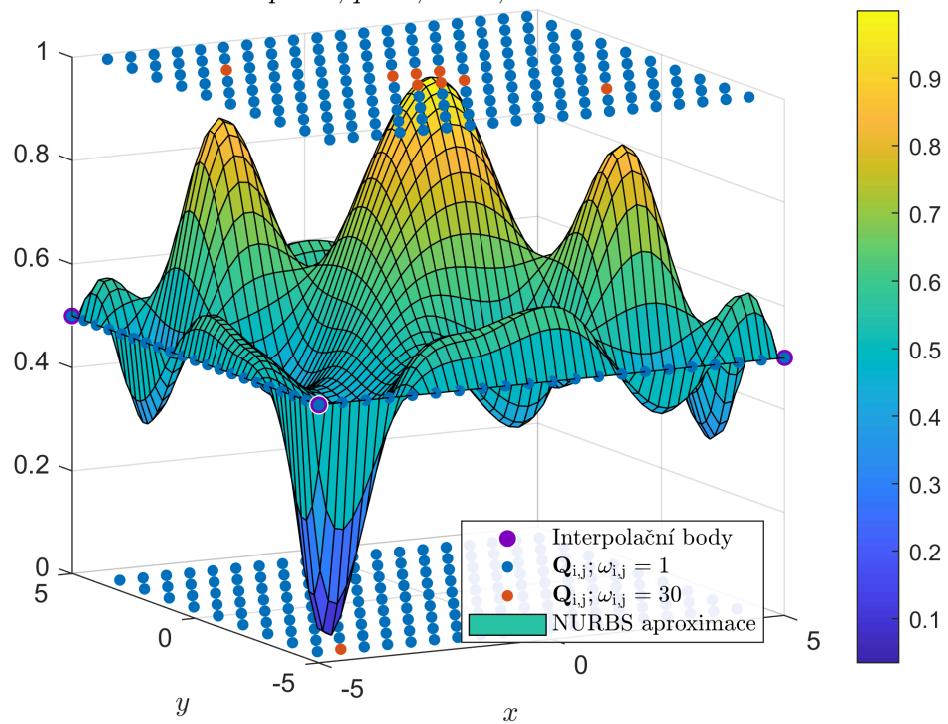
Ukázka approximace povrchu pro různé váhy bodů

$$p = 3, q = 3, n = 3, m = 3$$



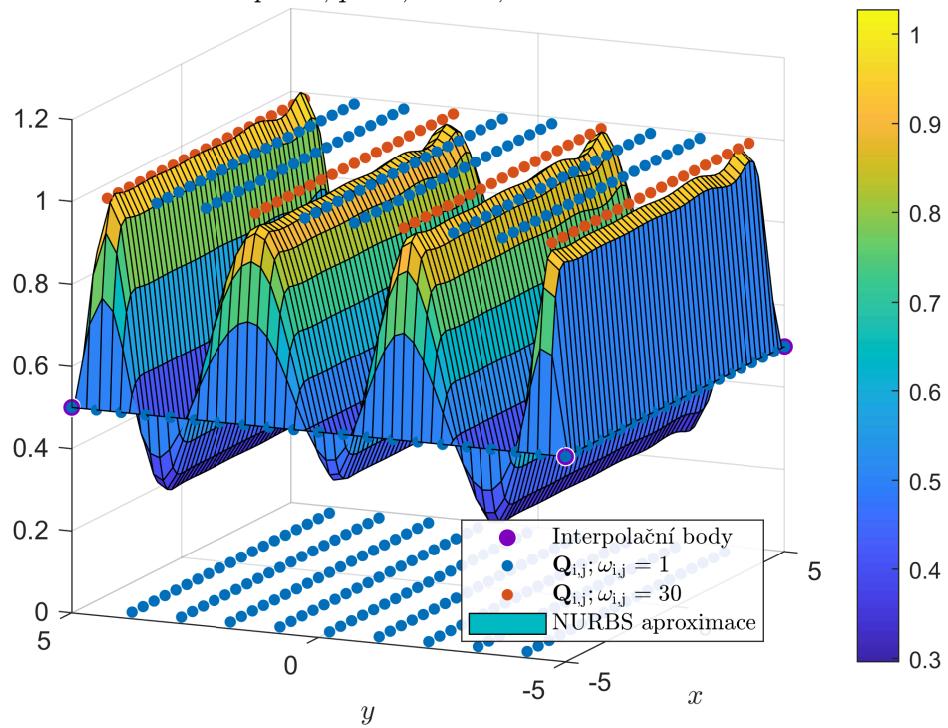
Obrázek 23: Aproximace povrchu pro různé váhy č. 2

Ukázka aproximace povrchu pro různé váhy bodů
 $p = 3, q = 3, n = 7, m = 7$



Obrázek 24: Aproximace povrchu pro různé váhy č. 3

Ukázka aproximace povrchu pro různé váhy bodů
 $p = 3, q = 3, n = 10, m = 10$



Obrázek 25: Aproximace povrchu pro různé váhy č. 4

3.6.1 NURBS aproximace povrchu s užitím Gaussovy funkce

Metoda v sekci [NURBS aproximace povrchu](#) vyžaduje, aby approximační body ležely v mřížce (3D i 4D případ), a proto není vhodná pro účely této práce. Proto jsem přišel s následující metodou, která sice není postavena na žádném matematickém kritériu (jako například nejmenší čtverce v předchozích aproximacích), ale nevyžaduje, aby approximační body ležely v mřížce, a poskytuje postačující výsledky.

Metoda využívá váženého průměrování a Gaussovy funkce $f(x, y)$:

$$f(x, y) = \exp \left(- \left(\frac{(x - x_0)^2}{2\sigma_x^2} + \frac{(y - y_0)^2}{2\sigma_y^2} \right) \right) \quad (64)$$

kde

- x_0 – střední hodnota osy x
- y_0 – střední hodnota osy y
- σ_x – variance osy x
- σ_y – variance osy y

Užitím Gaussovy funkce je zajištěno, že váha approximovaného bodu se vzdáleností klesá k nule (viz obrázky č. [26](#) a [29](#)), díky tomu je možné provádět vážený průměr přes všechny body^[15].

Algoritmus vyžaduje libovolnou množinu původních bodů \mathbf{Q}_i , množinu approximačních bodů $\bar{\mathbf{Q}}_j$ s množinou příslušných vah ω_j a již zmíněné parametry σ_x , σ_y ^[16]. Kroky algoritmu jsou následující:

1. Zvol další approximační bod $\bar{\mathbf{Q}} = [x_{ap}, y_{ap}, z_{ap}]$ a jeho váhu w
2. Projdi všechny body $\mathbf{Q}_i = [x, y, z]$ a pro každý proved:

$$(a) \bar{\omega} = \omega \cdot f(x, y) \quad x_0 = x_{ap}, \quad y_0 = y_{ap}$$

(Výpočet upravené váhy na základě vzdálenosti od approximovaného bodu)

$$(b) \bar{z} = \frac{z + z_{ap} \cdot \bar{\omega}}{\bar{\omega} + 1}$$

(Výpočet nové hodnoty z na základě váženého průměru)

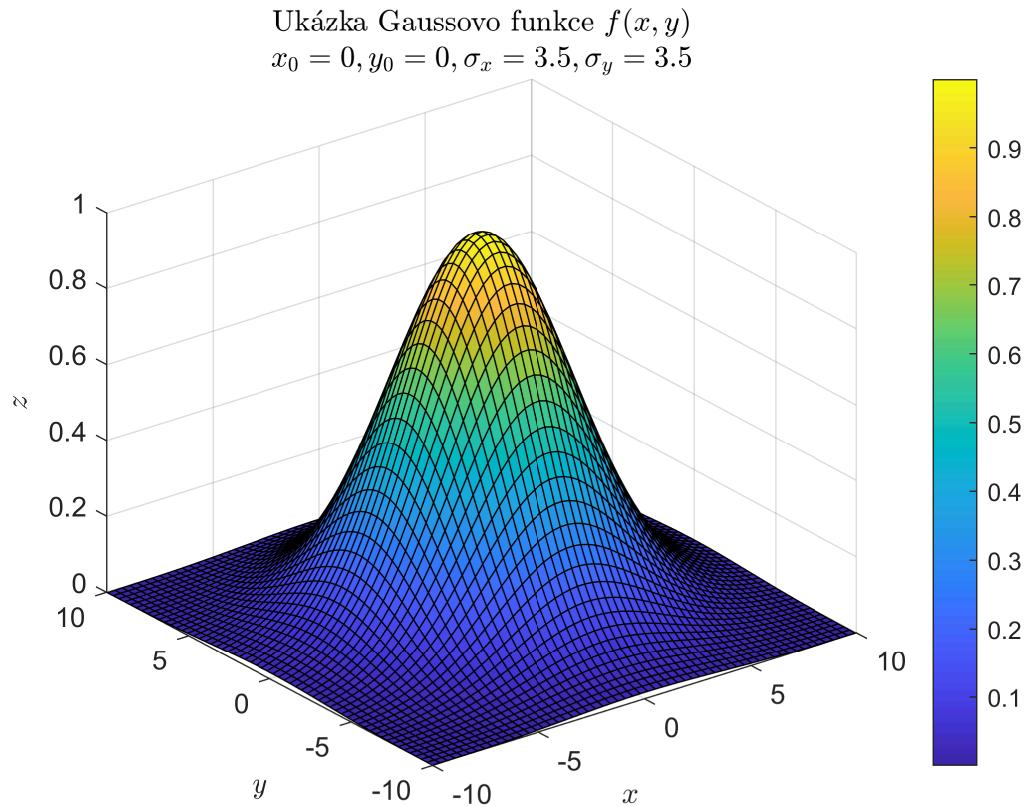
^[15]V případě nadměrného množství bodů lze průměrování omezit na okolí approximovaného bodu pro snížení výpočetní náročnosti.

^[16]V případě potřeby algoritmus lze snadno upravit tak, že parametry σ_x a σ_y mohou být zadány individuálně pro každý approximační bod $\bar{\mathbf{Q}}_j$

(c) Nahrad z za \bar{z}

3. Vrat se na krok 1, pokud si neprošel všechny approximační body, jinak konec

Pomocí parametrů σ_x, σ_y lze upřednostnit approximaci v ose x nebo y . Tento efekt lze vidět na obrázcích č. [27a](#), [27b](#), [27c](#), [27d](#), [28a](#), [28b](#), [28c](#), [28d](#), kde je obsažena i samotná ukázka takto navržené approximace^[17].

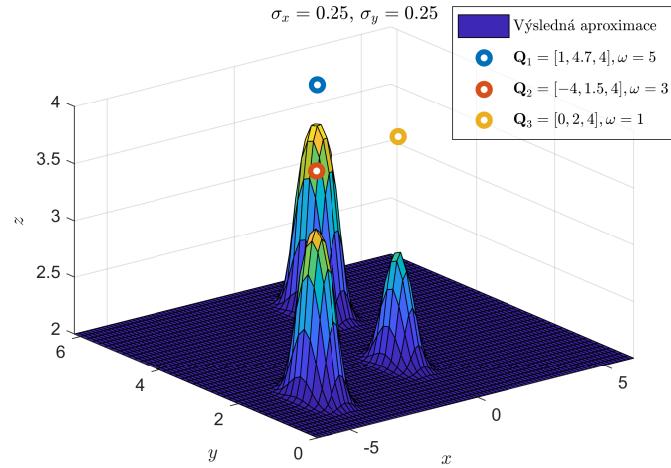


Obrázek 26: Ukázka Gaussovo funkce $f(x, y)$

^[17]Přestože ukázky obsahují povrchy tvořeny body v mřížce, není to vyžadováno výpočetním algoritmem.

Aproximace pro povrch $Z = 2$ s užitím Gaussovo funkce

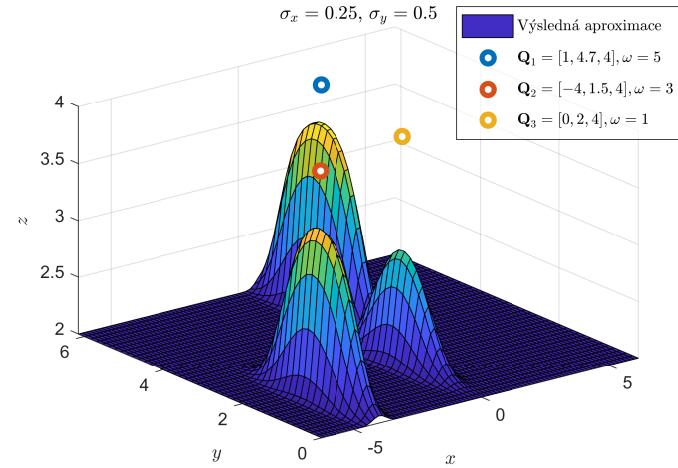
Aproximační body: Q_1, Q_2, Q_3



(a) Aproximace pro různé parametry Gaussovo funkce č. 1

Aproximace pro povrch $Z = 2$ s užitím Gaussovo funkce

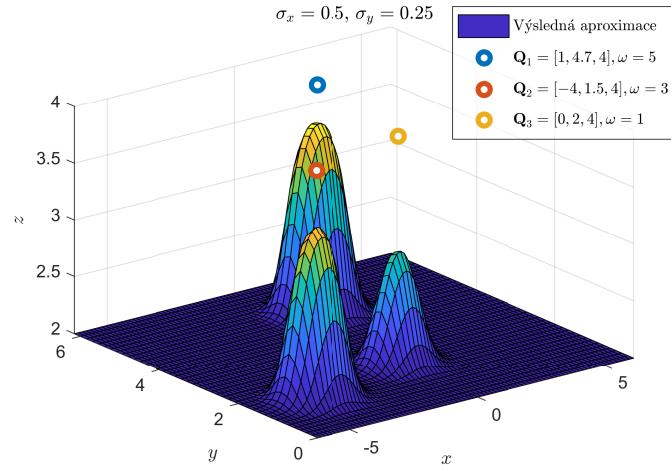
Aproximační body: Q_1, Q_2, Q_3



(b) Aproximace pro různé parametry Gaussovo funkce č. 2

Aproximace pro povrch $Z = 2$ s užitím Gaussovo funkce

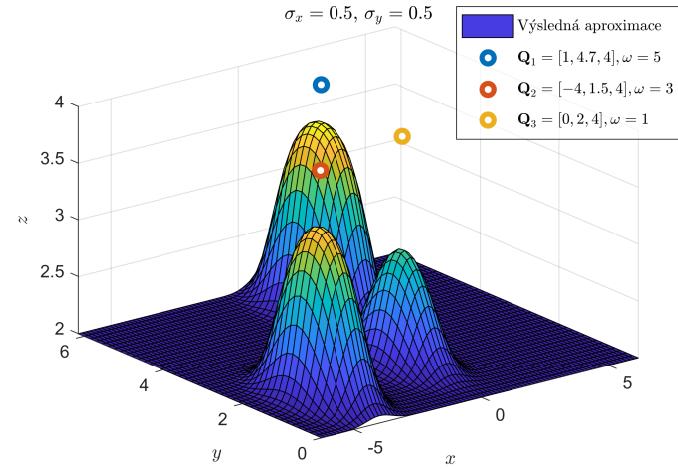
Aproximační body: Q_1, Q_2, Q_3



(c) Aproximace pro různé parametry Gaussovo funkce č. 3

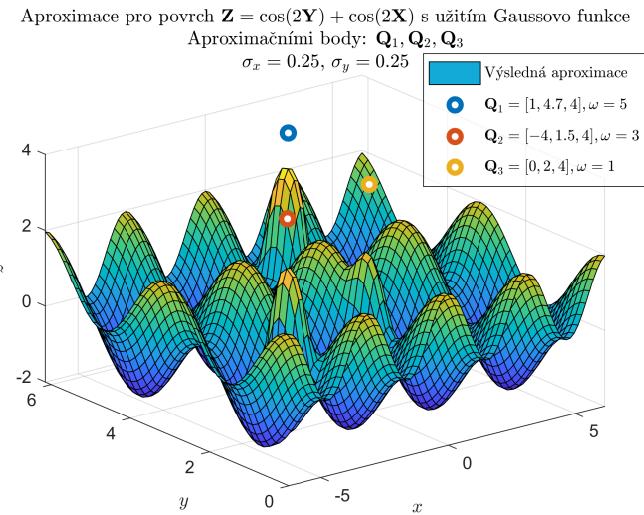
Aproximace pro povrch $Z = 2$ s užitím Gaussovo funkce

Aproximační body: Q_1, Q_2, Q_3

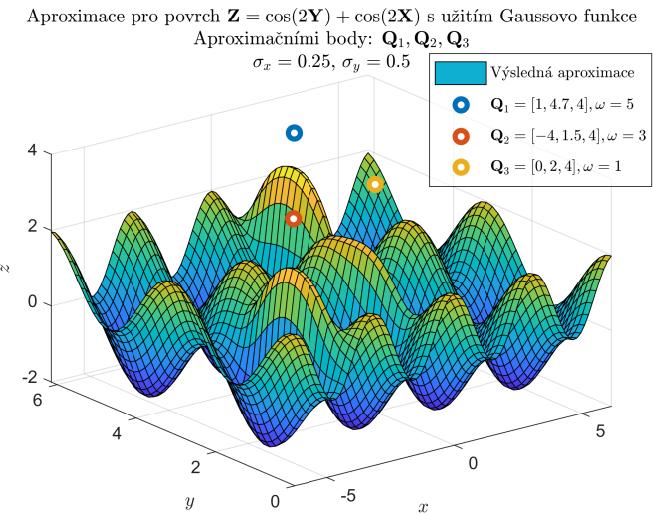


(d) Aproximace pro různé parametry Gaussovo funkce č. 4

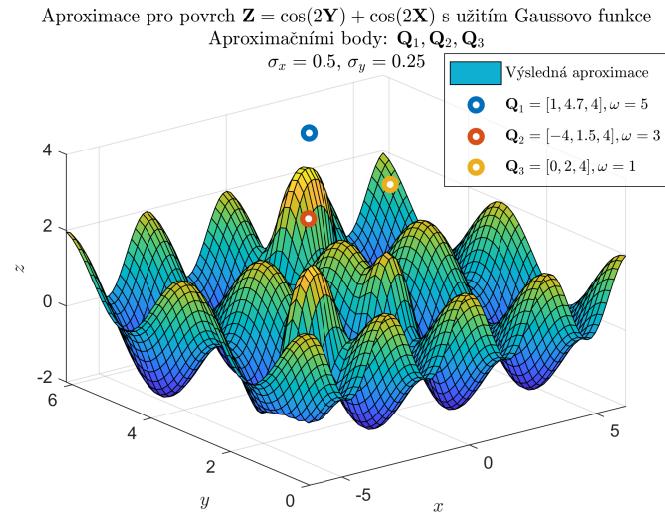
Obrázek 27: Ukázka approximace povrchu č. 1 s užitím Gaussovo funkce s různými parametry



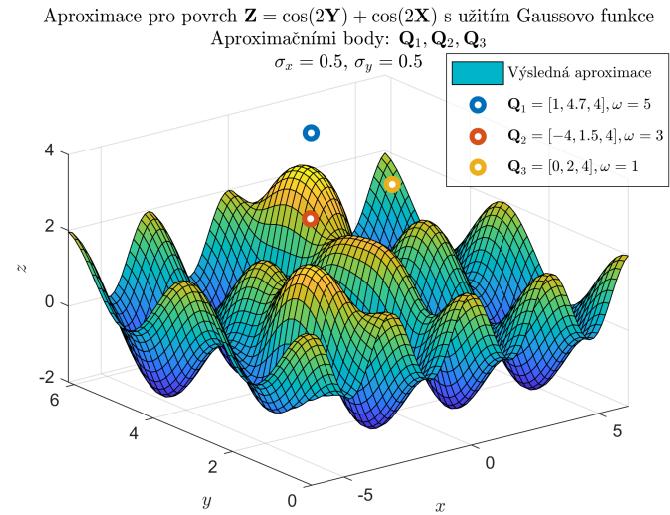
(a) Aproximace pro různé parametry Gaussovo funkce č. 5



(b) Aproximace pro různé parametry Gaussovo funkce č. 6



(c) Aproximace pro různé parametry Gaussovo funkce č. 7



(d) Aproximace pro různé parametry Gaussovo funkce č. 8

Obrázek 28: Ukázka approximace povrchu č. 2 s užitím Gaussovo funkce s různými parametry

3.6.2 NURBS aproximace 4D povrchu s užitím Gaussovy funkce

Opět se jedná o mnou navržený algoritmus dostačující pro účely této práce. Jelikož toto je vesměs nejdůležitější approximační funkce této práce, matematicky lépe podmíněný algoritmus může být otázkou dalšího výzkumu.

Pro approximaci ve 4D prostoru lze lehkými úpravami použít algoritmus zmíněný v sekci [NURBS aproximace povrchu s užitím Gaussovy funkce](#). Stačí rozšířit Gaussovou funkci $f(x, y)$ na funkci $f(x, y, w)$ ^[18] (viz animace č. 29), tj.:

$$f(x, y, w) = \exp \left(- \left(\frac{(x - x_0)^2}{2\sigma_x^2} + \frac{(y - y_0)^2}{2\sigma_y^2} + \frac{(w - w_0)^2}{2\sigma_w^2} \right) \right) \quad (65)$$

kde

- x_0 — střední hodnota osy x
- y_0 — střední hodnota osy y
- w_0 — střední hodnota osy w
- σ_x — variance osy x
- σ_y — variance osy y
- σ_w — variance osy w

Prvním krokem approximace je interpolace přes souřadnici w , stejně jako v sekci [Interpolace 4D nadpovrchu](#). Na takto získané povrchy pro příslušné hodnoty souřadnice w stačí aplikovat approximaci ze sekce [NURBS aproximace povrchu s užitím Gaussovy funkce](#).

Ukázka approximace s různými parametry pro approximační body $\bar{\mathbf{Q}}_i$ jsou na animacích č. 30, 31, kde:

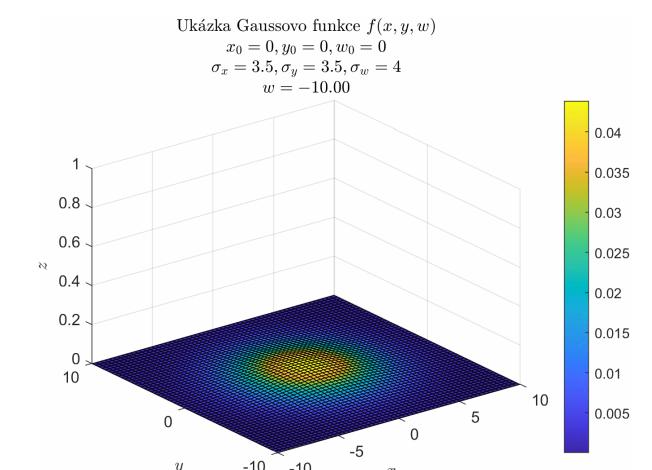
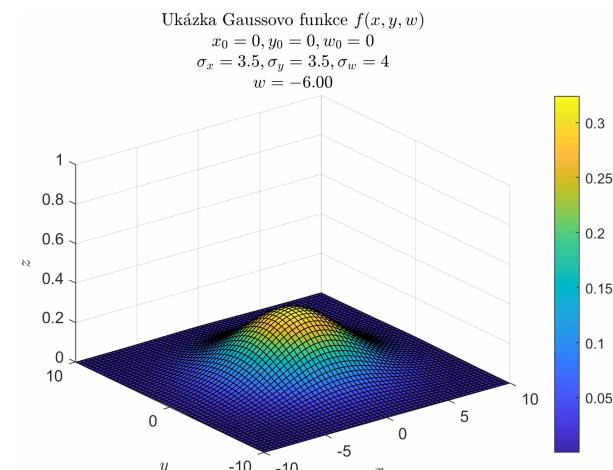
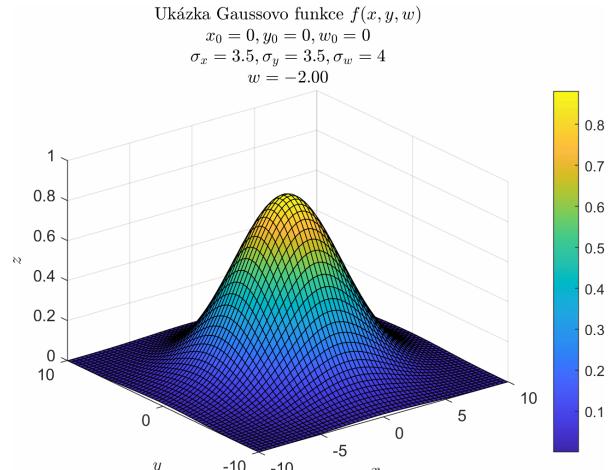
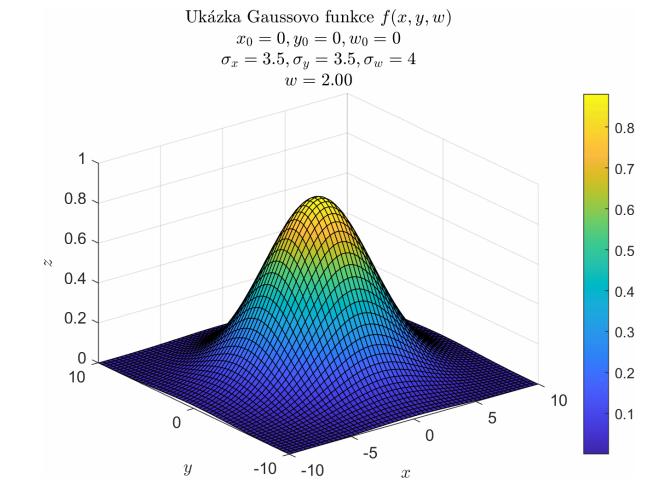
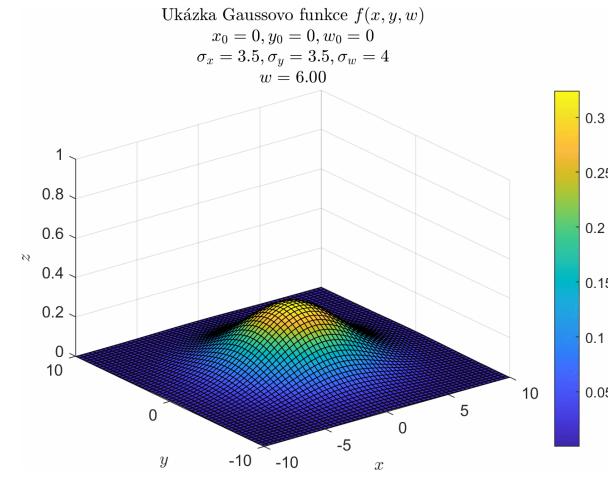
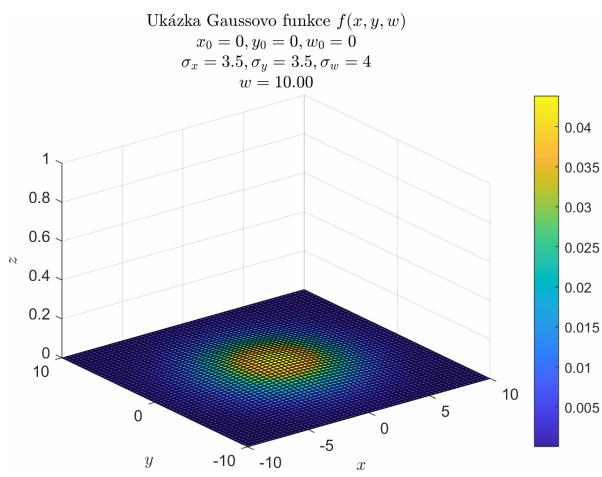
$$\bar{\mathbf{Q}}_1 = [x, y, z, w] = [-0.13319 \quad -0.13319 \quad 5 \quad 15] \quad (66)$$

$$\bar{\mathbf{Q}}_2 = [x, y, z, w] = [-4 \quad 1.5 \quad 4 \quad 16] \quad (67)$$

$$\bar{\mathbf{Q}}_3 = [x, y, z, w] = [-4 \quad -4 \quad 6 \quad 20] \quad (68)$$

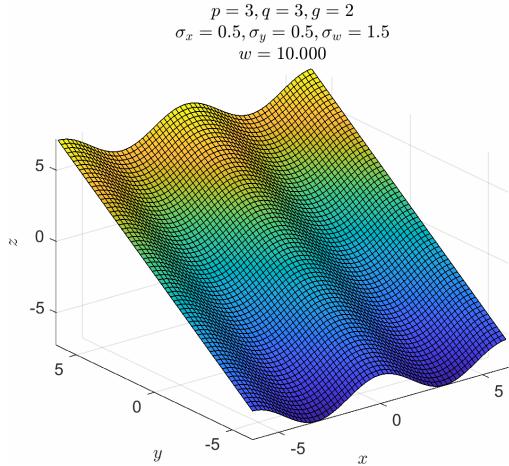
$$\bar{\mathbf{Q}}_4 = [x, y, z, w] = [4 \quad -3 \quad 7 \quad 25] \quad (69)$$

^[18]Jelikož chceme approximovat souřadnici z (stejně jako ve 3D případě), Gaussova funkce má jako třetí vstupní parametr souřadnici w

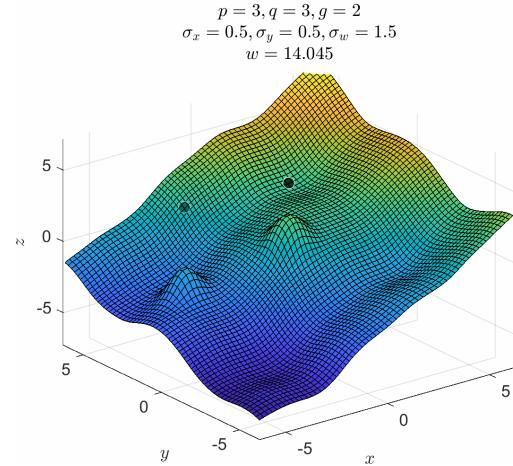


Obrázek 29: Ukázka průběhu Gaussovo funkce $f(x, y, w)$

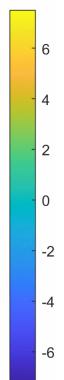
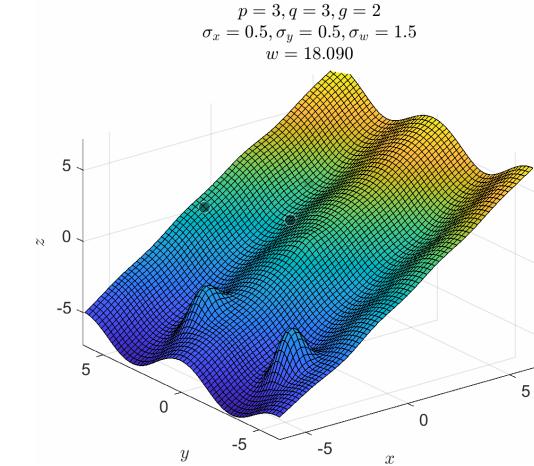
Aproximace bodů mezi povrchy č. 1



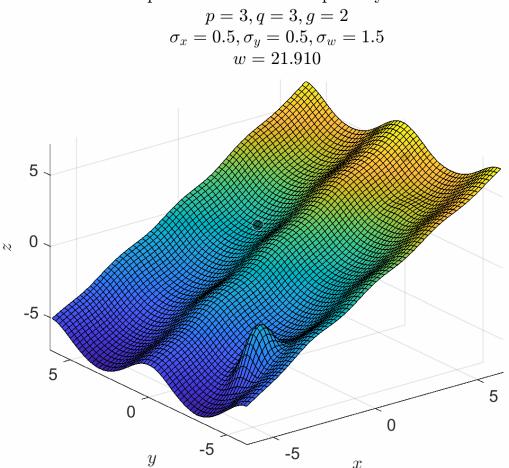
Aproximace bodů mezi povrchy č. 1



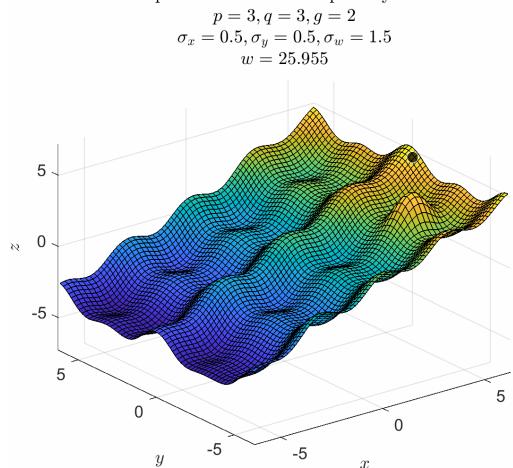
Aproximace bodů mezi povrchy č. 1



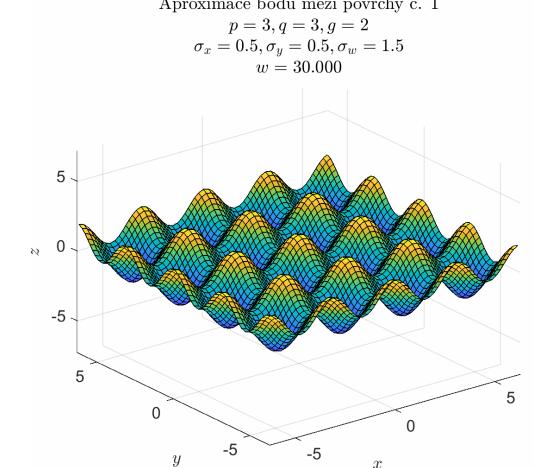
Aproximace bodů mezi povrchy č. 1



Aproximace bodů mezi povrchy č. 1

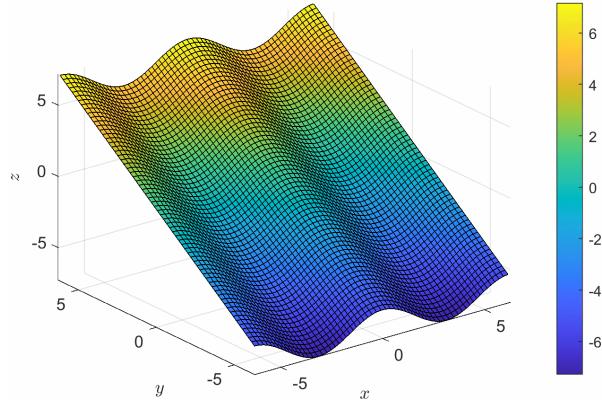


Aproximace bodů mezi povrchy č. 1

Obrázek 30: 4D approximace pro různé parametry Gaussovo funkce č. 1
(Průhlednost approximačních bodů odpovídá jejich vzdálenosti v ose w)

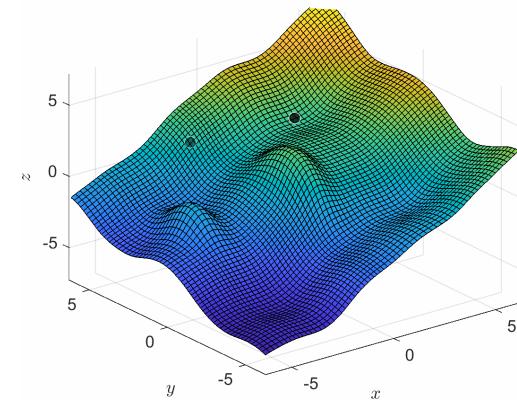
Aproximace bodů mezi povrchy č. 2

$$\begin{aligned} p &= 3, q = 3, g = 2 \\ \sigma_x &= 0.5, \sigma_y = 1, \sigma_w = 1.5 \\ w &= 10.000 \end{aligned}$$



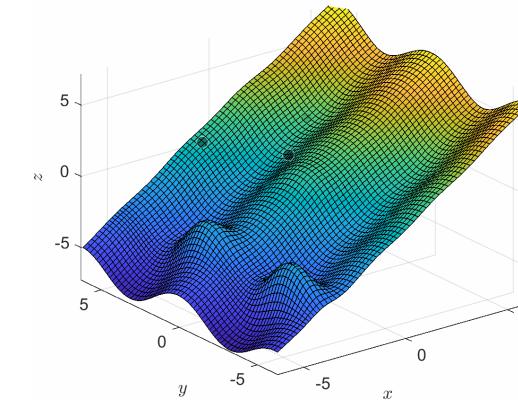
Aproximace bodů mezi povrchy č. 2

$$\begin{aligned} p &= 3, q = 3, g = 2 \\ \sigma_x &= 0.5, \sigma_y = 1, \sigma_w = 1.5 \\ w &= 14.045 \end{aligned}$$



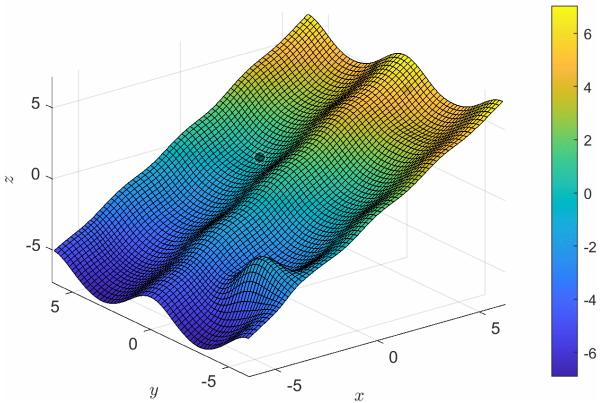
Aproximace bodů mezi povrchy č. 2

$$\begin{aligned} p &= 3, q = 3, g = 2 \\ \sigma_x &= 0.5, \sigma_y = 1, \sigma_w = 1.5 \\ w &= 18.090 \end{aligned}$$



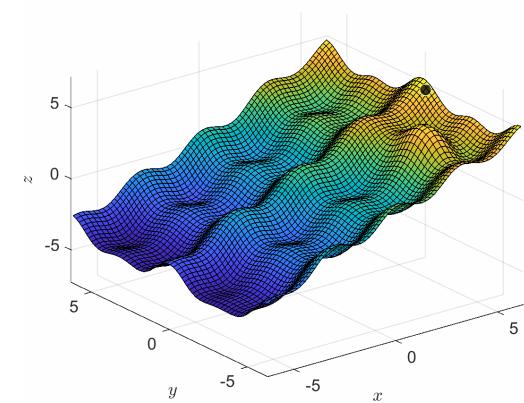
Aproximace bodů mezi povrchy č. 2

$$\begin{aligned} p &= 3, q = 3, g = 2 \\ \sigma_x &= 0.5, \sigma_y = 1, \sigma_w = 1.5 \\ w &= 21.910 \end{aligned}$$



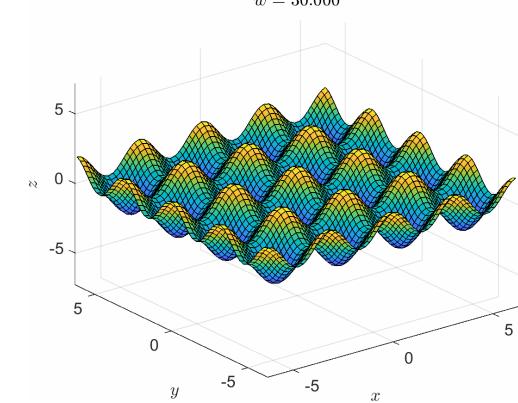
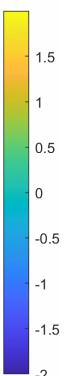
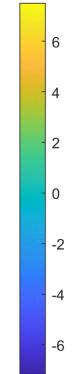
Aproximace bodů mezi povrchy č. 2

$$\begin{aligned} p &= 3, q = 3, g = 2 \\ \sigma_x &= 0.5, \sigma_y = 1, \sigma_w = 1.5 \\ w &= 25.955 \end{aligned}$$



Aproximace bodů mezi povrchy č. 2

$$\begin{aligned} p &= 3, q = 3, g = 2 \\ \sigma_x &= 0.5, \sigma_y = 1, \sigma_w = 1.5 \\ w &= 30.000 \end{aligned}$$

Obrázek 31: 4D approximace pro různé parametry Gaussovo funkce č. 2
(Průhlednost approximačních bodů odpovídá jejich vzdálenosti v ose w)

4 Návrh systému automatické kalibrace proudové tabulky

Tato kapitola se již zabývá aplikací NURBS teorie na náš problém — tj. užití NURBS teorie pro práci s kalibrační tabulkou. Dále se kapitola zaměřuje na sběr vhodných hodnot pro aktualizace kalibrační tabulky.

4.1 Aplikace NURBS teorie pro modelování kalibrační tabulky

V této kapitole je již použita NURBS interpolace a approximace na konkrétních datech. Nejprve je uvedena ukázka na 1 DoF modelu, poté na řezu kalibrační tabulkou, kde lze vidět cílenou approximaci, dále je uvedena 3D a 4D interpolace kompenzačních tabulek.

4.1.1 1 DoF model

Na obrázku č. 32 je ukázka cílené approximace křivky pro náš 1 DoF model systému. Approximační body jsme získali úpravou fyzikálních parametrů modelu, zvolené body jsou záměrně mimo “mřížku” od původních bodů. Ukázka obsahuje výsledky pro dvě různé váhy approximačních bodů.

4.1.2 2D řez kalibrační tabulkou

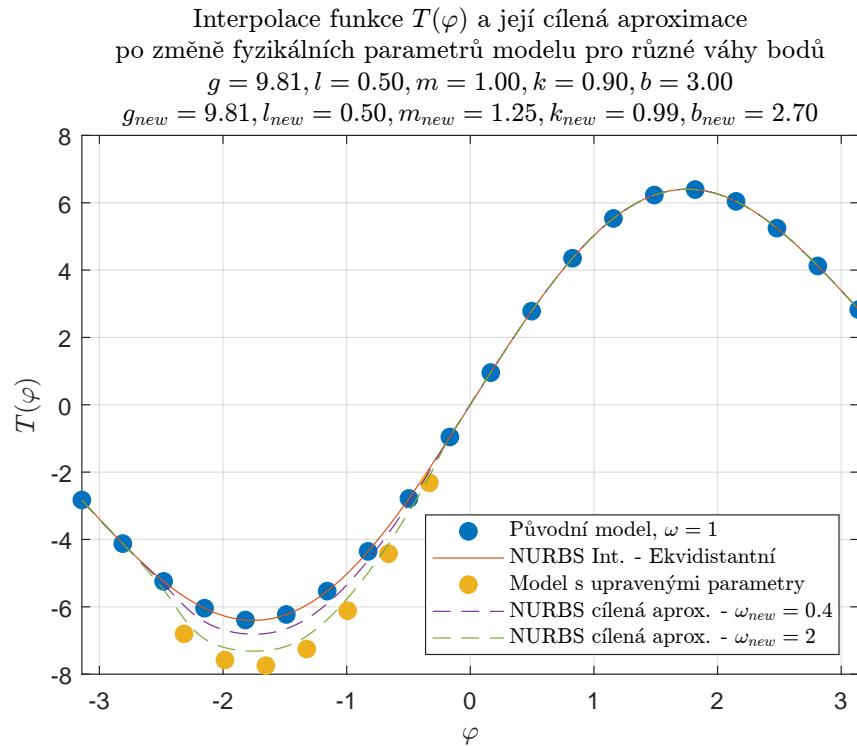
Na obrázcích č. 33, 34 je ukázka cílené approximace křivky na jedné z kalibračních tabulek pro fixní hodnotu osy *Prop*, kde je lépe ukázáno jak cílená approximace funguje v detailu. Řez tabulkou jsme získali právě zvolením fixní hodnoty kloubu *Prop*.

4.1.3 Interpolace kalibračních tabulek

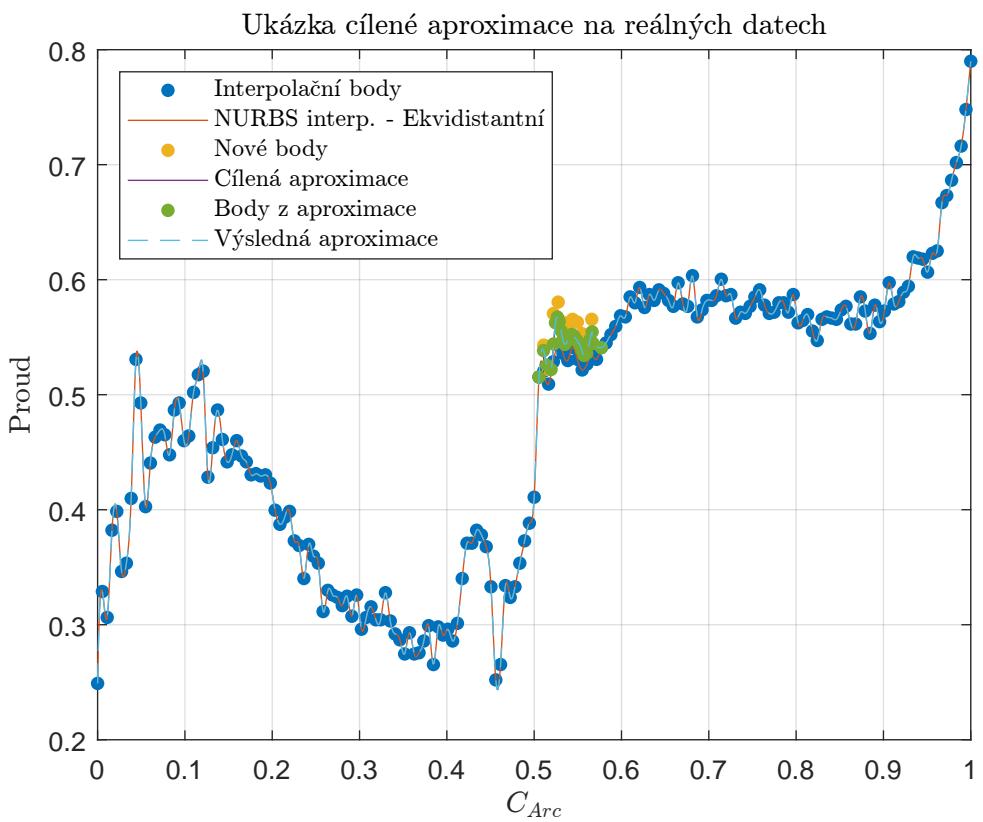
Na obrázcích č. 35, 36, 37, 38, napravo lze vidět aplikaci NURBS interpolace povrchu a nalevo lze vidět původní data. **Řád povrchu?**

4.1.4 Interpolace napříč kalibračními tabulkami

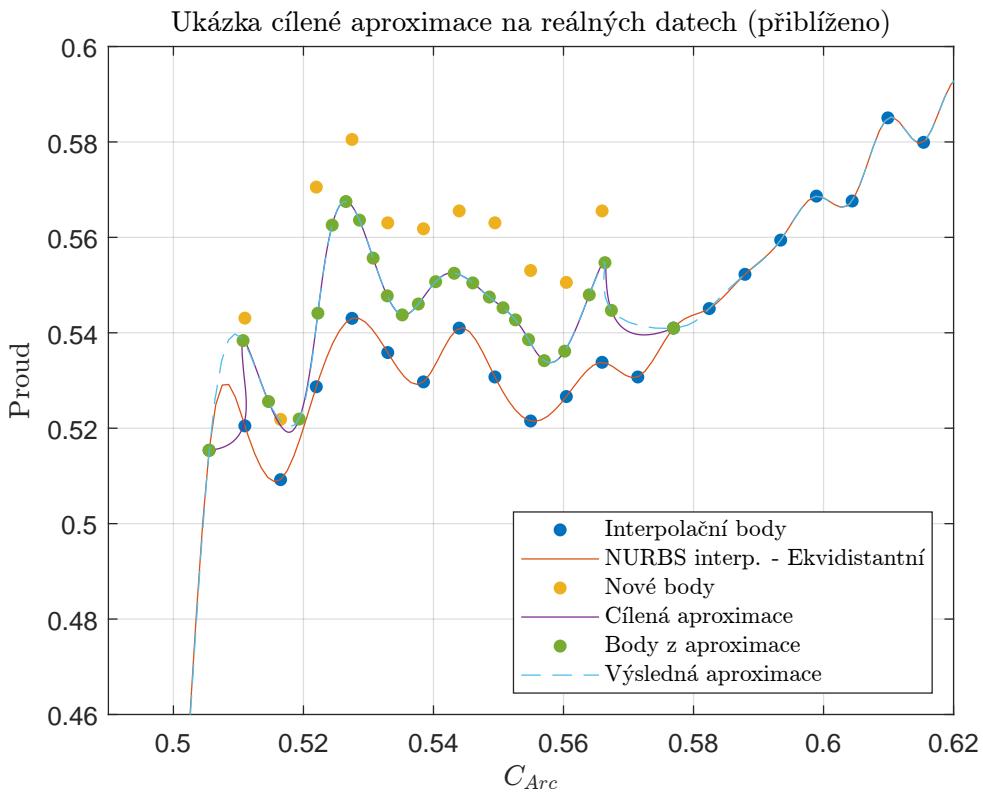
Jak již bylo zmíněno, máme 2 kalibrační tabulky pro dvě polohy kloubu Ids , a proto bychom rádi chtěli přes tuto souřadnici interpolovat, abychom získali kalibrační tabulky mezi těmito dvěma polohami. K tomuto využijeme NURBS interpolaci nadpovrchu ze sekce [Interpolace 4D nadpovrchu](#), výsledné animace jsou na obrázcích č. [40](#), [42](#).



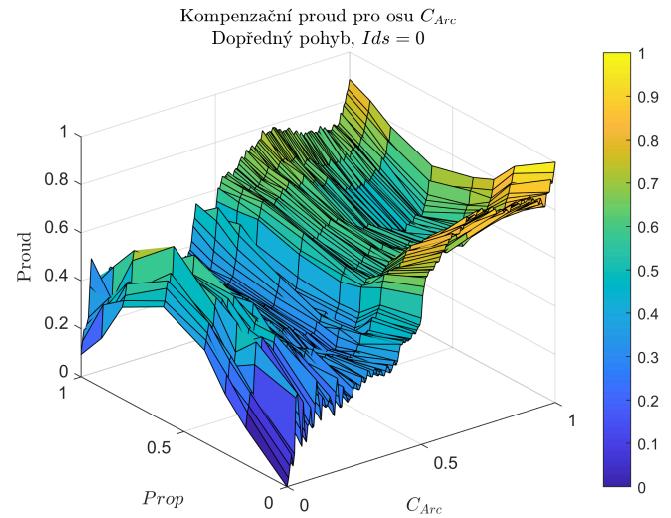
Obrázek 32: Cílená approximace křivky metodou nejmenších čtverců na 1 DoF modelu pro různé váhy nových bodů



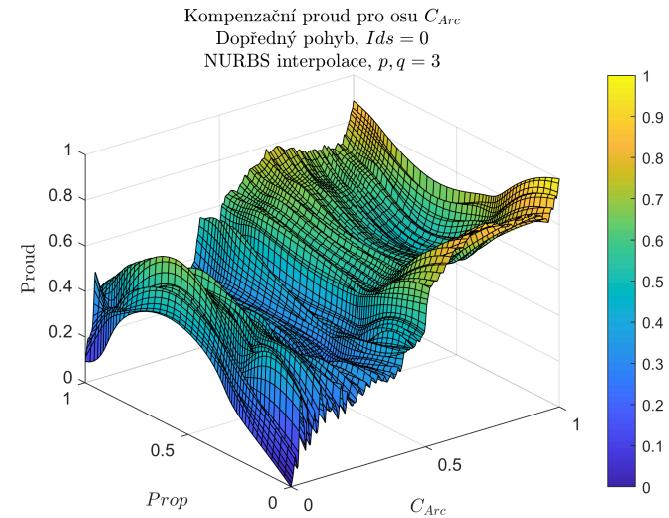
Obrázek 33: Cílená approximace křivky metodou nejmenších čtverců na reálných datech



Obrázek 34: Cílená approximace křivky metodou nejmenších čtverců na reálných datech (přiblíženo)

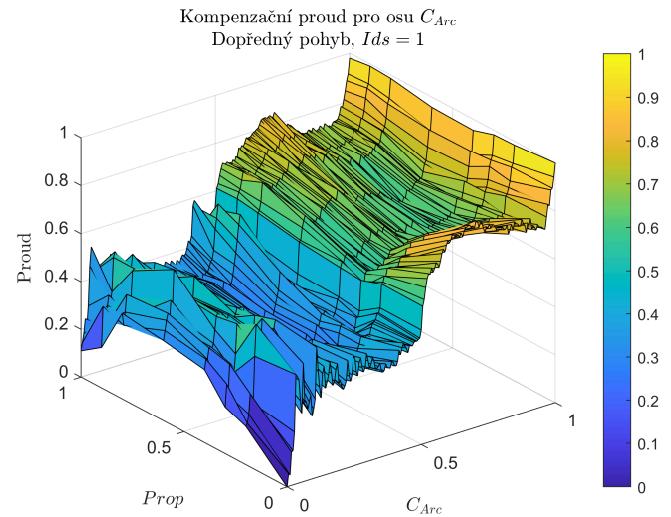


(a) Lineární interpolace kompenzační tabulky proudu č. 1

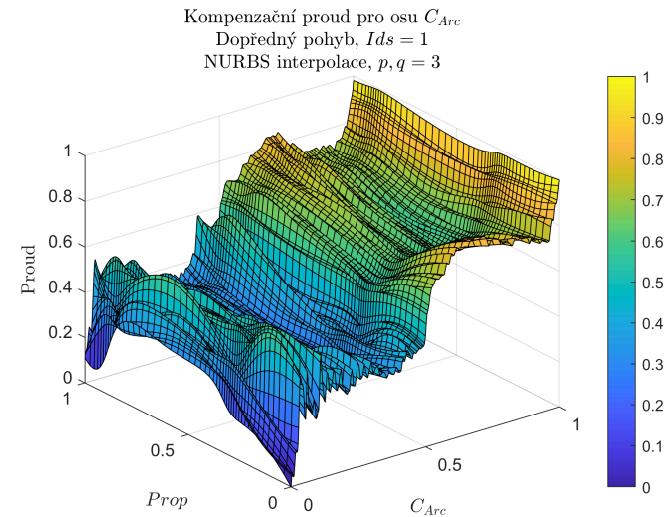


(b) NURBS interpolace kompenzační tabulky proudu č. 1

Obrázek 35: Ukázka interpolace na reálných datech č. 1

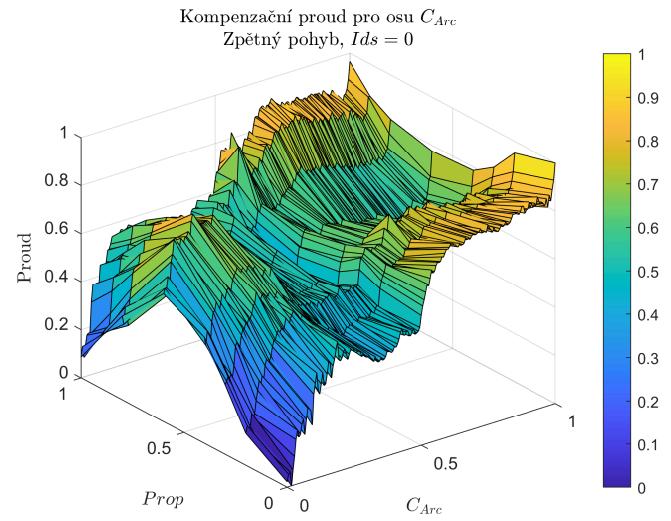


(a) Lineární interpolace kompenzační tabulky proudu č. 2

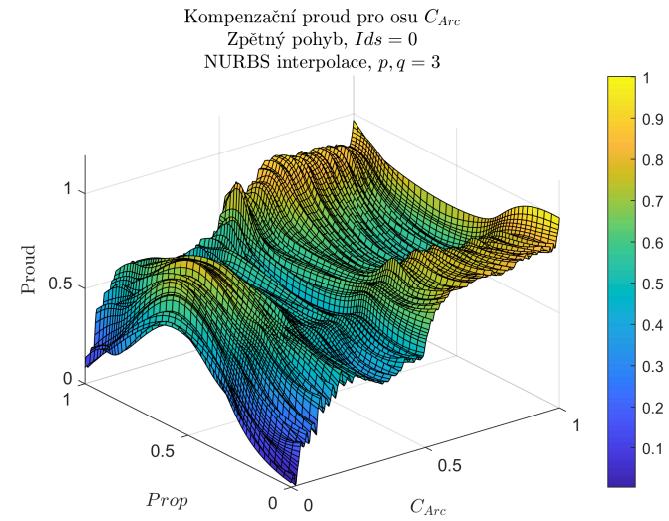


(b) NURBS interpolace kompenzační tabulky proudu č. 2

Obrázek 36: Ukázka interpolace na reálných datech č. 2

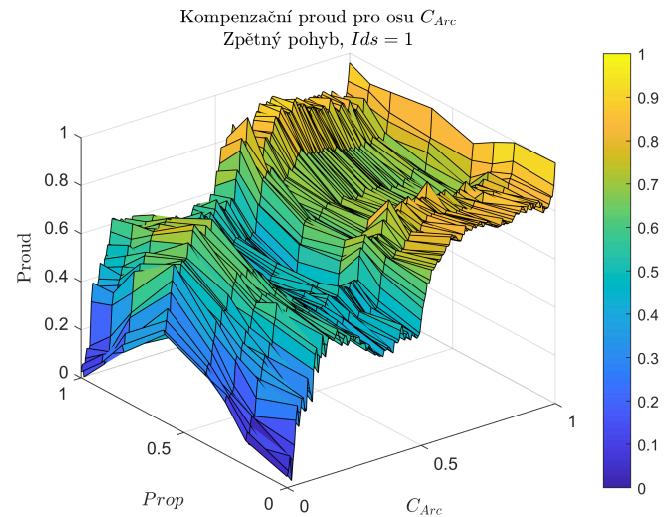


(a) Lineární interpolace kompenzační tabulky proudu č. 3

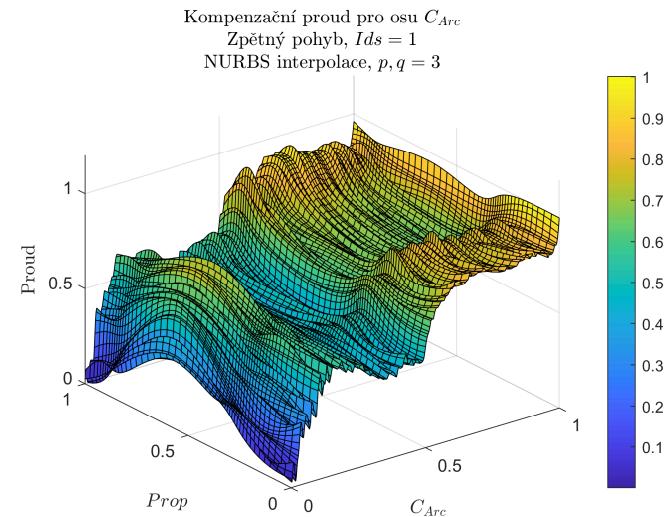


(b) NURBS interpolace kompenzační tabulky proudu č. 3

Obrázek 37: Ukázka interpolace na reálných datech č. 3

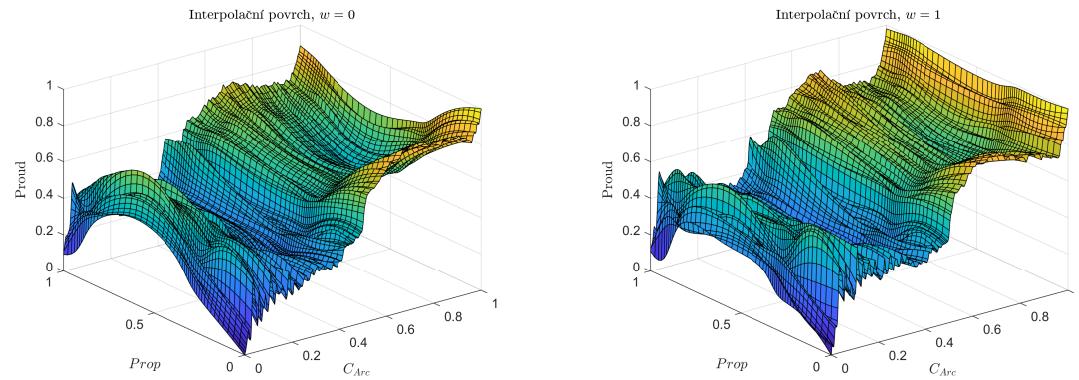


(a) Lineární interpolace kompenzační tabulky proudu č. 4

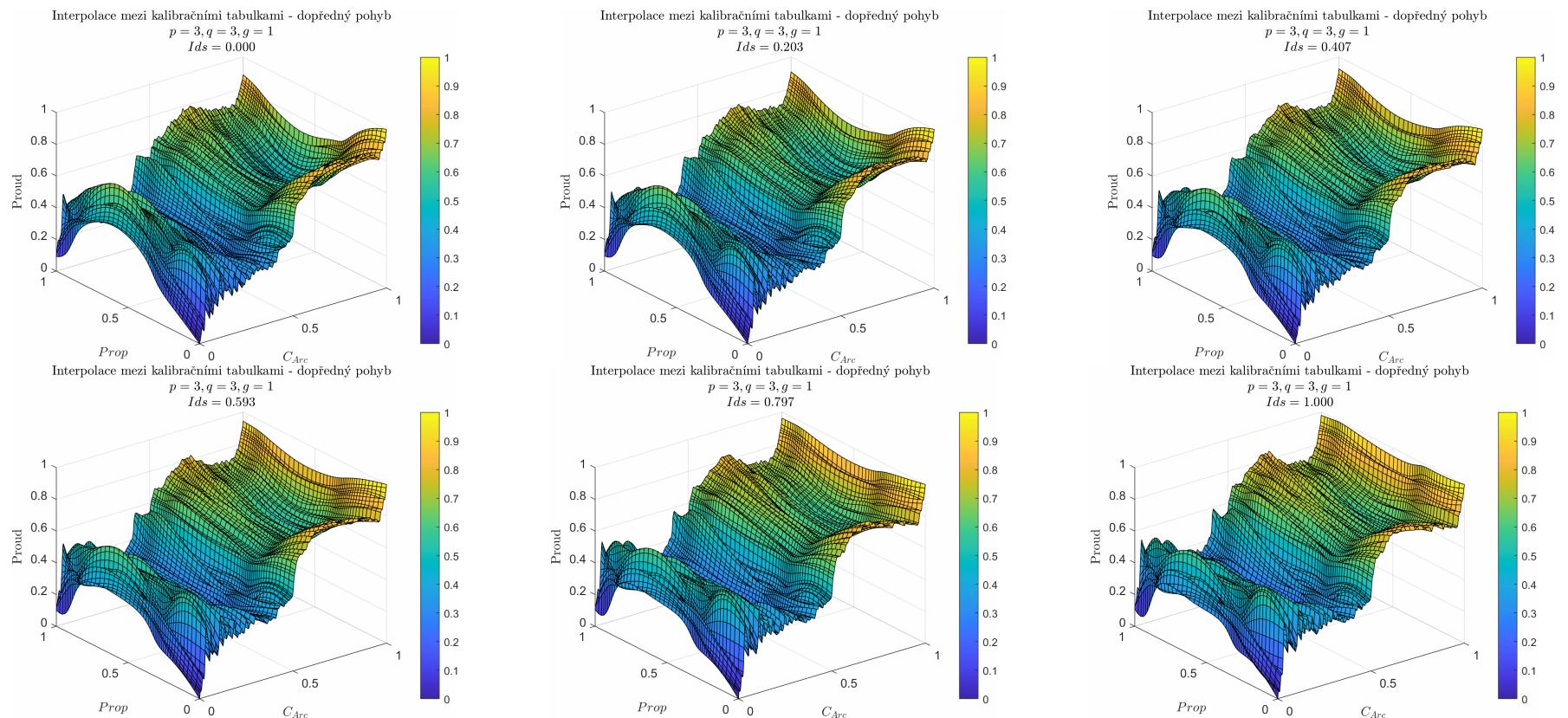


(b) NURBS interpolace kompenzační tabulky proudu č. 4

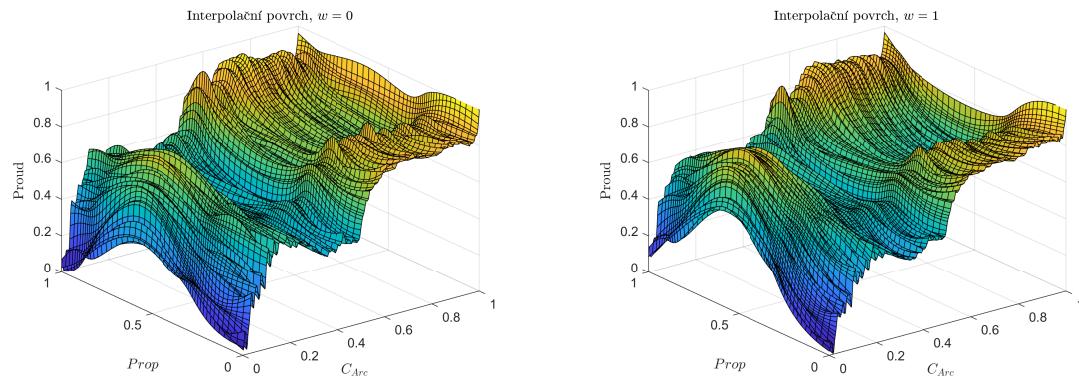
Obrázek 38: Ukázka interpolace na reálných datech č. 4



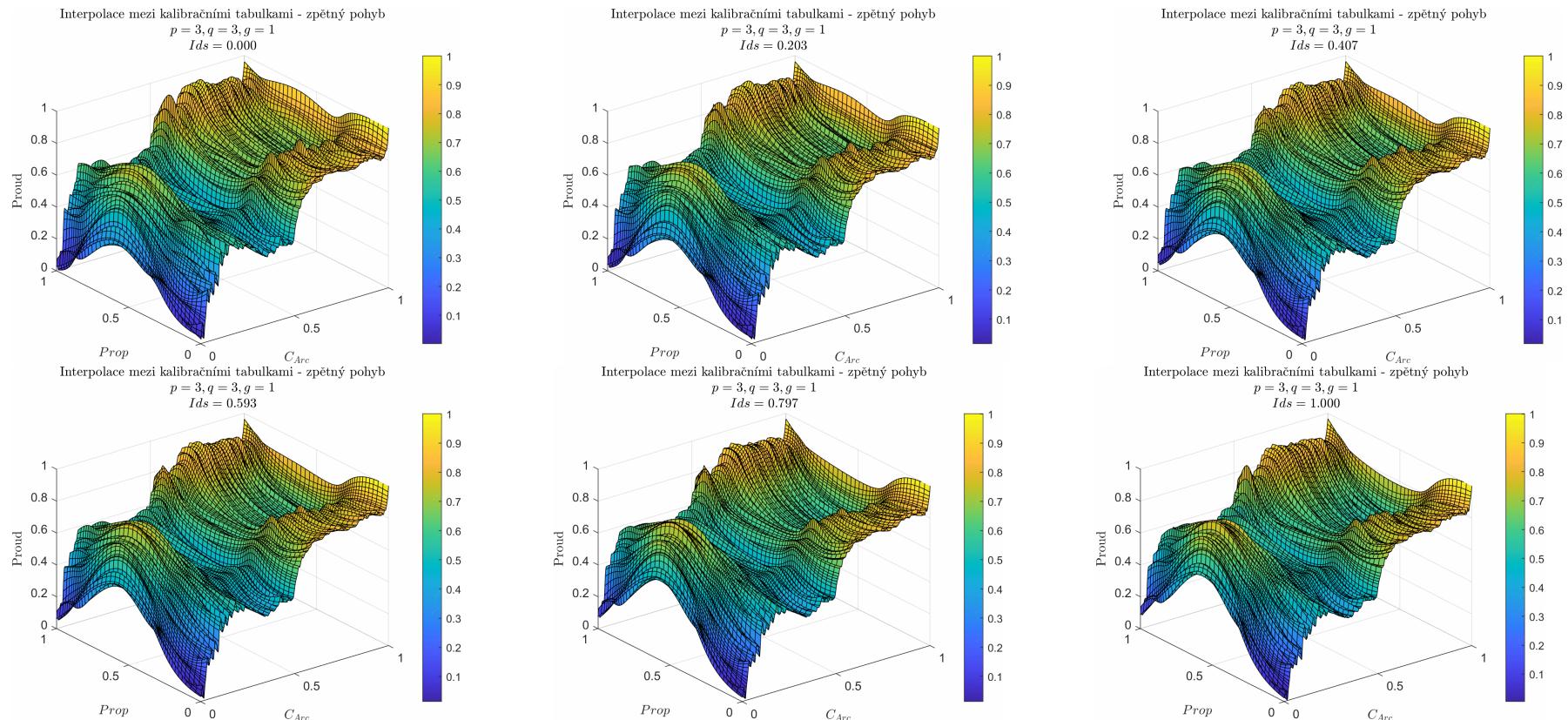
Obrázek 39: Interpolaci povrchy pro 4D interpolaci kompenzačních tabulek - dopředný pohyb



Obrázek 40: Ukázka průběhu interpolace mezi kompenzačními tabulkami přes souřadnici Ids — dopředný pohyb



Obrázek 41: Interpolaci povrhy pro 4D interpolaci kompenzačních tabulek - zpětný pohyb



Obrázek 42: Ukázka průběhu interpolace mezi kompenzačními tabulkami přes souřadnici Ids — zpětný pohyb

4.2 Analýza měřených dat

Tato kapitola je zaměřena na sběr vhodných dat pro aktualizace kalibračních tabulek. Společnost Phillips poskytla za účelem výzkumu a vývoje autonomního kalibračního systému tyto záznamy:

1. Záznam jedné osy manuálního ovládání stroje uživatelem
2. Záznam testů proudové odchylky na předdefinovaných trajektoriích

4.2.1 Analýza zaznamenaného pohybu jednoho kloubu

Cílem této procedury je hledat vhodné body pro aktualizaci kalibračních tabulek. Především hledáme trajektorie s pomalým pohybem a konstantní rychlostí, protože v této oblasti se nejvíce projevují statické a kvazistatické síly, které by kalibrační tabulka měla kompenzovat, a proto jsou pro nás relevantní.

V této části budeme uvažovat záznam číslo 1. Prozkoumáním záznamu lze zjistit, zda-li se daný kloub někdy pohybuje vhodnou rychlostí pro naše účely, tj.: neminimální^[19], shora omezenou^[20] a konstantní.

Body, které splňují tyto požadavky můžeme algoritmicky odhalit následovně:

1. Konstantní rychlosť — výpočtem diferencí naměřených hodnot jsme schopni sledovat absolutní změny v rychlosti v každém časovém okamžiku. Zvolením vhodného prahu^[21], lze takto určit kdy je rychlosť téměř konstantní.
2. Neminimální, shora omezená rychlosť — nalezneme hodnoty rychlosti, které jsou shora i zespoda omezeny vhodně zvolenými prahy^[21].

Užitím pouze filtru konstantní rychlosti dostáváme body, kdy rychlosť daného kloubu je skutečně téměř konstantní — viz obrázek č. 43. Nicméně zdaleka nevšechny tyto hodnoty jsou vhodné pro aktualizaci CCT, protože vybrané hodnoty obsahují značné výkyvy v proudovém měření.

Aplikací všech filtrů získáme body, které by se již daly využít pro úpravu CCT — viz obrázek č. 44. Je nutné dodat, že v tomto měření máme dostupné pouze 2 (poloha C_{Arc} ,

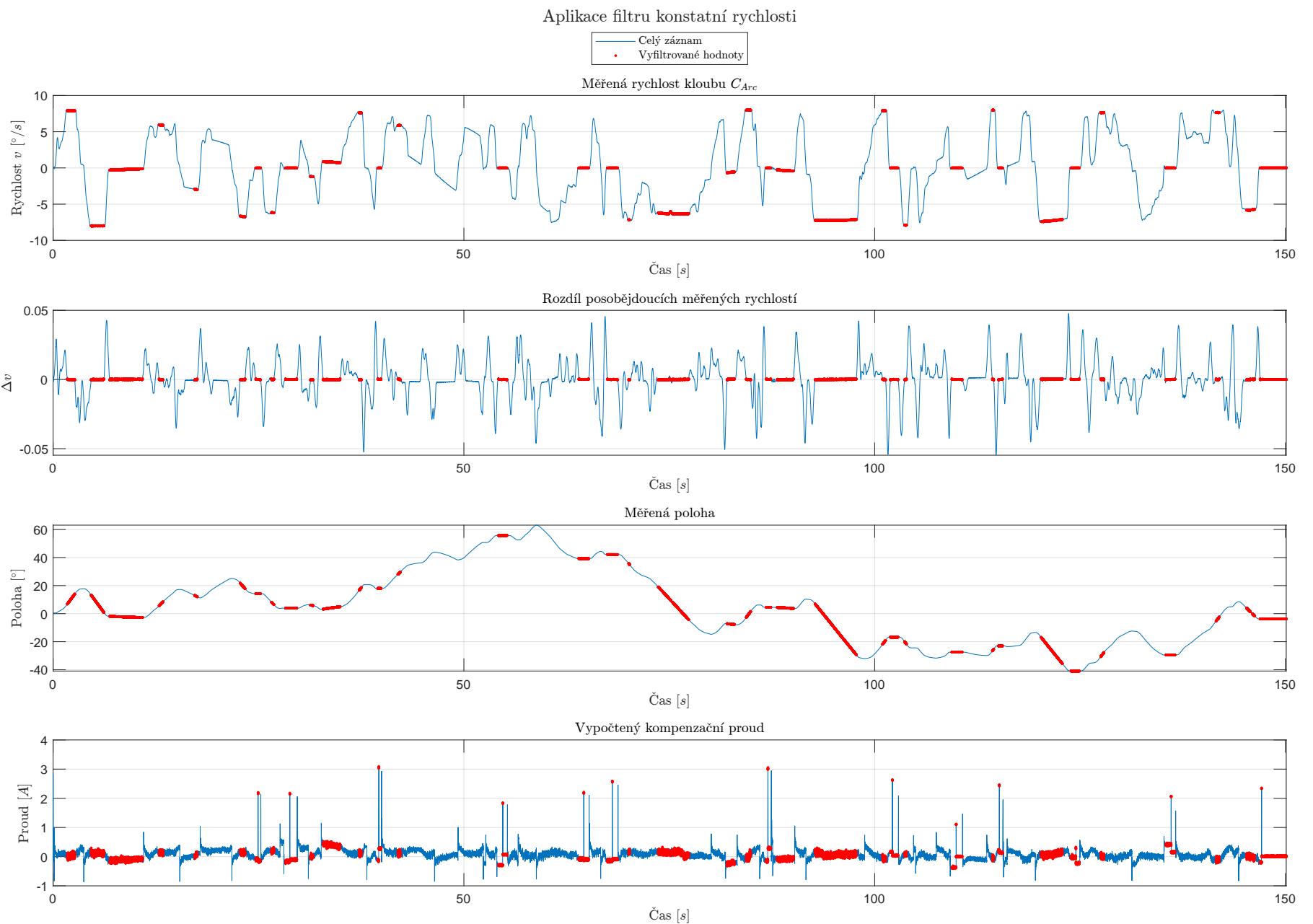
^[19]Při hodně malých rychlostech nejvíce působí tření, které by se značně podepsalo na měřeném proudu.

^[20]Kalibrační tabulka má největší vliv při pomalých pohybech, proto by i bylo vhodné tabulku upravovat na základě měření při takovýchto podobných rychlostech.

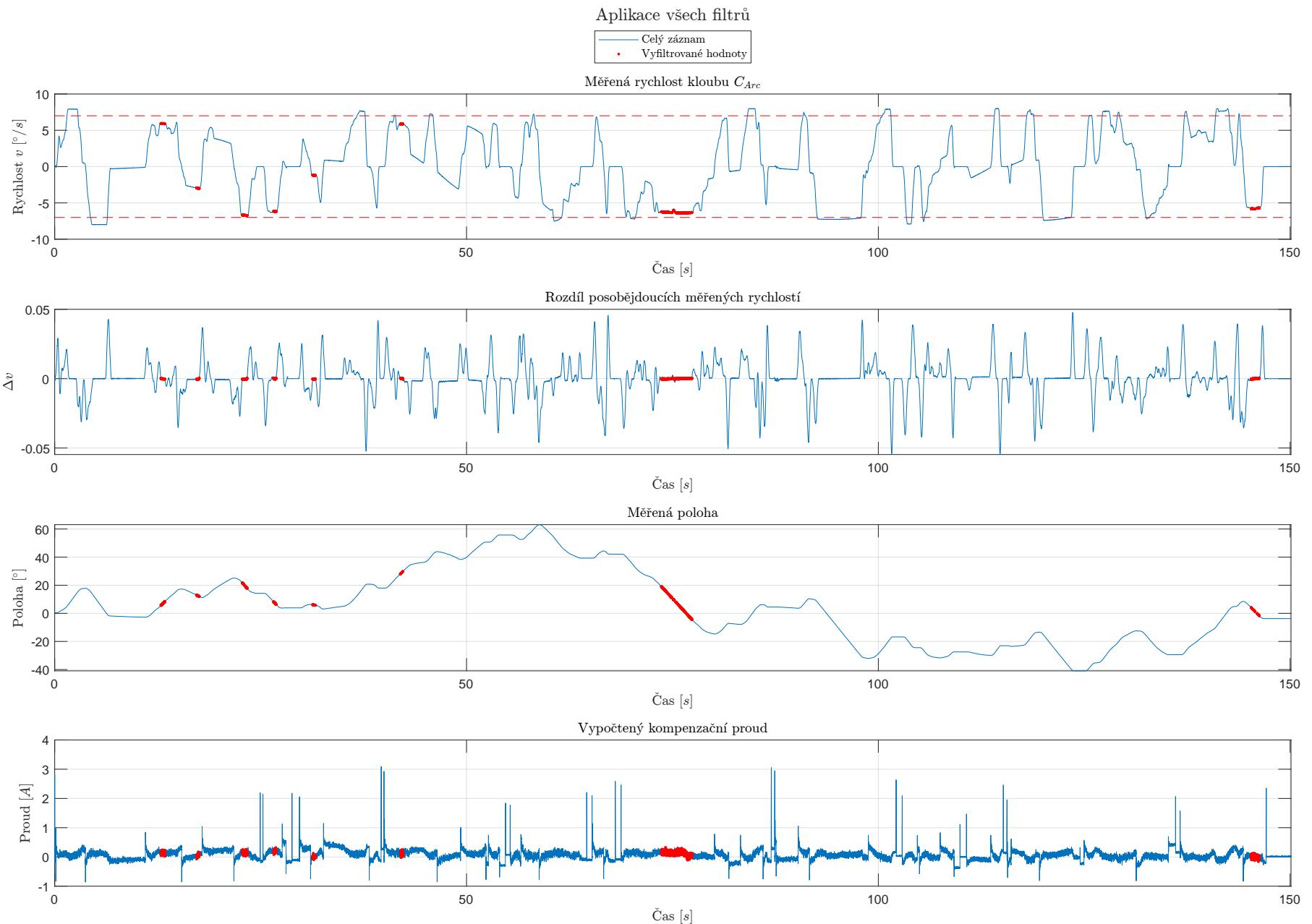
^[21]Pro účely této práce byly zvoleny hodnoty ručně na základě naměřených hodnot

kompenzační proud) ze 4 (poloha C_{Arc} , $Prop$, Ids a kompenzační proud) hodnot, které jsou potřeba pro získání přesné polohy v kompenzační tabulce, a tudíž na základě tohoto měření nejsme schopni přesně vyzkoušet aktualizaci CCT — potřebná data, na základě kterých by se aktualizace dala otestovat přímo, Phillips neposkytl.

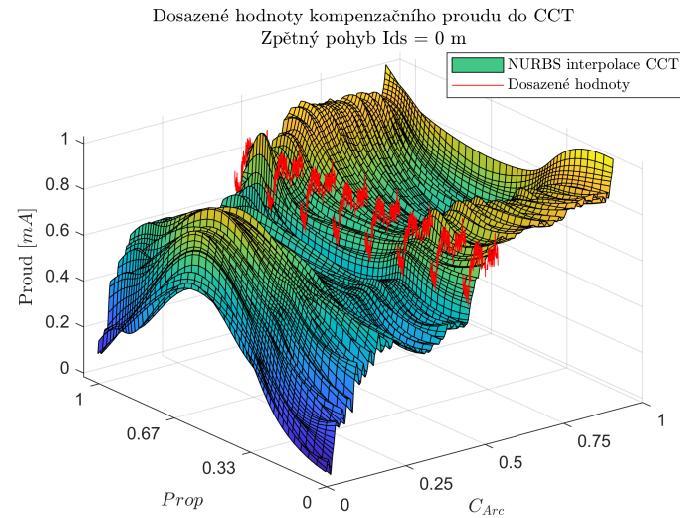
Přestože nevím 2 souřadnice, můžeme zkousit získané hodnoty dosadit do CCT. Konkrétně zkusíme dosadit nejdelší úsek vhodného pohybu nacházející se přibližně v polovině času záznamu. Z polohy lze vidět, že se jedná o zpětný pohyb, tím pádem vložíme hodnoty do našich dvou kalibračních tabulek pro zpětný pohyb. Z obrázků č. [45a](#) a [45b](#) lze odhadnout, že poloha kloubu $Prop$ byla okolo hodnoty 0.85 a poloha kloubu Ids byla přibližně 0. Pro tyto polohy je na obrázku č. [46](#) je již ukázka aktualizace CCT pomocí approximace povrchu užitím Gaussovy funkce.



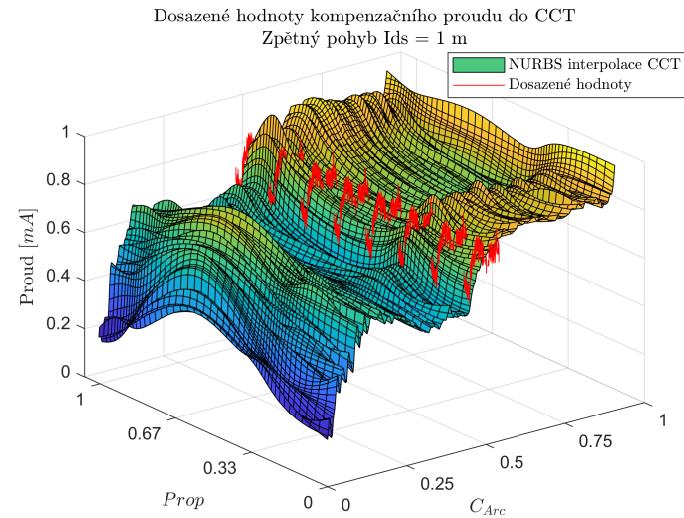
Obrázek 43: Ukázka aplikace filtru konstantní rychlosti



Obrázek 44: Ukázka aplikace všech filtrů

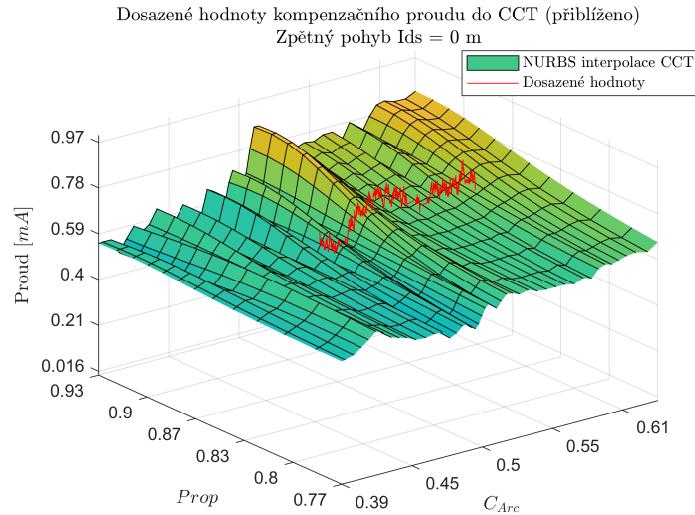


(a) Vložení hodnot do tabulky pro zpětný pohyb pro polohu kloubu $Ids = 0$ m

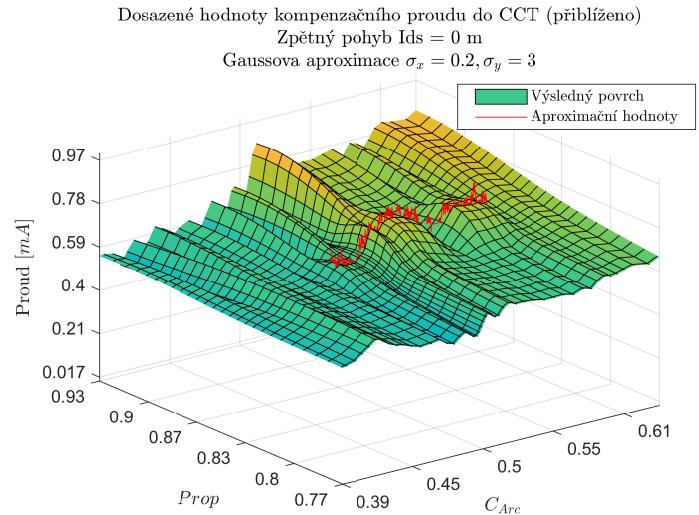


(b) Vložení hodnot do tabulky pro zpětný pohyb pro polohu kloubu $Ids = 1$ m

Obrázek 45: Vložení vhodných bodů ze záznamu měření jednoho kloubu



(a) Přiblížení dosazených hodnot



(b) Gaussova aproximace povrchu

Obrázek 46: Ukázka Gaussovo approximace povrchu pro hodnoty z měření jednoho kloubu

4.2.2 Analýza testu odchylky proudu

Tato sekce se zabývá záznamem číslo 2. Záznam je tvořen vícero soubory, každý odpovídající jednomu pohybu kloubu C_{Arc} pro fixní polohu $Prop$ a Ids . Stejně jako v sekci [Analýza zaznamenaného pohybu jednoho kloubu](#) se budeme zajímat o pomalý pohyb s konstantní rychlostí, vyhovující hodnoty pro jednotlivé typy trajektorií jsou zobrazeny na obrázcích č. [47a](#), [47b](#), [47c](#), [47d](#).

Ze záznamu jsme vypočetli nový kompenzační proud kalibrační tabulky I_{CCT}^{new} pomocí vzorečku:

$$I_{CCT}^{new} = I_{CCT} + I_{set} - I_{measured} \quad (70)$$

kde I_{CCT} značí kompenzační proud z CCT, I_{set} výstup regulátoru a $I_{measured}$ je měřený proud tekoucí pohonem.

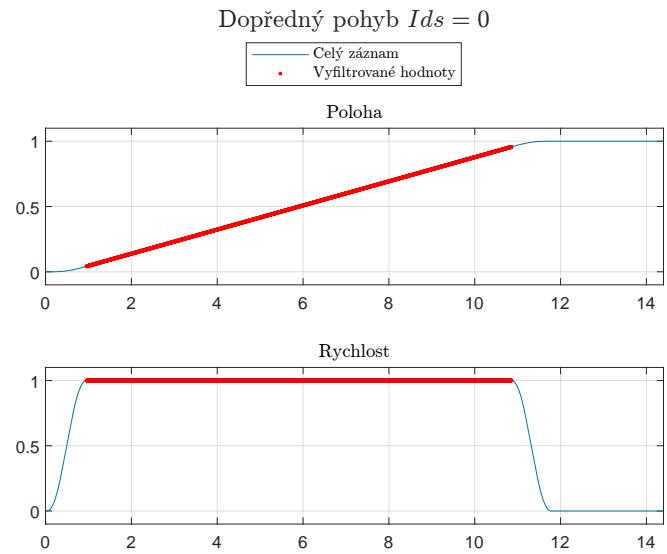
Takto vypočtené hodnoty jsou zobrazeny na obrázcích č. [48a](#), [48b](#), [48c](#), [48d](#). Jak lze vidět, výsledky jsou velmi zašuměné, přesto můžeme vypozorovat, že zašuměná data přibližně opisují kalibrační tabulku.

4.2.2.1 Zpracování zašuměných dat

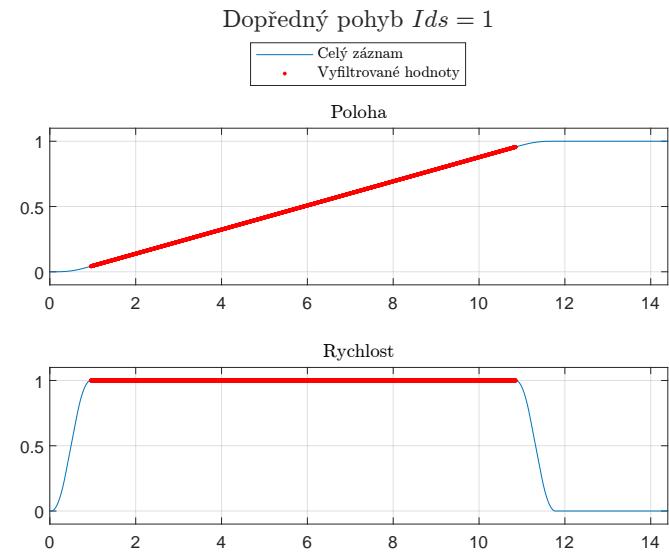
Na zašuměná data zkusíme aplikovat 2 různé způsoby zbavení se šumu:

1. Použití funkce Matlab `smooth`
2. Použití NURBS approximace křivky ze sekce [3.5](#)

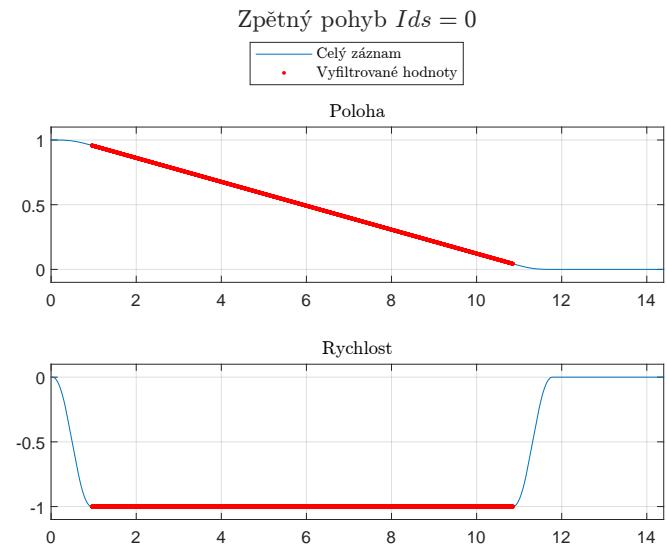
Výsledky těchto funkcí jsou na obrázcích č. [49a](#), [49b](#), [49c](#), [49d](#). Z takto upravených dat je již lépe vidět, že hodnoty nového kompenzačního proudu lemuje příslušnou CCT s výjimkou okolí souřadnice $Prop = 0.5$ a $C_{Arc} = 0.85$. Na tomto místě se hodnoty značně odchylují od původních hodnot kalibrační tabulky, toto je dále lépe vidět na obrázcích č. [50a](#), [50b](#), [50c](#), [50d](#), a [51a](#), [51b](#), [51c](#), [51d](#), kde je uvedena NURBS interpolace zpracovaných dat. Odchylka může být způsobena změnou fyzikálních parametrů manipulátoru a nebo Phillips používá nějaký svůj vlastní algoritmus, který data vyhodnocuje jiným způsobem. Nicméně pro naše účely se tyto odlišnosti hodí, protože nám umožní odzkoušení aktualizace CCT — viz kapitola [Aktualizace kalibrační tabulky](#).



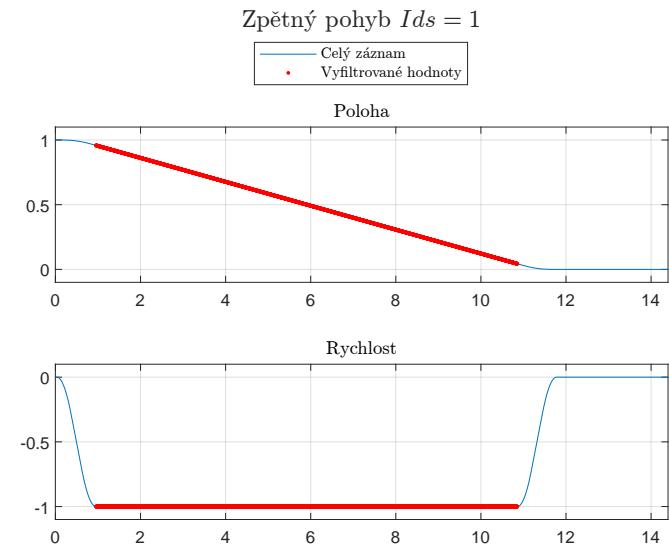
(a) Trajektorie kloubu C_{Arc} dopředného pohybu pro polohu $Ids = 0$



(b) Trajektorie kloubu C_{Arc} dopředného pohybu pro polohu $Ids = 1$

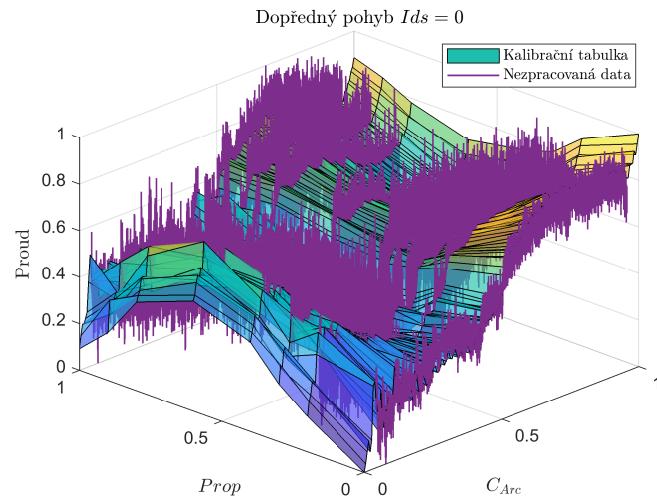


(c) Trajektorie kloubu C_{Arc} zpětného pohybu pro polohu $Ids = 0$

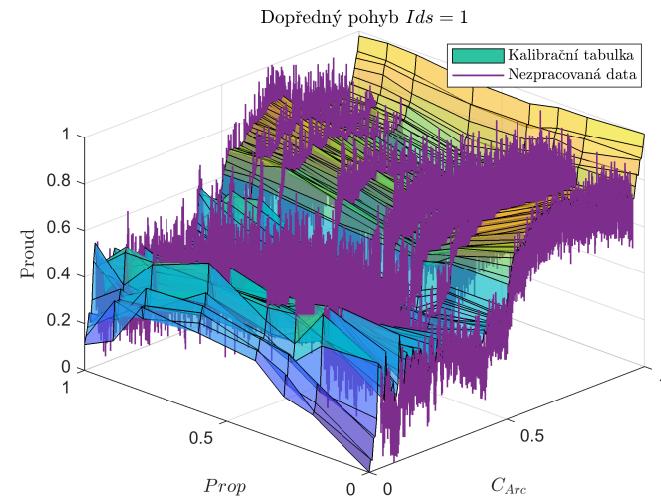


(d) Trajektorie kloubu C_{Arc} zpětného pohybu pro polohu $Ids = 1$

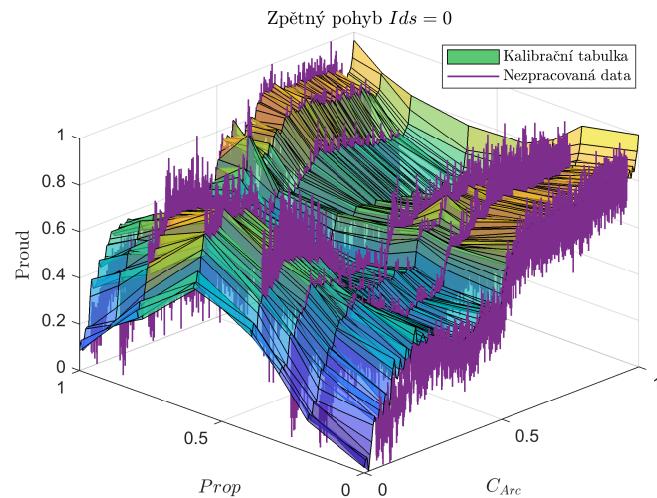
Obrázek 47: Ukázka trajektorií kloubu C_{Arc} pro 4 proudové kalibrační tabulky



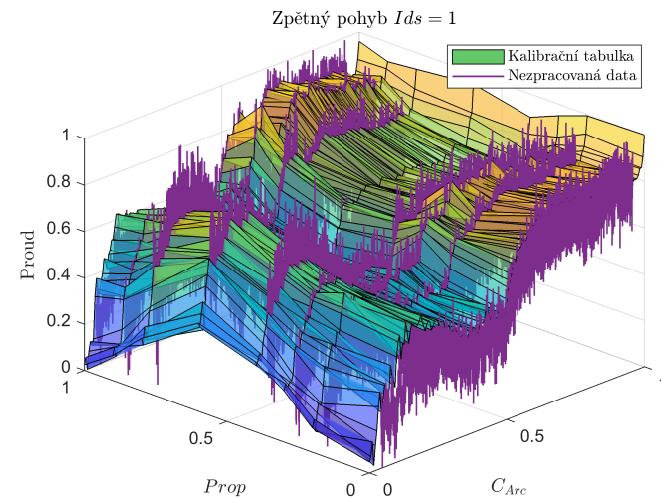
(a) Vypočtený kompenzační proud ze záznamu testovací trajektorie pro dopředný pohyb, $Ids = 0$



(b) Vypočtený kompenzační proud ze záznamu testovací trajektorie pro dopředný pohyb, $Ids = 1$

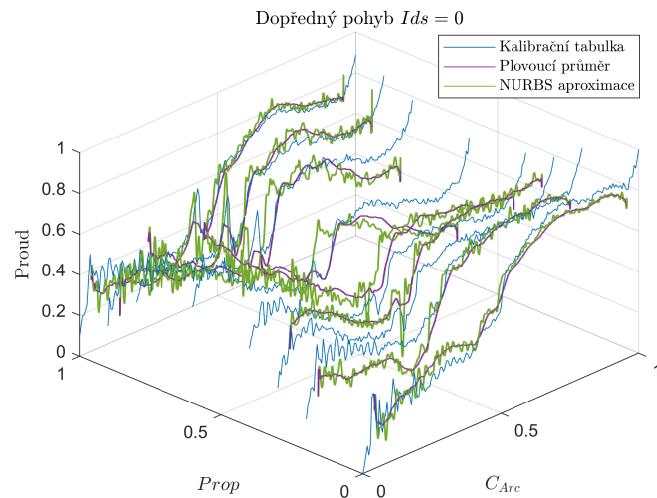


(c) Vypočtený kompenzační proud ze záznamu testovací trajektorie pro zpětný pohyb, $Ids = 0$

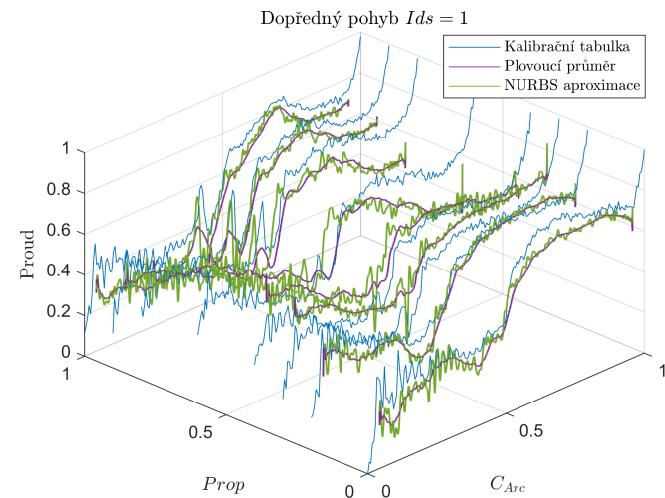


(d) Vypočtený kompenzační proud ze záznamu testovací trajektorie pro zpětný pohyb, $Ids = 1$

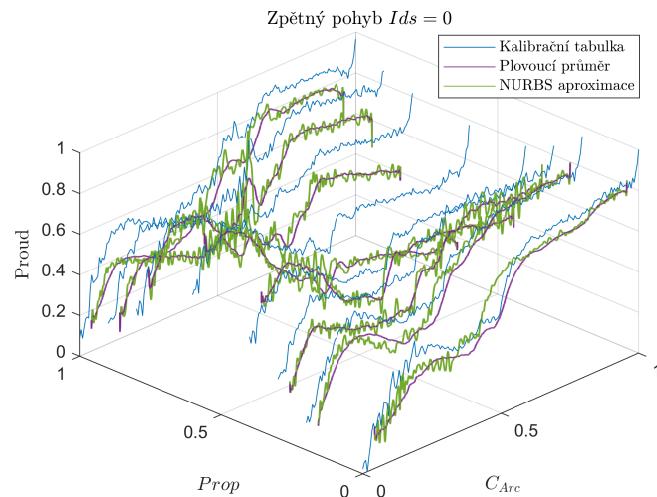
Obrázek 48: Vypočtený kompenzační proud pro testovací trajektorie



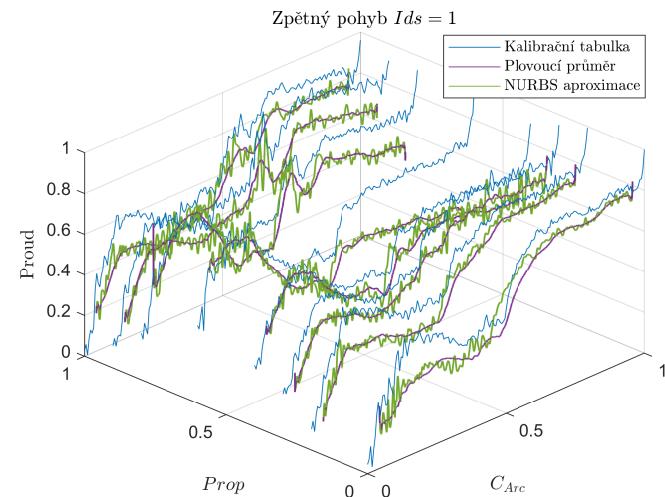
(a) Zpracovaný zašuměný záznam kompenzačního proudu pro dopředný pohyb, $Ids = 0$



(b) Zpracovaný zašuměný záznam kompenzačního proudu pro dopředný pohyb, $Ids = 1$

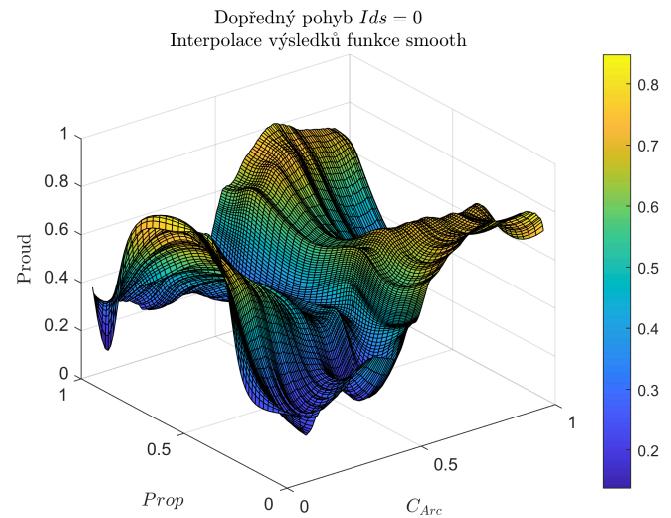


(c) Zpracovaný zašuměný záznam kompenzačního proudu pro zpětný pohyb, $Ids = 0$

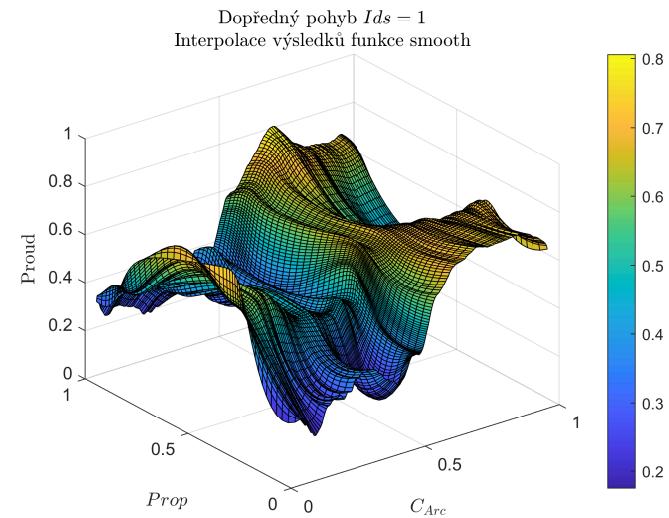


(d) Zpracovaný zašuměný záznam kompenzačního proudu pro zpětný pohyb, $Ids = 1$

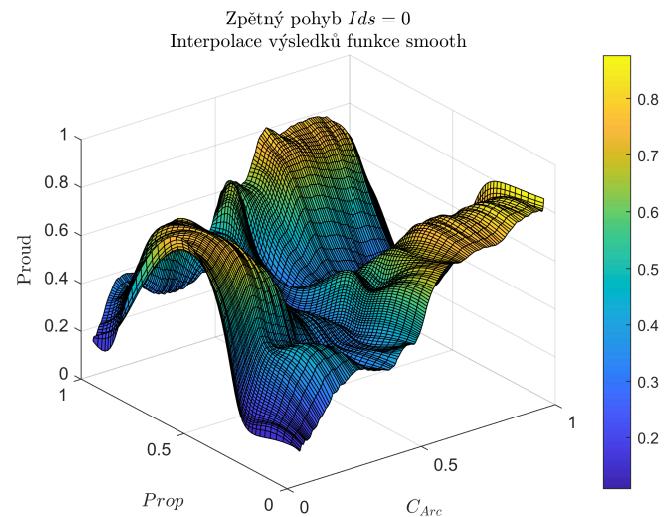
Obrázek 49: Zpracování zašuměných záznamů kompenzačního proudu pro testovací trajektorie



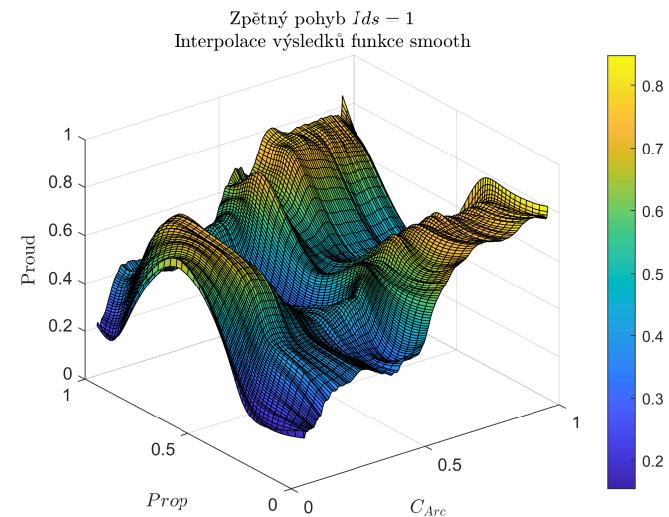
(a) NURBS interpolace zpracovaných dat pomocí funkce `smooth` pro dopředný pohyb, $Ids = 0$



(b) NURBS interpolace zpracovaných dat pomocí funkce `smooth` pro dopředný pohyb, $Ids = 1$

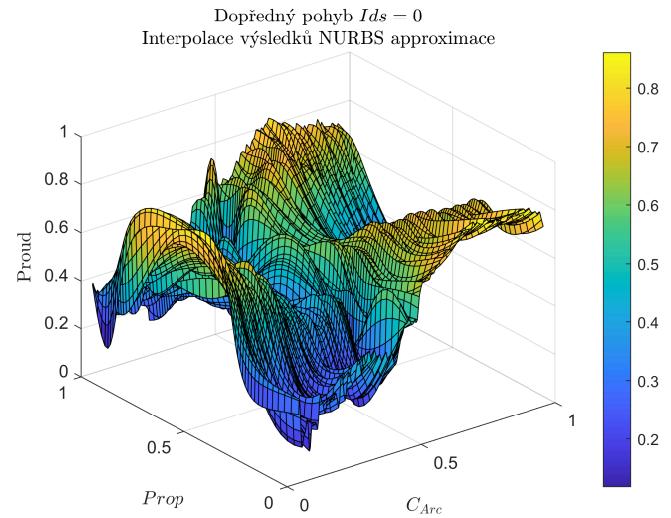


(c) NURBS interpolace zpracovaných dat pomocí funkce `smooth` pro zpětný pohyb, $Ids = 0$

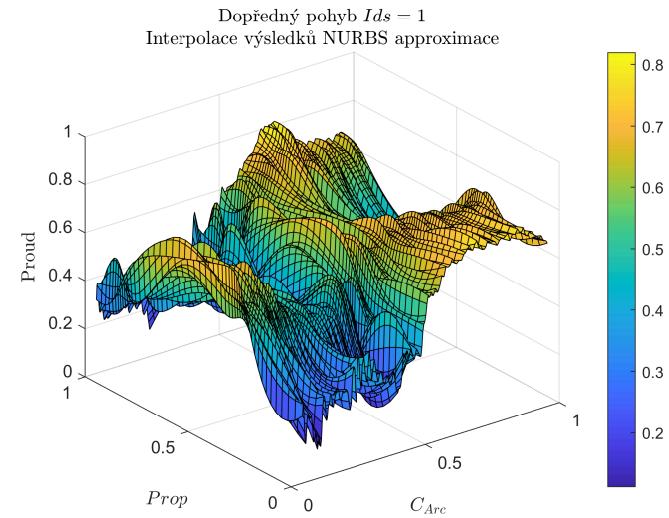


(d) NURBS interpolace zpracovaných dat pomocí funkce `smooth` pro zpětný pohyb, $Ids = 1$

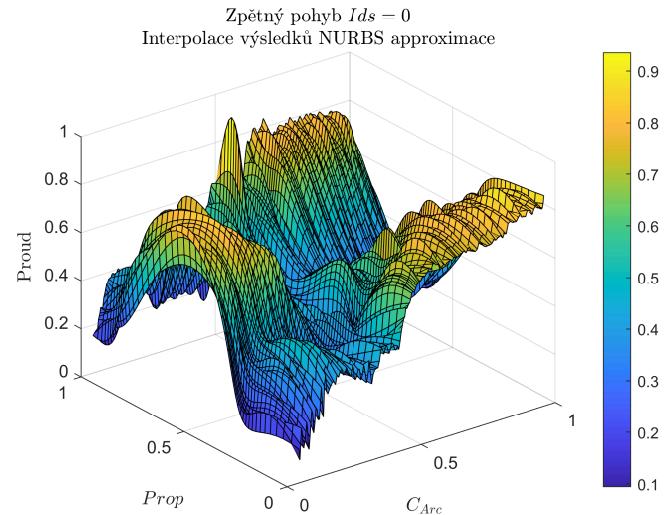
Obrázek 50: Proložení zpracovaných dat pomocí `smooth` NURBS interpolací



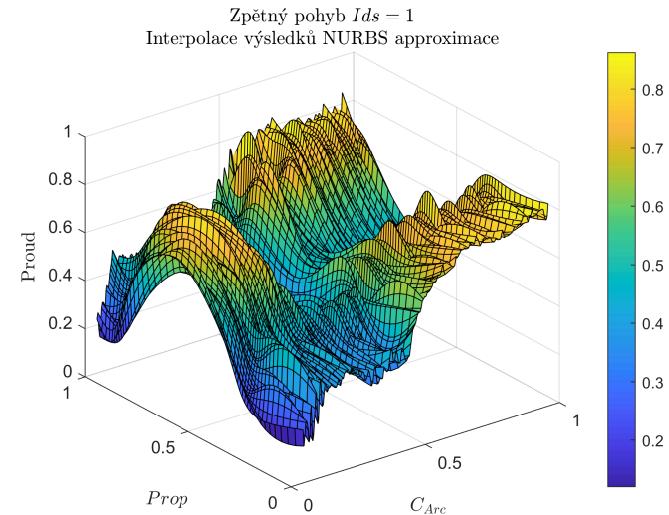
(a) NURBS interpolace zpracovaných dat pomocí NURBS approximace pro dopředný pohyb, $Ids = 0$



(b) NURBS interpolace zpracovaných dat pomocí NURBS approximace pro dopředný pohyb, $Ids = 1$



(c) NURBS interpolace zpracovaných dat pomocí NURBS approximace pro zpětný pohyb, $Ids = 0$



(d) NURBS interpolace zpracovaných dat pomocí NURBS approximace pro zpětný pohyb, $Ids = 1$

Obrázek 51: Proložení zpracovaných dat pomocí NURBS approximace NURBS interpolací

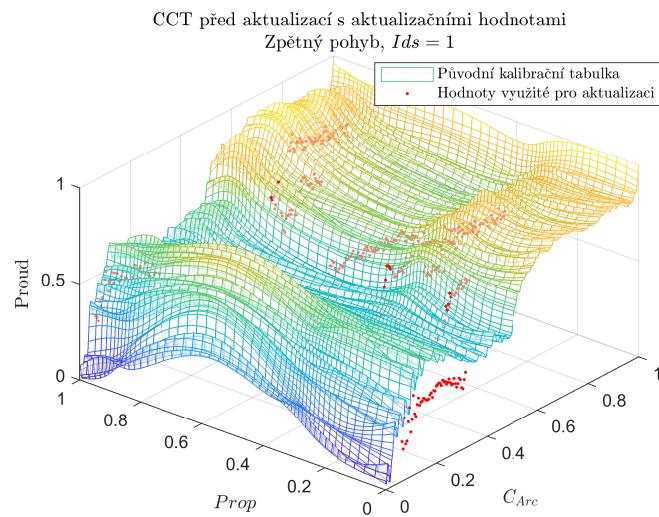
4.3 Aktualizace kalibrační tabulky

Na obrázcích č. [52a](#), [52b](#), [52c](#) a [52d](#) je ukázka výsledné aktualizace kalibrační tabulky užitím Gaussova funkce z kapitoly [NURBS approximace povrchu s užitím Gaussovy funkce](#). Jako zdrojová data pro aktualizační body, tj. nejnovější hodnoty potřebného kompenzačního proudu, jsou použity hodnoty z tabulky [NURBS interpolace zpracovaných dat pomocí NURBS approximace pro zpětný pohyb, \$Ids = 1\$](#) , kterou jsme získali analýzou záznamů pohybu manipulátoru. Byly vybrány takové body, na kterých je výsledný efekt vidět nejlépe z důvodu prezentace výsledků.

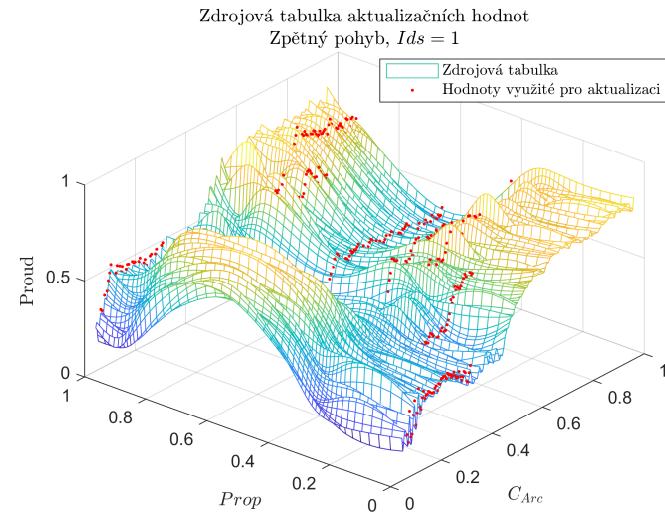
Jak lze z obrázku vidět, aktualizace tabulky poskytla uspokojivé výsledky. Nicméně Gaussova approximace v kombinaci s naší implementací sběru dat není bezvadná. Nasbírané body neobsahují hodnoty pro extrémy poloh kloubu C_{Arc} ^[22] (0 a 1) a to znamená, že zatím není možnost aktualizace těchto krajních hodnot.

Otestování aktualizace 4D proudové kalibrační tabulky užitím algoritmu z kapitoly [NURBS approximace 4D povrchu s užitím Gaussovy funkce](#) bohužel není možné z důvodu chybějícího záznamu manuálního ovládání stroje uživatelem pro více os.

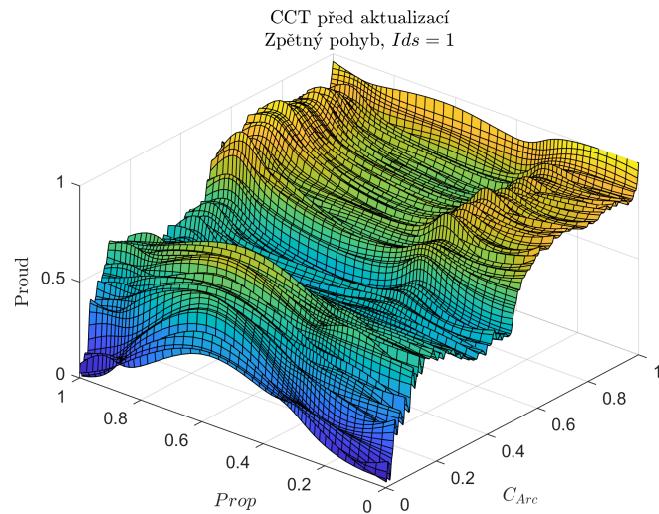
^[22]Toto je způsobeno naší filtrací konstantní rychlosti, jedná se o krajní případ, který je potřeba ještě dořešit.



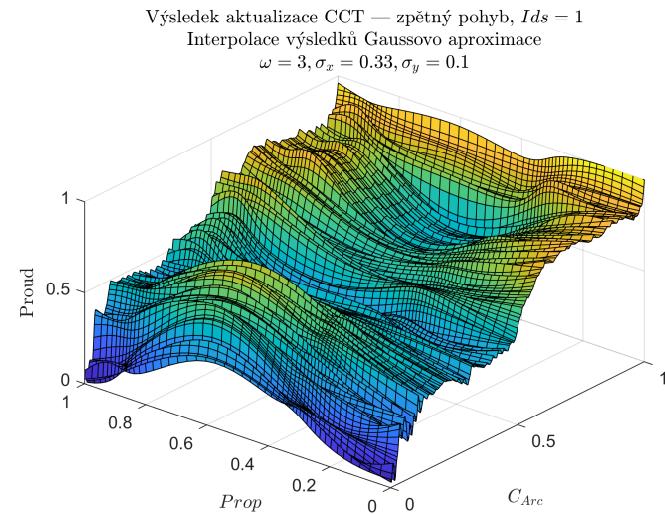
(a) Interpolace původní kalibrační tabulky s aktualizačními hodnotami



(b) Zdrojová tabulka aktualizačních hodnot — NURBS interpolace zpracovaných dat pomocí NURBS approximace pro zpětný pohyb, $Ids = 1$



(c) Interpolace původní kalibrační tabulky



(d) Výsledek aktualizace CCT s užitím Gaussovo approximace

Obrázek 52: Ukázka aktualizace CCT pro zpětný pohyb, $Ids = 1$ na základě měření získaného z analýzy testu odchylky proudu

5 Závěr

Cílem této práce bylo navrhnut automatickou aktualizaci proudové kalibrační tabulky na základě naměřených dat získaných za běžného užívání manipulátoru.

Nejprve jsme popsali řízený a řídící systém, včetně schématu regulační smyčky na kterém je ukázáno jak kalibrační tabulka spolupracuje s regulátorem. V další obsáhlé kapitole jsme rozebrali veškerou teorii k NURBS splinům, konkrétně NURBS 2D a 3D křivkám, 3D a 4D (nad)povrchům a k nim příslušné přístupy interpolace a approximace včetně jejich ukázek.

Poslední kapitola obsahuje již samostatné zpracování záznamů pohybu manipulátoru, které poskytla společnost Phillips za účelem tohoto výzkumu a vývoje. Podařilo se nám ověřit existenci vhodných bodů pro aktualizaci CCT při manuální operaci manipulátoru uživatelem. Pomocí těchto extrahovaných bodů jsme potom úspěšně sestavili vlastní verze kalibračních tabulek. Tyto nové verze tabulek jsme využily pro odzkoušení aktualizace původních verzí CCT.

změnit Na tuto práci by se dalo dále navázat dalším výzkumem, který by se pravděpodobně zabýval lepším přístupem k approximaci 3D/4D (nad)povrchu, který je tvořen body neležícími v mřížce — tj. například pokusit se na tento problém také aplikovat metodu nejmenších čtverců, aby vzniklé řešení bylo matematicky podmíněné. Dále je otázkou dalšího výzkumu rozsáhlejší sběr aktualizačních bodů pro CCT — momentálně námi navržený způsob sběru dat v kombinaci s naší 4D approximací pomocí Gaussova funkce téměř nikdy neaktualizuje body CCT ležící v mezních polohách kloubů.

Zdroje

- Phillips. *Azurion 7 C20*. 2023. URL: <https://www.philips.de/c-dam/corporate/newscenter/de/press-releases/2019/20190122-azurion-7-c20-mit-flexarm/phillips-azurion-7-c20-mit-flexarm-produkt2-un-hs-20190122-download.jpg> (cit. 14. 05. 2023).
- *Azurion 7 C20 with FlexArm Image-guided therapy system*. 2023. URL: <https://www.usa.philips.com/healthcare/product/HCNCVD207/azurion-7-c20-with-flexarm-image-guided-therapy-system> (cit. 14. 05. 2023).
- Prof. Dr. Les Piegl, Dr. Wayne Tiller (auth.) *The NURBS Book*. 2. vyd. Monographs in Visual Communication. Springer-Verlag Berlin Heidelberg, 1997. ISBN: 978-3-540-61545-3, 978-3-642-59223-2. URL: <https://link.springer.com/book/10.1007/978-3-642-59223-2>.