

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ

ÚSTAV MECHANIKY, BIOMECHANIKY A MECHATRONIKY



DIPLOMOVÁ PRÁCE

Řízení kinematické struktury Sliding Delta
Control of kinematic structure Sliding Delta

2015

Tomáš Kozák

Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu.

V Praze dne _____

podpis

Poděkování

Rád bych poděkoval vedoucímu své diplomové práce Ing. Janu Zavřelovi, Ph.D. za vloženou důvěru a optimismus. Velké díky si zaslouží také Ing. Pavel Steinbauer, Ph.D., jehož zásluhou jsem se dostal nejen k mechatronice, ale také ke spoustě praktických zkušeností během vlastního studia.

Největší poděkování ale míří směrem k mým rodičům, kteří mě celý život vytrvale podporují a umožňují mi věnovat se věcem, které mě baví.

Abstrakt

Tato diplomová práce je věnována řízení mechanismů s paralelní kinematickou strukturou a nadbytečnými pohony, které jsou zde zastoupeny robotem Sliding Delta. V první části je vytvořen stručný přehled metod, které lze k tomuto účelu použít a několik z nich je rozepsáno podrobněji. Jedná se o lineární kvadraticky optimální řízení, prediktivní řízení a momentové řízení. Krátce je zmíněn i návrh stavového pozorovatele, který je potřeba pro řízení laboratorního modelu mechanismu. Ve druhé části je sestaven matematický model jednoduchého 2D redundantního mechanismu, na kterém jsou popsány metody řízení vyzkoušeny. Po ověření jejich funkčnosti následuje rozšíření regulátorů pro použití na mechanismu Sliding Delta a znovu je simulačně ověřena jejich funkčnost. Nakonec je provedeno oživení laboratorního modelu a otestování prediktivního řízení v reálném čase. Výsledky experimentu jsou zhodnoceny a porovnány se simulacemi.

Abstract

This thesis is devoted to control of mechanisms with parallel kinematic structure and redundant actuation. Robot Sliding Delta is used as an example. Brief list of methods available for this purpose is made in the first chapter. Several approaches (linear-quadratic regulator, predictive regulator and computed torques method) are described in detail. State observer is mentioned as well, as it is needed for control of real laboratory model. Mathematical model of simple 2D redundant mechanism is created in the second chapter and it is used for tests of regulators mentioned earlier. Third chapter deals with application of created regulators on mechanism Sliding Delta. In the last chapter the laboratory model is completed and predictive regulator implemented as a real-time application. Experimental results are discussed and compared with simulations.

vložit originální zadání !!!!

Obsah

Seznam obrázků	ix
1 Úvod	1
1.1 Cíle bakalářské práce	2
2 Řízení redundantních kinematických struktur	3
2.1 Lineární kvadraticky optimální řízení (LQR)	5
2.2 Prediktivní řízení (MPC)	9
2.2.1 Diskretizace linearizovaného modelu	10
2.2.2 Vlastní návrh řídicího zákona	12
2.3 Stavový pozorovatel	14
2.4 Momentové řízení (CTC)	16
3 Kinematická struktura 2Delta	21
3.1 Sestavení matematického modelu	21
3.1.1 Inverzní úloha kinematiky	21
3.1.2 Úloha dynamiky	24
3.1.3 Úloha statiky	26
3.2 Implementace matematického modelu	28
3.2.1 Stavový popis a jeho linearizace	30
3.2.2 Vlastnosti linearizovaného modelu	31
3.3 Stabilizace v rovnovážné poloze	31
3.3.1 Stabilizace pomocí LQR	31
3.3.2 Stabilizace pomocí MPC	34
3.3.3 Stabilizace pomocí CTC	37
3.3.4 Shrnutí výsledků stabilizace	40
3.4 Sledování referenční trajektorie	40

3.4.1	Řízení pomocí LQR	41
3.4.2	Řízení pomocí MPC	42
3.4.3	Řízení pomocí CTC	42
3.4.4	Shrnutí a porovnání regulátorů	44
4	Kinematická struktura Sliding Delta	45
4.1	Matematický model Sliding Delta	46
4.1.1	Linearizace a vlastnosti matematického modelu	49
4.2	Aplikace zvolených metod řízení	51
4.2.1	Řízení pomocí LQR	52
4.2.2	Řízení pomocí MPC	53
4.2.3	Řízení pomocí CTC	57
4.2.4	Shrnutí a porovnání regulátorů	58
5	Oživení laboratorního modelu Sliding Delta	59
5.1	Použitý hardware a software	59
5.2	Nastavení a úprava modelu regulátoru	61
5.3	Výsledky a porovnání se simulací	64
6	Závěr	69
	Literatura	71
A	Parametry modelů mechanismů	I
B	Prvky matic stavového popisu 2Delta v různých polohách	V
C	Stabilizace mechanismu 2Delta	VII
D	Sledování trajektorie - 2Delta	XI
E	Sledování trajektorie - Sliding Delta	XV
F	Oživování laboratorního modelu	XIX
G	Experimentální výsledky	XXIII
H	Obsah přiloženého CD	XXVII

Seznam obrázků

2.1	Schéma decentralizovaného řízení s ošetřením přetahování.	4
2.2	Schéma centralizovaného řízení.	4
2.3	Principiální schéma řízení metodou Computed torques.	17
3.1	Schéma mechanismu 2Delta.	22
3.2	Uvolnění jednotlivých těles.	27
3.3	„dynamics“- model dynamiky mechanismu 2Delta.	29
3.4	„position“- řešení úlohy polohy mechanismu 2Delta.	29
3.5	„statics“- řešení úlohy statiky mechanismu 2Delta.	30
3.6	Schéma stabilizace mechanismu 2Delta pomocí LQR.	32
3.7	Průběh stabilizace stavů pomocí LQR.	33
3.8	Průběh omezených výstupů při stabilizaci pomocí LQR.	33
3.9	Schéma stabilizace mechanismu 2Delta pomocí MPC.	34
3.10	Schéma bloku MPC.	35
3.11	Průběh stabilizace stavů pomocí MPC.	36
3.12	Průběh omezených výstupů při stabilizaci pomocí MPC.	37
3.13	Schéma stabilizace mechanismu 2Delta pomocí CTC.	38
3.14	Schéma bloku 2Delta rozšířené o inverzní kinematiku.	38
3.15	Vnitřní schéma bloku CTC.	39
3.16	Průběh stabilizace stavů pomocí CTC.	39
3.17	Průběh omezených výstupů při stabilizaci pomocí CTC.	39
3.18	Zvolená stavová trajektorie pro mechanismus 2Delta.	41
3.19	Kompenzační blok pro řízení mechanismu 2Delta.	42
3.20	Průběh odchylek stavů při sledování trajektorie LQR.	43
3.21	Průběh odchylek stavů při sledování trajektorie MPC.	43
3.22	Průběh odchylek stavů při sledování trajektorie CTC.	43
4.1	Základní schéma mechanismu Sliding Delta.	45

4.2	Simulační model Sliding Delta.	49
4.3	Referenční trajektorie v souřadnicích xy.	51
4.4	Schéma stavového pozorovatele.	52
4.5	Průběh odchylek souřadnic platformy - LQR.	54
4.6	Průběh odchylek souřadnic pohonů - LQR.	54
4.7	Průběh odchylek souřadnic platformy - MPC.	55
4.8	Průběh odchylek souřadnic pohonů - MPC.	55
4.9	Průběh odchylek souřadnic platformy - CTC.	57
4.10	Průběh odchylek souřadnic pohonů - CTC.	57
4.11	Srovnání absolutních odchylek polohy pro různé regulátory.	58
5.1	Pracoviště pro oživení laboratorního modelu Sliding Delta.	60
5.2	Schéma ovládacího rozhraní v ControlDesku.	63
5.3	Žádaná a skutečná trajektorie platformy, poloměr 50 mm. V pravo pro zatíženou platformu.	64
5.4	Průběhy souřadnic pohonů při sledování kružnice s poloměrem 50 mm.	65
5.5	Odchytky souřadnic pohonů při sledování kružnice s poloměrem 50 mm.	65
5.6	Průběhy proudů při sledování kružnice s poloměrem 50 mm.	65
5.7	Průběhy souřadnic pohonů při sledování kružnice s poloměrem 50 mm, zatížená platforma.	67
5.8	Odchytky souřadnic pohonů při sledování kružnice s poloměrem 50 mm, zatížená platforma.	67
5.9	Průběhy proudů při sledování kružnice s poloměrem 50 mm, zatížená platforma.	67
5.10	Schéma mechanismu Sliding Delata s vykreslenými pohony.	68
A.1	Schéma mechanismu 2Delta.	II
A.2	Zjednodušené schéma Sliding Delta.	III
A.3	Zjednodušené schéma platformy a vozíku lineárního vedení.	III
B.1	Prvek A_{21} matice \mathbf{A} pro různé polohy.	V
B.2	Prvek B_{21} matice \mathbf{B} pro různé polohy.	VI
B.3	Prvek B_{22} matice \mathbf{B} pro různé polohy.	VI
C.1	Průběh odchylek stavů v průběhu stabilizace LQR.	VIII
C.2	Průběh odchylek stavů v průběhu stabilizace MPC.	VIII

C.3	Průběh odchylek stavů v průběhu stabilizace CTC.	IX
C.4	Detail průběhu odchylek stavů v průběhu stabilizace CTC.	IX
D.1	Simulační schéma pro sledování trajektorie LQR.	XII
D.2	Simulační schéma pro sledování trajektorie MPC.	XII
D.3	Simulační schéma pro sledování trajektorie CTC.	XIII
D.4	Průběh omezených výstupů při řízení pomocí LQR.	XIII
D.5	Průběh omezených výstupů při řízení pomocí MPC.	XIV
D.6	Průběh omezených výstupů při řízení pomocí CTC.	XIV
E.1	Simulační schéma pro sledování trajektorie LQR - SD.	XVI
E.2	Simulační schéma pro sledování trajektorie MPC - SD.	XVI
E.3	Simulační schéma pro sledování trajektorie CTC - SD.	XVII
E.4	Průběh omezených výstupů při řízení pomocí LQR - SD.	XVII
E.5	Průběh omezených výstupů při řízení pomocí MPC - SD.	XVIII
E.6	Průběh omezených výstupů při řízení pomocí CTC - SD.	XVIII
F.1	Laboratorní model mechanismu Sliding Delta.	XIX
F.2	Schéma regulátoru MPC pro běh v reálném čase.	XXI
G.1	Žádaná a skutečná trajektorie platformy, poloměr 100 mm. V pravo pro zatíženou platformu.	XXIII
G.2	Průběhy souřadnic pohonů při sledování kružnice s poloměrem 100 mm. .	XXIV
G.3	Průběhy souřadnic pohonů při sledování kružnice s poloměrem 100 mm, zatížená platforma.	XXIV
G.4	Odchytky souřadnic pohonů při sledování kružnice s poloměrem 100 mm.	XXV
G.5	Odchytky souřadnic pohonů při sledování kružnice s poloměrem 100 mm, zatížená platforma.	XXV
G.6	Průběhy proudů při sledování kružnice s poloměrem 100 mm.	XXVI
G.7	Průběhy proudů při sledování kružnice s poloměrem 100 mm, zatížená platforma.	XXVI

Kapitola 1

Úvod

Tato diplomová práce přímo navazuje na moji práci bakalářskou (KOZÁK, T., 2012), která byla věnována přehledu paralelních kinematických struktur (PKS) a jejich vlastnostem. Byl v ní také sestaven simulační model redundantního mechanismu se třemi translačními stupni volnosti - Sliding Delta. Pro potřeby bakalářské práce byl však tento model zjednodušen na neredundantní a byla provedena pouze simulace řízení bez praktického ověření na laboratorním modelu. Hlavním smyslem této diplomové práce tak je dokončit započatý projekt a přenést teoretické poznatky do fyzické reality.

Jako jedna z hlavních nevýhod paralelních kinematik, zejména pak těch redundantních, se uvádí vyšší náročnost jejich řízení. V případě neredundantních kinematik je to dáno především tím, že pro využití jejich plného potenciálu je třeba použít pokročilejší metody řízení než u sériových robotů a manipulátorů. Vyšší konstrukční náročnost a malý pracovní prostor by jinak byly obtížně obhajitelné. V případě redundantních variant pak může při klasických přístupech dojít až ke ztrátě říditelnosti v důsledku saturace pohonů. Ta je způsobena statickou přeurčeností systému a geometrickou nepřesností mechanismu, což vede ke vzájemnému přetahování pohonů místo k jejich spolupráci. Metody řízení PKS jsou proto již řadu let předmětem výzkumu. I přes mnohé potíže si paralelní roboty postupně našly své místo v průmyslové sféře a s vývojem v oblasti výpočetní techniky se jejich potenciál dále zvětšuje. V současné době jsou PKS až na výjimky nasazené jako manipulátory a těží tak především z velké dosažitelné dynamiky. Existuje ale i snaha zvýšit jejich přesnost a tuhost pro nasazení jako obráběcí stroje. Některé přístupy k řízení tohoto typu struktur budou vyzkoušeny právě na mechanismu Sliding Delta.

1.1 Cíle bakalářské práce

Cíle bakalářské práce vycházejí ze zásad pro vypracování, které jsou uvedeny v zadání:

1. Vypracujte přehled různých způsobů řízení paralelních kinematických struktur s nadbytečnými pohony (kapitola 2).
2. Vytvořte model jednoduchého redundantního mechanismu a proved'te simulační ověření funkčnosti vybraných způsobů řízení (kapitola 3).
3. Vytvořte simulační model mechanismu Sliding Delta a aplikujte na něj ověřené metody řízení (kapitola 4).
4. Oživte laboratorní model mechanismu Sliding Delta, ověřte na něm funkčnost navrženého řízení a jeho chování porovnejte se simulací (kapitola 5).

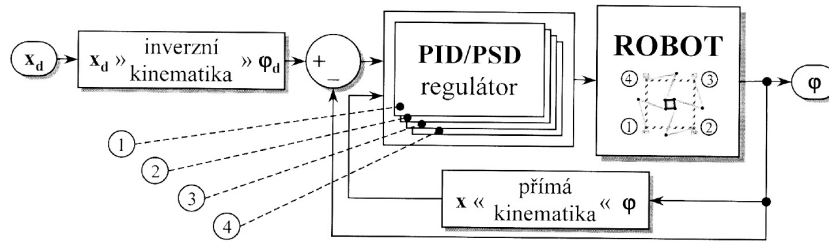
Kapitola 2

Řízení redundantních kinematických struktur

Jak již bylo zmíněno v úvodu, postupem času bylo k řízení PKS vyvinuto několik přístupů. Nejjednodušší a nejméně efektivní variantou je *decentralizované* řízení, které pohlíží na každý pohon jako na nezávislý systém s jedním vstupem a jedním výstupem (SISO¹). Každému pohonu tak náleží jeden nezávislý PID/PSD regulátor, který na základě inverzní kinematiky řídí příslušnou pohonovou souřadnici. Vliv chování zbytku soustavy je z pohledu regulátoru považován za poruchovou veličinu. Zřejmou nevýhodou tak je fakt, že jednotlivé regulátory nemají informaci o vzájemných interakcích a nemohou tedy spolupracovat. Logicky tak nelze dosáhnout kvalitního řízení. V případě redundantních systémů navíc vlivem geometrických nepřesností dochází ke stavu, kdy není možné přesně dosáhnout požadovaných poloh, což vede k nárůstu integračních členů regulátorů a bez ošetření tohoto jevu dochází k saturaci pohonů. To pak může vést až k poškození pohonů či samotného mechanismu. Řešením může být například omezení maximální velikosti integrační složky nebo lépe přepočítání vypočtených akčních zásahů vhodnou transformací. Bylo by také možné se bez integrační složky obejít úplně, ale vzhledem ke vzniku trvalé regulační odchylky nelze tento přístup použít k přesnému řízení.

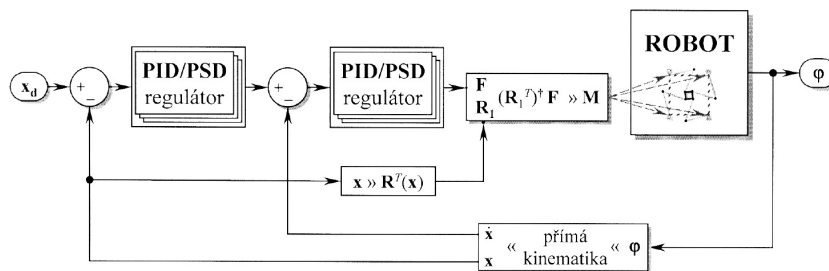
Protože jednotlivé pohony ve skutečnosti nezávislé nejsou, je nanejvýš vhodné nasadit řízení na vyšší (globální) úrovni - *centralizované*. To může stále pracovat s regulátory typu PID/PSD, ale zohledňuje kinematické vazby mezi tělesy mechanismu. První vrstva řízení se v tomto případě nachází na úrovni nezávislých souřadnic. Regulátor vypočte fik-

¹Single Input Single Output.



Obrázek 2.1: Schéma decentralizovaného řízení s ošetřením přetahování.

tivní akční zásahy, které přepočteme na skutečné akční zásahy očekávané na pohonech. Výhodou je, že v případě řízení na úrovni nezávislých souřadnic nemůže, vzhledem k jejich zaručené dosažitelnosti v pracovním prostoru, docházet k přetahování a následné saturaci pohonů. Ani tento přístup však nepřináší dostatečně efektivní řízení, které by bylo schopné vytěžit z paralelních struktur maximum.



Obrázek 2.2: Schéma centralizovaného řízení.

Takové řízení nám mohou poskytnout až globální přístupy vycházející ze znalosti dynamického modelu systému - *modelově orientované řízení*. Do této kategorie patří řada různých metod návrhu, například řízení inverzní dynamikou, klouzavé řízení (SMC²), kvadraticky optimální řízení (LQR³), stále populárnější prediktivní řízení (MPC⁴), nebo momentové řízení (CTC⁵). Všechny tyto přístupy jsou výpočetně náročnější, což ale vzhledem k dnes dostupnému hardware přestává být problém. Za cenu složitějšího návrhu a nutnosti dobře znát model systému však získáváme mnohem robustnější řízení, které pro redundantní systémy navíc umožňuje využít nadbytečný pohon ke splnění dodatečných požadavků. Můžeme tak například zajistit předepětí mechanismu, které vymezí

²Sliding Mode Control

³Linear-Quadratic Regulator.

⁴Model Predictive Control.

⁵Computed-Torques Control.

konstrukční vůle a zvýší tuhost. Pro řízení mechanismu Sliding Delta jsem pro účely této práce zvolil přístupy LQR, MPC a CTC a ty budou v následujících sekcích blíže představeny. (BELDA, K., 2002) (SKOPEC, T., 2004)

2.1 Lineární kvadraticky optimální řízení (LQR)

LQR je jedním z produktů teorie optimálního řízení (VALÁŠEK, M. a kol., 1995). Jeho návrh je postaven kolem lineárního stavového modelu a minimalizace kvadratického kritéria. Cílem je nalézt stabilizující zesílení stavové zpětné vazby \mathbf{K}

$$\mathbf{u}_t - \mathbf{u}_0 = -\mathbf{K}(\mathbf{x}_t - \mathbf{x}_0). \quad (2.1)$$

Kritérium J vážící rychlost stabilizace stavů a velikost akčních zásahů má následující tvar:

$$J = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt. \quad (2.2)$$

Jeho minimalizace na nekonečném horizontu vede na řešení algebraické Riccatiho rovnice. Pro potřeby této práce se nebudeme zabývat její teorií a řešením, protože pro návrh tohoto regulátoru je již v Matlabu přímo implementována funkce, která vše potřebné řeší. Je tak třeba pouze sestavit model mechanismu, linearizovat ho v okolí zvoleného pracovního bodu a zvolit váhové matice \mathbf{Q} (pozitivně definitní) a \mathbf{R} (pozitivně semidefinitní). Předpokladem navíc je plná říditelnost systému. Samotný příkaz se pak zapíše takto:

$$\mathbf{K} = \text{lqr}(\mathbf{A}, \mathbf{B}, \mathbf{Q}, \mathbf{R}).$$

LQR je regulátor robustní a měl by si být schopen poradit i s nepřesnostmi laboratorního modelu. Protože ale regulátor navrhujeme pro lineární model, všude mimo pracovní (linearizační) bod můžeme očekávat vznik trvalých regulačních odchylek. Pro přesnější regulaci je tak vhodné použít kompenzaci akčních zásahů vycházející ze statiky nebo lépe z inverzní dynamiky systému. Tato kompenzace je popsána spolu s implementací celého modelu v Simulinku v kapitole 3.

Popišme nyní postupně kroky návrhu. Vše začíná sestavením dynamického modelu mechanismu, k čemuž lze použít například Lagrangeovy rovnice smíšeného typu (LRST). V kapitole 3 jsou postupně popsány a provedeny všechny kroky jejich sestavení a převodu na popis pomocí nezávislých souřadnic, výsledkem čehož je následující maticová rovnice:

$$\ddot{\mathbf{q}} = (\mathbf{R}^T \mathbf{M} \mathbf{R})^{-1} (\mathbf{R}^T \mathbf{F} + \mathbf{R}^T \mathbf{G} - \mathbf{R}^T \mathbf{M} \dot{\mathbf{R}} \dot{\mathbf{q}}). \quad (2.3)$$

Tento systém rovnic je dále potřeba převést na stavový popis. Abychom to mohli provést, rozdělme pravou stranu na dvě části, kde jedna bude obsahovat vstupy $\mathbf{u} = \mathbf{F}$ a druhá zbytek:

$$\ddot{\mathbf{q}} = (\mathbf{R}^T \mathbf{M} \mathbf{R})^{-1} (\mathbf{R}^T \mathbf{G} - \mathbf{R}^T \mathbf{M} \dot{\mathbf{R}} \dot{\mathbf{q}}) + (\mathbf{R}^T \mathbf{M} \mathbf{R})^{-1} \mathbf{R}^T \mathbf{F}. \quad (2.4)$$

Takto rozdělenou rovnicí lze dále zapsat zjednodušeně jako

$$\ddot{\mathbf{q}} = \mathbf{f}_{(\mathbf{q}, \dot{\mathbf{q}})}^* + \mathbf{g}_{(\mathbf{q})}^* \mathbf{u}. \quad (2.5)$$

S takto připravenou rovnicí se podíváme jak vypadá samotný stavový popis. Jeho principem je převod soustavy rovnic n -tého řádu na soustavu n rovnic řádu prvního, se kterou se lépe pracuje a metody jejího řešení jsou také dobře známé. Samotný stavový popis se skládá ze dvou obecně nelineárních rovnic, diferenciální stavové

$$\dot{\mathbf{x}}_t = \mathbf{f}_{(\mathbf{x}_t)} + \mathbf{g}_{(\mathbf{x}_t)} \mathbf{u}_t \quad (2.6)$$

a algebraické výstupní

$$\mathbf{y}_t = \mathbf{h}_{(\mathbf{x}_t, \mathbf{u}_t)}. \quad (2.7)$$

Vektor \mathbf{x} představuje stavový vektor⁶, \mathbf{y} je vektor výstupů a \mathbf{u} vektor vstupů. Přejít od \mathbf{q} k \mathbf{x} je přitom pro provedení substitucí

$$\begin{aligned} \mathbf{x}_1 &= \mathbf{q}, \\ \mathbf{x}_2 &= \dot{\mathbf{q}}. \end{aligned} \quad (2.8)$$

⁶Stav je soubor vnitřních veličin systému. Z jeho znalosti a znalosti vnějšího působení na systém můžeme určit budoucí chování tohoto systému. Pro mechanický systém jsou stavovými veličinami obvykle nezávislé souřadnice a rychlosti.

Po derivaci těchto výrazů dostáváme zřejmě

$$\begin{aligned}\dot{\mathbf{x}}_1 &= \dot{\mathbf{q}} = \dot{\mathbf{x}}_2, \\ \dot{\mathbf{x}}_2 &= \ddot{\mathbf{q}}.\end{aligned}\tag{2.9}$$

Dosazením rovnice (2.5) do (2.6) tak získáváme významy jednotlivých členů stavové rovnice, jak vyplývají z rovnic dynamiky:

$$\begin{bmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_2 \\ \mathbf{f}^*(\mathbf{x}) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{g}^*(\mathbf{x}) \end{bmatrix} \mathbf{u}.\tag{2.10}$$

Výstupní rovnici (2.7) lze ještě upravit s přihlédnutím ke skutečnosti, že se zabýváme ryze dynamickými systémy⁷ a funkce \mathbf{g} je tak pouze funkcí stavů systému:

$$\mathbf{y}_t = \mathbf{h}_{(\mathbf{x}_t)}.\tag{2.11}$$

S takto definovaným nelineárním stavovým popisem můžeme přistoupit k jeho linearizaci. Jednou z možností, jak to provést, je zvoleném pracovním bodě použít Taylorův rozvoj se zanedbáním nelineárních členů:

$$\mathbf{f}_{(\mathbf{x}_t)} \cong \mathbf{f}_{(\mathbf{x}_0)} + \left. \frac{\partial \mathbf{f}_{(\mathbf{x}_t)}}{\partial \mathbf{x}_t} \right|_0 (\mathbf{x}_t - \mathbf{x}_0),\tag{2.12}$$

$$\mathbf{g}_{(\mathbf{x}_t)} \mathbf{u}_t \cong \mathbf{g}_{(\mathbf{x}_0)} \mathbf{u}_0 + \left. \frac{\partial \mathbf{g}_{(\mathbf{x}_t)} \mathbf{u}_t}{\partial \mathbf{u}_t} \right|_0 (\mathbf{u}_t - \mathbf{u}_0).\tag{2.13}$$

$$\mathbf{h}_{(\mathbf{x}_t)} \cong \mathbf{h}_{(\mathbf{x}_0)} + \left. \frac{\partial \mathbf{h}_{(\mathbf{x}_t)}}{\partial \mathbf{x}_t} \right|_0 (\mathbf{x}_t - \mathbf{x}_0).\tag{2.14}$$

Členy $\mathbf{f}_{(\mathbf{x}_0)}$, $\mathbf{g}_{(\mathbf{x}_0)}$ a $\mathbf{h}_{(\mathbf{x}_0)}$ porovnáme s původními výrazy (2.6) a (2.7), odkud je zřejmé, že

$$\dot{\mathbf{x}}_0 = \mathbf{f}_{(\mathbf{x}_0)} + \mathbf{g}_{(\mathbf{x}_0)} \mathbf{u}_0,\tag{2.15}$$

$$\mathbf{y}_0 = \mathbf{h}_{(\mathbf{x}_0)}.\tag{2.16}$$

⁷Systém, který nemá přímou vazbu mezi vstupy a výstupy.

Po dosazení do (2.6) a (2.7) a převedení $\dot{\mathbf{x}}_0$ a \mathbf{y}_0 na levé strany rovnic dostaneme finální tvar lineárních rovnic stavového popisu v odchylkovém tvaru:

$$\dot{\mathbf{x}}_t - \dot{\mathbf{x}}_0 = \mathbf{A}(\mathbf{x}_t - \mathbf{x}_0) + \mathbf{B}(\mathbf{u}_t - \mathbf{u}_0), \quad (2.17)$$

$$\mathbf{y}_t - \mathbf{y}_0 = \mathbf{C}(\mathbf{x}_t - \mathbf{x}_0) + \mathbf{D}(\mathbf{u}_t - \mathbf{u}_0). \quad (2.18)$$

Původní výrazy s parciálními derivacemi jsme přitom nahradili podle zvyklosti písmeny \mathbf{A} až \mathbf{C} . Matice \mathbf{A} je matice systémová, \mathbf{B} matice buzení (řízení) a \mathbf{C} je matice výstupní. Vektory $\dot{\mathbf{x}}_0$, \mathbf{x}_0 , \mathbf{u}_0 a \mathbf{y}_0 odpovídají rovnovážné poloze. Protože rovnovážná poloha je statická platí navíc, že $\dot{\mathbf{x}}_0 = \mathbf{0}$.

Protože však simulace a návrh řízení budeme provádět v Matlabu, můžeme si celý proces usnadnit využitím funkcí v tomto prostředí implementovaných. Samotný proces převodu na stavový popis a následné linearizace lze tak provést příkazem

$$[\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}] = \text{linmod}(\text{'dynamics'}, \mathbf{x}_0, \mathbf{u}_0).$$

Je tedy pouze potřeba vytvořit v Simulinku model dynamického systému („dynamics“) a zvolit vektory \mathbf{x}_0 a \mathbf{u}_0 pro rovnovážnou polohu.

Nesmíme zapomenout na podmínku, za které lze LQR (a obecně stavovou zpětnou vazbu) vůbec použít. Touto podmínkou je plná *řiditelnost* systému, která zaručuje, že existuje řízení \mathbf{u}_t , které v konečném čase převede systém z libovolného počátečního stavu do stavu rovnováhy. Lineární spojitý systém je pak plně říditelný, když je hodnota matice říditelnosti rovna řádu systému, tedy počtu prvků stavového vektoru \mathbf{x} . Matice říditelnosti je definována jako

$$\mathbf{R} = [\mathbf{B}, \mathbf{A}\mathbf{B}, \mathbf{A}^2\mathbf{B}, \dots, \mathbf{A}^{n-1}\mathbf{B}], \quad (2.19)$$

kde n je řád soustavy. Hodnota této matice v MATLABu zjistíme příkazem

$$\text{hR} = \text{rank}(\text{ctrb}(\mathbf{A}, \mathbf{B})).$$

2.2 Prediktivní řízení (MPC)

Modernější metoda MPC spočívá v predikci výstupů linearizovaného modelu v daných časových okamžicích. MPC kombinuje zpětnovazební a přímovazební řízení, kdy se provede predikce chování systému za základě znalosti požadované trajektorie a aktuálního stavu systému bez možnosti měřit následné aktuální stavy (přímá vazba). Z predikované posloupnosti akčních zásahů se však použije pouze první člen a celý výpočet se opakuje pro nový stav systému (zpětná vazba). Protože predikce probíhá v diskretních časových okamžicích, je nutné oproti LQR navíc provést diskretizaci stavového modelu. Proces linearizace je zde také potřeba provádět v každém výpočetním kroku, není proto účelné použít příkaz *linmod*, protože jeho provedení trvá příliš dlouho a není tak kompatibilní s požadavkem na dokončení výpočtu regulátoru v jednom časovém kroku. Využijeme tedy numerický proces, který spočívá v dekompozici založené na diferencích v určitém stavu, který se pak může (a bude) v průběhu řízení měnit. Původní stavovou rovnici (2.6) můžeme přepsat do následujícího tvaru:

$$\dot{\mathbf{x}}_t = \mathbf{f}_{(\mathbf{x}_t)} + \mathbf{B}_{(\mathbf{x}_t)} \mathbf{u}_t \quad (2.20)$$

To lze provést díky nezávislosti funkce \mathbf{g} na vstupech \mathbf{u} , což znamená, že matice $\mathbf{B}_{(\mathbf{x}_t)}$ je přímo rovna funkci $\mathbf{g}_{(\mathbf{x}_t)}$ a její aktuální linearizaci tak dostaneme přímo dosazením aktuálního stavu \mathbf{x}_t . Dekompozici budeme proto provádět pouze pro funkci $\mathbf{f}_{(\mathbf{x}_t)}$. Jejím cílem je nalézt následující rozklad:

$$\mathbf{f}_{(\mathbf{x})} = \mathbf{A}_{(\mathbf{x})} \mathbf{x} \quad (2.21)$$

Začneme s předpokladem, že existuje referenční stav \mathbf{x}_r , pro který platí, že

$$\mathbf{f}_{(\mathbf{x}_r)} = \mathbf{0} \quad (2.22)$$

Dále zvolíme pořadí dekompozice se zohledněním množství informací o vazbách. Pokud stav odkazuje pouze na stavovou proměnou, což je případ stavu \mathbf{x}_1 ⁸, jeho priorita pro dekompozici je nižší. Začneme tedy dekomponovat podle stavu \mathbf{x}_2 . Pro jednoduchost ukažme princip na systému s jedním stupněm volnosti, který má pouze dva stavy x_1 a x_2 :

$$\mathbf{A}_{(\mathbf{x})} = \left[\begin{array}{c} \frac{\mathbf{f}_{([x_1, x_{2r}])} - \mathbf{f}_{([x_{1r}, x_{2r}])}}{.x_1}, \quad \frac{\mathbf{f}_{([x_1, x_2])} - \mathbf{f}_{([x_1, x_{2r}])}}{.x_2} \end{array} \right] \quad (2.23)$$

⁸Jedná se o vektor všech nezávislých souřadnic mechanismu, které jsou obsaženy v rovnici (2.3).

Tečka před stavem při dělení značí v souladu se syntaxí Matlabu dělení po prvcích. Že předchozí rovnice platí lze dovézt roznásobením vektorem stavu, kdy dostaneme

$$\mathbf{f}_{(\mathbf{x})} = \mathbf{f}_{([x_1, x_{2r}])} - \mathbf{f}_{([x_{1r}, x_{2r}])} + \mathbf{f}_{([x_1, x_2])} - \mathbf{f}_{([x_1, x_{2r}])}. \quad (2.24)$$

Výrazy obsahující částečné referenční stavy se odečtou a $\mathbf{f}_{([x_{1r}, x_{2r}])}$ je podle předpokladu (2.22) roven vektorové nule. Navržená dekompozice bohužel přináší kvůli dělení stavy vést nepříjemnou situaci, kdy při nulovosti stavu dochází k dělení nulou. To lze ošetřit nahrazením nulových stavů nenulovými hodnotami, což můžeme provést, protože jejich nahrazení se projeví i ve funkci $\mathbf{f}_{(\mathbf{x})}$ a při dělení během dekompozice se tak vykompenzuje. Matice $\mathbf{C}_{(\mathbf{x})}$ je často pouze konstantní výběrovou maticí, není tedy třeba ji linearizovat. V opačném případě na ni aplikujeme podobnou dekompozici jako v případě matice $\mathbf{A}_{(\mathbf{x})}$.

2.2.1 Diskretizace linearizovaného modelu

Dalším nezbytným krokem na cestě k návrhu řídicího zákona je diskretizace stavového modelu, který jsme v předchozí sekci linearizovali. Pro diskretizaci budeme matice stavového popisu považovat za konstantní, vycházíme tedy ze tvaru

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u}. \quad (2.25)$$

Samotná diskretizace, vhodná pro použití algoritmy reálného času, je založena na řešení soustav diferenciálních rovnic prvního řádu, což je přesně případ naší rovnice (2.25). Výsledek takovéto rovnice hledáme ve tvaru

$$\mathbf{x}_{(t)} = \mathbf{\Phi}_{(t)} \mathbf{c}_{(t)}. \quad (2.26)$$

Vzhledem ke tvaru pravé strany rozdělíme řešení na hledání homogenního a partikulárního (relaxovaného) řešení. Začneme s homogenním, kdy považujeme vstupy \mathbf{u} za nulové:

$$\dot{\mathbf{x}} = \frac{d\mathbf{x}}{dt} = \mathbf{A} \mathbf{x}, \quad (2.27)$$

$$\int_{\mathbf{c}}^{\mathbf{x}} \frac{d\mathbf{x}}{\mathbf{x}} = \mathbf{A} \int_{t_0}^t dt,$$

$$\ln |\mathbf{x}| - \ln |\mathbf{c}| = \mathbf{A}(t - t_0),$$

$$\mathbf{x} = e^{\mathbf{A}(t-t_0)} \mathbf{c}. \quad (2.28)$$

Pro získání partikulárního řešení metodou variace konstant nyní dosadíme homogenní řešení (2.28) do původní rovnice (2.26), čímž dostáváme:

$$\dot{\mathbf{x}} = e^{\mathbf{A}(t-t_0)} \dot{\mathbf{c}} + \mathbf{A} e^{\mathbf{A}(t-t_0)} \mathbf{c} = \mathbf{A} e^{\mathbf{A}(t-t_0)} \mathbf{c} + \mathbf{B} \mathbf{u}. \quad (2.29)$$

Výraz se zjednoduší odečtením shodných členů a můžeme tak vyjádřit $\dot{\mathbf{c}}$:

$$\dot{\mathbf{c}}(t) = (e^{\mathbf{A}(t-t_0)})^{-1} \mathbf{B} \mathbf{u},$$

$$\mathbf{c}(t) = \mathbf{c}(t_0) + \int_{t_0}^t e^{\mathbf{A}(t_0-\tau)} \mathbf{B} \mathbf{u}(\tau) d\tau. \quad (2.30)$$

Tento výraz nyní můžeme dosadit do homogenního řešení (2.28):

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)} \mathbf{c}(t_0) + \int_{t_0}^t e^{\mathbf{A}(t-t_0)} e^{\mathbf{A}(t_0-\tau)} \mathbf{B} \mathbf{u}(\tau) d\tau, \quad (2.31)$$

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)} \mathbf{c}(t_0) + \int_{t_0}^t e^{\mathbf{A}(t-\tau)} \mathbf{B} \mathbf{u}(\tau) d\tau. \quad (2.32)$$

Nyní dosazením $t = t_0$ zjišťujeme, že $\mathbf{x}(t_0) = \mathbf{c}(t_0)$ a tak spojitá stavová odezva systému vypadá následovně:

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)} \mathbf{x}(t_0) + \int_{t_0}^t e^{\mathbf{A}(t-\tau)} \mathbf{B} \mathbf{u}(\tau) d\tau. \quad (2.33)$$

Odtud již konečně můžeme získat diskrétní stavovou funkci, pokud položíme

$$t_0 = k\delta, \quad t = (k+1)\delta.$$

$$\mathbf{x}_{(k+1)} = e^{\mathbf{A}\delta} \mathbf{x}_{(k)} + \int_{k\delta}^{(k+1)\delta} e^{\mathbf{A}((k+1)\delta-\tau)} \mathbf{B} d\tau \mathbf{u}_{(k)}. \quad (2.34)$$

Poslední rovnici formálně upravíme do klasického tvaru:

$$\mathbf{x}_{(k+1)} = \mathbf{A}_D \mathbf{x}_{(k)} + \mathbf{B}_D \mathbf{u}_{(k)}. \quad (2.35)$$

Matici \mathbf{A}_D lze nahradit Taylorovým rozvojem (e^x), přičemž výpočet ukončíme po dosažení požadované přesnosti ϵ :

$$\mathbf{A}_D = e^{\mathbf{A}\delta} = \mathbf{1} + \mathbf{A}\delta + \frac{\mathbf{A}^2\delta^2}{2!} + \dots + \frac{\mathbf{A}^r\delta^r}{r!} + O\delta^{r+1}. \quad (2.36)$$

Matici \mathbf{B}_D nahradíme podobně, jen navíc kvůli integrálu použijeme substituci:

$$\begin{aligned} \mathbf{A}((k+1)\delta - \tau) &= m \\ -\mathbf{A} d\tau &= dm \end{aligned} \quad (2.37)$$

$$-\mathbf{A}^{-1} \int_{\mathbf{A}\delta}^{\mathbf{0}} e^m dm = -\mathbf{A}^{-1}(\mathbf{1} - e^{\mathbf{A}\delta}) \quad (2.38)$$

$$\mathbf{B}_D = -\mathbf{A}^{-1}(\mathbf{1} - e^{\mathbf{A}\delta})\mathbf{B} = (\mathbf{1}\delta + \frac{\mathbf{A}\delta^2}{2!} + \dots + \frac{\mathbf{A}^{r-1}\delta^r}{r!} + O\delta^{r+1})\mathbf{B}. \quad (2.39)$$

2.2.2 Vlastní návrh řídicího zákona

Máme nyní konečně k dispozici diskretní stavový model

$$\begin{aligned} \mathbf{x}_{(k+1)} &= \mathbf{A}_D \mathbf{x}_{(k)} + \mathbf{B}_D \mathbf{u}_{(k)}, \\ \mathbf{y}_{(k)} &= \mathbf{C}_D \mathbf{x}_{(k)}. \end{aligned} \quad (2.40)$$

Návrh samotného regulátoru začneme sestavením posloupnosti stavů a výstupů pro N kroků do budoucna, při čemž vycházíme ze sestaveného diskrétního stavového popisu (2.40):

$$\begin{aligned}
\mathbf{y}^{(k)} &= \mathbf{C}_D \mathbf{x}^{(k)} \\
\hat{\mathbf{x}}^{(k+1)} &= \mathbf{A}_D \mathbf{x}^{(k)} + \mathbf{B}_D \mathbf{u}^{(k)} \\
\hat{\mathbf{y}}^{(k+1)} &= \mathbf{C}_D \mathbf{A}_D \mathbf{x}^{(k)} + \mathbf{C}_D \mathbf{B}_D \mathbf{u}^{(k)} \\
\hat{\mathbf{x}}^{(k+2)} &= \mathbf{A}_D^2 \mathbf{x}^{(k)} + \mathbf{A}_D \mathbf{B}_D \mathbf{u}^{(k)} + \mathbf{B}_D \mathbf{u}^{(k+1)} \\
\hat{\mathbf{y}}^{(k+2)} &= \mathbf{C}_D \mathbf{A}_D^2 \mathbf{x}^{(k)} + \mathbf{C}_D \mathbf{A}_D \mathbf{B}_D \mathbf{u}^{(k)} + \mathbf{C}_D \mathbf{B}_D \mathbf{u}^{(k+1)} \\
\hat{\mathbf{x}}^{(k+3)} &= \mathbf{A}_D^3 \mathbf{x}^{(k)} + \mathbf{A}_D^2 \mathbf{B}_D \mathbf{u}^{(k)} + \mathbf{A}_D \mathbf{B}_D \mathbf{u}^{(k+1)} + \mathbf{B}_D \mathbf{u}^{(k+2)} \\
\hat{\mathbf{y}}^{(k+3)} &= \mathbf{C}_D \mathbf{A}_D^3 \mathbf{x}^{(k)} + \mathbf{C}_D \mathbf{A}_D^2 \mathbf{B}_D \mathbf{u}^{(k)} + \mathbf{C}_D \mathbf{A}_D \mathbf{B}_D \mathbf{u}^{(k+1)} + \mathbf{C}_D \mathbf{B}_D \mathbf{u}^{(k+2)} \\
&\vdots \\
\hat{\mathbf{x}}^{(k+N)} &= \mathbf{A}_D^N \mathbf{x}^{(k)} + \mathbf{A}_D^{N-1} \mathbf{B}_D \mathbf{u}^{(k)} + \dots + \mathbf{B}_D \mathbf{u}^{(k+N-1)} \\
\hat{\mathbf{y}}^{(k+N)} &= \mathbf{C}_D \mathbf{A}_D^N \mathbf{x}^{(k)} + \mathbf{C}_D \mathbf{A}_D^{N-1} \mathbf{B}_D \mathbf{u}^{(k)} + \dots + \mathbf{C}_D \mathbf{B}_D \mathbf{u}^{(k+N-1)}
\end{aligned} \tag{2.41}$$

Po vybrání rovnic pro výstupy a jejich přepsání do maticového tvaru dostáváme pro predikci $\hat{\mathbf{y}}$ rovnici:

$$\hat{\mathbf{y}} = \mathbf{F} \mathbf{x}^{(k)} + \mathbf{G} \mathbf{u}. \tag{2.42}$$

Významy jednotlivých členů této rovnice jsou přitom následující:

$$\hat{\mathbf{y}} = \begin{bmatrix} \hat{\mathbf{y}}^{(k+1)} \\ \hat{\mathbf{y}}^{(k+2)} \\ \vdots \\ \hat{\mathbf{y}}^{(k+N)} \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \mathbf{u}^{(k)} \\ \mathbf{u}^{(k+1)} \\ \dots \\ \mathbf{u}^{(k+N-1)} \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} \mathbf{w}^{(k+1)} \\ \mathbf{w}^{(k+2)} \\ \dots \\ \mathbf{w}^{(k+N)} \end{bmatrix} \tag{2.43}$$

$$\mathbf{F} = \begin{bmatrix} \mathbf{C}_D \mathbf{A}_D \\ \mathbf{C}_D \mathbf{A}_D^2 \\ \vdots \\ \mathbf{C}_D \mathbf{A}_D^N \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{C}_D \mathbf{B}_D & 0 & \dots & 0 \\ \mathbf{C}_D \mathbf{A}_D \mathbf{B}_D & \mathbf{C}_D \mathbf{B}_D & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_D \mathbf{A}_D^{N-1} \mathbf{B}_D & \mathbf{C}_D \mathbf{A}_D^{N-2} \mathbf{B}_D & \dots & \mathbf{C}_D \mathbf{B}_D \end{bmatrix}. \tag{2.44}$$

Vektor \mathbf{w} obsahuje budoucí požadované výstupy, představuje tedy trajektorii na predikčním horizontu.

Dále je potřeba navrhnout optimalizační kritérium, zvolíme ho tedy například jako obdobu kritéria pro LQR. Při dobré znalosti modelu má následující tvar:

$$J = (\hat{\mathbf{y}} - \mathbf{w})^T \mathbf{Q} (\hat{\mathbf{y}} - \mathbf{w}) + \mathbf{u}^T \mathbf{R} \mathbf{u}. \quad (2.45)$$

Jeho minimalizace pak vede na rovnici pro výpočet vstupů:

$$\mathbf{u} = (\mathbf{G}^T \mathbf{Q} \mathbf{G} + \mathbf{R})^{-1} \mathbf{G}^T \mathbf{Q} (\hat{\mathbf{y}} - \mathbf{F} \mathbf{x}_{(k)}). \quad (2.46)$$

Z vektoru \mathbf{u} vybereme pouze první prvek $\mathbf{u}_{(k)}$ a ten použijeme jako vstup do systému. Zbytek zahodíme a v dalším kroku výpočet opakujeme. Podmínkou je, aby výpočet na predikčním horizontu proběhl v čase kratším, než je diskretizační perioda. Postup linearizace a diskretizace je čerpán z (BELDA, K., 2002), odvození řídicího zákona pak z (ROUBAL, J. et al., 2005).

2.3 Stavový pozorovatel

V předešlých částech, týkajících se návrhu regulátorů LQR a MPC, se operuje se stavovým vektorem \mathbf{x} . Ten se v našem případě skládá z nezávislých souřadnic a rychlostí, což jsou souřadnice a rychlosti platformy. Ty ale nejsme schopni přímo měřit. Měřit můžeme pouze souřadnice pohonů a je proto nutné na jejich základě nějakým způsobem potřebný stavový vektor získat (odhadnout). Ukazuje se, že ze znalosti modelu mechanismu, vstupů a výstupů lze tento vektor odhadnout - pozorovat - pomocí *stavového pozorovatele* (VALÁŠEK, M. a kol., 1995). Vycházíme opět z linearizovaného stavového modelu⁹.

$$\Delta \dot{\mathbf{x}} = \mathbf{A} \Delta \mathbf{x} + \mathbf{B} \Delta \mathbf{u}, \quad (2.47)$$

$$\Delta \mathbf{y} = \mathbf{C} \Delta \mathbf{x}. \quad (2.48)$$

⁹Pro kratší zápisy volíme $\Delta \mathbf{x} = \mathbf{x} - \mathbf{x}_0$, a $\Delta \mathbf{u} = \mathbf{u} - \mathbf{u}_0$

Rovnice pozorovatele pak lze zapsat jako

$$\Delta \dot{\hat{\mathbf{x}}} = \mathbf{A}\Delta\hat{\mathbf{x}} + \mathbf{B}\Delta\mathbf{u} + \mathbf{L}\mathbf{e}_y, \quad (2.49)$$

$$\Delta\hat{\mathbf{y}} = \mathbf{C}\Delta\hat{\mathbf{x}}. \quad (2.50)$$

Stříškou jsou označeny odhadované vektory a chyby odhadů \mathbf{x} a \mathbf{y} jsou

$$\begin{aligned} \mathbf{e}_y &= \mathbf{y} - \hat{\mathbf{y}}, \\ \mathbf{e}_x &= \mathbf{x} - \hat{\mathbf{x}}. \end{aligned} \quad (2.51)$$

Dosazením (2.47) a (2.49) do časové derivace chyby odhadu stavu dostáváme

$$\dot{\mathbf{e}}_x = \mathbf{A}\Delta\mathbf{x} - \mathbf{A}\Delta\hat{\mathbf{x}} - \mathbf{L}\mathbf{e}_y. \quad (2.52)$$

Postupnými úpravami se pak dostaneme k rovnici dynamiky chyby odhadu stavu:

$$\begin{aligned} \dot{\mathbf{e}}_x &= \mathbf{A}\mathbf{e}_x - \mathbf{L}\mathbf{e}_y, \\ \dot{\mathbf{e}}_x &= \mathbf{A}\mathbf{e}_x - \mathbf{L}(\mathbf{C}\Delta\mathbf{x} - \mathbf{C}\Delta\hat{\mathbf{x}}), \\ \dot{\mathbf{e}}_x &= (\mathbf{A} - \mathbf{L}\mathbf{C})\mathbf{e}_x. \end{aligned} \quad (2.53)$$

Aby měl stavový pozorovatel smysl, je třeba aby chyba odhadu \mathbf{e}_x konvergovala k nule pro libovolný počáteční odhad. Vlatní čísla matice $(\mathbf{A} - \mathbf{L}\mathbf{C})$ proto musí být asymptoticky stabilní. Pro dobré sledování je navíc vhodné, aby póly pozorovatele byly rychlejší, než póly samotného stabilizovaného systému. To lze zařídit jejich vhodným umístěním¹⁰ pomocí příkazu *place*.

¹⁰Volbu vlastních čísel regulátoru a pozorovatele lze na základě separačního principu provádět nezávisle na sobě.

I zde se navíc vyskytuje podmínka použitelnosti metody, v tomto případě jde o *pozorovatelnost*. Ta nám říká, zda je možné na základě měření vstupů a výstupů systému na konečném časovém intervalu jednoznačně určit stav tohoto systému na počátku měření. Číselně pro pozorovatelnost platí podobná podmínka jako pro říditelnost, ovšem místo matice říditelnosti sestavujeme matici pozorovatelnosti:

$$\mathbf{P} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix}. \quad (2.54)$$

V Matlabu tuto podmínku ověříme příkazem:

$$hP = \text{rank}(\text{obsv}(\mathbf{A}, \mathbf{C})).$$

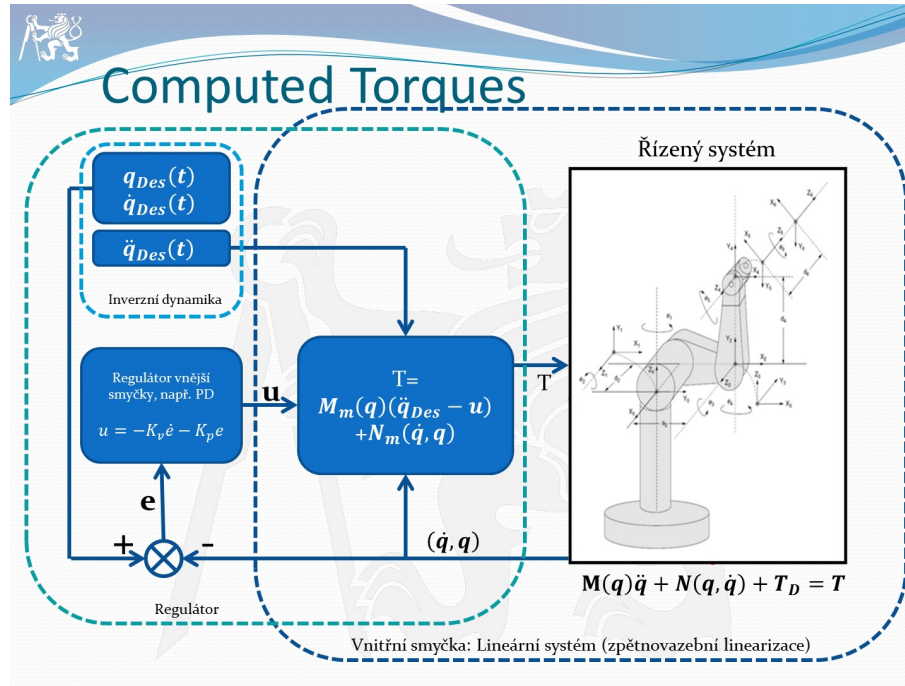
2.4 Momentové řízení (CTC)

Nakonec se podívejme na metodu CTC, která jako jediná z vybraných neoperuje se stavy. Jedná se o speciální případ zpětnovazební linearizace nelineárních systémů, který získal popularitu již v osmdesátých letech. Nasazení této metody na redundantně aktuované paralelní mechanismy však naráží na problémy s nadbytečným pohonem a jím způsobenou neurčitost silového řešení. V této práci budeme proto vycházet ze článku (MÜLLER, A. and HUFNAGEL, T., 2012), který navrhuje použitelný postup.

Celá metoda je obecně založena na využití inverzní dynamiky, která tvoří vnitřní smyčku řízení, a PID regulátoru operujícího na pohonových souřadnicích. Ideové schéma je zachyceno na obr. 2.3¹¹. Pro názornost je tento obrázek přiložen i přes to, že se týká sériového robota. Souřadnice \mathbf{q} v tomto představují právě pohonové souřadnice, které jsou pro sériovou strukturu zároveň souřadnicemi nezávislými.

Začneme tedy s odvozením řídicího zákona podle zmíněného článku, ve kterém použijeme značení odpovídající modelům vytvořeným dále v této práci. Vycházet budeme z Lagran-

¹¹Výukové materiály k předmětu Řízení mechanické systémy.



Obrázek 2.3: Principiální schéma řízení metodou Computed torques.

geových rovnic smíšeného typu, jejichž sestavení se nalézá v kapitole 3:

$$\mathbf{M}\ddot{\mathbf{s}} = \mathbf{J}^T \boldsymbol{\lambda} + \mathbf{G} + \mathbf{F}. \quad (2.55)$$

Definujme také počty jednotlivých souřadnic:

- n – počet zobecněných souřadnic \mathbf{s} ,
- m – počet závislých (pohonových) souřadnic \mathbf{z} ,
- δ – počet nezávislých souřadnic,
- r – počet vazbových rovnic.

Soustavu (2.55) (n rovnic pro $n + r$ neznámých) dále převedeme na soustavu m rovnic a zároveň z ní eliminujeme Lagrangeovy multiplikátory $\boldsymbol{\lambda}$. K tomu využijeme redukující projekci \mathbf{N}_J ¹² do jádra matice \mathbf{J} , která je dána následovně:

$$\mathbf{N}_J = \mathbf{I}_n + \mathbf{J}^+ \mathbf{J}. \quad (2.56)$$

Přitom \mathbf{J}^+ představuje pravou pseudoinverzi matice \mathbf{J} :

$$\mathbf{J}^+ = \mathbf{J}^T (\mathbf{J}\mathbf{J}^T)^{-1}. \quad (2.57)$$

¹²rozměr $n \times n$, hodnost δ

Protože platí, že $\mathbf{N}_J = \mathbf{N}_J^T$ a $\mathbf{J}\mathbf{N}_J = \mathbf{0}$ ¹³, dojde ke zmíněné eliminaci vektoru $\boldsymbol{\lambda}$. Projekční matici \mathbf{N}_J můžeme dále rozdělit na části odpovídající nezávislým (pasivním) respektive závislým (aktivním, pohonovým) souřadnicím:

$$\mathbf{N}_J = \begin{bmatrix} \tilde{\mathbf{P}} \\ \tilde{\mathbf{A}} \end{bmatrix} \begin{array}{l} (n-m) \times n \\ m \times n \end{array} \quad (2.58)$$

Po aplikaci této transformace na (2.55) dostáváme pohybové rovnice ve tvaru:

$$\mathbf{N}_J(\mathbf{M}\ddot{\mathbf{s}} - \mathbf{G}) = \tilde{\mathbf{A}}^T \mathbf{c}. \quad (2.59)$$

Zde \mathbf{c} představuje vektor akčních zásahů na pohonech, pro který platí

$$\mathbf{N}_J^T \mathbf{u} = \tilde{\mathbf{A}}^T \mathbf{c}. \quad (2.60)$$

Systém rovnic (2.59) je platný ve všech konfiguracích systému kromě singulárních poloh, protože využívá všechny souřadnice. Inverzní dynamika pro vstupy \mathbf{c} může být nyní řešena pseudoinverzí matice $\tilde{\mathbf{A}}^T$.

$$\mathbf{c} = (\tilde{\mathbf{A}}^T)^+ \mathbf{N}_J(\mathbf{M}\ddot{\mathbf{s}} - \mathbf{G}). \quad (2.61)$$

Provedení této pseudoinverze však není triviální problém, neboť matice $\tilde{\mathbf{A}}^T$ má hodnost $\delta < m$ a její řešení tak není jednoznačné. Řešení také musí být numericky efektivní, aby bylo možné ho použít pro řízení v reálném čase, není proto účelné použít SVD rozklad. V (MÜLLER, A. and HUFNAGEL, T., 2012) je k tomuto účelu odvozeno explicitní řešení pseudoinverze poddefinované matice ve tvaru:

$$(\tilde{\mathbf{A}}^T)^+ = \begin{bmatrix} (\tilde{\mathbf{A}}_1^T)^+ [\mathbf{I}_n - \tilde{\mathbf{A}}_2^T (\mathbf{I}_{m-\delta} + \mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T (\tilde{\mathbf{A}}_1^T)^+] \\ (\mathbf{I}_{m-\delta} + \mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T (\tilde{\mathbf{A}}_1^T)^+ \end{bmatrix}. \quad (2.62)$$

Jednotlivé matice, které se v tomto výrazu vyskytují, jsou přitom odvozeny z matice $\tilde{\mathbf{A}}$. Nejprve se tato matice rozdělí na dvě submatice

$$\tilde{\mathbf{A}} = \begin{bmatrix} \tilde{\mathbf{A}}_1 \\ \tilde{\mathbf{A}}_2 \end{bmatrix} \begin{array}{l} \delta \times n \\ (m-\delta) \times n \end{array} \quad (2.63)$$

¹³ \mathbf{N}_J je ortogonálním doplňkem \mathbf{J} .

Matice $\tilde{\mathbf{A}}_1$ odpovídá volbě δ nezávislých pohonových souřadnic v dané konfiguraci mechanismu a musí být zvolena tak, aby měla plnou řádkovou hodnotu (δ). Vzhledem k redundanci, která eliminuje singularity v pracovním prostoru mechanismu, tato volba může být vždy provedena, ale není jednoznačná. Obecně je proto nutné v každém kroku simulace, případně samotného řízení, hodnotu matice $\tilde{\mathbf{A}}_1$ kontrolovat a v případě potřeby tuto matici zvolit jedním z alternativních způsobů. Pokles hodnoty značí výskyt singularity, kdy pohony odpovídající volbě matice $\tilde{\mathbf{A}}_1$ nejsou schopny mechanismus v dané konfiguraci uřídit. Dále ještě platí:

$$(\tilde{\mathbf{A}}_1^T)^+ = (\tilde{\mathbf{A}}_1 \tilde{\mathbf{A}}_1^T)^{-1} \tilde{\mathbf{A}}_1, \quad (2.64)$$

$$\mathbf{B} = (\tilde{\mathbf{A}}_1^T)^+ \tilde{\mathbf{A}}_2^T. \quad (2.65)$$

Konečně tedy můžeme přistoupit k návrhu samotného řídicího zákona, který až na značení odpovídá i obr. 2.3:

$$\mathbf{c} = (\tilde{\mathbf{A}}^T)^+ \mathbf{N}_J (\mathbf{M} \mathbf{v} - \mathbf{G}). \quad (2.66)$$

V této rovnici vektor \mathbf{v} obsahuje požadovaná zrychlení zobecněných souřadnic a PID regulaci odchylek těchto souřadnic od požadované trajektorie:

$$\mathbf{v} = \ddot{\mathbf{s}}_{des} - \mathbf{K}_p \mathbf{e} - \mathbf{K}_d \dot{\mathbf{e}} - \mathbf{K}_i \int \mathbf{e} dt, \quad (2.67)$$

$$\mathbf{e} = \mathbf{s} - \mathbf{s}_{des}. \quad (2.68)$$

Zesílení PID smyčky¹⁴ jsou představovány diagonálními maticemi koeficientů. Stabilita řízení je zajištěna, pokud jsou všechny koeficienty kladné. Jak bylo uvedeno v úvodu této sekce, metoda CTC si vystačí pouze s pohonovými souřadnicemi \mathbf{z} , které jsme schopni měřit. Ve výrazu (2.68) se však vyskytují souřadnice \mathbf{s} , které obsahují i neměřitelné souřadnice platformy \mathbf{q} . S tímto problémem se vypořádáme tak, že ve všech zesilovacích maticích PID smyčky vynulujeme první členy, které se týkají pasivních souřadnic:

$$\begin{aligned} \mathbf{K}_p &= \text{diag}([0, \dots, 0, K_{p1}, \dots, K_{pm}]), \\ \mathbf{K}_d &= \text{diag}([0, \dots, 0, K_{d1}, \dots, K_{dm}]), \\ \mathbf{K}_i &= \text{diag}([0, \dots, 0, K_{i1}, \dots, K_{im}]). \end{aligned} \quad (2.69)$$

¹⁴Proporcionální \mathbf{K}_p , diferenciální \mathbf{K}_d a integrální \mathbf{K}_i .

Tím dosáhneme toho, že řízeny jsou pouze odchylky na pohonech a můžeme tedy vektor chyb přepsat následujícím způsobem:

$$\mathbf{e} = \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} - \begin{bmatrix} \mathbf{0} \\ \mathbf{z}_{des} \end{bmatrix}. \quad (2.70)$$

Kapitola 3

Kinematická struktura 2Delta

V této kapitole se budeme věnovat mechanismu *2Delta* - 2D variantě Sliding Delty. Jeho základní schéma se všemi parametry a souřadnicemi je zachyceno na obr. 3.1. Je zřejmé, že má pouze jeden stupeň volnosti a dva pohony (jeden nadbytečný), čímž je zajištěna vysoká přehlednost rovnic a snadná kontrola chování modelu. Vyzkoušíme na něm některé přístupy k řízení redundantně poháněných paralelních struktur a získané poznatky a zkušenosti pak aplikujeme při návrhu řízení komplikovanější Sliding Delty.

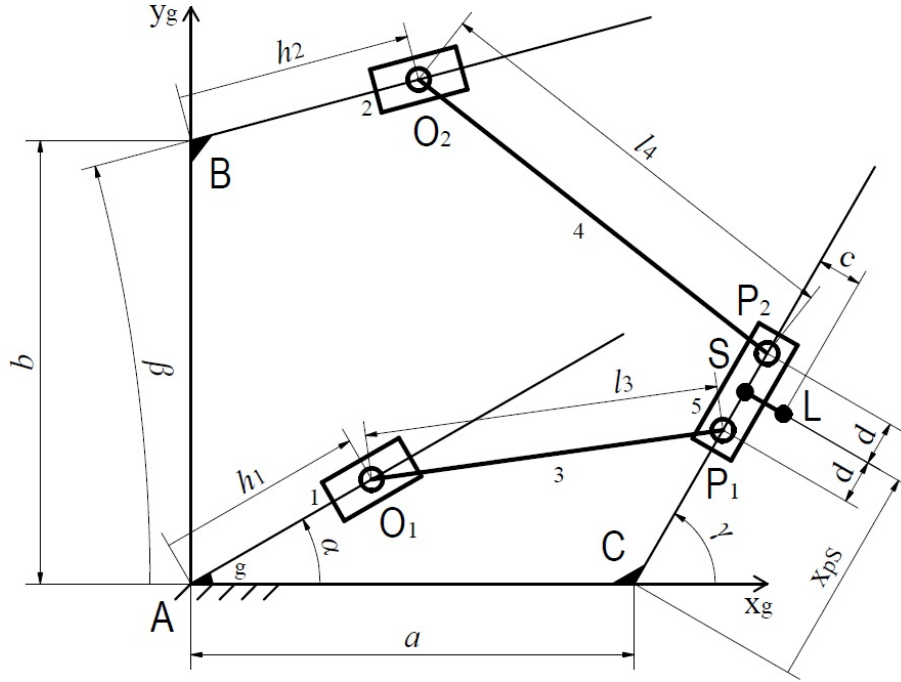
3.1 Sestavení matematického modelu

Sestavení matematického modelu bude sestávat z vyřešení úloh inverzní kinematiky, dynamiky a statiky. Dynamický model dále převedeme na stavový popis a linearizujeme ve zvolené pracovní poloze. Pro sestavení rovnic je použit stejný přístup jako při tvorbě modelu Sliding Delty v mé bakalářské práci - zanedbáme hmotnost spojnic a využijeme jejich konstatní délky. Teorie tvorby modelů je čerpána z (STEJSKAL, V., VALÁŠEK, M., 1996).

3.1.1 Inverzní úloha kinematiky

V první řadě je třeba zvolit nezávislé (\mathbf{q}) a závislé (\mathbf{z}) souřadnice:

$$\mathbf{q} = \begin{bmatrix} q \end{bmatrix} = \begin{bmatrix} x_{pS} \end{bmatrix}, \quad \mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix}. \quad (3.1)$$



Obrázek 3.1: Schéma mechanismu 2Delta.

Rovnice vazeb mezi platformou 5 a vozíky 1 a 2 zapíšeme v globálním systému $[x_g y_g]$:

$$\|\mathbf{r}_{gP_1} - \mathbf{r}_{gO_1}\| = l_3, \quad \|\mathbf{r}_{gP_2} - \mathbf{r}_{gO_2}\| = l_4. \quad (3.2)$$

Vyjádření souřadnic jednotlivých radiusvektorů je zde velmi snadné i bez využití transformačních matic, píšeme proto přímo

$$\mathbf{r}_{gP_1} = \begin{bmatrix} (x_{pS} - d) \cos \gamma + a \\ (x_{pS} - d) \sin \gamma \\ 1 \end{bmatrix}, \quad \mathbf{r}_{gO_1} = \begin{bmatrix} h_1 \cos \alpha \\ h_1 \sin \alpha \\ 1 \end{bmatrix}, \quad (3.3)$$

$$\mathbf{r}_{gP_2} = \begin{bmatrix} (x_{pS} + d) \cos \gamma + a \\ (x_{pS} + d) \sin \gamma \\ 1 \end{bmatrix}, \quad \mathbf{r}_{gO_2} = \begin{bmatrix} h_2 \cos \beta \\ h_2 \sin \beta + b \\ 1 \end{bmatrix}.$$

Po dosazení do rovnic (3.2) a přepsání do anulovaného tvaru dostáváme:

$$f_1 = 0 = l_3^2 - [(x_{pS} - d) \cos \gamma + a - h_1 \cos \alpha]^2 - [(x_{pS} - d) \sin \gamma - h_1 \sin \alpha]^2, \quad (3.4)$$

$$f_2 = 0 = l_4^2 - [(x_{pS} + d) \cos \gamma + a - h_2 \cos \beta]^2 - [(x_{pS} + d) \sin \gamma - h_2 \sin \beta - b]^2.$$

Dále lze tyto rovnice pro další použití zapsat vektorově:

$$\mathbf{f}(\mathbf{z}, \mathbf{q}) = \mathbf{0}. \quad (3.5)$$

Z rovnic (3.4) bohužel nelze přímo vyjádřit nezávislé souřadnice a úlohu polohy bude tedy nutné řešit iteračně. Vlastní implementace bude objasněna v následující kapitole. Nyní však můžeme přistoupit k úlohám rychlostí a zrychlení, které řešíme pomocí derivací vazbových podmínek (Jacobiho matic):

$$\mathbf{J}_z \dot{\mathbf{z}} + \mathbf{J}_q \dot{\mathbf{q}} = \mathbf{0}, \quad (3.6)$$

$$\mathbf{J}_z \ddot{\mathbf{z}} + \frac{d\mathbf{J}_z}{dt} \dot{\mathbf{z}} + \mathbf{J}_q \ddot{\mathbf{q}} + \frac{d\mathbf{J}_q}{dt} \dot{\mathbf{q}} = \mathbf{J}_z \ddot{\mathbf{z}} + \mathbf{J}_q \ddot{\mathbf{q}} + \mathbf{j}_{qz} = \mathbf{0}.$$

Pro zjednodušení vyjádření Jacobiho matic si nejprve vyjádříme pomocné proměnné a jejich časové derivace:

$$\begin{aligned} xPom_1 &= (x_{pS} - d) \cos \gamma - h_1 \cos \alpha + a, \\ yPom_1 &= (x_{pS} - d) \sin \gamma - h_1 \sin \alpha, \\ xPom_2 &= (x_{pS} + d) \cos \gamma - h_2 \cos \beta + a, \\ yPom_2 &= (x_{pS} + d) \sin \gamma - h_2 \sin \beta - b, \end{aligned} \quad (3.7)$$

$$\begin{aligned} x\dot{P}om_1 &= \dot{x}_{pS} \cos \gamma - \dot{h}_1 \cos \alpha, \\ y\dot{P}om_1 &= \dot{x}_{pS} \sin \gamma - \dot{h}_1 \sin \alpha, \\ x\dot{P}om_2 &= \dot{x}_{pS} \cos \gamma - \dot{h}_2 \cos \beta, \\ y\dot{P}om_2 &= \dot{x}_{pS} \sin \gamma - \dot{h}_2 \sin \beta. \end{aligned}$$

Samotné Jacobiho matice pak mají následující tvar:

$$\mathbf{J}_q = \begin{bmatrix} -2 xPom_1 \cos \gamma - 2 yPom_1 \sin \gamma \\ -2 xPom_2 \cos \gamma - 2 yPom_2 \sin \gamma \end{bmatrix}, \quad (3.8)$$

$$\mathbf{J}_z = \begin{bmatrix} 2 xPom_1 \cos \alpha + 2 yPom_1 \sin \alpha & 0 \\ 0 & 2 xPom_2 \cos \beta + 2 yPom_2 \sin \beta \end{bmatrix}.$$

Po jejich časové derivaci a přenásobení příslušnými souřadnicemi podle rovnice (3.6) dostáváme ještě vektor \mathbf{j}_{qz} :

$$\mathbf{j}_{qz} = 2 \begin{bmatrix} (-x\dot{P}om_1 \cos \gamma - y\dot{P}om_1 \sin \gamma) \dot{x}_{pS} + (x\dot{P}om_1 \cos \alpha + y\dot{P}om_1 \sin \alpha) \dot{h}_1 \\ (-x\dot{P}om_2 \cos \gamma - y\dot{P}om_2 \sin \gamma) \dot{x}_{pS} + (x\dot{P}om_2 \cos \beta + y\dot{P}om_2 \sin \beta) \dot{h}_2 \end{bmatrix}. \quad (3.9)$$

Nakonec soustavu rovnic (3.6) převedeme na tvar pro výpočet závislých rychlostí ($\dot{\mathbf{z}}$) a zrychlení ($\ddot{\mathbf{z}}$), ve kterém je později implementujeme:

$$\dot{\mathbf{z}} = -\mathbf{J}_{\mathbf{z}}^{-1}\mathbf{J}_{\mathbf{q}}\dot{\mathbf{q}}, \quad (3.10)$$

$$\ddot{\mathbf{z}} = -\mathbf{J}_{\mathbf{z}}^{-1}(\mathbf{J}_{\mathbf{q}}\ddot{\mathbf{q}} + \mathbf{j}_{\mathbf{qz}}).$$

3.1.2 Úloha dynamiky

Dalším krokem je sestavení pohybových rovnic, k čemuž opět využijeme stejný přístup jako při modelování Sliding Delt - Lagrangeovy rovnice smíšeného typu (LRST) transformované do nezávislých souřadnic. Výchozí tvar těchto rovnic je

$$\frac{d}{dt} \frac{\partial E_K}{\partial \dot{s}_j} - \frac{\partial E_K}{\partial s_j} = \sum_{k=1}^r \lambda_k \frac{\partial f_k}{\partial s_j} + S_j, \quad j = 1, 2, \dots, n. \quad (3.11)$$

Kinetická energie se spočte ze vztahu

$$E_K = \sum_i \left(\frac{1}{2} m_i \mathbf{v}_i^T \mathbf{v}_i + \frac{1}{2} \boldsymbol{\omega}_i^T \mathbf{I}_i \boldsymbol{\omega}_i \right). \quad (3.12)$$

Závislé a nezávislé souřadnice volíme stejně jako v kinematice a sestavíme z nich vektor souřadnic smíšených:

$$\mathbf{s} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} = \begin{bmatrix} \mathbf{q} \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} x_{pS} \\ h_1 \\ h_2 \end{bmatrix}. \quad (3.13)$$

Nyní s pomocí zvolených souřadnic vyjádříme kinetickou energii soustavy. Pro zjednodušení výpočtů zanedbáme hmotnosti spojnic 3 a 4, jak bylo zmíněno již v úvodu kapitoly. V úvahu tak připadají pouze translační pohyby všech tří vozíků:

$$E_K = \frac{1}{2} m_1 \dot{h}_1^2 + \frac{1}{2} m_2 \dot{h}_2^2 + \frac{1}{2} m_5 \dot{x}_{pS}^2. \quad (3.14)$$

Díky vhodné volbě souřadnic můžeme nyní i bez provádění jednotlivých derivací v rovnici (3.11) psát pohybové rovnice v jednoduché maticové formě:

$$\mathbf{M}\ddot{\mathbf{s}} = \mathbf{J}^T \boldsymbol{\lambda} + \mathbf{S}. \quad (3.15)$$

V této rovnici \mathbf{M} představuje matici hmotnosti:

$$\mathbf{M} = \begin{bmatrix} m_5 & 0 & 0 \\ 0 & m_1 & 0 \\ 0 & 0 & m_2 \end{bmatrix}, \quad (3.16)$$

\mathbf{J}^T představuje transponovanou Jacobiho matici¹, $\boldsymbol{\lambda}$ vektor Lagrangeových multiplikátorů a \mathbf{S} vektor zobecněných sil. Pro sestavení matice \mathbf{J} použijeme rovnice (3.4) z kinematického řešení, respektive přímo dílčí Jacobiho matice \mathbf{J}_q a \mathbf{J}_z (3.8):

$$\mathbf{J} = [\mathbf{J}_q, \mathbf{J}_z]. \quad (3.17)$$

Dále s využitím principu virtuálních prací sestavíme vektor \mathbf{S} :

$$\sum_i S_i \delta s_i = (F_1 - m_1 g \cos \alpha) \delta h_1 + (F_2 - m_2 g \cos \beta) \delta h_2 - m_5 g \cos \gamma. \quad (3.18)$$

Po rozepsání podle souřadnic a rozdělení na akční a pasivní síly dostáváme

$$\mathbf{S} = \mathbf{F} + \mathbf{G}, \quad (3.19)$$

$$\mathbf{F} = \begin{bmatrix} 0 \\ F_1 \\ F_2 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} -m_5 g \sin \gamma \\ -m_1 g \sin \alpha \\ -m_2 g \sin \beta \end{bmatrix}. \quad (3.20)$$

Rovnice (3.15) však nelze řešit přímo, obsahují příliš mnoho neznámých. Posledním krokem je tak jejich převod na popis pomocí nezávislých souřadnic, při čemž zároveň eliminujeme Lagrangeovy multiplikátory. Jedná se o mechanický proces obsahující množství kroků, které jsou podrobně rozepsány již v mé bakalářské práci, zde proto jen velmi stručně:

$$\dot{\mathbf{s}} = \mathbf{R} \dot{\mathbf{q}}, \quad (3.21)$$

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \quad (3.22)$$

$$\dot{\mathbf{q}} = \mathbf{B} \dot{\mathbf{s}}, \quad (3.23)$$

$$\dot{\mathbf{s}} = \begin{bmatrix} \mathbf{J} \\ \mathbf{B} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{R}^* & \mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \dot{\mathbf{q}} \end{bmatrix}, \quad (3.24)$$

¹Matice parciálních derivací vazbových rovnic podle zvolených souřadnic \mathbf{s} .

$$\mathbf{R}\dot{\mathbf{q}} = \begin{bmatrix} 0 \\ -\mathbf{J}_z^{-1}\mathbf{j}_{qz} \end{bmatrix}. \quad (3.25)$$

Během všech simulací je při použití této metody nutné sledovat vhodnost volby souřadnic, která je charakterizována podmíněností transformační matice \mathbf{J}_B :

$$\mathbf{J}_B = \begin{bmatrix} \mathbf{J} \\ \mathbf{B} \end{bmatrix}. \quad (3.26)$$

Finální tvar soustavy pohybových rovnic, vhodný pro simulaci a návrh řízení, je tedy následující:

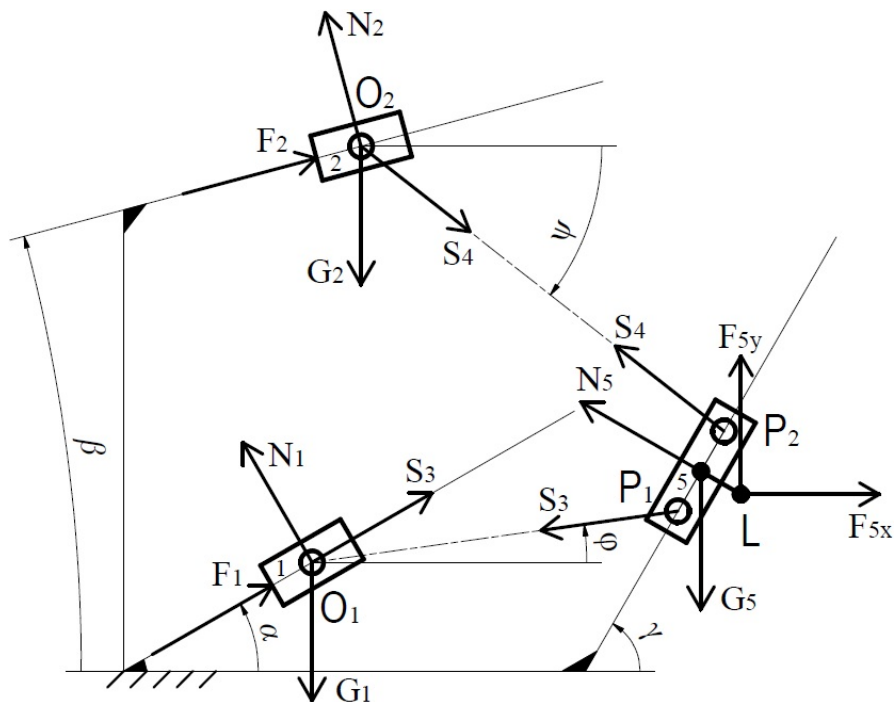
$$\ddot{\mathbf{q}} = (\mathbf{R}^T\mathbf{M}\mathbf{R})^{-1}(\mathbf{R}^T\mathbf{F} + \mathbf{R}^T\mathbf{G} - \mathbf{R}^T\mathbf{M}\dot{\mathbf{R}}\dot{\mathbf{q}}). \quad (3.27)$$

3.1.3 Úloha statiky

Konečně je třeba vyřešit i úlohu statiky, protože bude nutné nalézt síly pro rovnováhu ve zvolené poloze, ve které budeme model mechanismu následně linearizovat. Začneme tedy uvolněním vozíků a platformy (spojnice zanedbáváme), které je zachyceno na obr. 3.2. Momenty jsou zachyceny v lineárních vedeních a proto je neuvažujeme.

Pro každé těleso můžeme v rovině psát dvě rovnice rovnováhy, jednu ve směru osy "x" a druhou ve směru osy "y", máme tedy k dispozici celkem 6 rovnic pro 7 neznámých. To odpovídá naší situaci a znamená to, že jednu z akčních sil (F_1 a F_2) budeme muset pro statický výpočet nějak zvolit. Pro jednoduchost definujme F_2 jako konstantu. Síly F_{5x} a F_{5y} jsou zátěžné síly vzniklé například při obrábění a budeme s nimi pracovat jako s poruchovými veličinami. Do výpočtu zde zahrneme jejich střední hodnoty. V maticovém zápisu pak vypadají zmíněné rovnice následovně:

$$\begin{bmatrix} 0 \\ G_1 \\ 0 \\ G_2 \\ -F_{5x} \\ G_5 - F_{5y} \end{bmatrix} = \begin{bmatrix} \cos \alpha & 0 & -\sin \alpha & 0 & 0 & \cos \varphi & 0 \\ \sin \alpha & 0 & \cos \alpha & 0 & 0 & \sin \varphi & 0 \\ 0 & \cos \beta & 0 & -\sin \beta & 0 & 0 & \cos \psi \\ 0 & \sin \beta & 0 & \cos \beta & 0 & 0 & -\sin \psi \\ 0 & 0 & 0 & 0 & -\sin \gamma & -\cos \varphi & -\cos \psi \\ 0 & 0 & 0 & 0 & \cos \gamma & -\sin \varphi & \sin \psi \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ N_1 \\ N_2 \\ N_5 \\ S_3 \\ S_4 \end{bmatrix}. \quad (3.28)$$



Obrázek 3.2: Uvolnění jednotlivých těles.

Úhly natočení spojnic φ a ψ spočteme ze souřadnic jejich koncových bodů (prvky radiusvektorů ve výrazech (3.3)):

$$\varphi = \arctan \frac{y_{gP_1} - y_{gO_1}}{x_{gP_1} - x_{gO_1}}, \quad \psi = \arcsin \frac{y_{gO_2} - y_{gP_2}}{x_{gP_2} - x_{gO_2}}. \quad (3.29)$$

Abychom mohli soustavu (3.28) řešit, je nutné na pravé straně eliminovat sílu F_2 a její příspěvek přesunout na stranu levou, čímž se matice soustavy stane čtvercovou a tedy invertovatelnou:

$$\mathbf{G}_S = \mathbf{A}_S \mathbf{F}_S \quad (3.30)$$

$$\mathbf{G}_S - \mathbf{A}_S^D F_2 = \mathbf{A}_S^* \mathbf{F}_S^* \quad (3.31)$$

Hvězdičkované matice a vektory přitom vypadají takto:

$$\mathbf{A}_S^* = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 & \cos \varphi & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 & \sin \varphi & 0 \\ 0 & 0 & -\sin \beta & 0 & 0 & \cos \psi \\ 0 & 0 & \cos \beta & 0 & 0 & -\sin \psi \\ 0 & 0 & 0 & -\sin \gamma & -\cos \varphi & -\cos \psi \\ 0 & 0 & 0 & \cos \gamma & -\sin \varphi & \sin \psi \end{bmatrix}, \quad (3.32)$$

$$\mathbf{F}_S^* = \begin{bmatrix} F_1 \\ N_1 \\ N_2 \\ N_5 \\ S_3 \\ S_4 \end{bmatrix}, \quad \mathbf{A}_S^{*D} = \begin{bmatrix} 0 \\ 0 \\ \cos \beta \\ \sin \beta \\ 0 \\ 0 \end{bmatrix}.$$

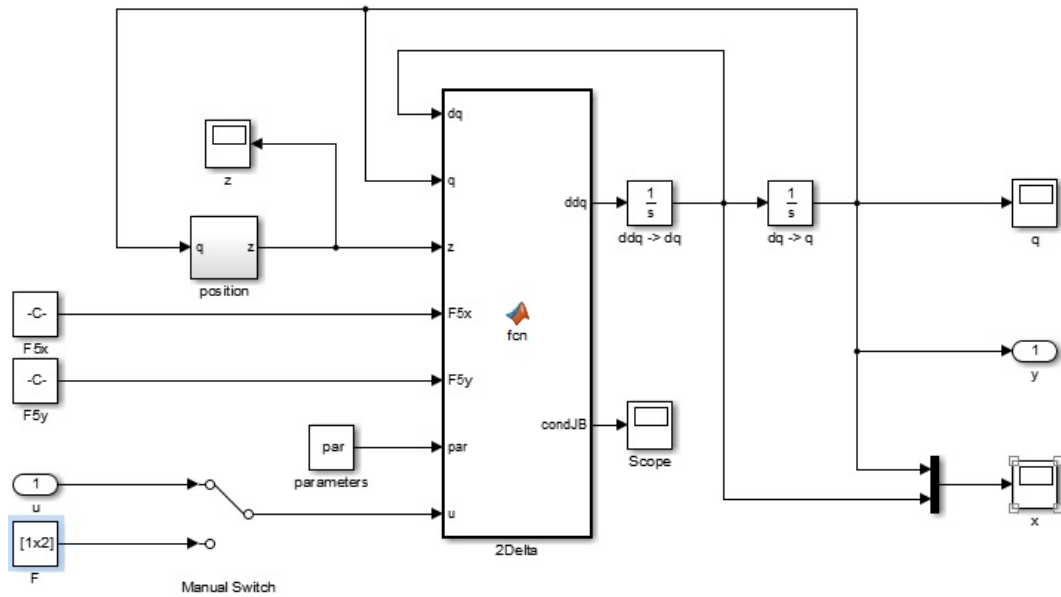
Vlastní řešení je pak velice snadné:

$$\mathbf{F}_S^* = \mathbf{A}_S^{*-1}(\mathbf{G}_S - \mathbf{A}_S^{*D}F_2) \quad (3.33)$$

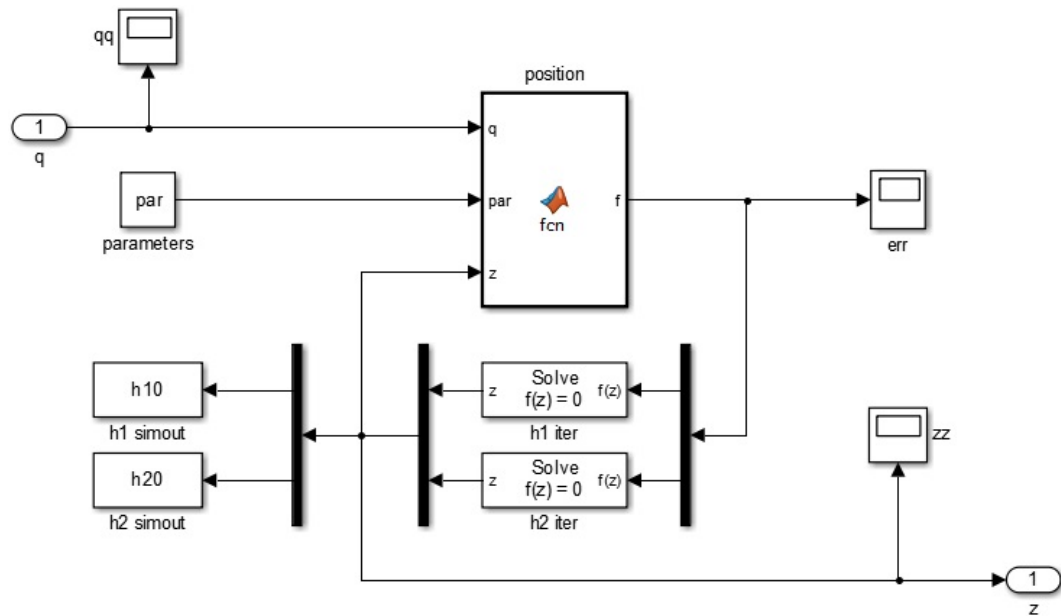
Nyní máme sestaveny všechny potřebné rovnice a můžeme přistoupit k jejich implementaci, návrhu řízení a simulacím.

3.2 Implementace matematického modelu

Začneme implementací rovnic dynamiky systému (obr. 3.3), k čemuž využijeme blok *MATLAB function*. Do tohoto simulinkového bloku se zapisují funkce v jazyce Matlabu, což nám umožňuje prakticky krok za krokem sledovat postup odvozený v předchozí kapitole. Kvůli potřebě Jacobiho matic je nutné již zde implementovat i část kinematiky. Jak již bylo uvedeno, úloha polohy zde nemá analytické řešení a je tedy nutné ji řešit iteračně. K tomu slouží subsytém „position“ (obr. 3.4). Pro urychlení konvergence iterace závislých souřadnic \mathbf{z} je v každém kroku použit výsledek z kroku předchozího. Pro zápis těchto hodnot do pracovního prostoru Matlabu slouží bloky *simout*. Poměrně důležitou roli hraje v celém modelu blok „parameters“, který obsahuje vektor všech parametrů mechanismu. Tento vektor je pro přehlednost a snadnou úpravu vytvořen jako samostatný skript v Matlabu. Zvolené parametry jsou vypsány v příloze A.

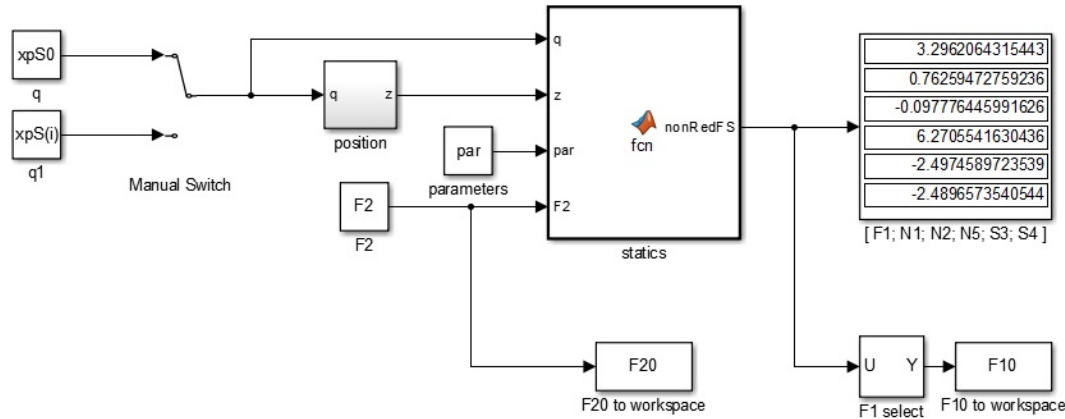


Obrázek 3.3: „dynamics“ - model dynamiky mechanismu 2Delta.



Obrázek 3.4: „position“ - řešení úlohy polohy mechanismu 2Delta.

Výpočet sil pro stabilizaci mechanismu v pracovním bodě, daném souřadnicí x_{pS_0} , je implementován ve zvláštním modelu (obr. 3.5).



Obrázek 3.5: „statics“ - řešení úlohy statiky mechanismu 2Delta.

3.2.1 Stavový popis a jeho linearizace

Poloha pro linearizaci byla zvolena pomocí souřadnice x_{pS_0} . Pro konkrétní polohu platformy byly potom sestaveny vektory pro stavový popis odpovídající zvolené poloze:

$$\mathbf{x}_0 = \begin{bmatrix} 50 \\ 0 \end{bmatrix} \text{ mm}, \quad \mathbf{y}_0 = \begin{bmatrix} 50 \end{bmatrix} \text{ mm}, \quad \mathbf{u}_0 = \begin{bmatrix} 3,2962 \\ 2,000 \end{bmatrix} \text{ N}. \quad (3.34)$$

V této poloze budeme následně model stabilizovat navrženým řízením. Lineární stavový model mechanismu 2Delta je pak v této poloze dán čtveřicí matic:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 109,5063 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 1,6620 & -0,5941 \end{bmatrix}, \quad (3.35)$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 0 & 0 \end{bmatrix}.$$

Není překvapivé, že matice \mathbf{D} skutečně vyšla nulová a matice \mathbf{C} pouze vybírá první prvek stavového vektoru. Zajímavé ale je, že celá dynamika je v tomto modelu zredukována do třech čísel, kterými jsou prvek A_{21} matice \mathbf{A} a prvky B_{21} a B_{22} matice \mathbf{B} . Pro zajímavost byla ještě provedena linearizace pro různé jiné polohy mechanismu. Na obr. B.1 až obr. B.3 v příloze B je vidět, že v okolí zvoleného bodu je chování systému značně nelineární, a to zejména pro polohy na levo od něj. Na pravo od tohoto bodu má pohon „1“ (obr. B.2) lineárnější průběh, ovšem pohon „2“ (obr. B.3) zde bohužel přechází přes singularitu - $B_{22} = 0$. Celkově lze říci, že mechanismus 2Delta nemá při zvolených rozměrech právě příkladné vlastnosti. Měl by díky tomu ale dobře otestovat navržené regulátory.

3.2.2 Vlastnosti linearizovaného modelu

Základními vlastnostmi linearizovaného systému jsou jeho vlastní čísla, pozorovatelnost a říditelnost. Vlastní čísla charakterizují stabilitu. Jejich záporná reálná část znamená, že je systém stabilní, nulová značí mez stability a kladná nestabilitu. Výskyt imaginární části ukazuje na kmitavý systém. Vlastní čísla mechanismu 2Delta se zvolenými rozměry při linearizaci ve zvolené poloze jsou následující

$$\text{eig}(A) = \begin{bmatrix} 10,4645 \\ -10,4645 \end{bmatrix}. \quad (3.36)$$

Systém je tedy nekmitavý, ale nestabilní. Pro jeho stabilizaci bude nutný stabilizující regulátor, což splňují všechny dříve popsané metody jeho návrhu. Pozorovatelnost i říditelnost systému jsou obě úplné (hodnota rovna dvěma), můžeme tedy přistoupit k návrhu parametrů regulátorů a k simulacím.

3.3 Stabilizace v rovnovážné poloze

Návrh všech regulátorů začneme hledáním vhodných parametrů jejich nastavení pro stabilizaci ve zvolené rovnovážné poloze. Teprve po jejich nalezení a ověření jejich funkčnosti rozšíříme simulace o sledování referenční trajektorie. V případě mechanismu 2Delta nebudeme pro zjednodušení používat stavového pozorovatele, pouze chceme ověřit přístupy k návrhu regulátorů. Navíc vzhledem k případné konstrukci by bylo realizovatelné přímé odměřování nezávislé souřadnice i rychlosti platformy.

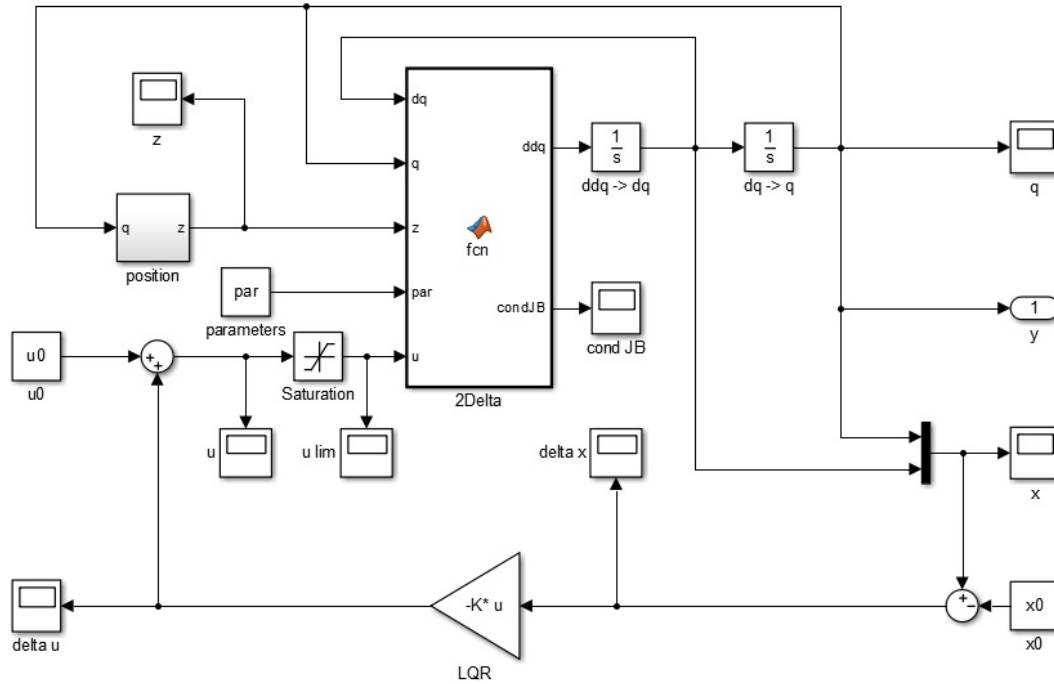
3.3.1 Stabilizace pomocí LQR

Po několika pokusech byly navrženy hodnoty váhových matic, které výrazně upřednostňují stabilizaci polohy před rychlostí a požadavky na vstupy:

$$\mathbf{Q} = \begin{bmatrix} 10^6 & 0 \\ 0 & 10^3 \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (3.37)$$

Těmto hodnotám pak odpovídá Kalmanovo zesílení \mathbf{K} :

$$\mathbf{K} = \begin{bmatrix} 1001,2 & 44,2 \\ -358,1 & -15,8 \end{bmatrix}. \quad (3.38)$$



Obrázek 3.6: Schéma stabilizace mechanismu 2Delta pomocí LQR.

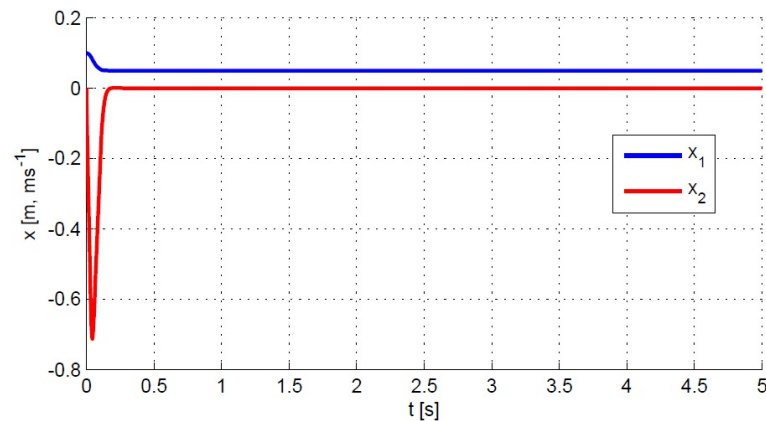
Zkontrolujeme nyní stabilitu navrženého regulovaného systému:

$$\dot{\mathbf{x}}_t = (\mathbf{A} - \mathbf{BK})(\mathbf{x}_t - \mathbf{x}_0), \quad (3.39)$$

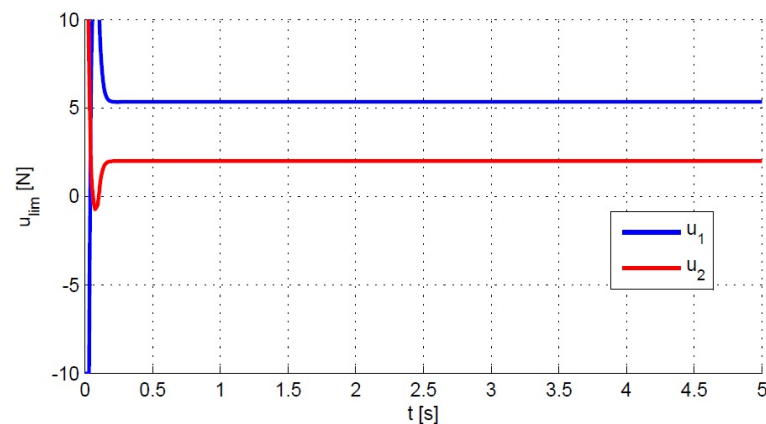
kde $(\mathbf{A} - \mathbf{BK})$ je systémová matice uzavřené regulační smyčky. Pro posouzení stability je tedy třeba nalézt vlastní čísla této nové matice:

$$\text{eig}(\mathbf{A} - \mathbf{BK}) = \begin{bmatrix} -41,4458 + 7,1160i \\ -41,4458 - 7,1160i \end{bmatrix}. \quad (3.40)$$

Všechna vlastní čísla nyní leží v levé polorovině komplexní roviny, potvrzuje se tedy stabilizující vliv navrženého řízení.



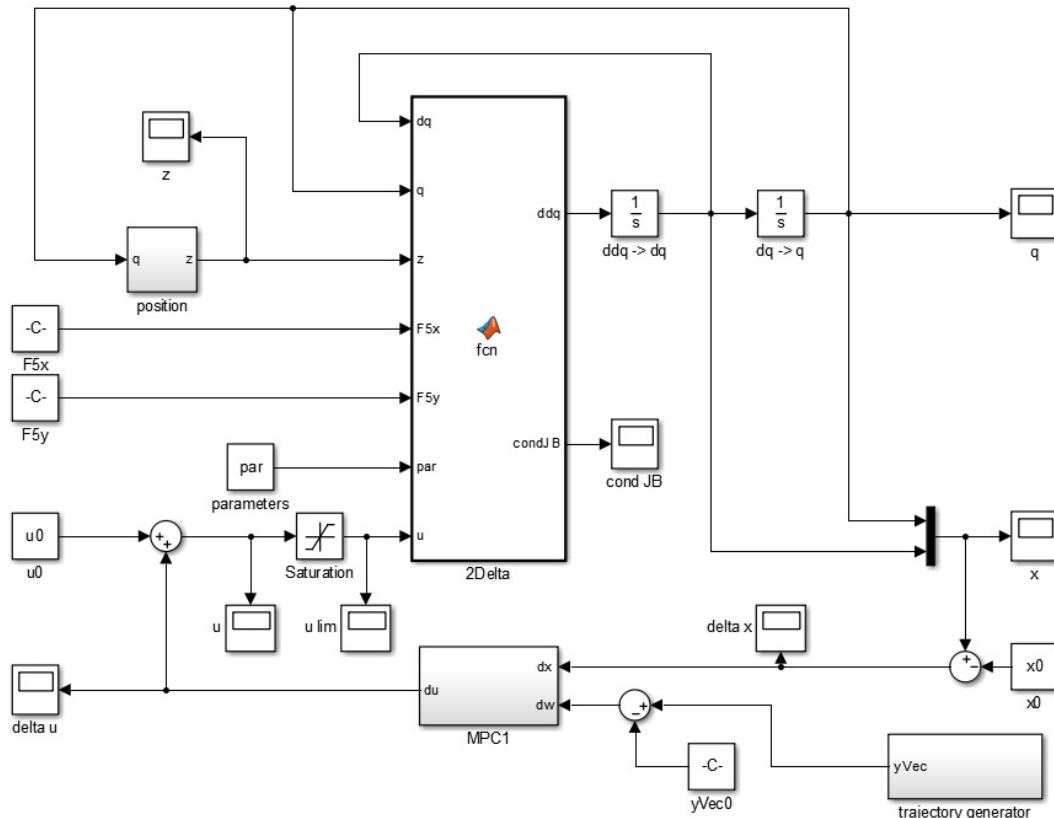
Obrázek 3.7: Průběh stabilizace stavů pomocí LQR.



Obrázek 3.8: Průběh omezených výstupů při stabilizaci pomocí LQR.

Na obr. 3.6 je zachyceno simulační schéma stabilizace. Na obrázku obr. 3.7 je znázorněn průběh stabilizace stavů \mathbf{x} , kde je vidět, že systém z počáteční polohy ($x_{pS_0} + 0,05\text{mm}$) rychle směřuje do zvolené rovnovážné polohy bez zbytečného překmitu. Na průběhu stavů (obr. 3.8) je pak vidět průběh vstupů \mathbf{u} aplikovaných regulátorem. Tyto vstupy jsou omezeny na interval $(-10\text{N}, +10\text{N})$, což je oprávněné realistické omezení maximálních možností pohonů. Na začátku stabilizace vzhledem ke značné počáteční výchylce systém na toto omezení narazí, poté je ale již vše v mezích. Takto navržený regulátor tedy zřejmě dobře stabilizuje náš systém.

Z průběhu vstupů je také vidět, že vhodnou volbou síly F_2 jsme do systému vnesli určité předpětí - obě síly mají stejné znaménko, tedy působí proti sobě. Toho budeme využívat i v následujících simulacích.

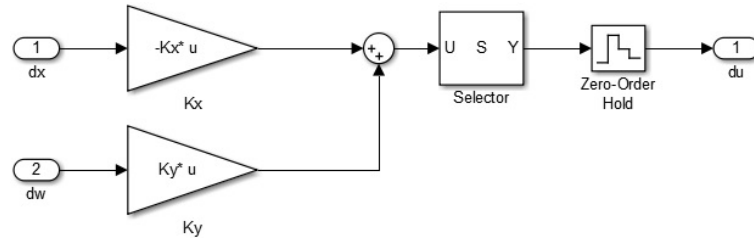


Obrázek 3.9: Schéma stabilizace mechanismu 2Delta pomocí MPC.

3.3.2 Stabilizace pomocí MPC

Vzhledem k náročnosti implementace tohoto typu řízení, zejména pak případný přechod do reálného času, který je v omezeném čase mimo naše možnosti, bylo rozhodnuto vyzkoušet alespoň jeho zjednodušenou variantu. Nebudeme tedy linearizovat model v každém výpočetním kroku, ale pouze ve zvolené pracovní poloze, podobně jako v případě LQR. Účelem bude vyzkoušet si sestavení matic pro predikci výstupů a následně i generování trajektorie v potřebném tvaru. Mimo pracovní polohu můžeme kvůli tomuto zjednodušení očekávat trvalé regulační odchylky vyplývající z nelinearity mechanismu, se kterými se však pro účely této práce smíříme. Výhodou je možnost si vše předpočítat a během simulace a případného řízení již pouze násobit matice.

Schéma pro stabilizaci mechanismu 2Delta pomocí zjednodušeného MPC je zachyceno na obr. 3.9, na obr. 3.10 je pak struktura samotného bloku regulátoru.



Obrázek 3.10: Schéma bloku MPC.

Jak je vidět, regulátor se skládá ze součtu

$$\mathbf{u} = \mathbf{K}_y \hat{\mathbf{y}} - \mathbf{K}_x \mathbf{x}^{(k)}. \quad (3.41)$$

Matice \mathbf{K}_y a \mathbf{K}_x byly získány roznásobením z původní rovnice (2.46), platí tedy:

$$\begin{aligned} \mathbf{K}_y &= (\mathbf{G}^T \mathbf{Q} \mathbf{G} + \mathbf{R})^{-1} \mathbf{G}^T \mathbf{Q}, \\ \mathbf{K}_x &= (\mathbf{G}^T \mathbf{Q} \mathbf{G} + \mathbf{R})^{-1} \mathbf{G}^T \mathbf{Q} \mathbf{F}. \end{aligned} \quad (3.42)$$

Všechny matice, které se v těchto výrazech vyskytují, mají význam shodný s tím použitým při odvozování MPC v sekci 2.2. Bohužel se v celém textu vyskytuje mnoho rovnic týkajících se různých problematik, které standardně využívají stejná písmena pro značení různých proměnných. Budeme se je ale snažit držet oddělené tak, aby nedocházelo ke zmatení. Vymýšlet pro každou matici nebo vektor jedinečné názvy by totiž nebylo účelné.

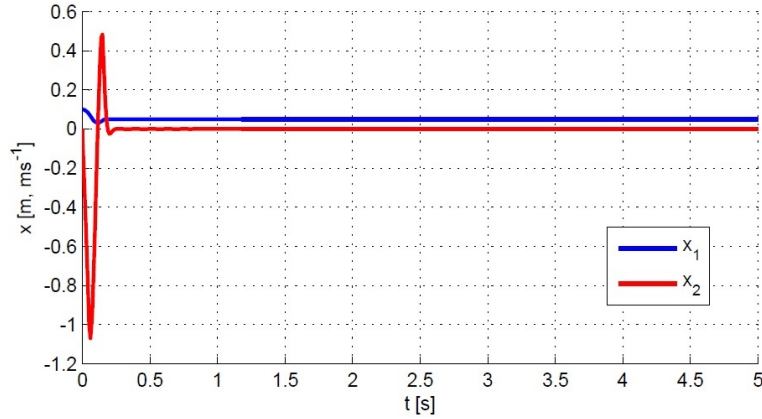
Jak bylo napsáno úvodem, linearizace modelu proběhne stejně jako v případě LQR pouze jednou - ve zvolené poloze dané souřadnicí x_{pS_0} . Matice stavového popisu zůstávají tudíž také stejné (viz. (3.35)). Co je třeba provést dále je diskretizace tohoto stavového modelu, což lze v Matlabu provést příkazy:

$$\begin{aligned} \text{sys} &= \text{ss}(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}), \\ [\mathbf{A}_D, \mathbf{B}_D, \mathbf{C}_D, \mathbf{D}_D] &= \text{ssdata}(\text{c2d}(\text{sys}, T_s)). \end{aligned}$$

Zvolíme tedy vzorkovací periodu $T_s = 0,01\text{s}$ a získáváme matice diskrétního stavového popisu²:

$$\begin{aligned} \Delta \mathbf{x}^{(k+1)} &= \mathbf{A}_D \Delta \mathbf{x}^{(k)} + \mathbf{B}_D \Delta \mathbf{u}^{(k)}, \\ \Delta \mathbf{y}^{(k)} &= \mathbf{C}_D \Delta \mathbf{x}^{(k)} + \mathbf{D}_D \Delta \mathbf{u}^{(k)}. \end{aligned} \quad (3.43)$$

²Stejně jako výchozí spojité stavový popis i tento diskrétní pracuje s odchylkami od rovnovážných hodnot.



Obrázek 3.11: Průběh stabilizace stavů pomocí MPC.

$$\mathbf{A}_D = \begin{bmatrix} 1,0065 & 0,01 \\ 1,3000 & 1,0065 \end{bmatrix}, \quad \mathbf{B}_D = \begin{bmatrix} 0,0001 & 0 \\ 0,0167 & -0,0060 \end{bmatrix}, \quad (3.44)$$

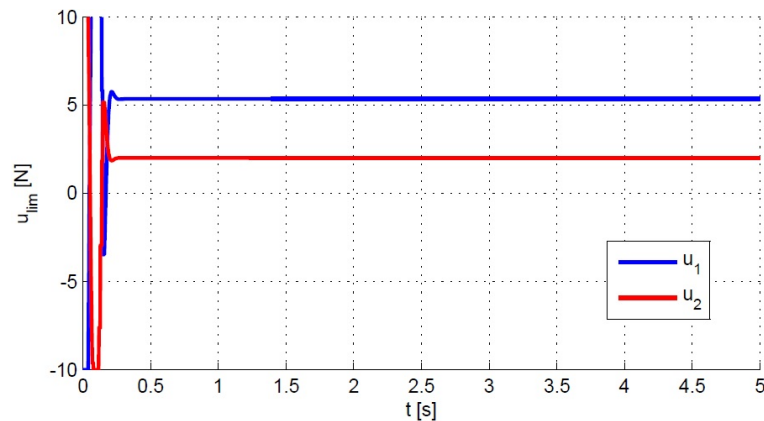
$$\mathbf{C}_D = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad \mathbf{D}_D = \begin{bmatrix} 0 & 0 \end{bmatrix}.$$

Dále je potřeba pro návrh MPC zvolit počet kroků predikce, stanovme tedy $N = 10$. V našem případě tato volba není příliš kritická, ale v případě důsledného postupu s linealizací v každém výpočetním kroku prudce roste výpočetní náročnost úlohy. To je dáno prováděním inverzí matic, jejichž velikost na N přímo závisí. Poslední volba se týká váhových matic \mathbf{Q} a \mathbf{R} . Ty volíme jako diagonální a vzhledem k jejich rozměrům pro jejich zápis použijeme pseudo-notaci podobnou Matlabu:

$$\mathbf{Q} = 10^7 * \text{eye}(N * \text{size}(\mathbf{C}_D, 1)),$$

$$\mathbf{R} = \text{eye}(N * \text{size}(\mathbf{u}_0, 2)).$$

Dále postupujeme sestavením potřebných matic a konečně výpočtem zesílení \mathbf{K}_y a \mathbf{K}_x podle rovnic (3.42). Kvůli svým rozměrům zde tyto matice uvedeny nejsou, nemělo by to velký názorný smysl. Přejděme tedy rovnou k výsledkům stabilizace. Na obr. 3.11 je vidět, že stabilizace je o něco agresivnější než v případě LQR, což je z části způsobeno volbou váhových matic. Větší překmit rychlosti ale příliš nevadí, po krátké chvíli na začátku simulace, kdy se pohony dostanou do saturace (obr. 3.12), se systém stabilizuje. I toto řízení tedy vypadá nadějně.



Obrázek 3.12: Průběh omezených výstupů při stabilizaci pomocí MPC.

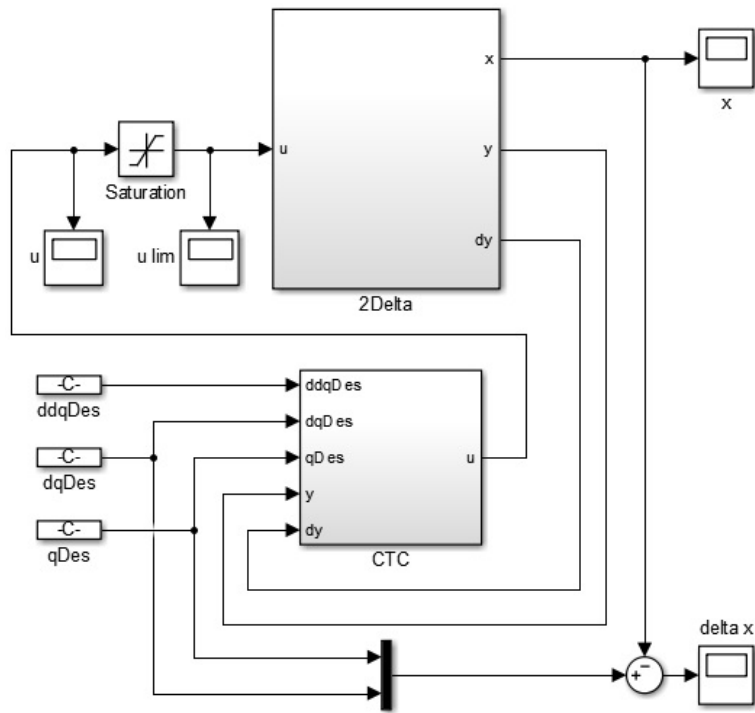
3.3.3 Stabilizace pomocí CTC

Nakonec se podíváme na stabilizaci pomocí CTC, jejíž simulační schéma je na obr. 3.13. Protože CTC pracuje s pohonovými souřadnicemi a my jsme doposud pracovali pouze se souřadnicemi platformy (nezávislé \mathbf{q}), ve kterých integrujeme přímou dynamiku 2Delta, bylo zde potřeba model doplnit o inverzní kinematiku pro získání závislých pohonových souřadnic \mathbf{z} . Toto rozšíření je vidět na obr. 3.14. Opět se zde podobně jako v jiných místech objevuje dvoje značení, kdy z pohledu kinematiky se sice jedná o závislé souřadnice \mathbf{z} , z pohledu modelu mechanismu jde ale v tomto případě o výstupy \mathbf{y} . Na obr. 3.15 je dále zobrazena vnitřní struktura regulátoru CTC, kde je vidět vnější integrační smyčka pro chybu \mathbf{e} . Průběh stabilizace (obr. 3.16) vypadá podobně jako v případě MPC, jen trochu pomaleji. Na obr. C.4 je pak v detailu vidět, že systém se rychle dostane do blízkosti zvolené polohy a pak již pomaleji směřuje k její přesné hodnotě. Tomu odpovídají i průběhy akčních sil na obr. 3.17. Ani při několikeré kontrole rovnic nebyla odhalena chyba a toto chování se nepodařilo odstranit žádnou volbou zesílení PID regulátoru. Tato zesílení byla experimentálně zvyšována, dokud se zlepšovala odezva systému (rychlost stabilizace) a nakonec byla nastavena takto:

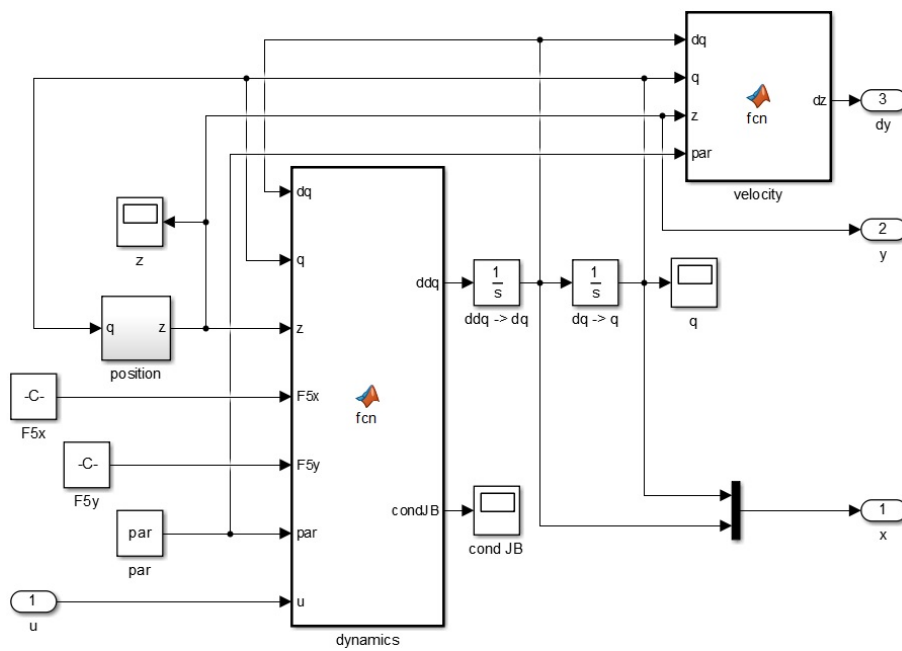
$$\mathbf{K}_p = 20000 * \text{diag}([0, 1, 1]),$$

$$\mathbf{K}_d = 500 * \text{diag}([0, 1, 1]),$$

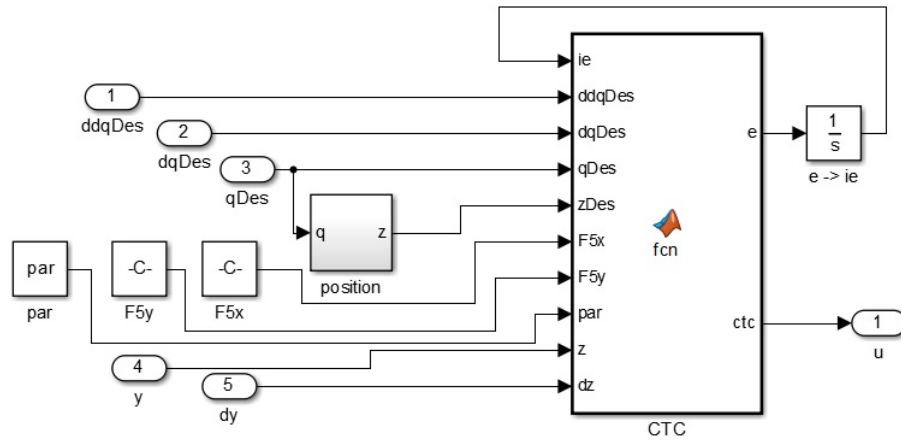
$$\mathbf{K}_i = 20000 * \text{diag}([0, 1, 1]).$$



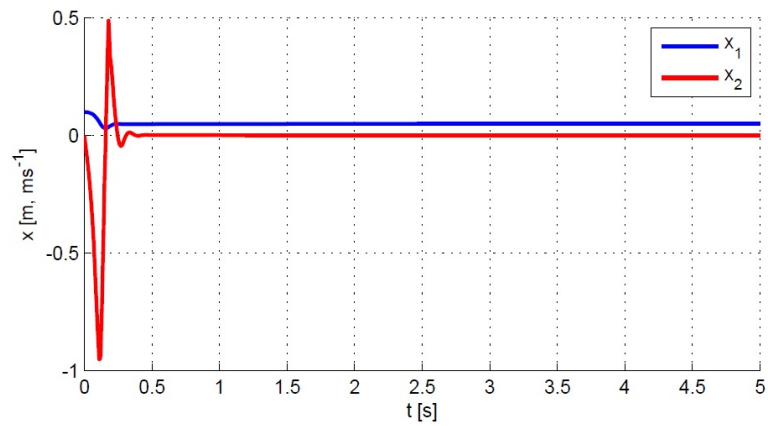
Obrázek 3.13: Schéma stabilizace mechanismu 2Delta pomocí CTC.



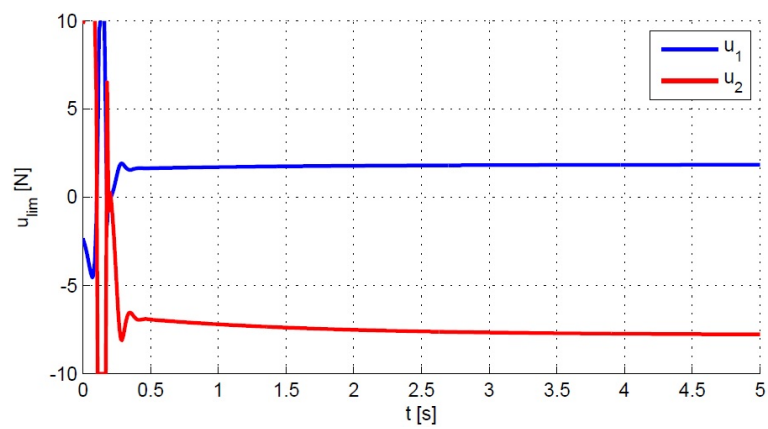
Obrázek 3.14: Schéma bloku 2Delta rozšířené o inverzní kinematiku.



Obrázek 3.15: Vnitřní schéma bloku CTC.



Obrázek 3.16: Průběh stabilizace stavů pomocí CTC.



Obrázek 3.17: Průběh omezených výstupů při stabilizaci pomocí CTC.

3.3.4 Shrnutí výsledků stabilizace

Jak je vidět z průběhů stabilizací, všechny implementované regulátory jsou schopny mechanismus 2Delta rozumným způsobem převést do zvolené polohy. Tento přechod z počátečních podmínek lze také chápat jako odezvu na skokovou změnu trajektorie. V případech LQR a MPC je do systému navíc zavedeno určité předpětí. V případě CTC by vnesení předpětí znamenalo například vytvoření projekce sil na platformu, získání virtuální výslednice na ní, a následné přepočtení zpět na pohony. Všechny soubory potřebné pro provedení všech simulací v této práci jsou obsaženy na přiloženém disku. V příloze C jsou dále pro porovnání stabilizačních účinků jednotlivých přístupů grafy průběhů odchylek stavů od požadovaných hodnot³. Z nich jako nejúčinnější vychází stabilizace pomocí LQR. Průběhy samozřejmě silně závisí na zvolených koeficientech jednotlivých regulátorů. Ty byly voleny experimentálně pro zajištění stabilizace a bylo by možné je dále optimalizovat pro dosažení lepších výsledků.

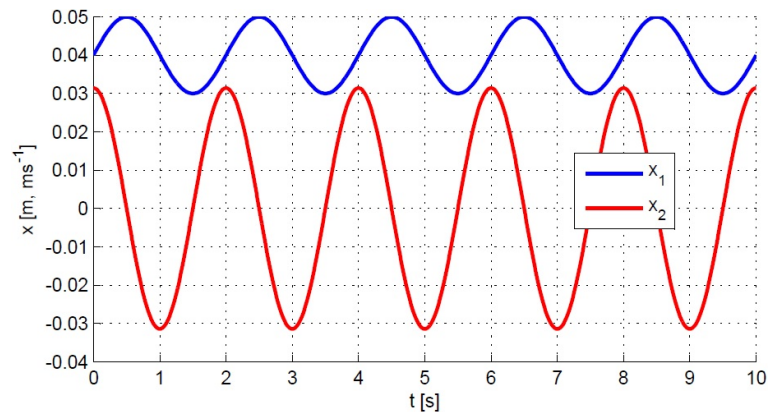
3.4 Sledování referenční trajektorie

Po úspěšném ověření stabilizačních funkcí všech regulátorů můžeme přistoupit k úpravě simulačních schémat pro řízení mechanismu po referenční trajektorii. Zvolená stavová trajektorie je zachycena na obr. 3.18. Matematicky se jedná o následující funkce:

$$\begin{aligned}y &= x_{pS_0} - 0,01 + 0,01 \sin(2\pi \cdot 0,5 \cdot t), \\ \dot{y} &= 0,01 \cdot 2\pi \cdot 0,5 \cdot \cos(2\pi \cdot 0,5 \cdot t).\end{aligned}$$

Generování této trajektorie se lehce různí pro jednotlivé regulátory. Pro LQR generujeme celou stavovou trajektorii, pro MPC generujeme pouze funkci y , ale na N kroků do budoucnosti. Pro CTC navíc musíme generovat i druhou derivaci \dot{y} . Tuto druhou derivaci navíc potřebujeme i pro kompenzační člen v případě LQR. Vzhledem k tomu, že v předchozí kapitole je již zobrazena většina potřebných Simulinkových schémat, schémata pro sledování trajektorie vložíme pro zpřehlednění textu do přílohy D. V této příloze jsou vloženy i grafy průběhů akčních zásahů. Pro zpřehlednění také nebudeme dále uvádět grafy průběhů stavů, ale pouze průběhy jejich odchylek od požadovaných hodnot. Všechny tyto grafy mají pro porovnání shodný rozsah os a zaměřují se primárně na odchylku polohy.

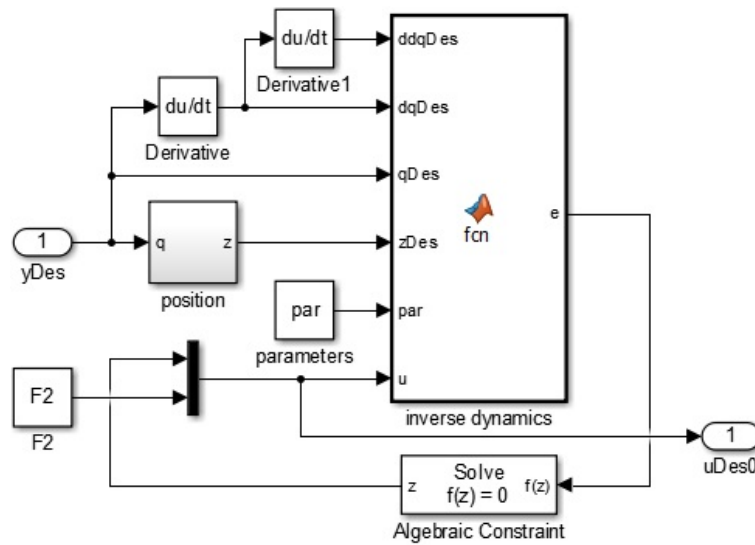
³V těchto grafech dx znamená odchylku od požadovaných hodnot, nikoli derivaci.



Obrázek 3.18: Zvolená stavová trajektorie pro mechanismus 2Delta.

3.4.1 Řízení pomocí LQR

Pro sledování referenční trajektorie pomocí LQR posíláme do regulátoru místo vektoru \mathbf{x}_0 vektor aktuálních požadovaných stavů \mathbf{x}_{Des} . To samo by stačilo, pokud by náš systém byl lineární. Bohužel tomu tak není a proto by docházelo k trvalým regulačním odchylkám, způsobeným návrhem regulátoru pro linearizovaný systém. Je tedy vhodné spočítat kompenzační vektor \mathbf{u}_0 , k čemuž využijeme inverzní dynamickou úlohu. Pro požadovaný stav systému iteračně vypočteme síly pro dosažení tohoto stavu v bloku „dynamic compensation“ (obr. 3.19). Prakticky tak vlastně dochází k řízení inverzní dynamikou se stabilizací odchylek pomocí LQR. Tato kombinace by měla dosahovat lepších výsledků než samotné LQR. Simulační schéma je zachyceno na obr. D.1. Na obr. D.4 je pak zachycen průběh akčních zásahů. Je vidět, že průběh sledování je rozumný s malou odchylkou (obr. 3.20). Dynamická kompenzace způsobuje kromě odstranění trvalých odchylek také to, že vstup F_2 je konstantní a nepodílí se tak na samotném řízení - pouze udržuje předepětí. To je způsobeno tím, že v úloze inverzní dynamiky kvůli redundanci tuto sílu volíme.



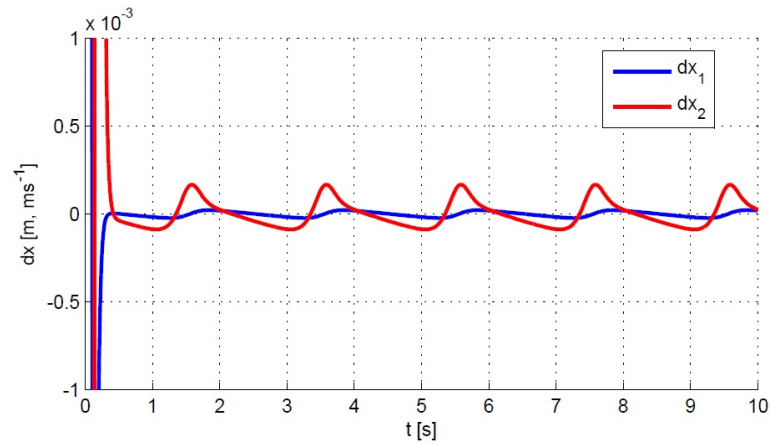
Obrázek 3.19: Kompenzační blok pro řízení mechanismu 2Delta.

3.4.2 Řízení pomocí MPC

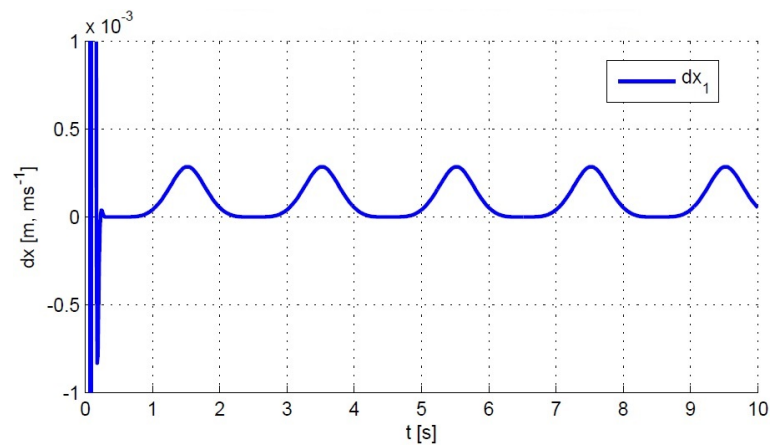
Simulační schéma pro sledování trajektorie pomocí MPC (obr. D.2) je až na generátor trajektorie prakticky stejné, jako v případě stabilizace tímto regulátorem. Pro správný výpočet odchylky od požadované trajektorie je zde ale vzhledem k tomu, že požadovanou trajektorii generujeme podle předpisu (2.43), porovnávat aktuální výstup s o jednu periodu T_S zpožděným požadovaným výstupem. Z průběhu této odchylky (obr. 3.21) je vidět, že sledování není tak přesné jako v případě LQR. Může za to použité zjednodušení návrhu MPC, tedy neprovádění linearizace v každém kroku řízení. Průběh akčních zásahů, který je na obr. D.5, ukazuje, že v tomto případě síla F_2 není konstantní, ale osciluje kolem požadované hodnoty. Částečně se tak kromě udržování předpětí podílí i na samotném řízení.

3.4.3 Řízení pomocí CTC

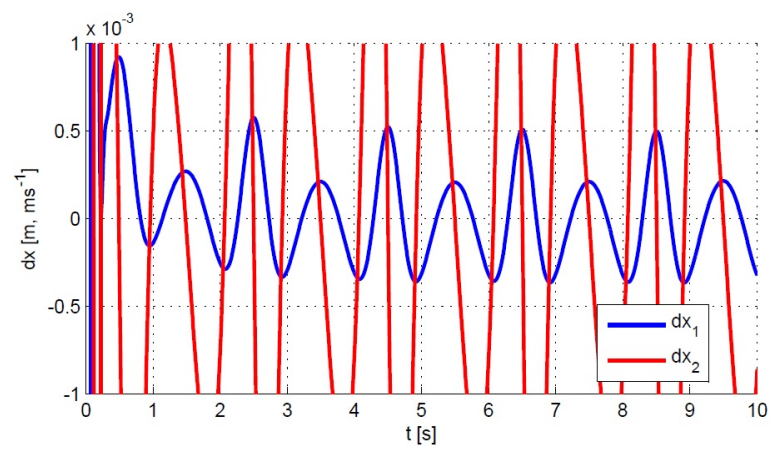
I v tomto případě je simulační schéma pro sledování referenční trajektorie (obr. D.3) až na generátor trajektorie shodné se schématem pro stabilizaci. Z průběhu odchylek na obr. 3.22 je zřejmé, že ke sledování dochází, ovšem se značnou odchylkou. Jak bylo popsáno výše, v rovnicích nebyla odhalena chyba a není tedy zřejmé, čím je to způsobeno. Pravděpodobně budou hrát roli nepřilíš dobré vlastnosti mechanismu 2Delta, na které bylo poukázáno v souvislosti s linearizací modelu.



Obrázek 3.20: Průběh odchylek stavů při sledování trajektorie LQR.



Obrázek 3.21: Průběh odchylek stavů při sledování trajektorie MPC.



Obrázek 3.22: Průběh odchylek stavů při sledování trajektorie CTC.

3.4.4 Shrnutí a porovnání regulátorů

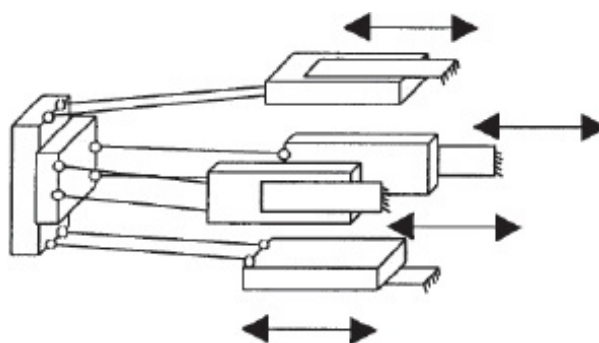
Pomocí všech implementovaných regulátorů se podařilo s různou mírou úspěšnosti docílit sledování definované referenční trajektorie. Podle očekávání po provedení simulací stabilizací i zde vychází nejlépe metoda LQR, v tomto případě doplněná o dynamickou kompenzaci. V případech LQR a MPC se jedná o aplikaci řízení navrženého pro linealizovaný systém na systém nelineární, jsou proto pozorovány předpokládané odchylky. V případě CTC z neznámé příčiny dochází k odchýlkám značným. Všechny regulátory nyní zkusíme aplikovat na složitější mechanismus Sliding Delta.

Kapitola 4

Kinematická struktura Sliding Delta

Po provedení simulačních ověření funkčnosti zvolených regulátorů se konečně dostáváme k samotnému mechanismu Sliding Delta. Jedná se, jak název napovídá, o delta robota, který byl popsán například v (ŠIKA, Z. et al., 2003), který je navíc redundantní s jedním nadbytečným pohonem. Koncepce delta robotů využívá paralelogramy pro zamezení rotací platformy a ty díky tomu mají v základní podobě¹ pouze tři translační stupně volnosti. Tím se dosáhne možnosti použít pouze tři pohony, což značně uvolňuje pracovní prostor a omezuje možné kolize členů mechanismu. Vzhledem k zamýšlenému použití zejména jako rychlé manipulátory navíc absence rotací většinou nevadí. Přidáním nadbytečného pohonu se v tomto případě většinou dosahuje vyšší dynamiky, v našem případě půjde ale spíše o pokus o odstranění vůlí vnesením předpětí.

¹Existují i různé variace s rotačním stupněm volnosti nebo hybridní provedení se šesti stupni volnosti.



Obrázek 4.1: Základní schéma mechanismu Sliding Delta.

4.1 Matematický model Sliding Delta

Již bylo zmíněno, že matematický model mechanismu Sliding Delta byl sestaven v bakalářské práci (KOZÁK, T., 2012). Postupně v ní byly vytvořeny modely inverzní kinematiky, přímé dyanamiky a statiky. Pro přehlednost zde uvedme základní vztahy, aby bylo jasné, s jakými rovnicemi a proměnnými se pracuje v simulačních schématech. Vycházím ze schématického modelu na obr. A.2 (příloha A). Velmi důležitá je zde volba souřadnic:

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} x_{gO_9} \\ y_{gO_9} \\ z_{gO_9} \end{bmatrix}, \quad \mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} z_{g1} \\ z_{g2} \\ z_{g3} \\ z_{g4} \end{bmatrix}. \quad (4.1)$$

I přes skutečnost, že závislost a nezávislost souřadnic závisí na formulaci konkrétní úlohy (přímá/inverzní), my v celém textu uvažujeme o pohonových souřadnicích jako o závislých. Kinematická úloha polohy byla řešena maticovou metodou, úlohy rochlostí a zrychlení pak podle (VALÁŠEK, M. et al., 2006). Výsledkem tohoto postupu jsou rovnice pro výpočet závislých souřadnic:

$$\begin{aligned} z_{g1} &= z_{gO_9} - d_1 - \sqrt{l_1^2 - \left(x_{gO_9} + \frac{a_p}{2} - a\right)^2 - \left(y_{gO_9} - \frac{b}{2}\right)^2}, \\ z_{g2} &= z_{gO_9} - d_2 - h_p - \sqrt{l_2^2 - \left(x_{gO_9} - \frac{a}{2}\right)^2 - \left(y_{gO_9} - \frac{b_p}{2}\right)^2}, \\ z_{g3} &= z_{gO_9} - d_3 - \sqrt{l_3^2 - \left(x_{gO_9} - \frac{a_p}{2}\right)^2 - \left(y_{gO_9} - \frac{b}{2}\right)^2}, \\ z_{g4} &= z_{gO_9} - d_4 + h_p - \sqrt{l_4^2 - \left(x_{gO_9} - \frac{a}{2}\right)^2 - \left(y_{gO_9} + \frac{b_p}{2} - b\right)^2}. \end{aligned} \quad (4.2)$$

Dále také samozřejmě Jacobiho matice \mathbf{J}_q a \mathbf{J}_z a vektor \mathbf{j}_{qz} :

$$\mathbf{J}_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{J}_q = \begin{bmatrix} -\frac{xPom_1}{odm_1} & -\frac{yPom_1}{odm_1} & -1 \\ -\frac{xPom_2}{odm_2} & -\frac{yPom_2}{odm_2} & -1 \\ -\frac{xPom_3}{odm_3} & -\frac{yPom_3}{odm_3} & -1 \\ -\frac{xPom_4}{odm_4} & -\frac{yPom_4}{odm_4} & -1 \end{bmatrix}. \quad (4.3)$$

$$\mathbf{j}_{\mathbf{qz}} = \begin{bmatrix} -\frac{(\mathbf{xPom}_1 \cdot \dot{x}_{gO_9} + \mathbf{yPom}_1 \cdot \dot{y}_{gO_9})^2}{\text{odm}_1^3} & -\frac{\dot{x}_{gO_9}^2 + \dot{y}_{gO_9}^2}{\text{odm}_1} \\ -\frac{(\mathbf{xPom}_2 \cdot \dot{x}_{gO_9} + \mathbf{yPom}_2 \cdot \dot{y}_{gO_9})^2}{\text{odm}_2^3} & -\frac{\dot{x}_{gO_9}^2 + \dot{y}_{gO_9}^2}{\text{odm}_2} \\ -\frac{(\mathbf{xPom}_3 \cdot \dot{x}_{gO_9} + \mathbf{yPom}_3 \cdot \dot{y}_{gO_9})^2}{\text{odm}_3^3} & -\frac{\dot{x}_{gO_9}^2 + \dot{y}_{gO_9}^2}{\text{odm}_3} \\ -\frac{(\mathbf{xPom}_4 \cdot \dot{x}_{gO_9} + \mathbf{yPom}_4 \cdot \dot{y}_{gO_9})^2}{\text{odm}_4^3} & -\frac{\dot{x}_{gO_9}^2 + \dot{y}_{gO_9}^2}{\text{odm}_4} \end{bmatrix}. \quad (4.4)$$

Pomocné výrazy obsažené v těchto maticích mají následující význam:

$$\begin{aligned} \text{odm}_1 &= \sqrt{l_1^2 - \left(x_{gO_9} + \frac{a_p}{2} - a\right)^2 - \left(y_{gO_9} - \frac{b}{2}\right)^2}, \\ \text{odm}_2 &= \sqrt{l_2^2 - \left(x_{gO_9} - \frac{a}{2}\right)^2 - \left(y_{gO_9} - \frac{b_p}{2}\right)^2}, \\ \text{odm}_3 &= \sqrt{l_3^2 - \left(x_{gO_9} - \frac{a_p}{2}\right)^2 - \left(y_{gO_9} - \frac{b}{2}\right)^2}, \\ \text{odm}_4 &= \sqrt{l_4^2 - \left(x_{gO_9} - \frac{a}{2}\right)^2 - \left(y_{gO_9} + \frac{b_p}{2} - b\right)^2}. \end{aligned} \quad (4.5)$$

$$\mathbf{xPom} = \left[\left(x_{gO_9} + \frac{a_p}{2} - a\right), \left(x_{gO_9} - \frac{a}{2}\right), \left(x_{gO_9} - \frac{a_p}{2}\right), \left(x_{gO_9} - \frac{a}{2}\right) \right], \quad (4.6)$$

$$\mathbf{yPom} = \left[\left(y_{gO_9} - \frac{b}{2}\right), \left(y_{gO_9} - \frac{b_p}{2}\right), \left(y_{gO_9} - \frac{b}{2}\right), \left(y_{gO_9} + \frac{b_p}{2} - b\right) \right].$$

Pro úlohu dynamiky je ze zvolených závislých a nezávislých souřadnic sestaven vektor souřadnic zobecněných a k němu jsou složeny i odpovídající Jacobiho matice \mathbf{J} a matice hmotnosti \mathbf{M} :

$$\mathbf{s} = \begin{bmatrix} \mathbf{q} \\ \mathbf{z} \end{bmatrix}, \quad (4.7)$$

$$\mathbf{J} = [\mathbf{J}_{\mathbf{q}}, \mathbf{J}_{\mathbf{z}}], \quad (4.8)$$

$$\mathbf{M} = \text{diag}([m_9, m_9, m_9, m_1, m_2, m_3, m_4]). \quad (4.9)$$

V Lagrangeových rovnicích smíšeného typu rovnou rozepíšeme vektor zobecněných sil \mathbf{S} na aktivní (\mathbf{F}) a pasivní (\mathbf{G}) složky.

$$\mathbf{M}\ddot{\mathbf{s}} = \mathbf{J}^T \boldsymbol{\lambda} + \mathbf{G} + \mathbf{F}. \quad (4.10)$$

Vektor akčních zásahů \mathbf{F} obsahuje pouze síly na pohonech, vektor \mathbf{G} pak tíhy jednotlivých těles. Aby se zvýšila variabilita modelu a bylo možné snadno změnit polohu mechanismu (vertikálně, svisle, ...), využíváme pro zadání směru gravitační síly vektor směrových cosinů v globálním systému (VALÁŠEK, M. et al., 2007):

$$\mathbf{g}_{cos} = \begin{bmatrix} \cos \alpha_g \\ \cos \beta_g \\ \cos \gamma_g \end{bmatrix}. \quad (4.11)$$

Úhly α_g , β_g a γ_g svírá směrový vektor \mathbf{g}_{cos} postupně s osami x_g , y_g a z_g . Dvěma nejpravděpodobnějším konfiguracím mechanismu (gravitace proti ose y_g nebo ve směru z_g), odpovídají

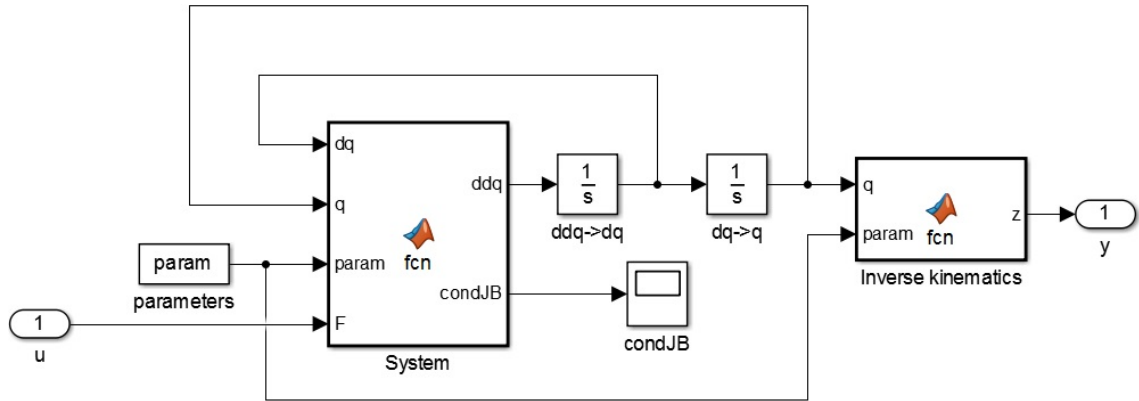
$$\mathbf{g}_{cos-y} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}, \quad \mathbf{g}_{cos-z} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \quad (4.12)$$

Silové vektory mají tedy následující tvar:

$$\mathbf{F} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} m_9 g \cos \alpha_g \\ m_9 g \cos \beta_g \\ m_9 g \cos \gamma_g \\ m_1 g \cos \gamma_g \\ m_2 g \cos \gamma_g \\ m_3 g \cos \gamma_g \\ m_4 g \cos \gamma_g \end{bmatrix}. \quad (4.13)$$

Dále byl dynamický model převeden na popis pomocí nezávislých souřadnic, za které byly zvoleny souřadnice platformy. Samotný převod byl stručně popsán v sekci 3.1.2. Všechny matice a vektory mají stejný význam v případě obou mechanismů, liší se pouze jejich rozměry a hodnoty. Výběrová matice \mathbf{B} má v tomto případě tvar následující:

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (4.14)$$



Obrázek 4.2: Simulační model Sliding Delta.

Dále byla vyřešena úloha statiky, nebo možná správněji silová rovnováha, abychom získali síly pro linearizaci v pracovním bodě. V bakalářské práci byl ale čtvrtý pohon (síla F_4) považován za pasivní vozík a počítali jsme tak s neredundantní variantou mechanismu. Rozšíření o nadbytečný pohon znamená podobně jako v případě mechanismu 2Delta možnost jednu sílu zvolit. Volíme tedy $F_{40} = 5$ N, čímž do mechanismu vneseme předpětí. Byla také zvolena orientace mechanismu shodná se schématem na obr. A.2, gravitace tak působí proti ose y_g . Pro linearizaci byl zvolen střed pracovního prostoru a hodnoty stavů, vstupů a výstupů v tomto bodě jsou:

$$\mathbf{x}_0 = \begin{bmatrix} 234 \\ 234 \\ 599 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ mm}, \quad \mathbf{y}_0 = \begin{bmatrix} 126,9101 \\ 126,9101 \\ 126,9101 \\ 126,9101 \end{bmatrix} \text{ mm}, \quad \mathbf{u}_0 = \begin{bmatrix} -14,8600 \\ 24,7201 \\ -14,8600 \\ 5 \end{bmatrix} \text{ N}. \quad (4.15)$$

4.1.1 Linearizace a vlastnosti matematického modelu

Linearizaci opět provedeme pomocí nástrojů Matlabu, k čemuž využijeme simulační model implementovaný v Simulinku. Ten je pro Sliding Deltu zachycen na obr. 4.2. Pomocí inverzní kinematiky respektujeme skutečnost, že nejsme schopni měřit souřadnice platformy, ale pouze souřadnice pohonů. Provedením linearizace pak dostáváme následující

matice stavového popisu:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -15,0018 & 0 & 0 & 0 & 0 & 0 \\ 0 & 15,2067 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -0,4526 & 0 & 0,4526 & 0 \\ 0 & 0,4588 & 0 & -0,4588 \\ 0,5790 & 0,5790 & 0,5790 & 0,5790 \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} -0,4651 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0,4651 & 1 & 0 & 0 & 0 \\ 0,4651 & 0 & 1 & 0 & 0 & 0 \\ 0 & -0,4651 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

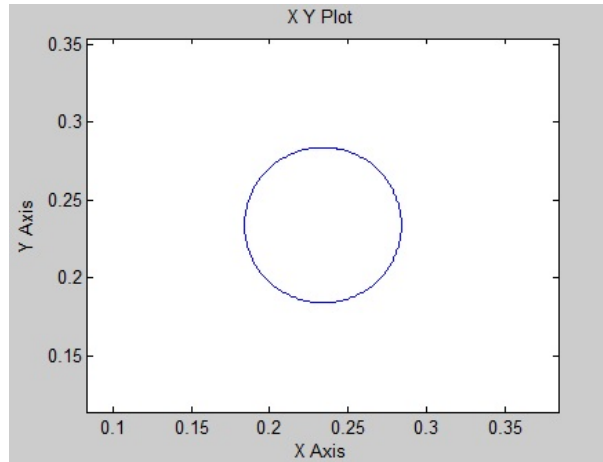
(4.16)

Nyní se podíváme na vlastnosti lineárního modelu našeho systému. Vlastní čísla jsou podle očekávání nestabilní, bez stabilizujícího řízení se tak neobejdeme:

$$\text{eig}(\mathbf{A}) = \begin{bmatrix} 0 \\ 0 \\ 0 + 3,8732i \\ 0 - 3,8732i \\ 3,8996 \\ -3,8996 \end{bmatrix}. \quad (4.17)$$

Pozorovatelnost a říditelnost systému jsou i v případě mechanismu Sliding Delta úplné, stabilizující řízení tedy navrhnout lze². Stejně tak je možné navrhnout stavového pozorovatele pro odhad stavů systému pro potřeby regulátorů LQR a MPC.

²Alespoň v určitém okolí pracovního bodu.



Obrázek 4.3: Referenční trajektorie v souřadnicích xy.

4.2 Aplikace zvolených metod řízení

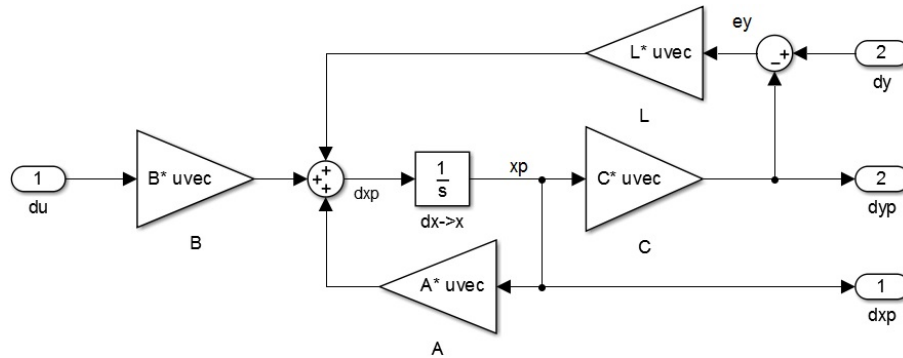
Protože základní vlastnosti jednotlivých regulátorů byly ověřeny na mechanismu 2Delta, nebudeme se dále zabývat samostatnou stabilizací. Přistoupíme rovnou k sestavení simulačních schémat pro sledování trajektorie. Pro LQR a MPC navíc navrhne stavového pozorovatele, bez kterého nemají tyto regulátory pro náš mechanismus smysl. Generátory trajektorie jsou obdobné jako v předchozí kapitole pro 2Deltu, rozdíl je pouze v počtu souřadnic. Jako referenční trajektorie platformy byla zvolena kružnice v rovině $x_g y_g$, se středem ve středu pracovního prostoru:

$$\begin{aligned} q_1^* &= q_{10} + X \sin(w_x t), \\ q_2^* &= q_{20} + Y \cos(w_y t), \\ q_3^* &= q_{30}. \end{aligned} \quad (4.18)$$

Výchozí poloha mechanismu je volena ve středu, mechanismus tak bude nejprve muset na kruhovou trajektorii „najat“. Otestuje se tak i odezva na skok. Aby se minimalizovala možnost kolize laboratorního modelu, byly parametry trajektorie nastaveny poměrně konzervativně:

$$\begin{aligned} w_x &= w_y = \pi \text{ s}^{-1}, \\ X &= Y = 50 \text{ mm}. \end{aligned} \quad (4.19)$$

Na obr. 4.3 je referenční trajektorie vykreslena v pohledu proti ose z_g (čelní pohled). Rozsahy os odpovídají pracovnímu prostoru mechanismu Sliding Delta, je tedy vidět,



Obrázek 4.4: Schéma stavového pozorovatele.

že si udržujeme rezervu od jeho okrajů. Po zprovoznění laboratorního modelu a jeho odladění bude možné vyzkoušet parametry trajektorie zvyšovat. Podle potřeby jednotlivých regulátorů a pro zobrazování různých veličin během simulace počítáme derivace složek trajektorie a také jejich převod na souřadnice pohonů pomocí inverzní kinematiky.

Většina simulačních schémat je pro zpřehlednění textu umístěna v příloze E, stejně jako průběhy sil na pohonech. I v tomto případě jsou vstupy omezeny, na $(-50N, +50N)$. Tyto limity byly předběžně nastaveny na základě parametrů použitých motorů a lineárních vedení (viz. kapitola 5).

4.2.1 Řízení pomocí LQR

I v případě mechanismu Sliding Delta byly experimentálně navrženy váhové matice upřednostňující stabilizaci polohy před rychlostí a požadavky na vstupy:

$$\mathbf{Q} = \text{diag}([10^7, 10^7, 10^7, 10^3, 10^3, 10^3]), \quad \mathbf{R} = \text{diag}([1, 1, 1, 1]). \quad (4.20)$$

Těmto hodnotám pak odpovídá Kalmanovo zesílení \mathbf{K} :

$$\mathbf{K} = \begin{bmatrix} -2220 & 0 & 1581 & -74 & 0 & 40 \\ 0 & 2253 & 1581 & 0 & 74 & 40 \\ 2220 & 0 & 1581 & 74 & 0 & 40 \\ 0 & -2253 & 1581 & 0 & -74 & 40 \end{bmatrix}. \quad (4.21)$$

Vlastní čísla stabilizovaného systému jsou při učiněných volbách následující:

$$\text{eig}(A - BK) = \begin{bmatrix} -33,2730 + 30,2856i \\ -33,2730 - 30,2856i \\ -33,7467 + 30,2178i \\ -33,7467 - 30,2178i \\ -46,5443 + 38,6755i \\ -46,5443 - 38,6755i \end{bmatrix}. \quad (4.22)$$

Systém je tedy stabilní a poměrně rychlý. Pro návrh stavového pozorovatele musíme vybrat póly, které jsou rychlejší než nejrychlejší pól stabilizovaného systému. Zvolme je tedy rychlejší alespoň dvakrát:

$$p = -2 [50, 51, 52, 60, 61, 62]. \quad (4.23)$$

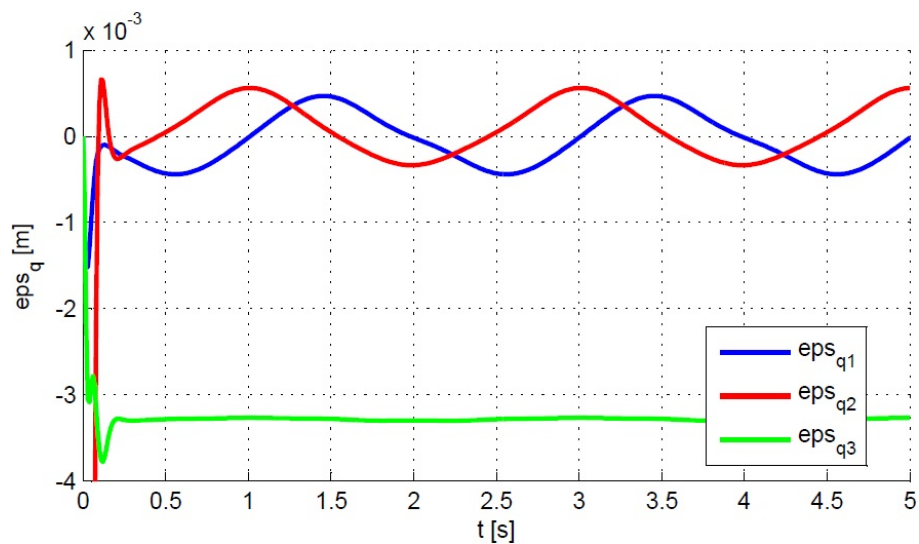
Pro jejich umístění a tedy získání matice pozorovatele \mathbf{L} (rovnice obr. 2.49) je v Matlabu implementována funkce *place*:

$$\mathbf{L} = \text{place}(A', C', p)'$$

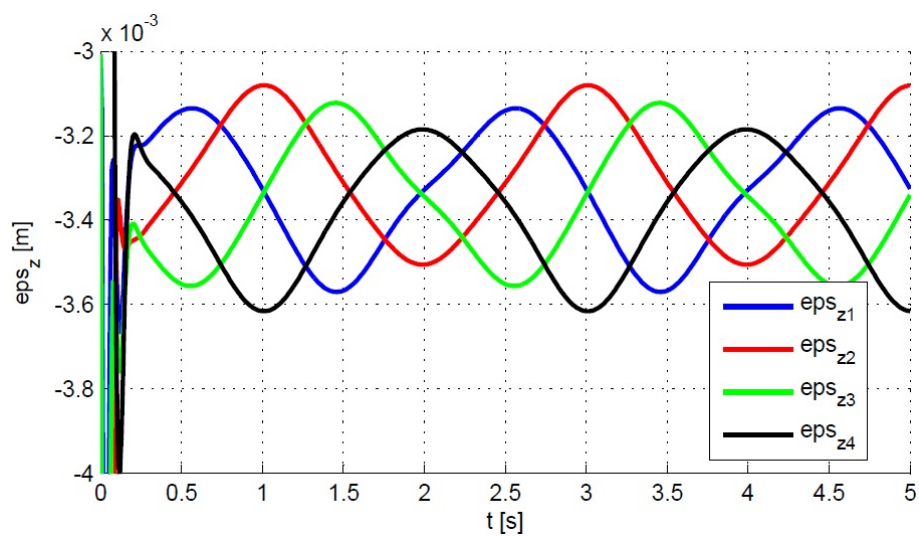
Na obr. 4.4 je zachyceno schéma pozorovatele tak, jak je implementováno v Simulinku. Na obr. E.1 je pak kompletní schéma pro simulaci řízení mechanismu Sliding Delta po trajektorii pomocí LQR. Průběh odchylek poloh platformy je znázorněn na obr. 4.5. Je zřejmé, že se zde vyskytuje určitá trvalá odchylka regulace. Tak je způsobena stavovým pozorovatelem, který byl navržen pro lineární systém, a tak mimo bod linearizace nedává přesné odhady. Na obr. 4.6 jsou zachyceny odchylky souřadnic pohonů a na obr. E.4 je opět vidět vliv kompenzace na akční síly.

4.2.2 Řízení pomocí MPC

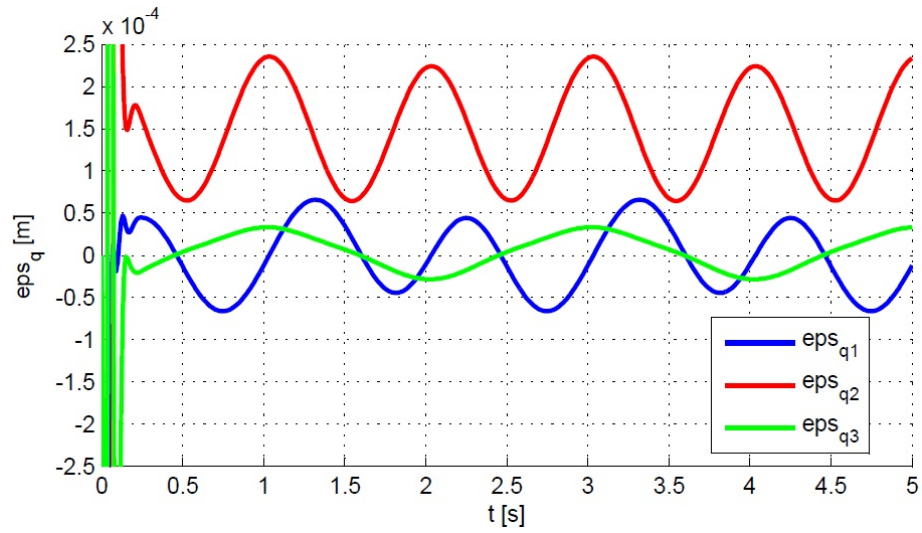
Při sestavování regulátoru MPC bylo nejprve opět nutné provést diskretizaci lineárního modelu. Vzorkovací periodu volíme $T_s = 0,01s$, abychom měli jistotu, že se provedení potřebných výpočtů stihne během této periody i na méně výkonném hardware.



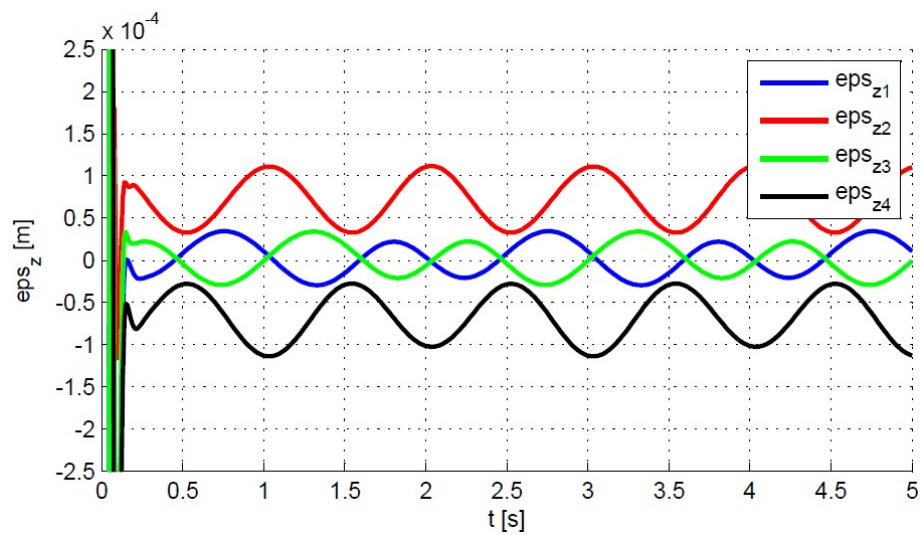
Obrázek 4.5: Průběh odchylek souřadnic platformy - LQR.



Obrázek 4.6: Průběh odchylek souřadnic pohonů - LQR.



Obrázek 4.7: Průběh odchylek souřadnic platformy - MPC.



Obrázek 4.8: Průběh odchylek souřadnic pohonů - MPC.

Dostáváme tedy následující matice:

$$\mathbf{A}_D = \begin{bmatrix} 0,9993 & 0 & 0 & 0,01 & 0 & 0 \\ 0 & 1,0008 & 0 & 0 & 0,01 & 0 \\ 0 & 0 & 1,0000 & 0 & 0 & 0,01 \\ -0,1500 & 0 & 0 & 0,9993 & 0 & 0 \\ 0 & 0,1521 & 0 & 0 & 1,0008 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0000 \end{bmatrix}, \quad (4.24)$$

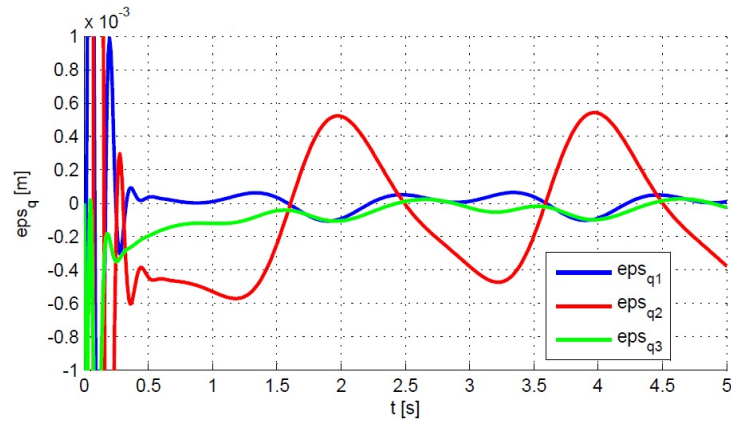
$$\mathbf{B}_D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -0,0045 & 0 & 0,0045 & 0 \\ 0 & 0,0046 & 0 & -0,0046 \\ 0,0058 & 0,0058 & 0,0058 & 0,0058 \end{bmatrix}.$$

Matice \mathbf{C}_D a \mathbf{D}_D jsou stejné jako ve spojitém případě. Dále volíme predikční horizont $N = 10$ kroků a váhové matice pro kritérium:

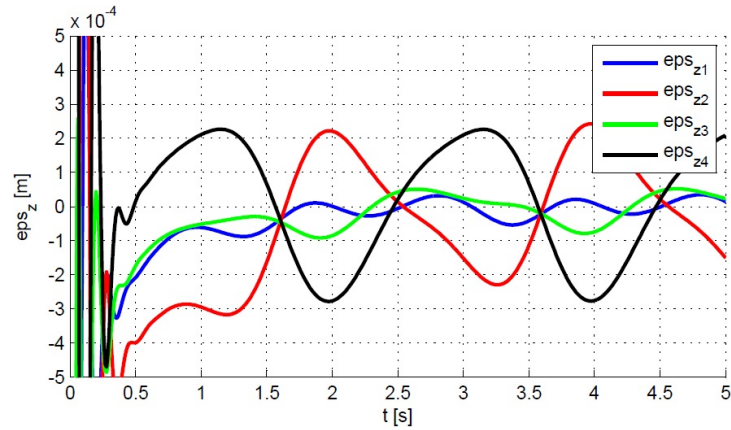
$$\mathbf{Q} = 10^8 * \text{eye}(N * \text{size}(\mathbf{C}_D, 1)),$$

$$\mathbf{R} = \text{eye}(N * \text{size}(\mathbf{u}_0, 2)).$$

Výsledky této simulace jsou zachyceny na obr. 4.7 a obr. 4.8. Oproti očekávání se zde nevyskytují velké trvalé odchylky, a to ani přes nasazení stavového pozorovatele. Při testování se naopak ukázalo, že vliv lineárního návrhu MPC a pozorovatele se vzájemně vykompenzoval a tyto odchylky se tak téměř vynulovaly. Na obr. E.4 je stejně jako v případě mechanismu 2Delta vidět, že na řízení se podílejí všechny pohony.



Obrázek 4.9: Průběh odchylek souřadnic platformy - CTC.



Obrázek 4.10: Průběh odchylek souřadnic pohonů - CTC.

4.2.3 Řízení pomocí CTC

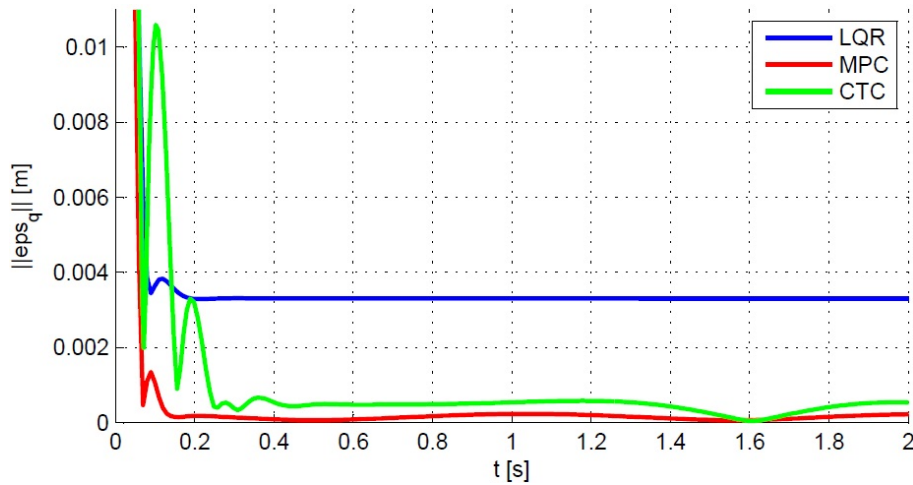
Poslední provedenou simulací byl testován regulátor CTC. I zde jsme postupně narazili na horní limit nastavení parametrů PID regulátoru, přičemž konečné použité hodnoty zesílení jsou:

$$\mathbf{K}_p = 25000 * \text{diag}([0, 0, 0, 1, 1, 1, 1]) ,$$

$$\mathbf{K}_d = 500 * \text{diag}([0, 0, 0, 1, 1, 1, 1]) ,$$

$$\mathbf{K}_i = 50000 * \text{diag}([0, 0, 0, 1, 1, 1, 1]) .$$

Ani v tomto případě nepozorujeme trvalé regulační odchylky a to dokonce ani malé. Na obr. 4.9 je vidět, že odchylky x a z souřadnic platformy jsou na podobné úrovni jako při řízení pomocí MPC, tedy řádově 0,1 mm. V ose y je odchylka o řád větší, zřejmě díky působení gravitace v tomto směru a tedy nutnosti kompenzovat toto zatížení.



Obrázek 4.11: Srovnání absolutních odchylek polohy pro různé regulátory.

4.2.4 Shrnutí a porovnání regulátorů

I díky přípravě a testování na mechanismu 2Delta se nám podařilo odsimulovat řízení Sliding Delta pomocí všech zvolených typů regulátorů. Všechny tyto regulátory mají stabilizující vliv a jsou schopny s určitou přesností sledovat referenční trajektorii. Původně nejlepší LQR kvůli nasazení stavového pozorovatele začalo vykazovat poměrně velkou trvalou regulační odchylku v ose z . Naopak u MPC došlo díky tomu k vykompenzování těchto odchylek a jeho přesnost je tak poměrně dobrá, zvláště s přihlédnutím ke zvoleným zjednodušením při jeho návrhu. Regulátor CTC také vykazuje lepší chování, než v případě 2Delta. Přičítáme to podle původního odhadu lepším kinematickým vlastnostem mechanismu Sliding Delta, jehož rozměry byly optimalizovány mimo jiné i s ohledem na dobrou manipulovatelnost (ŠIKA, Z. et al., 2003).

Na obr. 4.11 jsou pro porovnání vykresleny absolutní odchylky (přímé vzdálenosti) od požadované trajektorie pro všechny tři regulátory. Z tohoto obrázku je patrné, že všechny tři regulátory jsou podobně rychlé, jen CTC lehce zakmitává při úvodním skoku. LQR drží odchylku konstantní, regulace je tedy až na trvalou hodnotu této odchylky velmi dobrá. Nejlépe ze simulací vychází i ve své zjednodušené formě regulátor MPC, který byl nakonec zvolen pro testování na laboratorním modelu.

Kapitola 5

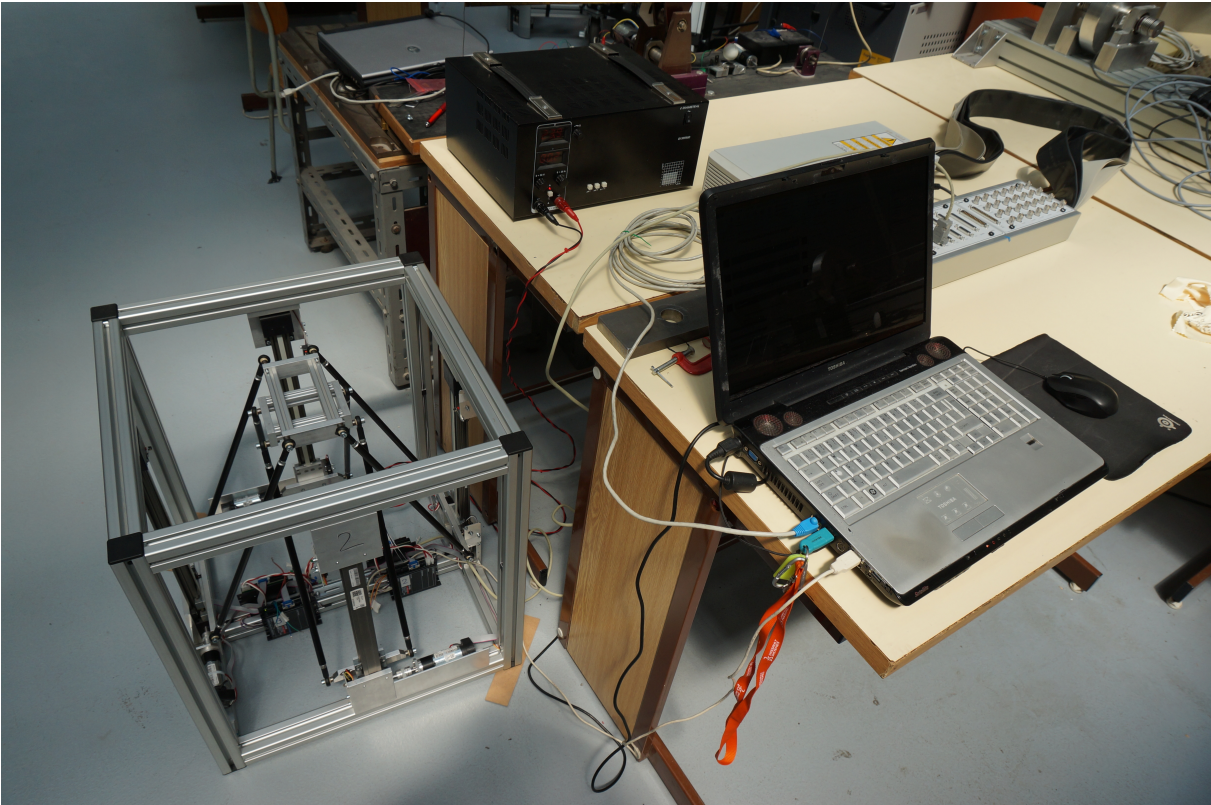
Oživení laboratorního modelu Sliding Delta

Při procesu ožívování laboratorního modelu je třeba postupně provést několik kroků. Těmi jsou volba hardware a software pro samotné řízení, fyzické zapojení hardware, konfigurace nastavení a také úprava Simulinkových modelů regulátoru do tvaru vhodného pro sestavení a běh v reálném čase.

5.1 Použitý hardware a software

Model je již od sestavení osazen stejnosměrnými motory Maxon s převodovkami a rotačními enkodéry¹, pro jejich řízení budou proto použity motorové moduly od stejného výrobce. Jedná se o řídicí jednotky EPOS2 24/5, které umožňují řízení v polohovém, rychlostním nebo proudovém režimu a komunikují prostřednictvím sběrnice CAN. Jejich konfigurace se provádí pomocí programu EPOS Studio po připojení modulů k PC pomocí USB kabelu. Pro vlastní řízení byl zvolen real-time procesor dSpace ds1103 v expansion boxu, ke kterému je připojen konektorový panel. Použit je tedy i příslušný software ControlDesk ve verzi 5.1, pomocí kterého lze ovládat běžící regulátor. Součástí tohoto programu je i instalační balík pro Matlab/Simulink verze 2013a, který obsahuje knihovnu bloků pro použití v reálném čase a umožňuje sestavení celého řídicího schématu pro běh na procesorech dSpace. Mechanismus je také osazen koncovými spínači Omron D3V-165-1C25, které zatím nejsou zapojeny a čekají na budoucí využití.

¹Sestava Maxon s objednacím číslem 244746.



Obrázek 5.1: Pracoviště pro oživení laboratorního modelu Sliding Delta.

Podívejme se nyní podrobněji na pohony. Jedná se o sestavy motorů A-max 32, planetových převodovek GP 32 a enkodérů HEDL 5540. Prostřednictvím pružné spojky jsou dále poháněny řemenice lineárních vedení DryLin ZLW od společnosti IGUS. V tab. F.1 jsou z katalogových listů vypsány parametry, které byly pro oživení mechanismu Sliding Delta použity. Z nominálního momentu motorů byl původně v části 4.2 odvozen limit pro maximální velikost akčních sil, a to sice jako dvojnásobek nominální síly vyvoditelné na vozík lineárního vedení:

$$\begin{aligned}
 F_{lim} &= 2 \cdot T_{nom} \cdot n_g / R, \\
 F_{lim} &= 2 \cdot 45,5 \text{ mNm} \cdot 4,8 / 8,59 \text{ mm}, \\
 F_{lim} &\cong 50 \text{ N}.
 \end{aligned}
 \tag{5.1}$$

Trvale by však síly neměly přesahovat polovinu limitu. Tento předběžný výpočet nezhledňuje požadavky na dynamiku motorů, pro reálné řízení bude tedy použito omezení proudové. Při správném nastavení jednotek EPOS by tak nemelo hrozit poškození motorů.

Tabulka 5.1: Konfigurace CAN zpráv.

název	COB-ID	typ	data	hodnota	velikost
CAN START	0x000	RX async.	byte 0	0	1 B
			byte 1	0	1 B
MODE SET	0x400	RX async.	mode	235	1 B
			continuous current	[mA]	2 B
			output current	[mA]	2 B
CONTROLWORD	0x300	RX async.	control word	6 = OFF 15 = ON	2 B
SYNC	0x080	RX async.			0 B
SET current	$0x20i$ $i \in \langle 1, 4 \rangle$	RX sync.	setting value	[mA]	2 B
GET position	$0x28i$ $i \in \langle 1, 4 \rangle$	TX sync.	position act. value	[counts]	4 B
			velocity act. value	[rpm]	1 B

5.2 Nastavení a úprava modelu regulátoru

Nejprve byly nakonfigurovány motorové jednotky EPOS, což v první řadě obnášelo správně zadat parametry motorů a enkodérů do objektového slovníku. Následovala autokalibrace konstant zpětnovazebního řízení proudové smyčky. Nakonec bylo potřeba nastavit baudrate (1 Mbaud) a sestavit zprávy pro komunikaci po sběrnici CAN². Každé jednotce byla přiřazena adresa (1 až 4), aby se od sebe navzájem odlišily. Konfigurace zpráv je zachycena v tab. 5.1. Po zapnutí řídicích jednotek je třeba nastartovat komunikaci CAN (CAN START), zvolit proudový režim a nastavit limity (MODE SET). Pro přechod do operačního režimu (aktivní výstup) a zpět slouží zpráva CONTROLWORD s různým obsahem. Všechny řídicí jednotky jsou synchronizovány pomocí synchronizačního objektu SYNC. Je zde důležité upozornit na jednotky použité pro proud, polohu a rychlost. Poloha je udávána v počtech přírůstků enkodéru (counts³), rychlost je v rpm (ot/min) a proud v mA.

²Controller Area Network - původně automobilový, dnes již obecně průmyslový komunikační standard využívající protokol založený na zprávách posílaných po dvou vodičové sběrnici.

³Při čtyřnásobné přesnosti dvoukanalového (A, B) enkodéru jde v našem případě o 4×500 přírůstků na otáčku.

Další na řadě byla úprava Simulinkového schématu regulátoru (obr. F.2). Nejprve byl odstraněn blok systému a do schématu byly umístěny základní bloky z knihovny dSpace RTI (real-time interface). Dalším krokem bylo nakonfigurování bloků CAN zpráv podle tab. 5.1. Aby bylo možné propojit regulátor se vstupy (aktuální polohy pohonů), bylo nutné provést převod jednotek. Pro připojení na výstupy jsem se rozhodl použít alespoň jednoduchý model stejnosměrného motoru s permanentními magnety. Ten je popsán dvěma rovnicemi, elektrickou

$$U = L \frac{dI}{dt} + RI + k_v \omega \quad (5.2)$$

a mechanickou

$$J \frac{d\omega}{dt} = k_T I - \eta \omega - \tau. \quad (5.3)$$

Zde U je napětí na svorkách, I proud, ω úhlová rychlost motoru, L indukčnost vinutí, R odpor vinutí, k_v rychlostní konstanta, J moment setrvačnosti, k_T momentová konstanta, η třecí koeficient a τ mechanický moment na hřídeli motoru. Protože je použit proudový regulátor, stačí nyní pouze z rovnice (5.3) vyjádřit proud:

$$I = \frac{J}{k_T} \dot{\omega} + \frac{\eta}{k_T} \omega + \frac{1}{k_T} \tau. \quad (5.4)$$

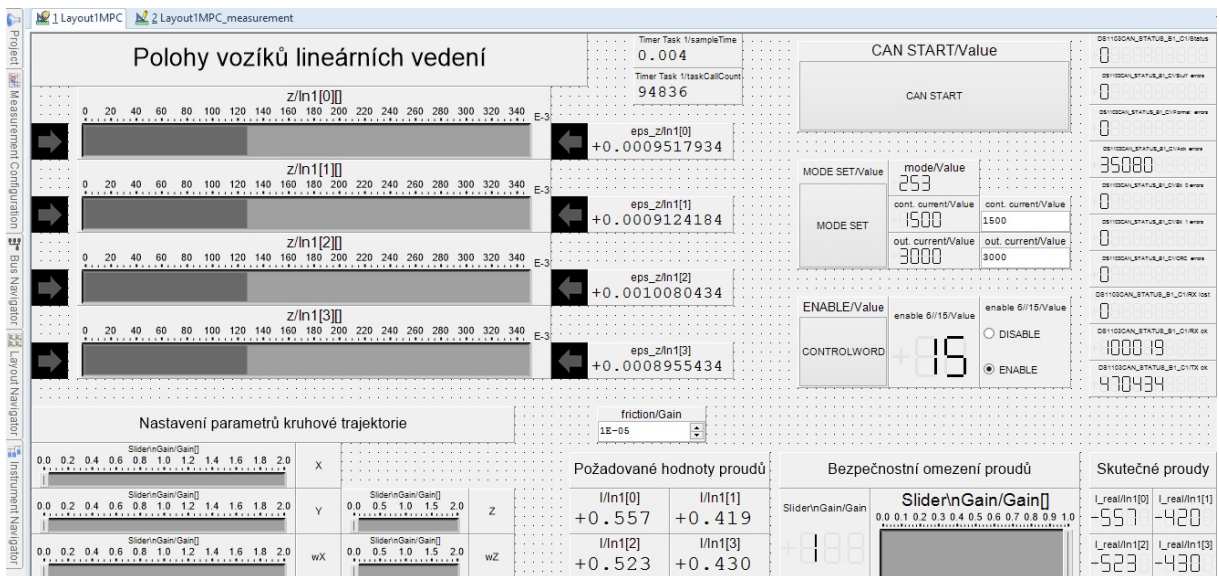
Požadované rychlosti a zrychlení motoru vycházejí z referenční trajektorie a mechanické momenty spočteme ze sil na výstupu regulátoru pomocí převodů:

$$\tau = \mathbf{F} \cdot \mathbf{r} / n_g. \quad (5.5)$$

Další úpravou regulátoru oproti původním simulacím byla změna konfigurace mechanismu tak, aby gravitace působila proti ose z_g :

$$\mathbf{g}_{cos_z} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}. \quad (5.6)$$

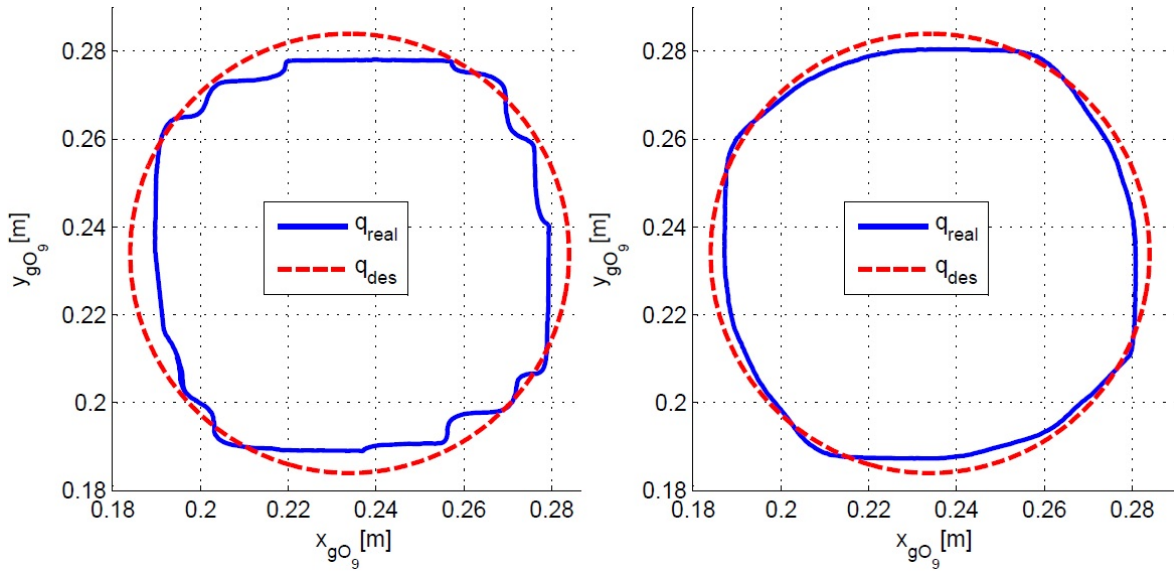
Důvodem bylo přetěžování některých motorů v jiných konfiguracích. Aby se dále zlepšilo využití motorů, bylo také nutné upustit od předepínání mechanismu, neboť se ukázalo, že pohony na to nejsou dostatečně výkonné (vliv zanedbaného tření). Síla F_{40} byla proto změněna na 4 N. Poslední změna se týká výpočetní periody, která byla po několika experimentech snížena na 4 ms.



Obrázek 5.2: Schéma ovládacího rozhraní v ControlDesku.

Takto upravené schéma je již připraveno pro běh na procesoru dSpace. V programu ControlDesk bylo tedy vytvořeno schéma pro ovládání regulátoru (obr. 5.2). Postup použití začíná zapnutím procesoru dSpace a jednotek EPOS. Mechanismus se při tom musí nacházet ve spodní poloze se všemi pohony opřeny v dorazech. Tím je zajištěna inicializace počítadel enkodérů v nulových pozicích. Dále v Matlabu provedeme sestavení schématu a nahrání real-time aplikace do procesoru. Nakonec aktivujeme komunikaci CAN a převedeme motorové jednotky do operačního režimu. Tím je řízení připraveno k experimentům. Nyní lze nastavit parametry trajektorie a posuvníkem „Bezpečnostní omezení proudů“ následně snižujeme omezení výstupních proudů a mechanismus se začne pohybovat. Toto omezení je řešeno jako proporcionální zesílení výstupu regulátoru a původně bylo navrženo jako ochrana před agresivním chováním regulátoru po zapnutí. Při experimentech se ale ukázalo jako účelné zvětšit rozsah tohoto zesílení z jedné⁴ až na hodnotu 4, řízení tak lépe překonává pasivní odpory, které byly při návrhu regulátoru zanedbány.

⁴Na obr. 5.2 je zachycena původní varianta se zesílením do jedné.

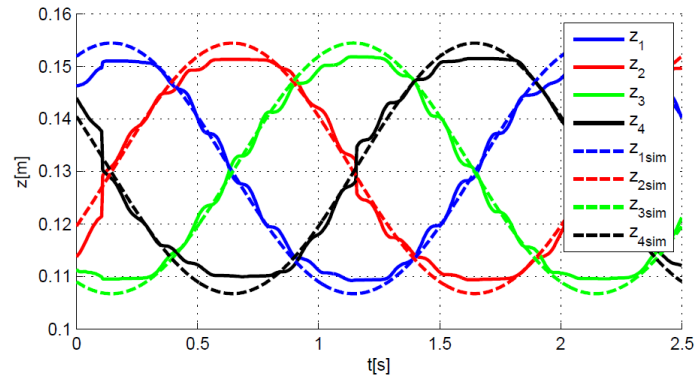


Obrázek 5.3: Žádaná a skutečná trajektorie platformy, poloměr 50 mm.
V pravo pro zatíženou platformu.

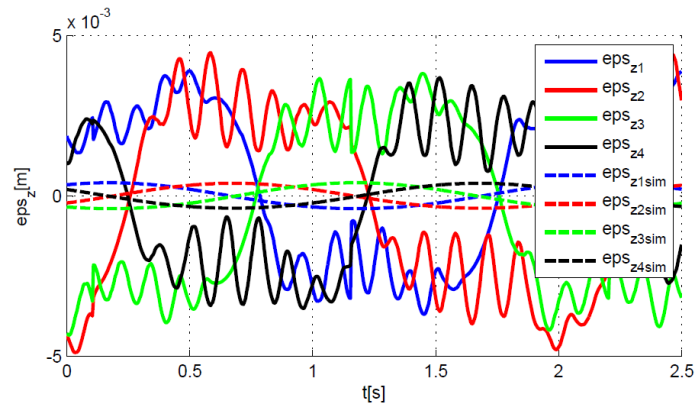
5.3 Výsledky a porovnání se simulací

Data použitá v této kapitole jsou získána prostřednictvím funkcí programu ControlDesk. Zvolil jsem záznam absolutních poloh pohonů \mathbf{z} , jejich odchylky od požadovaných poloh \mathbf{eps}_z a proudy \mathbf{I} , které reprezentují akční zásahy. Protože se oproti původním simulacím změnilo několik nastavení, byl vytvořen upravený simulační skript „SDtrajectoryMPC_OBSV_comp“ pro získání dat k porovnávání s experimenty, rozšířený o výpočet proudů. Pro vykreslení trajektorií platformy bylo navíc nezbytné vyřešit úlohu přímé kinematiky. Stejně jako v případě inverzní kinematiky nemá ani tato úloha analytické řešení, a s ohledem na nadbytečný pohon ani řešení jednoznačné. Výpočet tak spočívá v minimalizaci chyb splnění vazbových rovnic (4.2), k čemuž využívám funkci *fminsearch*. Spolu se samotným vykreslováním trajektorií se o to stará skript „DirectKinematics“ a funkce „DirKin“. Vykreslování přímo naměřených dat a dat ze simulací obstarává skript „Plotter“.

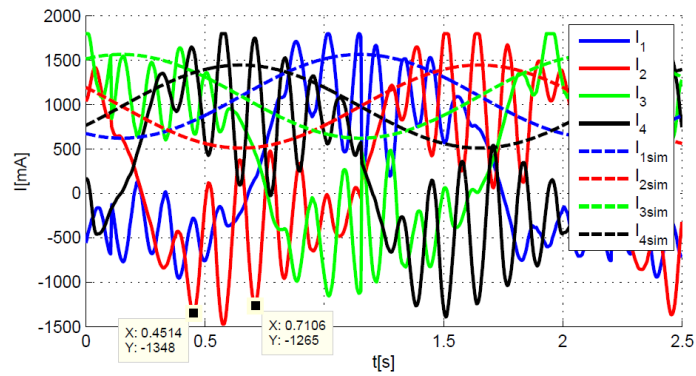
Jako první testovací trajektorii jsem použil kružnici navrženou v části 4.2. Porovnání odpovídajících simulačních a experimentálních dat je zachyceno na obr. 5.4 až obr. 5.6. Je zřejmé, že oproti simulaci dochází k větším odchylkám, a to zhruba o řád. Velikost odchylek lze částečně vysvětlit zanedbáním pasivních odporů při modelování. Tomu odpovídají i větší potřebné proudy pro zajištění pohybu mechanismu.



Obrázek 5.4: Průběhy souřadnic pohonů při sledování kružnice s poloměrem 50 mm.



Obrázek 5.5: Odchytky souřadnic pohonů při sledování kružnice s poloměrem 50 mm.



Obrázek 5.6: Průběhy proudů při sledování kružnice s poloměrem 50 mm.

Rozkmitání se jeví jako rezonance mechanismu, kterou regulátor není schopen utlumit. To je způsobeno předpokladem, že mechanismus je absolutně tuhý, a tedy zanedbáním poddajností při modelování. Podle průběhu oscilací lze soudit, že dochází ke kruhovým kmitům v rovině xy . Kmity jsou totiž pro sousední pohony vždy posunuty o půl periody. Vlastní frekvence tohoto módu závisí na poloze platformy, ale přibližně ji lze určit z obr. 5.6. Časový rozdíl dvou vyznačených bodů odpovídá dvěma periodám, vlastní frekvenci tedy spočtu jako

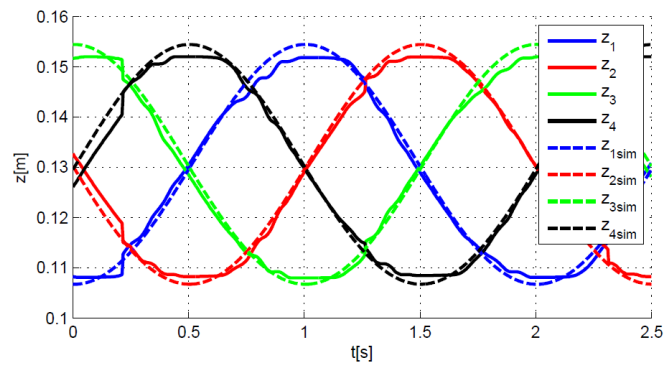
$$f_1 = \frac{2}{(0,7106 \text{ s} - 0,4514 \text{ s})} = 7,71 \text{ Hz.} \quad (5.7)$$

Pro ověření, že se skutečně jedná o rezonanci, byla platforma zatížena závažím o hmotnosti 2,3 kg a experiment byl opakován pro stejnou trajektorii. Na obr. 5.3 je zachyceno porovnání skutečných a požadovaných trajektorií platformy pro oba experimenty. Na obr. 5.7 až obr. 5.9 jsou pak vykresleny průběhy měřených veličin při pohybu se zátěží. Je zřejmé, že došlo k přeladění soustavy a navíc i ke snížení odchylek poloh pohonů oproti původnímu stavu. Z průběhů těchto odchylek a z průběhů proudů je také vidět, že dochází k vybudení jiného módu. Kmity jsou nyní synchronní pro všechny pohony. Mechanismus tedy kmitá ve směru osy z , přičemž příslušná vlastní frekvence je:

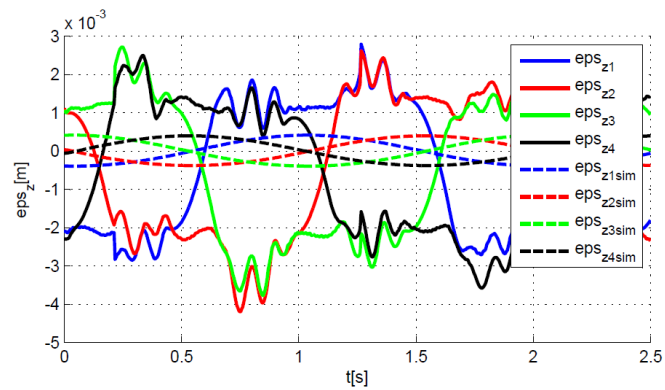
$$f_2 = \frac{2}{(1,359 \text{ s} - 1,191 \text{ s})} = 11,90 \text{ Hz.} \quad (5.8)$$

Snížení vlastních frekvencí přidáním zátěže způsobilo, že původní mód již není vybuzen. Rezonance v nově vybuzeném módu je navíc lépe tlumená, což lze vidět na snížení odchylek. Celkově se chování mechanismu po zatížení zlepšilo, ovšem za cenu přetěžování motorů. Pro lepší představu o průbězích experimentů jsou na přiloženém CD obsaženy jejich videozáznamy.

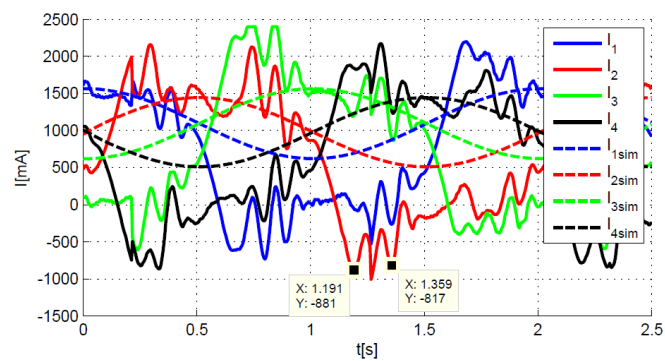
Stejná sada měření byla provedena ještě pro trajektorii ve tvaru kružnice s poloměrem 100 mm, příslušná data jsou zachycena v příloze G. Výsledky jsou podobné těm v této kapitole, ovšem s tím rozdílem, že dochází k ještě většímu přetěžování pohonů. V krajních polohách pracovního prostoru totiž většinu zátěže nese vždy nejbližší motor.



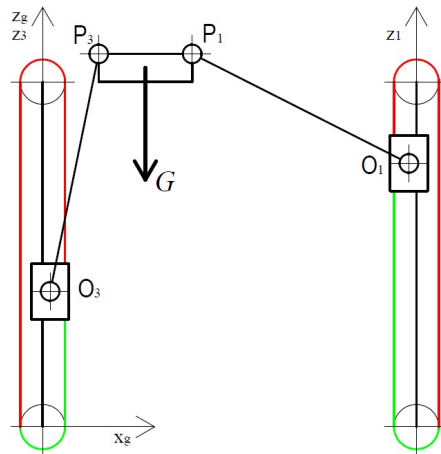
Obrázek 5.7: Průběhy souřadnic pohonů při sledování kružnice s poloměrem 50 mm, zatížená platforma.



Obrázek 5.8: Odchyly souřadnic pohonů při sledování kružnice s poloměrem 50 mm, zatížená platforma.



Obrázek 5.9: Průběhy proudů při sledování kružnice s poloměrem 50 mm, zatížená platforma.



Obrázek 5.10: Schéma mechanismu Sliding Delata s vykreslenými pohony.

K objasnění zdrojů kmitání, které zásadně ovlivňuje přesnost a stabilitu mechanismu, slouží obr. 5.10. Na něm je v pohledu z boku zachycena konfigurace použítá při experimentech. Barevně jsou zde vyznačeny řemeny lineárních pohonů. Při udržování zobrazené polohy jsou natahovány červené části řemenů, zelené jsou odlehčovány. Pokud tedy budeme sledovat cestu přenosu sil, zjistíme, že je velmi dlouhá. Přes jeden řemen, dva kulové klouby a spojnice na každé straně platformy. Pro určení dominantních poddajností by bylo třeba provést analýzu jednotlivých součástí, lze ale soudit, že největší vliv budou mít použité polyuretanové řemeny. Jejich aktivní (zatížená) část může v nejhorsím případě dosahovat délek i přes 600 mm. Spojnice jsou sice vyrobeny z trubek z uhlíkového laminátu, jejich délka ovšem není malá. I ty tak budou pravděpodobně přispívat k poddajnosti soustavy, kterou zřejmě není možné bez důsledků zanedbat.

Samotný výskyt poddajností by však k rozkmitání mechanismu nestačil, je potřeba ho vybudit. Zdrojem tohoto buzení se na základě pozorování a zkoumání naměřených dat zdá být reverzace pohonů. Při té totiž dochází jak ke skokové změně napnutí řemenů (záměna aktivních a pasivních částí) a tak k přestavení vůlí v převodovkách a ve vozících lineárních vedení. Navíc je vzhledem k pasivním odporům potřeba, aby došlo k překonání statického tření při změně směru pohybu. Značným zdrojem pasivních odporů jsou zřejmě použité kulové klouby igubal KBRM-06 od společnosti IGUS, které jsou vyrobeny z plastů. Jako jediné z kloubů uvažovaných při konstrukci mechanismu však měly dostatečný rozsah pohyblivosti. Bohužel, nároky na kloubové spoje jsou další často zmiňovanou nevýhodou paralelních kinematik. Jako nevhodná se jeví také velká délka příček paralelogramů ve srovnání s šířkou vozíků lineárních vedení.

Kapitola 6

Závěr

Postupně byly v této práci řešeny jednotlivé zadané úkoly. Nejprve byl vytvořen stručný přehled přístupů k řízení mechanismů s paralelní kinematickou strukturou a nadbytečnými pohony. Metody kvadraticky optimálního řízení, prediktivního řízení a momentového řízení byly následně rozepsány do té míry, aby bylo možné na navrhout a imlementovat příslušné regulátory. Podobně byl popsán i návrh stavového pozorovatele potřebného pro realizaci řízení na laboratorním modelu. Dále byl navržen jednoduchý 2D redundantní mechanismus 2Delta, pro který byl sestaven matematický model a následně i vytvořeny vybrané regulátory. Jejich funkčnost byla simulačně ověřena při stabilizaci mechanismu v požadované poloze a při sledování referenční trajektorie. Po tomto kroku byl z mé bakalářské práce převzat matematický model redundantního mechanismu se třemi stupni volnosti Sliding Delta, který byl rozšířen o nadbytečný pohon. Regulátory navržené v předchozí kapitole byly upraveny pro použití na tento složitější mechanismus. Simulačně bylo opět ověřeno chování systému při sledování referenční trajektorie. Jako nejlepší se nakonec i přes značná zjednodušení ukázal prediktivní regulátor, který byl dále upraven pro běh v reálném čase a řízení laboratorního modelu mechanismu Sliding Delta. Tento model byl za pomoci zvoleného hardware oživen a jeho chování bylo porovnáno se simulacemi. Byly tedy splněny všechny požadavky na tuto práci kladené.

Výsledky experimentů s laboratorním modelem však ukázaly na několik problémů. Hlavním z nich je rozkmitávání mechanismu na rezonanční frekvenci, což je způsobeno obtížně odstranitelnými konstrukčními nedostatky. Že se skutečně jedná o vlastnost mechanismu a nikoli regulátoru bylo ukázáno přeladěním vlastní frekvence při zatížení platformy. Mechanismus totiž není vzhledem k použitým komponentám dostatečně tuhý. Předpoklad dokonalé tuhosti při modelování je tedy neplatný a navržený lineární regulátor tak není

schopen kmitání utlumit. Jako problematické se ukazuje i zanedbání pasivních odporů, kterých se v mechanismu vyskytuje více, než bylo očekáváno. To má vliv jak na přesnost polohování, tak na zvýšení potřebných akčních sil, na což nejsou dostatečně dimenzovány použité pohony. Pasivní odpory se nejvíce projevují při reverzaci směru pohybu vozíků, ke které nedojde okamžitě, ale až po překonání statického tření. To způsobuje odchylky od kruhovitosti trajektorie. Značné vůle mezi vozíky a kolejnicemi lineárních vedení spolu s nevhodným poměrem délek příček paralelogramů vůči šířkám vozíků také snižují přesnost a stabilitu mechanismu. Kvůli nedostatečné momentové zatížení motorů bylo nutné upustit i od zamýšleného předepětí mechanismu za účelem vymezení vůlí. Potvrzuje se tak vlastně další často zmiňovaný problém paralelních kinematik, jímž je vyšší konstrukční náročnost.

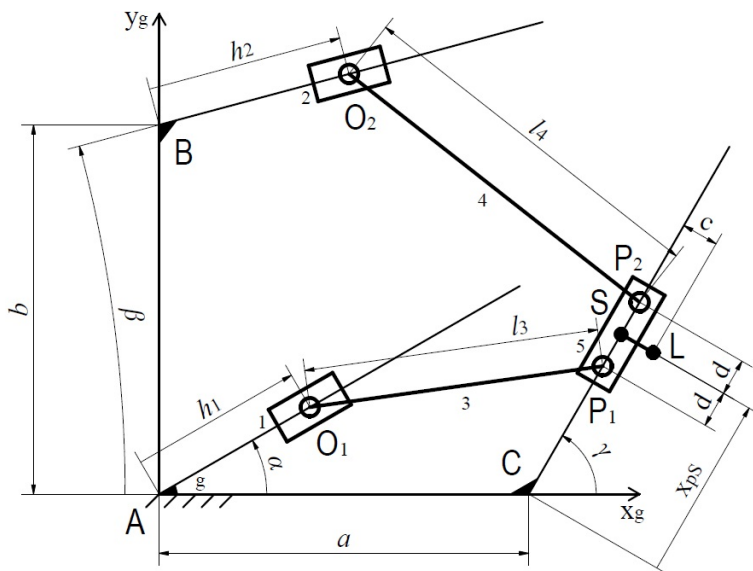
Co se týče dalšího možného postupu prací na mechanismu Sliding Delta, nabízejí se dvě cesty. Jednou z nich je použít jiné pohonné jednotky, které by vykazovaly vyšší tuhost jak v podélném tak v příčném směru. Nabízí se tedy použití kuličkových šroubů nebo přímo lineárních motorů. Navržené řízení by tak mohlo dobře pracovat i bez složitějšího modelování. Bohužel ale obě tyto varianty s sebou nesou vysoké náklady. Druhou cestou by bylo naopak zahrnout tuhost i pasivní odpory do návrhu regulátoru, což by obnášelo vytvoření alespoň jednoduchého MKP modelu. Výhodou by byla i výměna použitých převodovek za typ s větším převodovým poměrem nebo doplnění přídatného převodu do pomala. To by vedlo ke zvětšení dosažitelných akčních momentů. V obou případech by bylo dále vhodné navrhnout složitější MPC regulátor. Linearizace modelu by mohla probíhat vždy v určitých časových krocích, nebo by mohlo být navrženo regulátorů více pro různé stavy systému a mezi nimi by se pak podle potřeby přepínalo. Bylo by také možné prozkoumat možnosti nelineárního řízení. Z praktického pohledu by také bylo vhodné implementovat inicializační sekvenci pro převedení mechanismu do definované výchozí polohy a ošetřit možnosti nárazu pohonů do dorazů.

Literatura

- BELDA, K. (2002), *Řízení redundantních paralelních struktur robotických systémů*. Diplomová práce. Praha: ČVUT.
- ŠIKA, Z., VALÁŠEK, M., BAUMA, V. and VAMPOLA, T. (2003), Design of Redundant Parallel Robots by Multidisciplinary Virtual Modelling. *Virtual Nonlinear Multibody Systems*. Praha: Vydavatelství ČVUT.
- KOZÁK, T. (2012), *Paralelní kinematické struktury*. Bakalářská práce. Praha: ČVUT.
- MÜLLER, A. and HUFNAGEL, T. (2012), Model-based control of redundantly actuated parallel manipulators in redundant coordinates. *Robotics and Autonomous Systems 60*. Výzkumná zpráva. Dostupná také z WWW: <http://www.elsevier.com/locate/robot>.
- ROUBAL, J., PEKAŘ, J., PACHNER, D. and HAVLENA, V. (2005), *Moderní teorie řízení - cvičení*, Praha: Vydavatelství ČVUT.
- SKOPEC, T. (2004), *Syntéza různých řízení mechanismu s redundantní paralelní kinematikou Sliding Star*. Diplomová práce. Praha: ČVUT.
- STEJSKAL, V., VALÁŠEK, M. (1996), *Kinematics and Dynamics of Machinery*, New York: Marcel Dekker. ISBN 0-8247-9731-0.
- VALÁŠEK, M. a kol. (1995), *Mechatronika*, Praha: Vydavatelství ČVUT.
- VALÁŠEK, M., BAUMA, V. and ŠIKA, Z. (2006), *Mechanika B*, Praha: Vydavatelství ČVUT.
- VALÁŠEK, M., STEJSKAL, V. and BŘEZINA, J. (2007), *Mechanika A*, Praha: Vydavatelství ČVUT.

Příloha A

Parametry modelů mechanismů



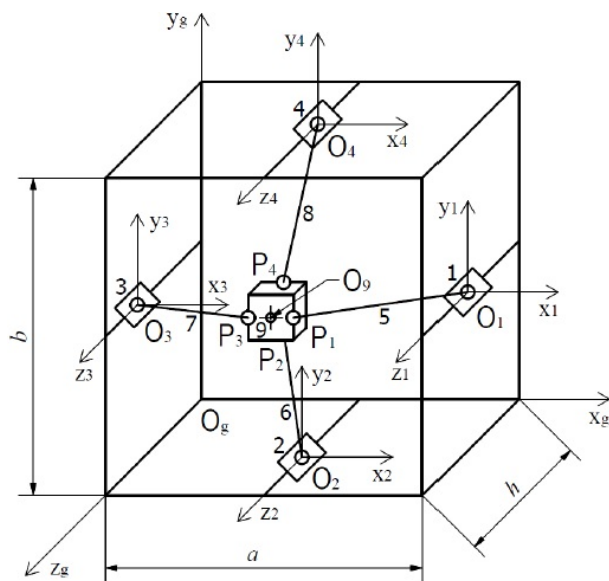
Obrázek A.1: Schéma mechanismu 2Delta.

Tabulka A.1: Parametry modelu 2Delta.

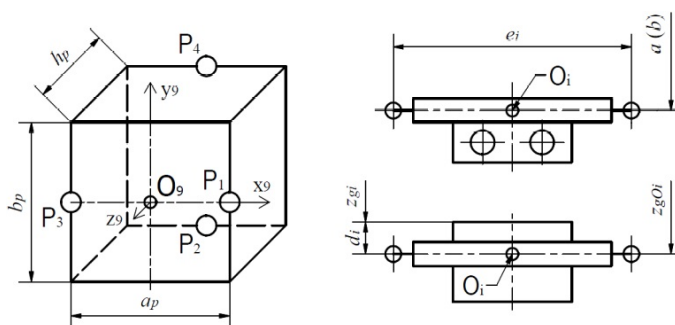
parametr	hodnota
a	0,1 m
b	0,1 m
c	0,01 m
d	0,01 m
l_3	0,08 m
l_4	0,1 m
α	30°
β	15°
γ	60°

parametr	hodnota
g	9,81 ms ⁻²
m_1	0,2 kg
m_2	0,2 kg
m_3	0,1 kg
m_4	0,1 kg
m_5	0,3 kg

parametr	hodnota
h_{10}	0,05 m
h_{20}	0,05 m
x_{ps0}	0,05 m
F_{20}	2 N
F_{5x}	0 N
F_{5y}	0 N



Obrázek A.2: Zjednodušené schéma Sliding Delta.



Obrázek A.3: Zjednodušené schéma platformy a vozíku lineárního vedení.

Tabulka A.2: Parametry laboratorního modelu.

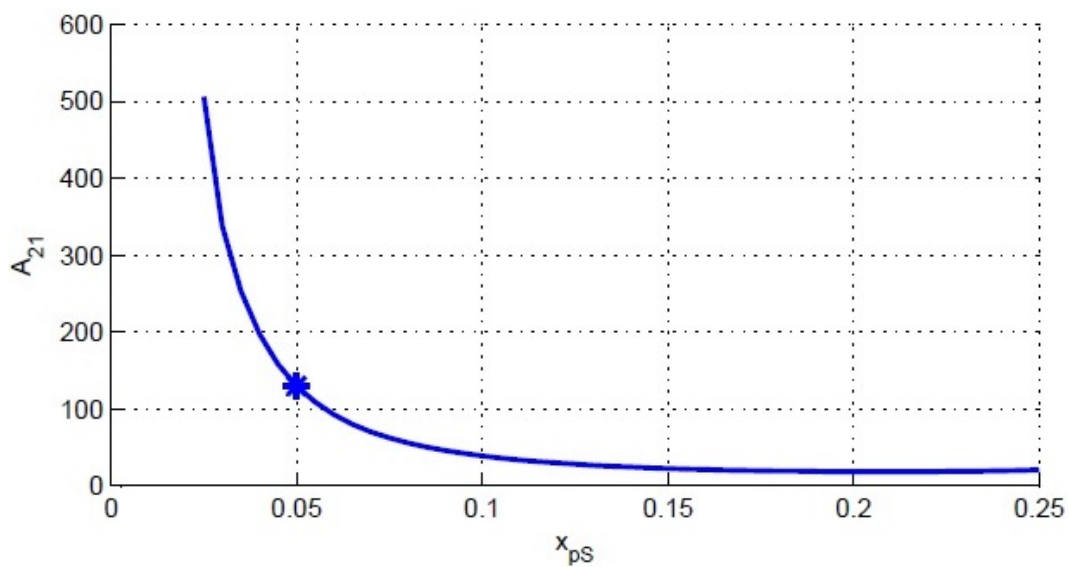
parametr	hodnota
a	468 mm
b	468 mm
h	350 mm
a_p	80 mm
b_p	80 mm
h_p	50 mm
d_1, d_3	55 mm
d_2, d_4	5 mm
$l_1 - l_4$	460 mm

parametr	hodnota
m_1, m_3	0,164 kg
m_2, m_4	0,132 kg
$m_5 - m_8$	0,100 kg
m_9	0,735

omezení	hodnota
x_{gO_9min}	84 mm
x_{gO_9max}	384 mm
y_{gO_9min}	114 mm
y_{gO_9max}	354 mm
F_{40}	5 N

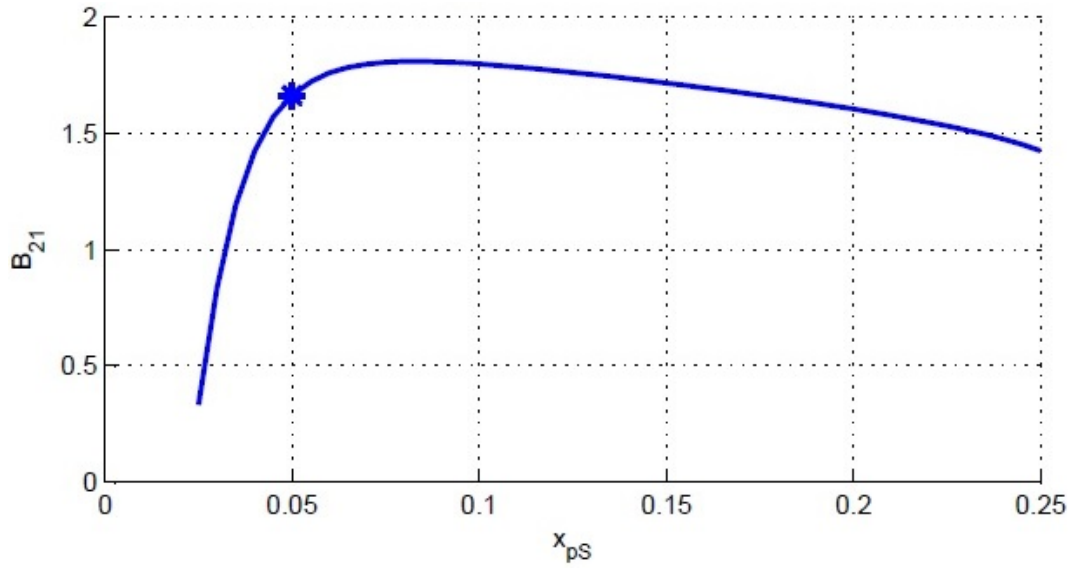
Příloha B

Prvky matic stavového popisu 2Delta v různých polohách

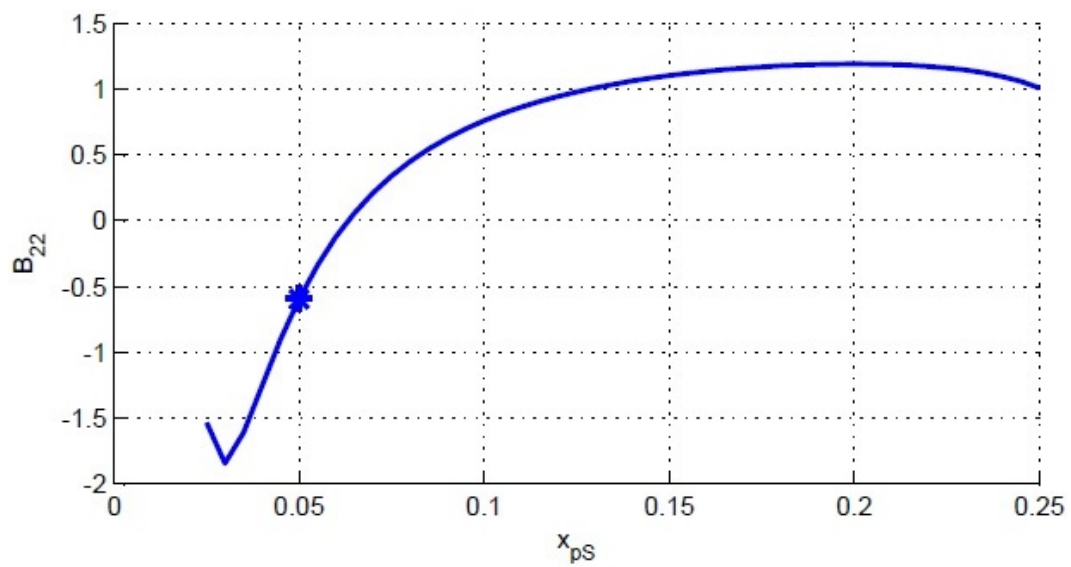


Obrázek B.1: Prvek A_{21} matice \mathbf{A} pro různé polohy.

VIPŘÍLOHA B. PRVKY MATICE STAVOVÉHO POPISU 2DELTA V RŮZNÝCH POLOHÁCH



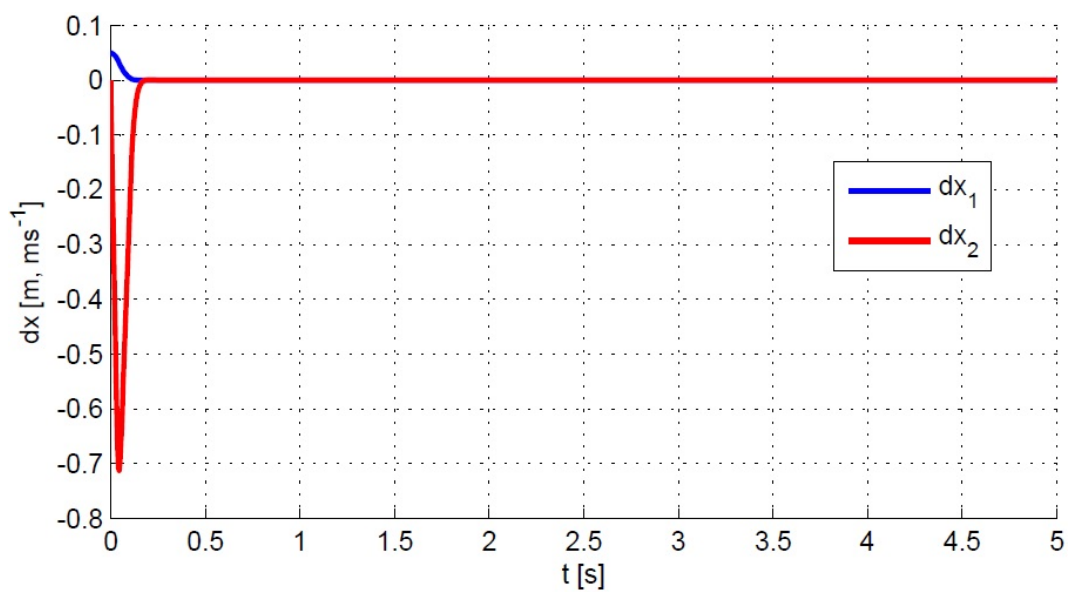
Obrázek B.2: Prvek B_{21} matice \mathbf{B} pro různé polohy.



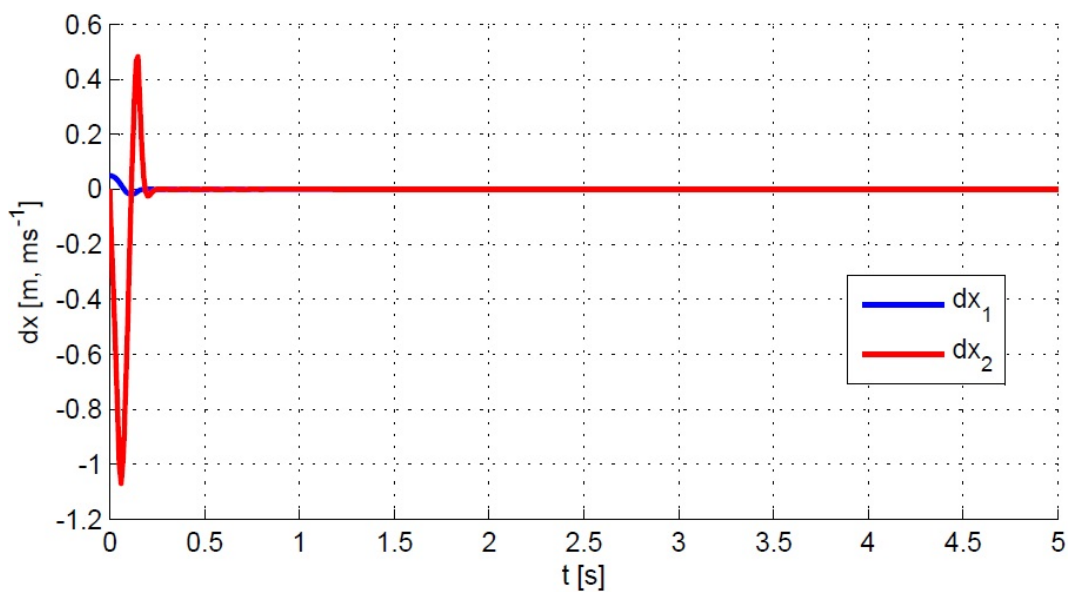
Obrázek B.3: Prvek B_{22} matice \mathbf{B} pro různé polohy.

Příloha C

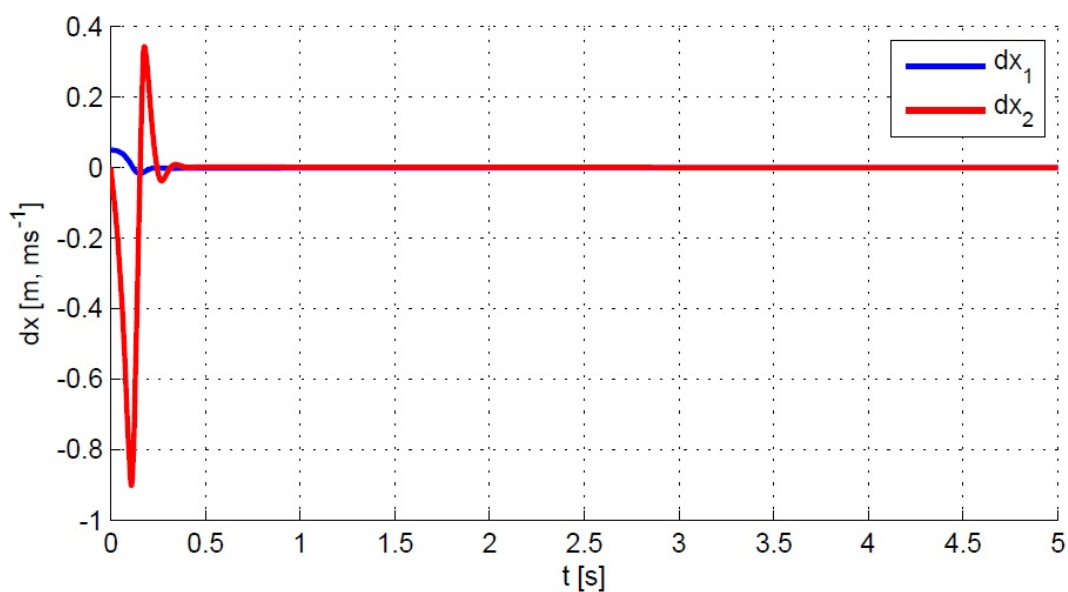
Stabilizace mechanismu 2Delta



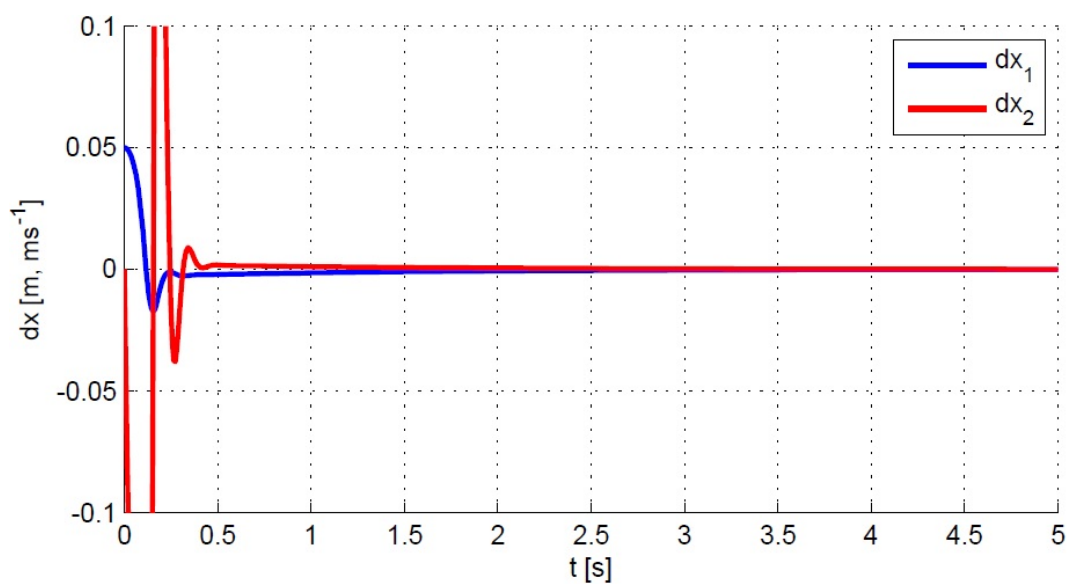
Obrázek C.1: Průběh odchylek stavů v průběhu stabilizace LQR.



Obrázek C.2: Průběh odchylek stavů v průběhu stabilizace MPC.



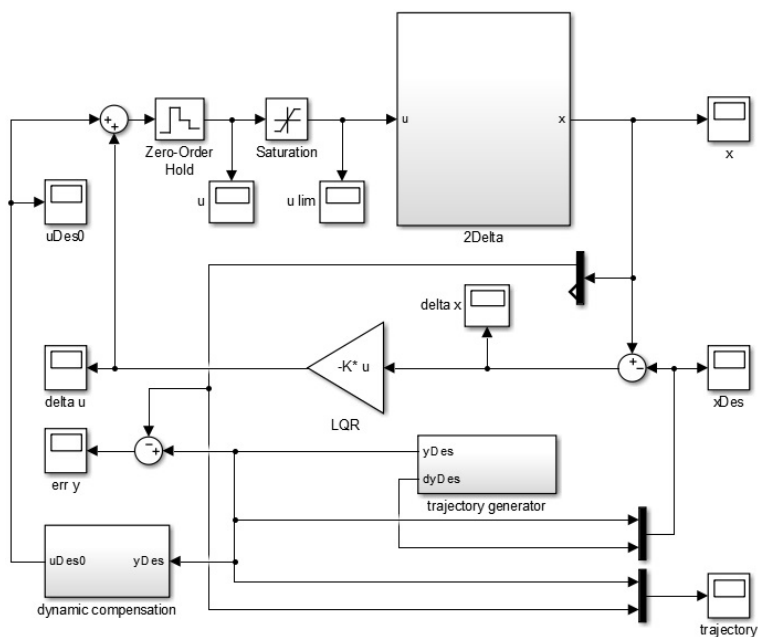
Obrázek C.3: Průběh odchylek stavů v průběhu stabilizace CTC.



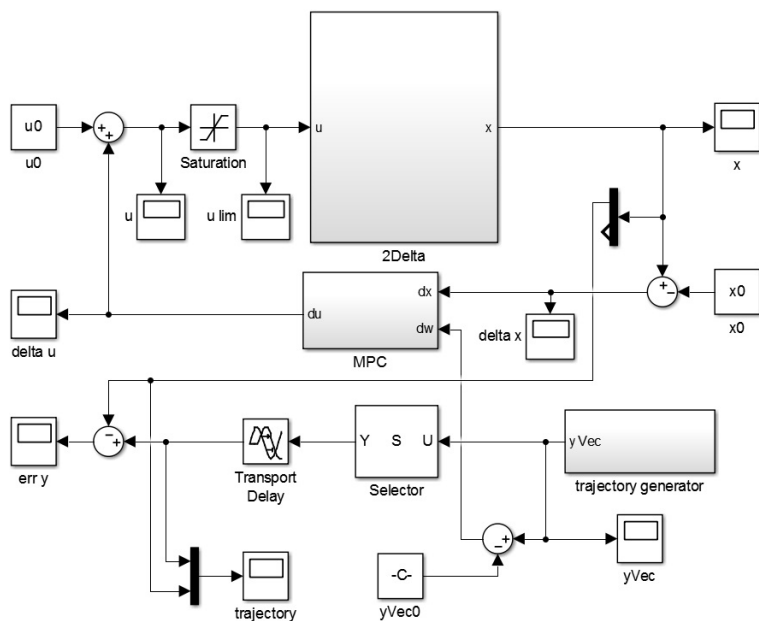
Obrázek C.4: Detail průběhu odchylek stavů v průběhu stabilizace CTC.

Příloha D

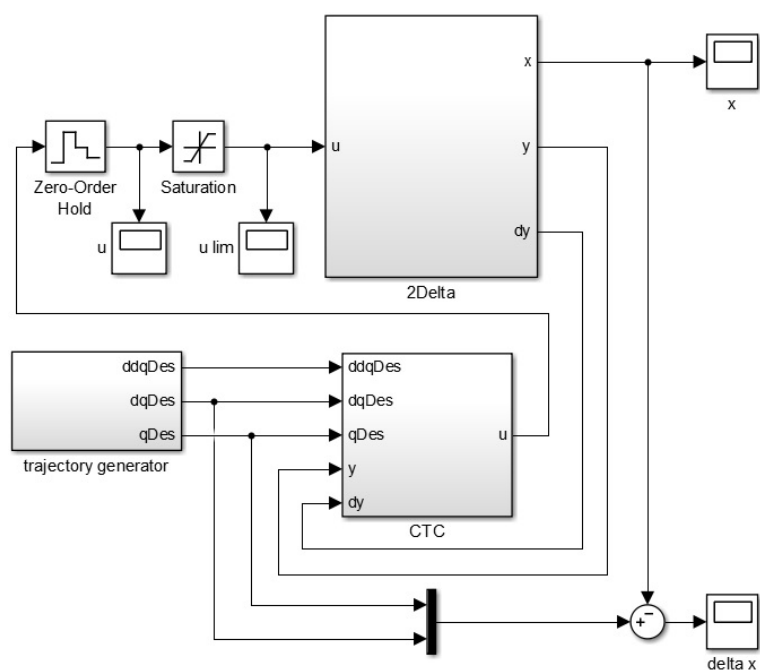
Sledování trajektorie - 2Delta



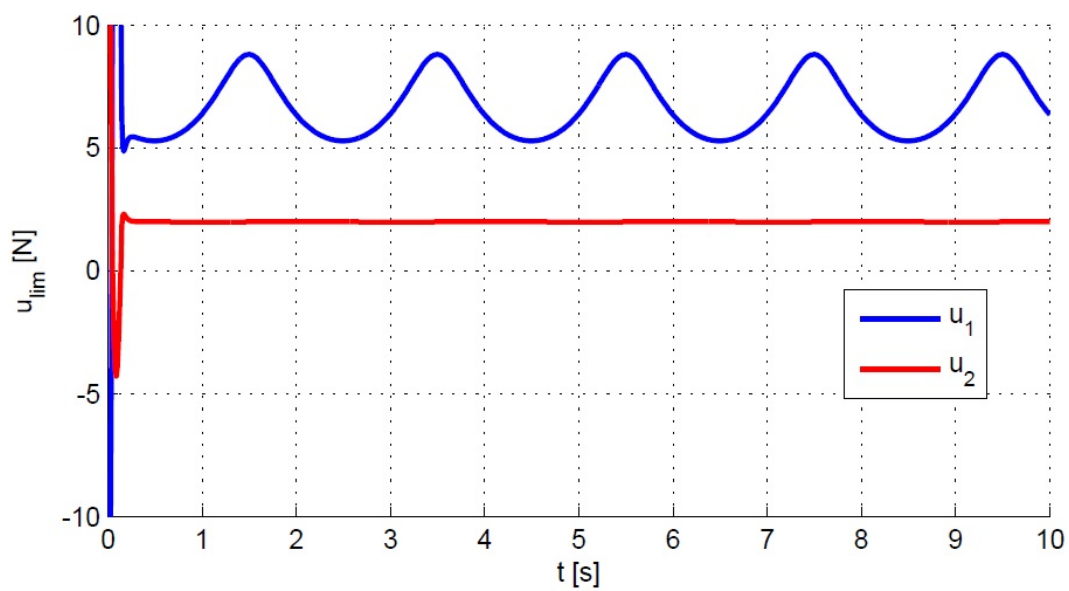
Obrázek D.1: Simulační schéma pro sledování trajektorie LQR.



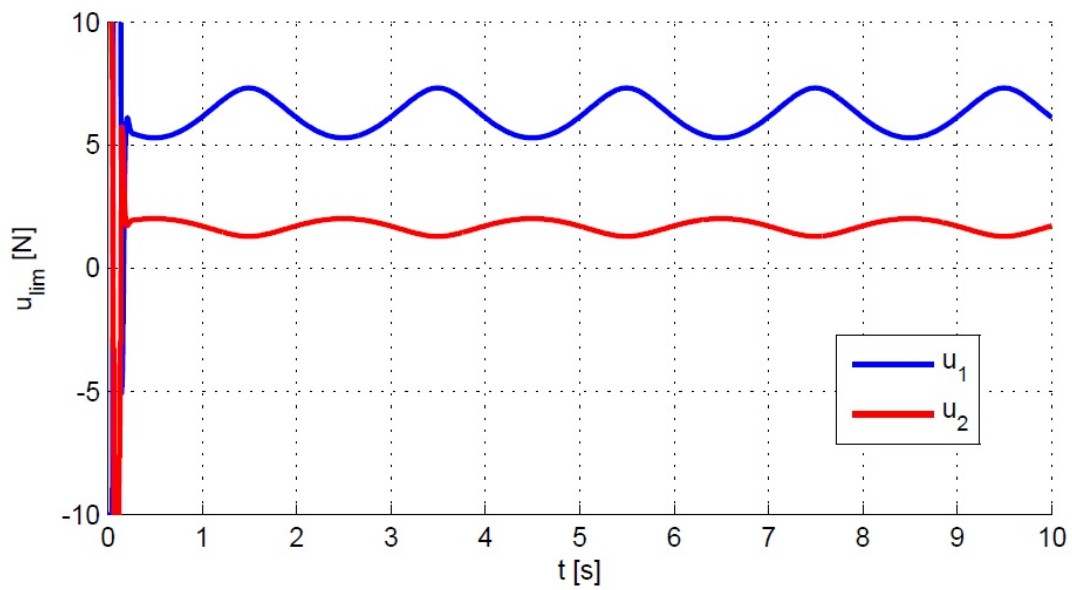
Obrázek D.2: Simulační schéma pro sledování trajektorie MPC.



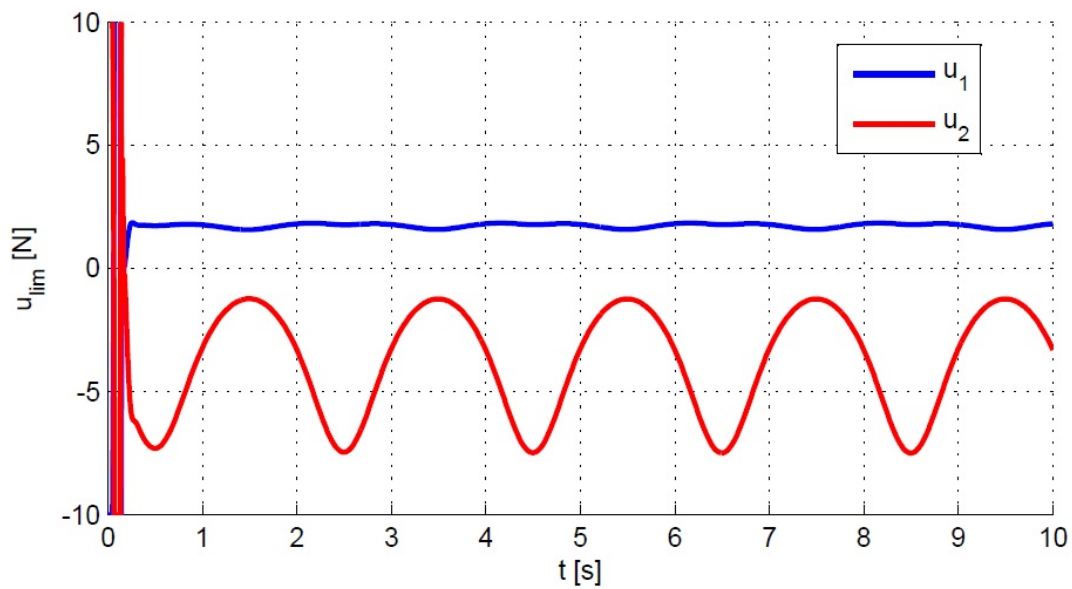
Obrázek D.3: Simulační schéma pro sledování trajektorie CTC.



Obrázek D.4: Průběh omezených výstupů při řízení pomocí LQR.



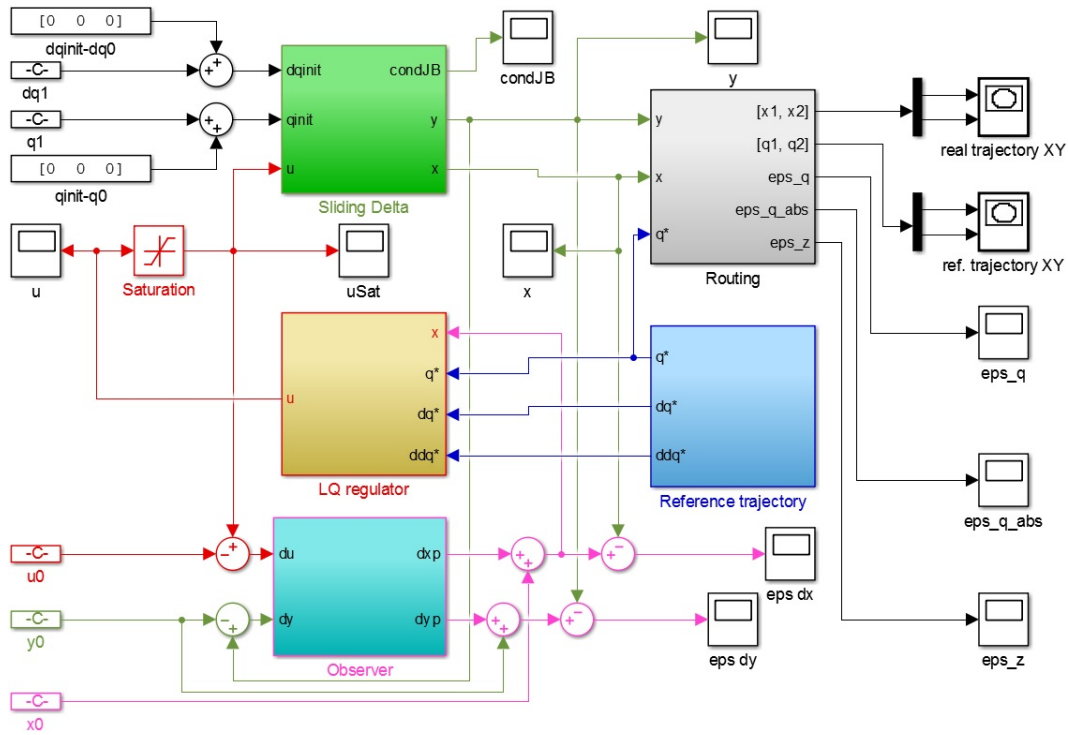
Obrázek D.5: Průběh omezených výstupů při řízení pomocí MPC.



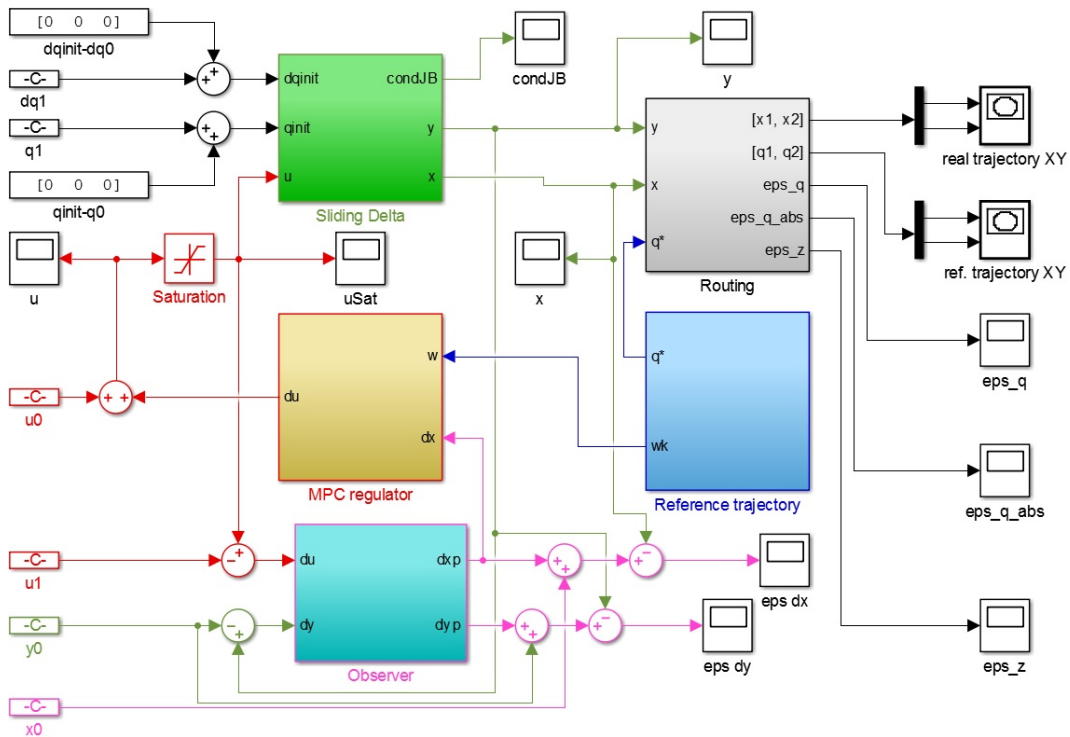
Obrázek D.6: Průběh omezených výstupů při řízení pomocí CTC.

Příloha E

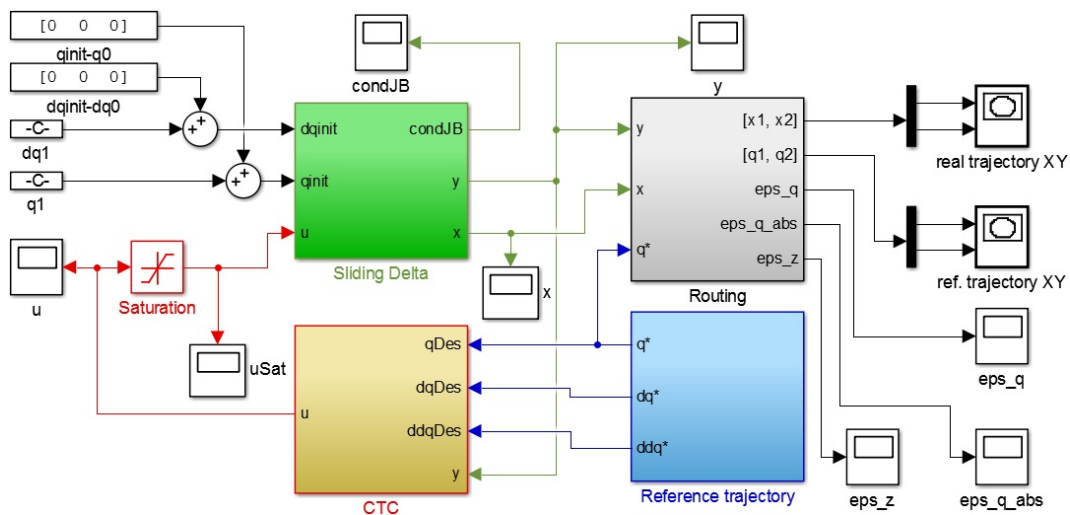
Sledování trajektorie - Sliding Delta



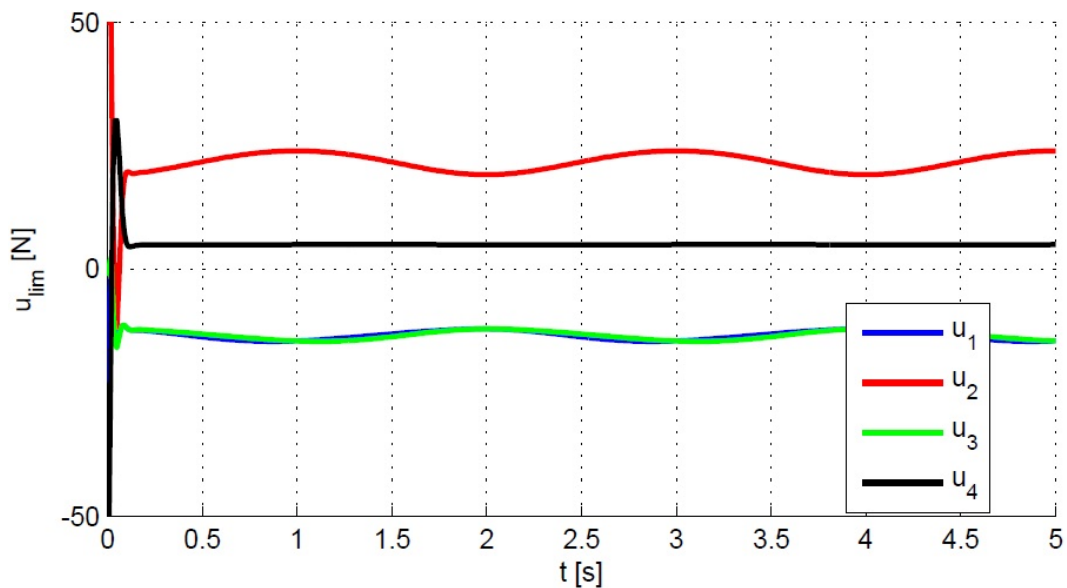
Obrázek E.1: Simulační schéma pro sledování trajektorie LQR - SD.



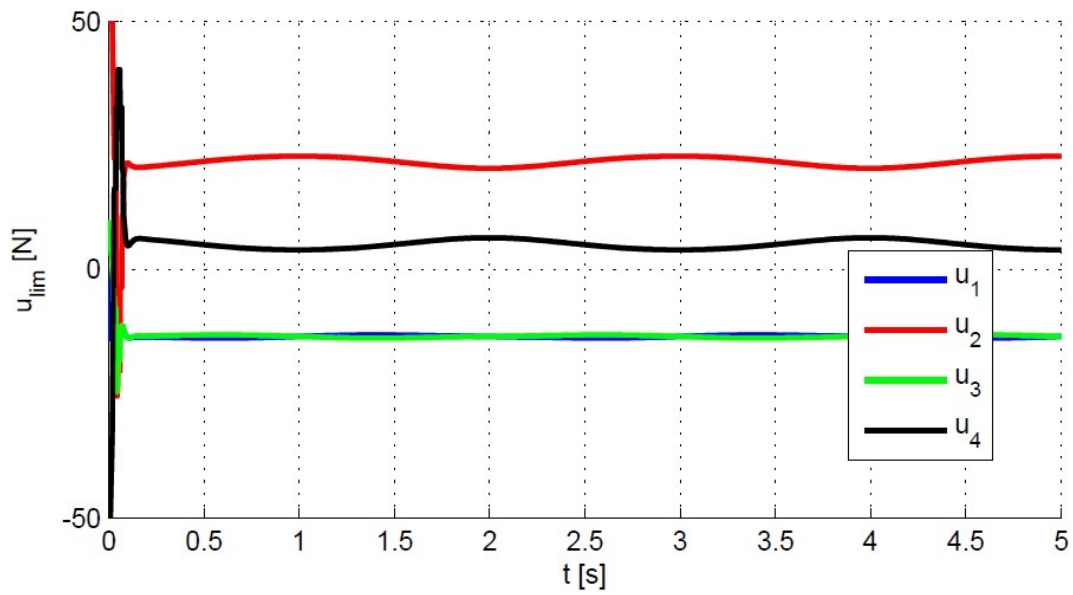
Obrázek E.2: Simulační schéma pro sledování trajektorie MPC - SD.



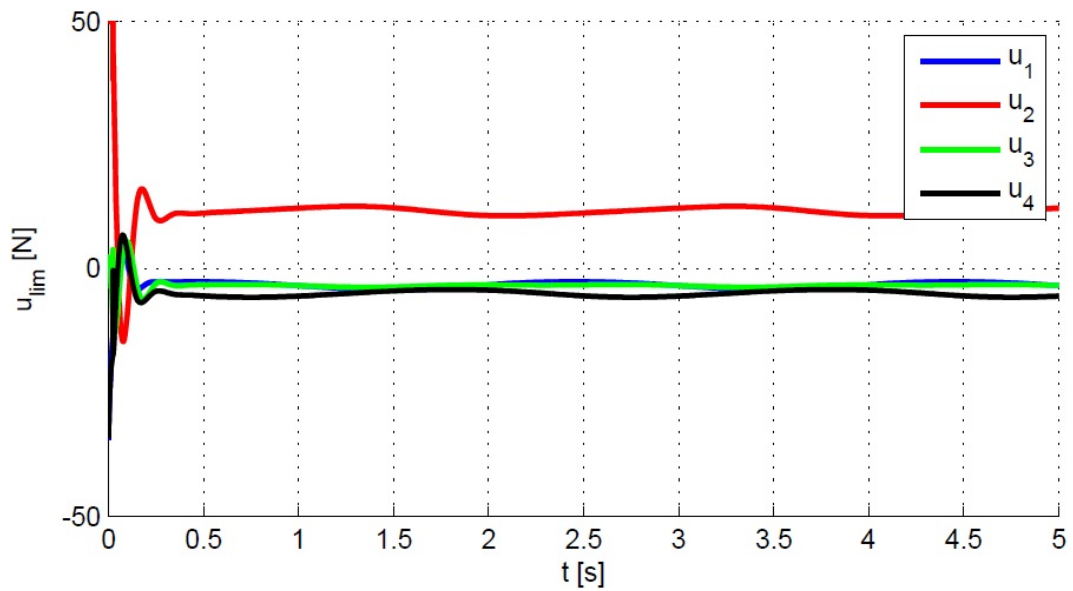
Obrázek E.3: Simulační schéma pro sledování trajektorie CTC - SD.



Obrázek E.4: Průběh omezených výstupů při řízení pomocí LQR - SD.



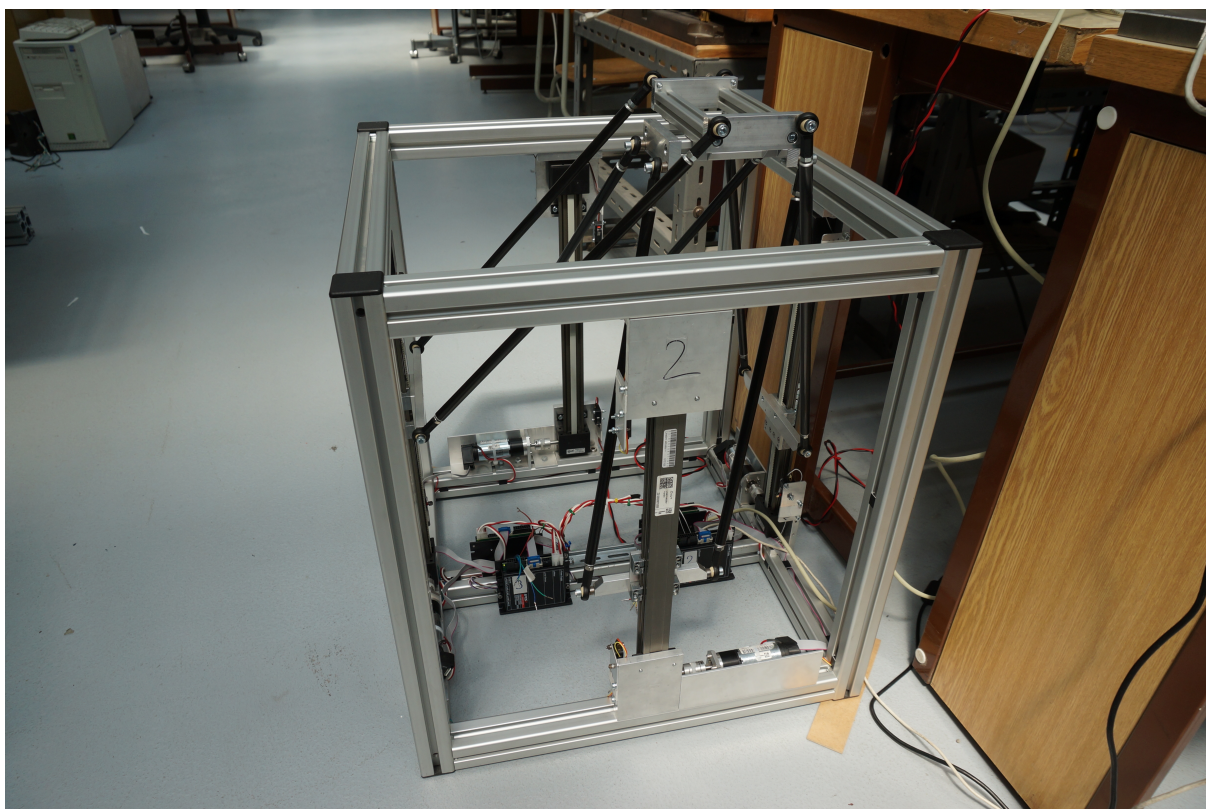
Obrázek E.5: Průběh omezených výstupů při řízení pomocí MPC - SD.



Obrázek E.6: Průběh omezených výstupů při řízení pomocí CTC - SD.

Příloha F

Oživování laboratorního modelu



Obrázek F.1: Laboratorní model mechanismu Sliding Delta.

Tabulka F.1: Parametry pohonů.

motor - A-max 32

výrobce		Maxon motor
objednací číslo		236669
výkon	P	20 W
napájecí napětí	U_m	24 V
nominální proud (max. trvalý)	I_{nom}	1,33 A
nominální moment (max. trvalý)	T_{nom}	45,5 mNm
momentová konstanta	k_T	35,2 mNm/A
moment setrvačnosti	J_m	45,3 gcm ²
teplotní časová konstanta	τ_t	18,3 s

rotační enkodér - HEDL 5540

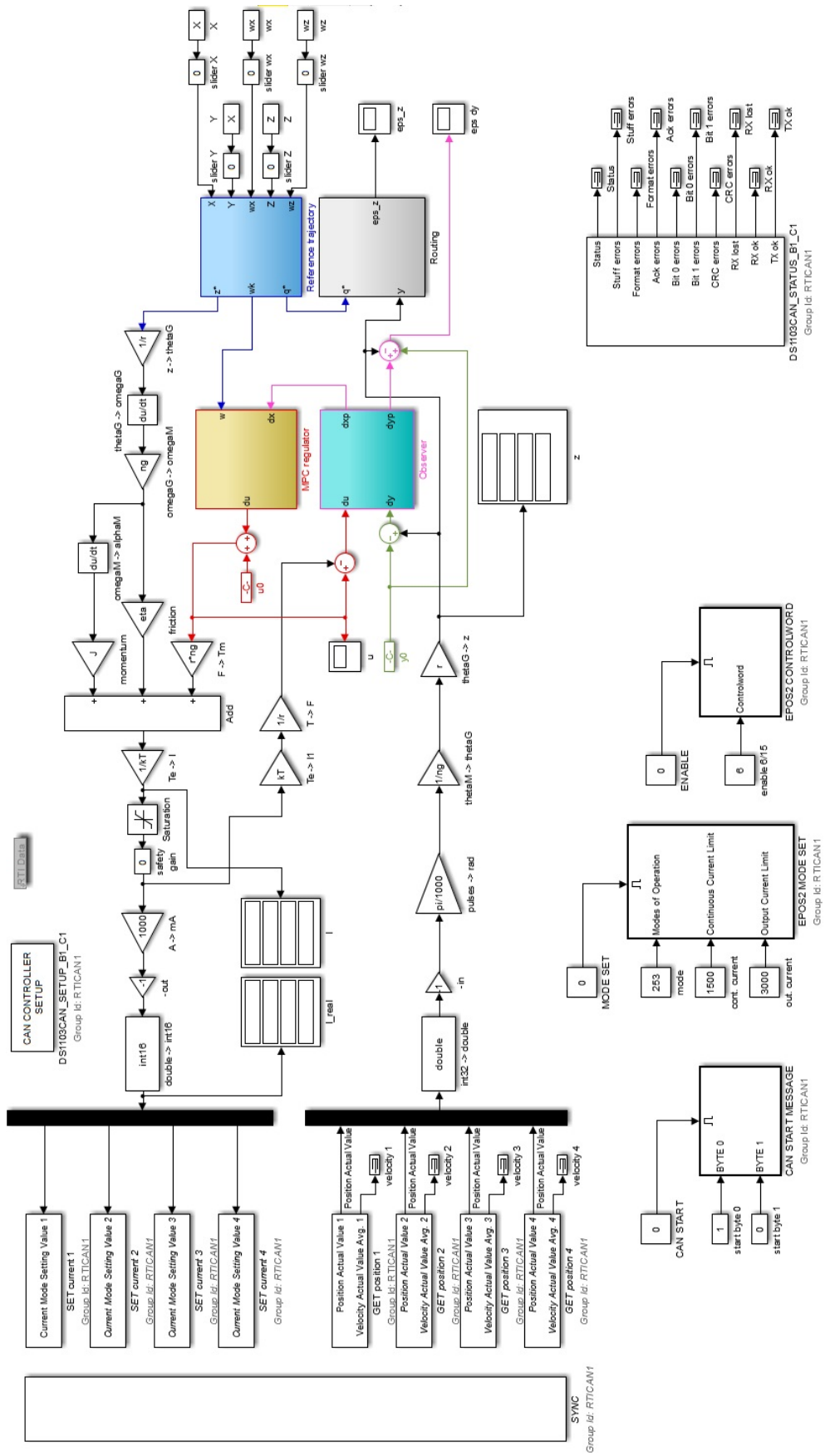
výrobce		Maxon motor
objednací číslo		110514
napájecí napětí	U_s	5 V
počet pulzů na otáčku	cpt	500
počet kanálů		3 (A, B, I)
moment setrvačnosti	J_m	0,6 gcm ²

převodovka - 32 A, Metal Version

výrobce		Maxon motor
objednací číslo		166156
převodový poměr	n_g	4,8
maximální moment	T_{max}	750 mNm
moment setrvačnosti	J_m	1,5 gcm ²

lineární vedení - DryLin ZLW

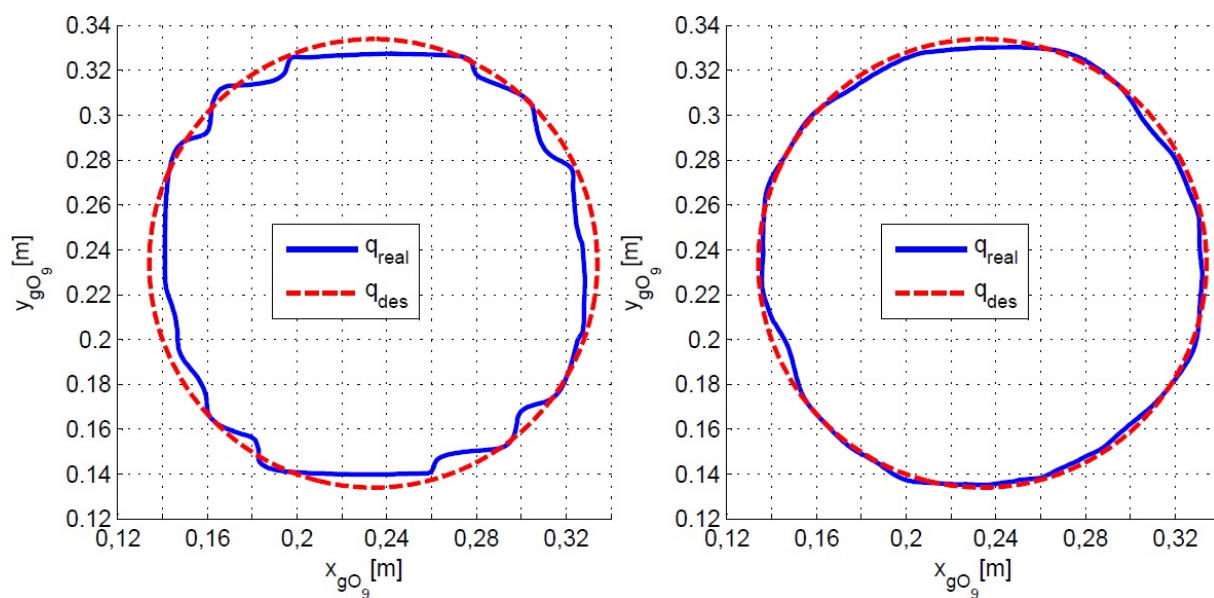
výrobce		IGUS
objednací číslo		ZLW-0630-02-S-60-R
převodový poměr	r	$54/2\pi$ mm/rad
poloměr řemenice	R	8,59 mm
předpětí řemenu	F_p	100 N



Obrázek F.2: Schéma regulátoru MPC pro běh v reálném čase.

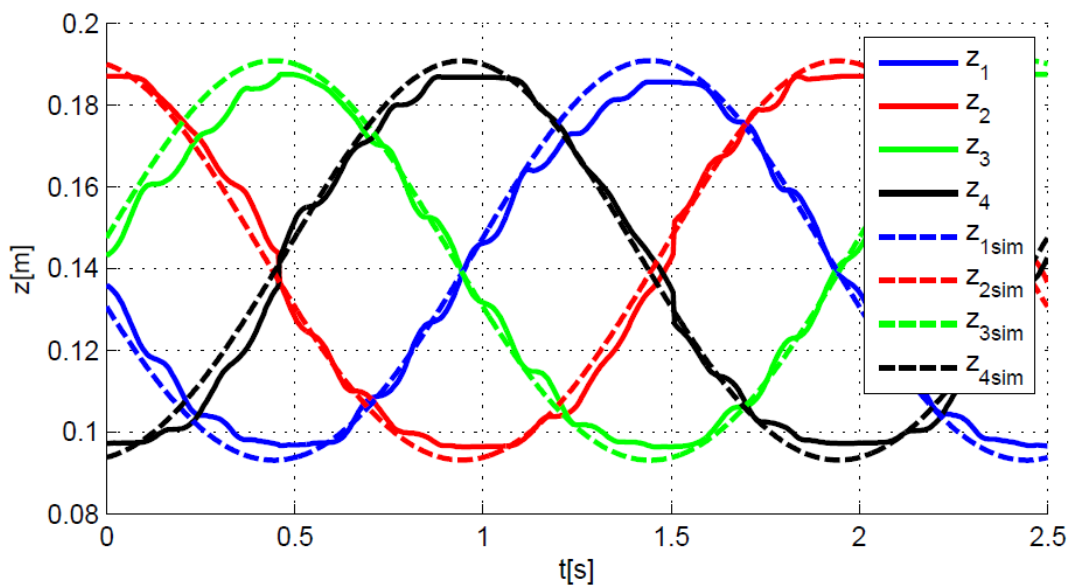
Příloha G

Experimentální výsledky

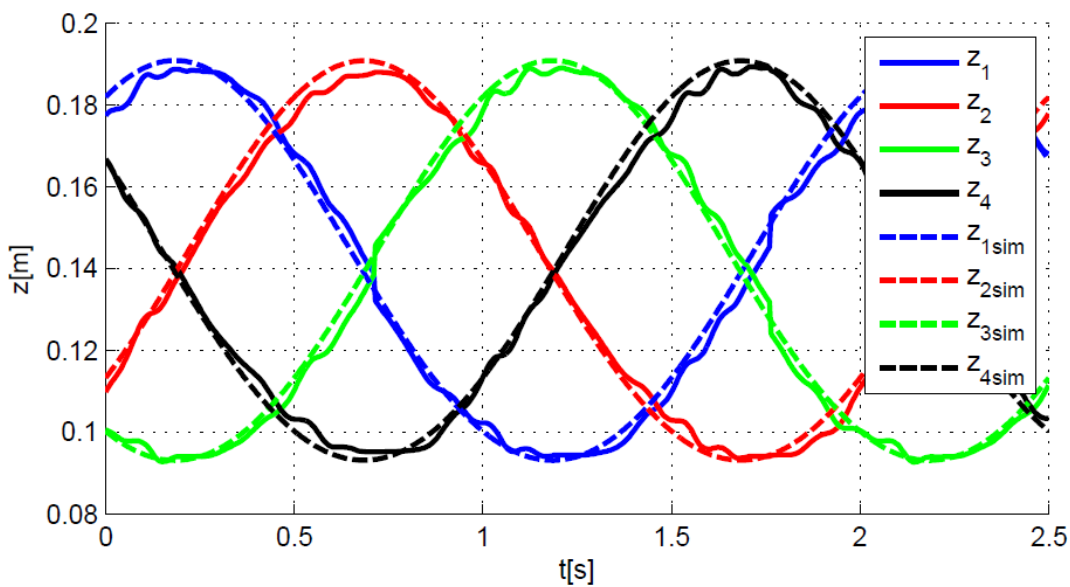


Obrázek G.1: Žádaná a skutečná trajektorie platformy, poloměr 100 mm.

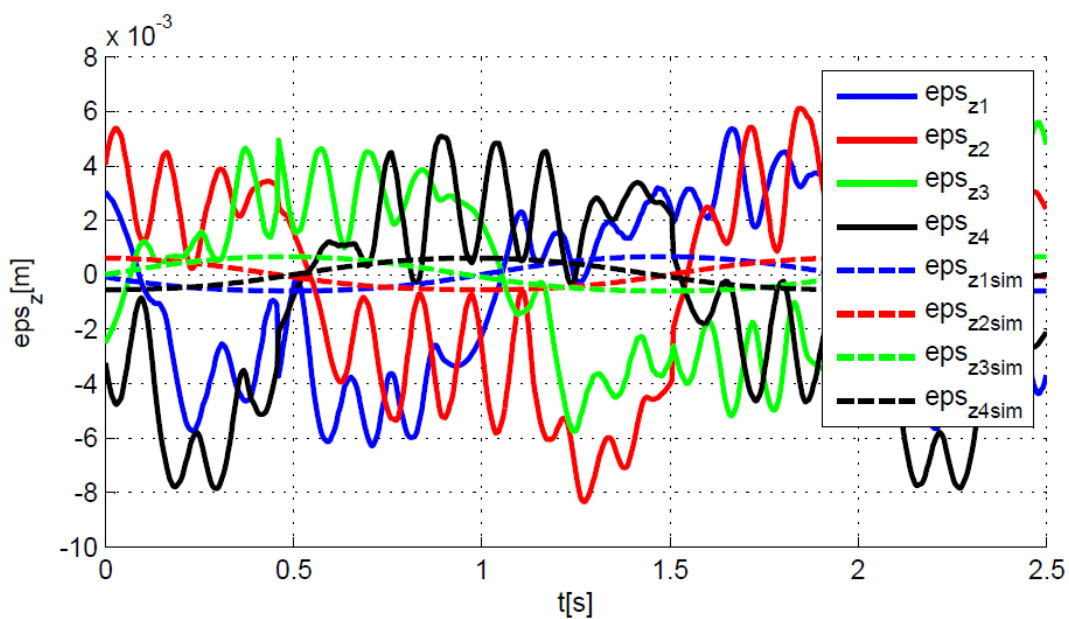
V pravo pro zatíženou platformu.



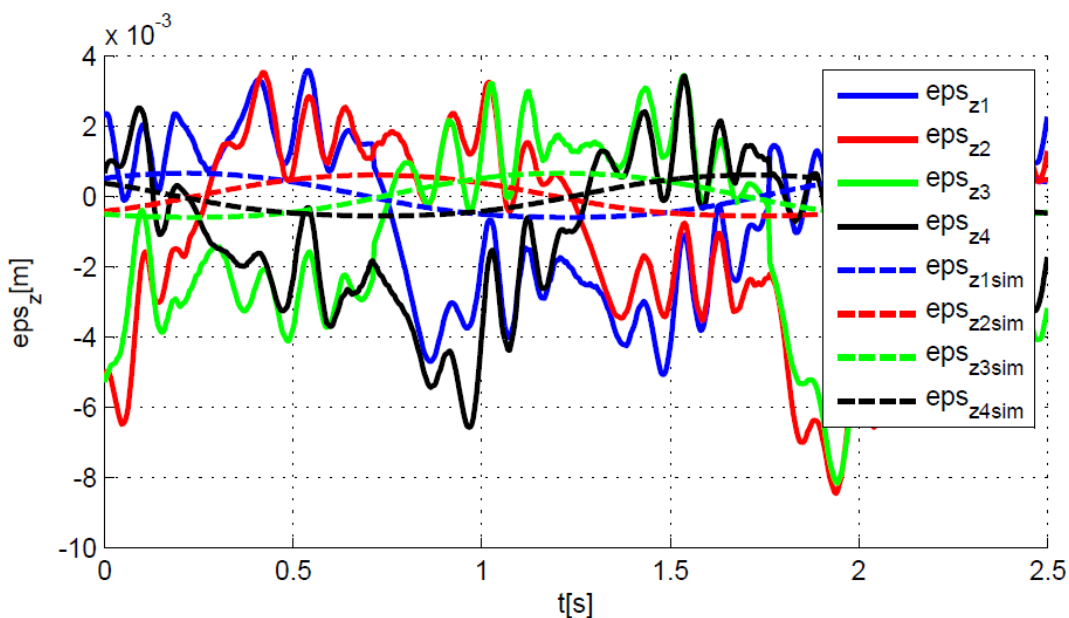
Obrázek G.2: Průběhy souřadnic pohonů při sledování kružnice s poloměrem 100 mm.



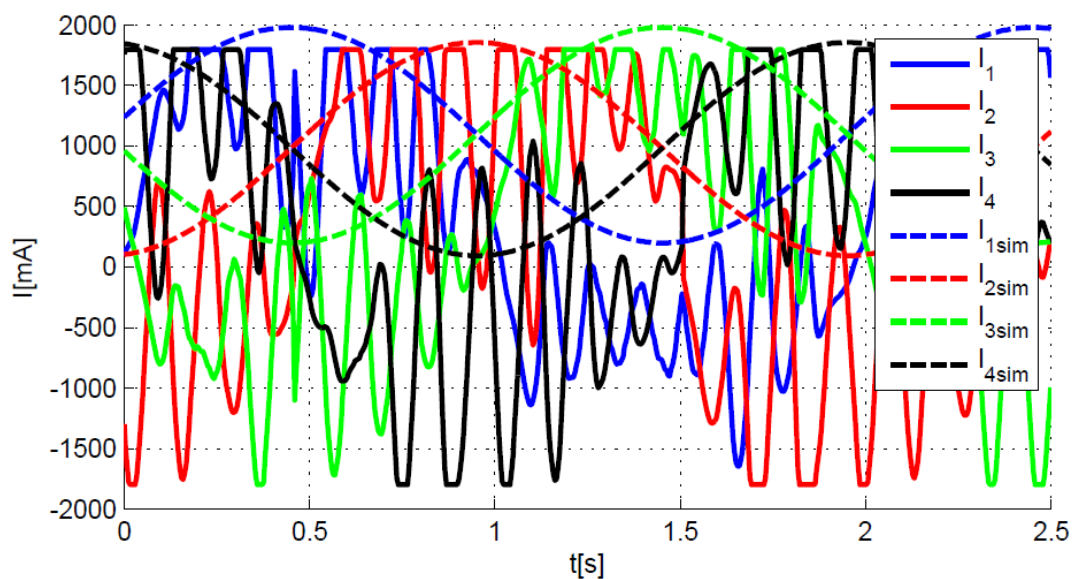
Obrázek G.3: Průběhy souřadnic pohonů při sledování kružnice s poloměrem 100 mm, zatížená platforma.



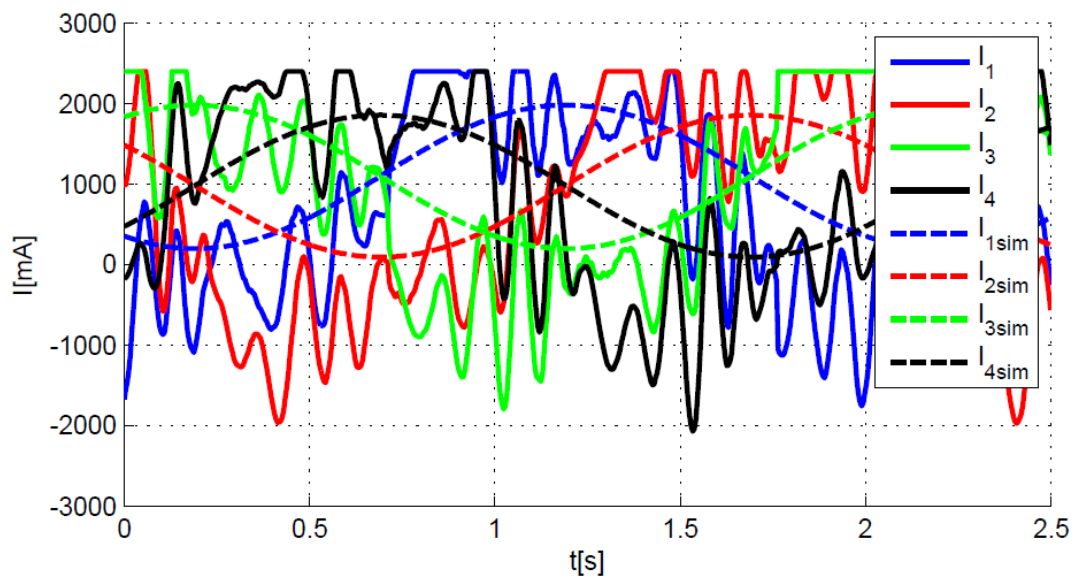
Obrázek G.4: Odchyly souřadnic pohonů při sledování kružnice s poloměrem 100 mm.



Obrázek G.5: Odchyly souřadnic pohonů při sledování kružnice s poloměrem 100 mm, zatížená platforma.



Obrázek G.6: Průběhy proudů při sledování kružnice s poloměrem 100 mm.



Obrázek G.7: Průběhy proudů při sledování kružnice s poloměrem 100 mm, zatížená platforma.

Příloha H

Obsah přiloženého CD

Přiložené CD obsahuje všechny soubory vytvořené v rámci této diplomové práce. Jedná se zejména o skripty a schémata z programu Matlab a řídicí projekt z programu ControlDesk. Přiloženy jsou také fotografie pracoviště a mechanismu Sliding Delta a videa z provedených experimentů. Pro případ navázání na tuto práci se na CD nachází také 3D model mechanismu vytvořený v programu Autodesk Inventor.

- **Diplomová práce.pdf**

- **Sliding Delta - 3D model:** Autodesk Inventor Professional 2015
 - \Worspaces\Worspace\finální model\final.iam

- **2Delta - simulace:** Matlab 2014a 32bit
 - skripty (.m)
 - schémata (.slx)

- **Sliding Delta - simulace:** Matlab 2014a 32bit
 - skripty (.m)
 - schémata (.slx)

- **Sliding Delta - experiment:** Matlab 2013a 32bit
 - fotografie (.jpg)
 - videa (.MTS)
 - zaznamenaná data (.mat)
 - dSpace projekt.zip (ControlDesk 5.1)
 - skripty (.m)
 - schémata (.slx)