

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2020

Bc. Dominik Friml



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

POROVNÁNÍ H-NEKONEČNO A LQG REGULACE

COMPARISON OF H-INFINITY AND LQG CONTROLLER

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Dominik Friml

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Lukáš Pohl, Ph.D.

BRNO 2020



Diplomová práce

magisterský navazující studijní obor **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

Student: Bc. Dominik Friml

ID: 186060

Ročník: 2

Akademický rok: 2019/20

NÁZEV TÉMATU:

Porovnání H-nekonečno a LQG regulace

POKYNY PRO VYPRACOVÁNÍ:

1. Proveďte rešerši obou přístupů k regulaci dynamické soustavy.
2. Implementujte výpočet H-nekonečno a LQG regulace v prostředí MATLAB.
3. Simulačně ověřte robustnost a rychlost obou typů regulace na vybrané dynamické soustavě.
4. Zhodnoťte kvalitu regulace obou přístupů s ohledem na náročnost výpočtu regulátoru a samotného akčního zásahu.

DOPORUČENÁ LITERATURA:

SKOGESTAD, Sigurd; POSTLETHWAITE, Ian. Multivariable feedback control: analysis and design. New York: Wiley, 2007.

Termín zadání: 3.2.2020

Termín odevzdání: 1.6.2020

Vedoucí práce: Ing. Lukáš Pohl, Ph.D.

doc. Ing. Václav Jirsík, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Práce se zabývá popisem MIMO systémů a porovnáním frekvenčních charakteristik, kromě jiných i charakteristikou singulárních hodnot. Jsou definovány \mathcal{H}_2 a \mathcal{H}_∞ normy pro signály a MIMO systémy. Dále je popsán způsob optimální regulace MIMO systému pomocí LQG, \mathcal{H}_2 a \mathcal{H}_∞ řízení, jejich obecné výhody, nevýhody a vzájemné podobnosti. Je implementován výpočet stavových \mathcal{H}_2 a \mathcal{H}_∞ regulátorů a rekonstruktorů. Pomocí \mathcal{H}_∞ kaskádní regulace, \mathcal{H}_2 optimálního LQG řízení a \mathcal{H}_∞ ekvivalentu LQG je vyřešena úloha inverzního kyvadla na vozíku. Výsledky jsou mezi sebou srovnány. Je přednesena teorie o možnosti návrhu identických \mathcal{H}_2 a \mathcal{H}_∞ stavových regulátorů.

KLÍČOVÁ SLOVA

\mathcal{H}_∞ , \mathcal{H}_2 , LQG, LMI, MIMO, MATLAB, Optimální regulace, Singulární hodnoty, Stavová regulace, Stavová rekonstrukce, \mathcal{H}_∞ stavová regulace, \mathcal{H}_∞ stavová rekonstrukce, \mathcal{H}_2 stavová regulace, \mathcal{H}_2 stavová rekonstrukce, \mathcal{H}_2 LQG, Inverzní kyvadlo na vozíku

ABSTRACT

This thesis deals with the MIMO system analysis, with a comparison of a different frequency characteristics, and including singular values characteristic. The \mathcal{H}_2 and \mathcal{H}_∞ norms are defined for signals and a MIMO systems. A MIMO optimal control using an LQG, an \mathcal{H}_2 and an \mathcal{H}_∞ control is defined and described. Their general advantages, disadvantages and similarities are summed up. An \mathcal{H}_2 and an \mathcal{H}_∞ state controller and state observer synthesis is implemented in a MATLAB using linear matrix inequalities. Control of an inverted pendulum is designed using the \mathcal{H}_∞ optimal cascade control, the \mathcal{H}_2 state control and the \mathcal{H}_∞ state control. The results are compared with each other. The theoretical possibility of same \mathcal{H}_2 and \mathcal{H}_∞ optimal state controller design is presented.

KEYWORDS

\mathcal{H}_∞ , \mathcal{H}_2 , LQG, LMI, MIMO, MATLAB, Optimal Control, Singular Values, State Control, State observer, \mathcal{H}_∞ State Control, \mathcal{H}_∞ State Observer, \mathcal{H}_2 State Control, \mathcal{H}_2 State Observer, \mathcal{H}_2 LQG, Inverted Pendulum on a Cart

FRIML, Dominik. *Porovnání H -nekonečno a LQG regulace*. Brno, 2020, 94 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce: Ing. Lukáš Pohl, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Porovnání H-nekonečno a LQG regulace“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno 1. 6. 2020

.....
podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Lukášovi Pohlovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Obsah

Úvod	13
1 Popis MIMO systému	14
1.1 Stavový popis	14
1.2 Matice operátorových přenosů jako popis systému	15
1.2.1 Matice operátorových přenosů ze stavového popisu systému	17
2 Směrovost systému	18
2.1 Vstupní a výstupní prostor	18
2.1.1 Příklady prostorů	18
2.2 Vstupní a výstupní signály jako vektory	19
2.2.1 Směr vstupního signálu MIMO systémů	20
2.3 Matice operátorových přenosů jako transformační matice	20
2.4 Jednotkový vstup	21
2.5 Směrovost systému	22
3 Frekvenční charakteristika MIMO systémů	24
3.1 Bodeho frekvenční charakteristika MIMO systémů	24
3.2 Frekvenční charakteristika vlastních čísel	25
3.3 Frekvenční charakteristika singulárních hodnot	27
4 Normy v teorii řízení	30
4.1 Normy signálu	30
4.1.1 \mathcal{H}_2 norma	30
4.1.2 \mathcal{H}_∞ norma	31
4.2 Normy systému	31
4.2.1 \mathcal{H}_2 norma	31
4.2.2 \mathcal{H}_2 norma pomocí singulárních hodnot	32
4.2.3 \mathcal{H}_∞ norma	32
5 Stavová regulace	34
5.1 Optimální stavová regulace	34
6 Rekonstrukce stavů	36
6.1 Optimální stavová rekonstrukce	37
7 LQG	38

8	\mathcal{H}_∞ a \mathcal{H}_2 regulace	39
8.1	\mathcal{H}_2 syntéza regulátoru	41
8.1.1	LQR pomocí \mathcal{H}_2 optimalizace	41
8.2	\mathcal{H}_∞ syntéza regulátoru	44
9	Implementace syntézy \mathcal{H}_2 a \mathcal{H}_∞ regulátoru	46
9.1	Suboptimální regulátor	46
9.2	\mathcal{H}_2 stavový regulátor pomocí LMI	46
9.3	\mathcal{H}_∞ stavový regulátor pomocí LMI	47
9.4	Implementace syntézy v prostředí MATLAB	48
9.4.1	Syntéza \mathcal{H}_2 regulátoru pomocí LMI v prostředí MATLAB	48
9.4.2	Syntéza \mathcal{H}_∞ regulátoru pomocí LMI v prostředí MATLAB	50
10	Regulovaná soustava	53
10.1	Model inverzního kyvadla na vozíku	53
10.2	Cíl úlohy	54
11	Řešení úlohy	56
11.1	\mathcal{H}_2 LQG řešení úlohy	56
11.2	\mathcal{H}_∞ řešení úlohy	58
11.3	\mathcal{H}_∞ LQG řešení úlohy	63
11.3.1	Návrh stavového \mathcal{H}_∞ optimálního regulátoru	63
11.3.2	Návrh stavového \mathcal{H}_∞ optimálního rekonstruktoru	66
11.3.3	\mathcal{H}_∞ LQG	68
12	Srovnání \mathcal{H}_2 a \mathcal{H}_∞ stavové regulace	69
	Závěr	71
	Literatura	73
	Seznam příloh	75
A	Obsah přiloženého CD	76
B	Odvození stavového popisu stavového rekonstruktoru	77
C	Další příklady výpočtu statického \mathcal{H}_2 optimálního regulátoru	78
D	Další příklady výpočtu statického \mathcal{H}_∞ optimálního regulátoru	80
E	Odvození matematického modelu inverzního kyvadla na vozíku	83

F	Návrh \mathcal{H}_∞ optimálního kaskádního regulátoru	88
G	Návrh \mathcal{H}_∞ optimálního stavového regulátoru	90
H	Návrh \mathcal{H}_∞ optimálního stavového rekonstruktoru	92
I	Převodění stavového rekonstruktoru na \mathcal{H}_∞ problém	94

Seznam obrázků

1.1	Grafické znázornění stavového popisu MIMO systému	15
1.2	Bodeho charakteristiky pro MIMO systém se dvěma vstupy a dvěma výstupy.	16
2.1	Reálný podprostor MIMO systému se dvěma vstupy a dvěma výstupy. Zobrazené vektory u_1, u_2, u_3, y_1, y_2 a y_3 jsou každý kombinací vstupů a výstupů. Jejich význam bude popsán níže.	18
2.2	Příklad signálu $\vec{u}(\omega)$ pro různé ω	20
2.3	Zobrazení $\mathbf{G}\vec{u}$ v Bodeho charakteristice pro první vstup a první výstup	21
2.4	Zobrazení vstupního a výstupního prostoru se zvýrazněným zesílením jako odpovědí na množinu jednotkových vstupů.	23
2.5	Kompaktní zobrazení zesílení systému pro všechny jednotkové vstupy u MIMO systému se dvěma vstupy a dvěma výstupy.	23
3.1	Bodeho frekvenční charakteristika pro MIMO systém se dvěma vstupy a dvěma výstupy.	24
3.2	Zobrazení zesílení systému pro $\omega_{selected}$ a bodů zobrazených v Bodeho frekvenční charakteristice.	25
3.3	Frekvenční charakteristika vlastních čísel matice operátorových přenosů	26
3.4	Vlastní čísla a vlastní vektory pro $\mathbf{G}(j\omega_{selected}) = \mathbf{G}$	26
3.5	Singulární rozklad \mathbf{G} pro konkrétní ω	28
3.6	Hodnoty singulárního rozkladu \mathbf{G} pro všechny ω	28
3.7	Porovnání všech frekvenčních charakteristik pro MIMO systém 18. řádu se dvěma vstupy a dvěma výstupy	29
3.8	Porovnání všech frekvenčních charakteristik pro konkrétní ω	29
5.1	Schematické zapojení stavového regulátoru	34
6.1	Blokové schéma stavového rekonstruktoru	36
7.1	Schematické zapojení LQG	38
8.1	Obecný regulační problém	39
8.2	Obecný regulační problém, detail	40
8.3	LQR definováno jako \mathcal{H}_2 problém	42
8.4	LQR definováno jako \mathcal{H}_2 problém s jednou váhou	43
8.5	Schéma obecného \mathcal{H}_∞ problému	45
9.1	Porovnání singulárních hodnot systémů s regulátorem navrženým příkazem <code>lqr</code> a vypočítaným pomocí LMI	49
9.2	Porovnání singulárních hodnot systémů s regulátorem navrženým třemi různými způsoby	52
10.1	Schematický náčrt modelu inverzního kyvadla	53
10.2	Schéma požadovaného řešení úlohy	54

11.1	Výsledek simulace navrženého \mathcal{H}_2 LQG regulátoru	57
11.2	Schematické zobrazení kaskádního Hinf optimálního regulátoru	58
11.3	Singulární charakteristiky signálů S a KS v porovnání s vahami $W_{S_\theta}(s)$ a $W_{KS}(s)$ při návrhu regulátoru odchytky \mathbf{K}_θ	59
11.4	Charakteristiky singulárních hodnot signálů S a KS v porovnání s va- hami $W_{S_x}(s)$ a $W_{KS}(s)$ při návrhu regulátoru polohy \mathbf{K}_x	61
11.5	Srovnání \mathcal{H}_2 optimálního LQG regulátoru (modře) a \mathcal{H}_∞ optimálního kaskádního regulátoru (červeně)	62
11.6	Schematické znázornění návrhu stavového regulátoru pomocí \mathcal{H}_∞ normy	63
11.7	Charakteristiky singulárních hodnot signálů S a KS v porovnání s příslušnými vahami a charakteristikami singulárních hodnot ekvi- valentních signálů \mathcal{H}_2 optimálního LQR.	65
11.8	Schematické znázornění návrhu stavového rekonstruktoru pomocí \mathcal{H}_∞ normy	66
11.9	Charakteristiky singulárních hodnot signálů S a KS v porovnání s pří- slušnými vahami a charakteristikami singulárních hodnot ekvivalent- ních signálů \mathcal{H}_2 optimálního stavového rekonstruktoru.	67
11.10	Srovnání \mathcal{H}_2 optimálního LQG regulátoru (modře) a \mathcal{H}_∞ optimálního LQG regulátoru (červeně)	68

Seznam tabulek

11.1	Tabulka výsledků simulace \mathcal{H}_{2LQG} regulátoru	57
11.2	Tabulka výsledků simulace kaskádního \mathcal{H}_{∞} optimálního regulátoru v porovnání s \mathcal{H}_{2LQG}	62

Seznam výpisů

6.1	Příklad návrhu stavového regulátoru \mathbf{K} a rekonstruktoru \mathbf{L} stejným příkazem v prostředí MATLAB	37
6.2	Příklad návrhu stavového optimálního regulátoru \mathbf{K} a rekonstruktoru \mathbf{L} stejným příkazem v prostředí MATLAB	37
9.1	Příklad výpočtu LQR regulátoru a ekvivalentního \mathcal{H}_2 stavového regulátoru v prostředí MATLAB.	48
9.2	Výsledky zobrazené po spuštění kódu 9.1	49
9.3	Příklad výpočtu statického \mathcal{H}_∞ optimálního regulátoru pomocí řešení LMI problému a příkazů <code>hinffi</code> a <code>hinfstruct</code>	50
9.4	Výsledky zobrazené po spuštění kódu 9.3	51
11.1	Návrh stavového regulátoru a rekonstruktoru	56
C.1	MATLAB skript s příklady výpočtu statického \mathcal{H}_2 optimálního regulátoru různými metodami	78
D.1	MATLAB skript s příklady výpočtu statického \mathcal{H}_∞ optimálního regulátoru různými metodami	80
F.1	MATLAB skript pro návrh \mathcal{H}_∞ optimálního kaskádního regulátoru	88
G.1	MATLAB skript pro návrh \mathcal{H}_∞ optimálního stavového regulátoru	90
H.1	MATLAB skript pro návrh \mathcal{H}_∞ optimálního stavového rekonstruktoru	92

Úvod

Regulace systémů s více vstupy a více výstupy je velmi známý problém, který je často řešen pomocí diagonalizace systému a následného SISO řízení. V této práci budou popsány přístupy k regulaci, kterými je možné se diagonalizací vyhnout.

Bude popsán způsob analýzy a popisu MIMO systému tak, aby bylo možné pracovat, navrhovat a chápat MIMO regulátory, ať už stavové, \mathcal{H}_2 , \mathcal{H}_∞ nebo jiné.

Cílem práce je navrhnout optimální zapojení stavového regulátoru a stavového rekonstruktoru zvané LQG (Linear Quadratic Gaussian) a srovnat je s \mathcal{H}_∞ regulací. Ukáže se, že se jedná o poměrně složitou úlohu, protože oba dva způsoby přístupu k návrhu i k následné regulaci jsou velmi odlišné.

Naštěstí je ale dokázáno, že návrh LQG je speciálním případem syntézy regulátorů pomocí minimalizace \mathcal{H}_2 normy. Mezi \mathcal{H}_2 a \mathcal{H}_∞ syntézou již takové rozdíly nejsou, což umožní srovnání dvou návrhů, jednoho \mathcal{H}_2 optimálního a jednoho \mathcal{H}_∞ optimálního.

Samotná \mathcal{H}_2 a \mathcal{H}_∞ syntéza bude také implementována v prostředí MATLAB. Způsobů syntézy je více. V této práci je implementována syntéza pomocí lineárních maticových nerovností.

1 Popis MIMO systému

V následujících kapitolách budou popsány způsoby použití lineární algebry pro popis MIMO systémů.

1.1 Stavový popis

MIMO systém, stejně jako SISO systém, je možné popsat stavovým popisem 1.1.

$$\begin{aligned}
 \dot{x}_1 &= a_{11}x_1 + \dots + a_{1n}x_n + b_{11}u_1 + \dots b_{1m}u_m \\
 \dot{x}_2 &= a_{21}x_2 + \dots + a_{2n}x_n + b_{21}u_2 + \dots b_{2m}u_m \\
 &\vdots \\
 \dot{x}_n &= a_{n1}x_2 + \dots + a_{nn}x_n + b_{n1}u_2 + \dots b_{nm}u_m \\
 \\
 y_1 &= c_{11}x_1 + \dots + a_{1n}x_n + d_{11}u_1 + \dots b_{1m}u_m \\
 y_2 &= c_{21}x_2 + \dots + a_{2n}x_n + d_{21}u_2 + \dots b_{2m}u_m \\
 &\vdots \\
 y_n &= c_{k1}x_2 + \dots + a_{kn}x_n + d_{k1}u_2 + \dots b_{km}u_m,
 \end{aligned} \tag{1.1}$$

kde u je vstup, m je počet vstupů a b_{nm} je prvek matice \mathbf{B} , tedy prvek určující sílu vazby mezi vstupem u_m a stavem \dot{x}_n . x je vnitřní stav systému, n je počet vnitřních stavů systému a a_{nn} je prvek matice \mathbf{A} , tedy prvek určující vazbu mezi stavem x_n a jeho derivací podle času \dot{x}_n . d_{km} je prvek matice \mathbf{D} , tedy prvek určující vazbu ze vstupu u_m na výstup y_k . Konečně c_{kn} je prvek matice \mathbf{C} a určuje vazbu mezi stavem x_n a výstupem y_k .

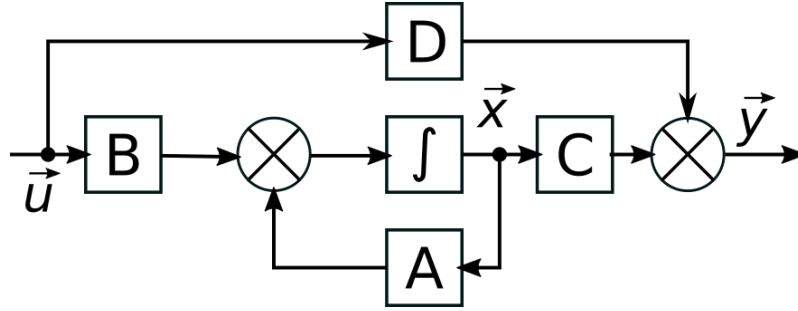
$$\begin{aligned}
 \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix} &= \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \\ a_{n1} & & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_{11} & \dots & b_{1m} \\ \vdots & \ddots & \\ b_{n1} & & b_{nm} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{bmatrix} \\
 \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix} &= \begin{bmatrix} c_{11} & \dots & c_{1n} \\ \vdots & \ddots & \\ a_{k1} & & c_{kn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} d_{11} & \dots & d_{1m} \\ \vdots & \ddots & \\ d_{k1} & & d_{km} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{bmatrix}
 \end{aligned} \tag{1.2}$$

Tento popis je sice vnitřní, tedy popisující systém úplně, ale práce s tímto popisem je často nepraktická. Proto se také používá jeho maticová forma 1.2, nebo

zkráceně 1.3.

$$\begin{aligned}\dot{\vec{x}} &= \mathbf{A}\vec{x} + \mathbf{B}\vec{u} \\ \vec{y} &= \mathbf{C}\vec{x} + \mathbf{D}\vec{u},\end{aligned}\tag{1.3}$$

Kde $\dot{\vec{x}}$ je sloupcový vektor derivací stavů, \vec{x} je sloupcový vektor stavů, \vec{u} je sloupcový vektor vstupů neboli vstup MIMO systému, a \vec{y} je sloupcový vektor výstupů, tedy výstup MIMO systému.¹



Obr. 1.1: Grafické znázornění stavového popisu MIMO systému

Při bližší studii stavového popisu MIMO systému vyjde najevo, že se jedná o dvě soustavy dvou rovnic. Proto je možné popis zjednodušit na jednu soustavu rovnic, pokud je znám počet vstupů m a počet výstupů k . Tímto způsobem vznikne kompaktní stavový popis.

$$\begin{bmatrix} \dot{\vec{x}} \\ \vec{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{u} \end{bmatrix} = \vec{y}_C = \mathbf{A}_C \vec{u}_C,\tag{1.4}$$

kde index C značí kompaktní vyjádření (z anglického compact).

1.2 Matice operátorových přenosů jako popis systému

Ve většině situací není potřeba detailně znát a pozorovat vnitřní stavy systému, jindy to jednoduše nebývá možné. Z toho důvodu se u SISO systémů často používá podíl Laplaceových obrazů výstupní a vstupní veličiny, tedy operátorový přenos.

$$G(s) = \frac{Y(s)}{U(s)}\tag{1.5}$$

¹V následujících kapitolách se bude předpokládat, že vektor je vždy sloupcový. Řádkový vektor bude naznačen pomocí transpozice \vec{x}^T .

Toto u MIMO, SIMO a MISO systémů nemusí být možné, protože počet vstupů ani počet výstupů nemusí být roven jedné. Proto se používá matice operátorových přenosů mezi jednotlivými vstupy a výstupy.

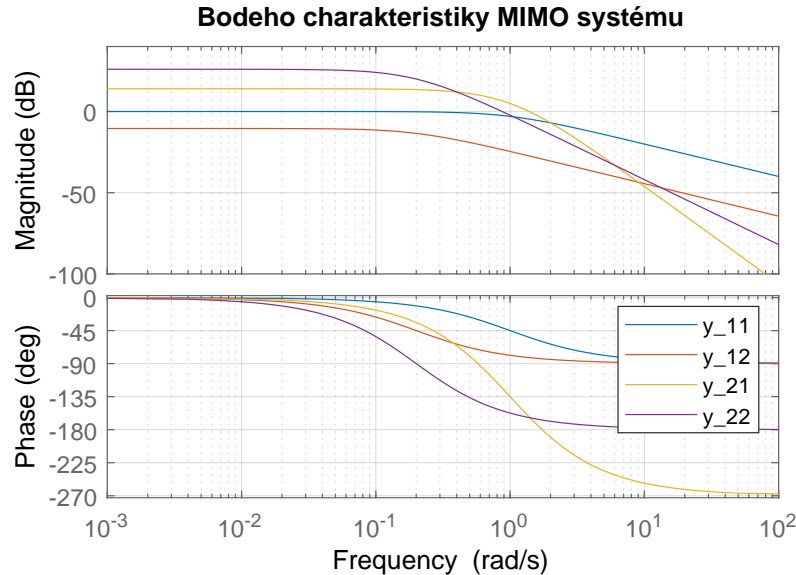
$$\mathbf{G}(s) = \begin{bmatrix} \mathbf{G}(s)_{11} = \frac{Y(s)_1}{U(s)_1} & \cdots & \mathbf{G}(s)_{1m} = \frac{Y(s)_1}{U(s)_m} \\ \vdots & \ddots & \vdots \\ \mathbf{G}(s)_{k1} = \frac{Y(s)_k}{U(s)_1} & \cdots & \mathbf{G}(s)_{km} = \frac{Y(s)_k}{U(s)_m} \end{bmatrix}, \quad (1.6)$$

kde $\mathbf{G}(s)_{km}$ je operátorový přenos z m -tého vstupu na k -tý výstup, $Y(s)_k$ je k -tý výstup a $U(s)_m$ je m -tý vstup.

Dosadíme-li za Laplaceův operátor $s = j\omega$, kde j je imaginární jednotka a ω je úhlová frekvence, dostaneme matici operátorových přenosů vhodnou pro vykreslení modulových a fázových frekvenčních charakteristik.

$$\mathbf{G}(s) \Big|_{s=j\omega} = \mathbf{G}_c(j\omega) = \frac{Y(j\omega)_c}{U(j\omega)_c}; \quad \mathbf{G}_c(j\omega) \in \mathbb{C} \quad (1.7)$$

V případě MIMO systémů se charakteristiky vykreslují pro každý přenos, tedy pro každý prvek matice $\mathbf{G}(s)$ zvlášť, jak je vidět na obrázku 1.2. Vznikne tedy tolik frekvenčních charakteristik, kolik má matice $\mathbf{G}(s)$ prvků, tedy $m \times k$ prvků. Výsledné charakteristiky je možné interpretovat jako charakteristiky konkrétního vstupu na konkrétní výstup za předpokladu, že ostatní vstupy jsou nulové.



Obr. 1.2: Bodeho charakteristiky pro MIMO systém se dvěma vstupy a dvěma výstupy.

1.2.1 Matice operátorových přenosů ze stavového popisu systému

Přenosovou funkci systému je možné získat z jeho stavového popisu. Způsob, jakým je to možné, lze odvodit z obrázku 1.1.

Pro SISO systém 1. řádu je možné přenosovou funkci $\mathbf{G}(s)$ vyjádřit z obrázku 1.1 pomocí blokové algebry jako

$$G(s) = C \frac{B}{s - A} + D, \quad (1.8)$$

kde s je Laplaceův operátor.

Rovnice 1.8 platí pouze pro SISO systémy prvního řádu, protože jedině v tomto případě jsou proměnné A , B , C a D skaláry. V případě MIMO, SIMO, MISO systému, nebo i SISO systému vyššího řádu než prvního, jsou proměnné \mathbf{A} , \mathbf{B} , \mathbf{C} a \mathbf{D} matice, proto není možné použít zlomek. Z toho důvodu se používá rovnice

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I}_m - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}; \quad \mathbf{I}_m = \text{diag}\{1, 1, \dots, 1\}, \quad (1.9)$$

kde \mathbf{I}_m je jednotková matice $m \times m$ a kde m je počet stavů systému.

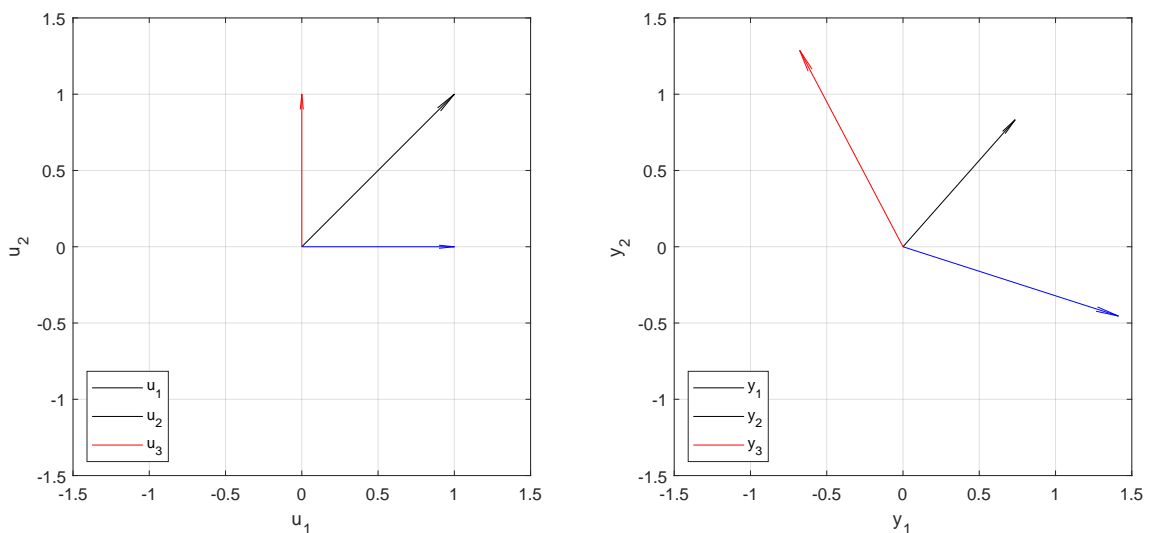
2 Směrnost systému

Jak bylo vysvětleno v předchozích kapitolách, matice operátorových přenosů $G(s)$ je vstupně-výstupní popis systému. Obecně má systém m vstupů a k výstupů, orientovaných do vektorů \vec{u} a \vec{y} . Jelikož se jedná o vektory, musí mít nejen velikost, ale i směr. Právě směr je důvod, proč je analýza MIMO systémů mnohem složitější než u SISO systémů, kde vstup i výstup je skalár.

2.1 Vstupní a výstupní prostor

V teorii řízení se můžeme bavit o prostoru stavovém, vstupním a výstupním. Stavový, vstupní nebo výstupní prostor umožňuje jednoznačný matematický popis zobrazující stav, kombinaci vstupů nebo kombinaci výstupů.

Zpravidla se jedná o komplexní ortonormální prostor, kde počet dimenzí odpovídá počtu stavů, vstupů nebo výstupů.¹



Obr. 2.1: Reálný podprostor MIMO systému se dvěma vstupy a dvěma výstupy. Zobrazené vektory u_1 , u_2 , u_3 , y_1 , y_2 a y_3 jsou každý kombinací vstupů a výstupů. Jejich význam bude popsán níže.

2.1.1 Příklady prostorů

Prostor může být intuitivní, tedy odpovídat reálné situaci, ale nemusí. V případě, kdy například dimenze vstupního prostoru přímo odpovídají dimenzím reálného prostoru, je jednoduché interpretovat vektory zobrazené v tomto prostoru.

¹Z důvodu složitosti zobrazení komplexního prostoru vyšší dimenze, bude dále vždy zobrazován pouze reálný podprostor.

Například popisujeme-li vozík, který se může pohybovat ve 2D prostoru, můžeme zvolit jako první vstup sílu působící na vozík ve směru osy x a jako druhý vstup sílu působící na vozík ve směru osy y . Vektor vstupů $\vec{u} = [1 \ 2]^T$ potom intuitivně vyjadřuje působení na vozík silou 1N ve směru osy x a silou 2N ve směru osy y .

Pokud jsou zároveň výstupními veličinami souřadnice vozíku ve 2D prostoru, lze opět odhadnout, jak se budou souřadnice vozíku, tedy výstupní vektor \vec{y} , měnit v čase.

Tento příklad je velice jednoduchý a intuitivní, zejména proto, že systém má vždy vazbu pouze z prvního vstupu na první výstup a z druhého vstupu na druhý výstup. Zároveň má velice dobře zvolené vstupní a výstupní proměnné tak, aby vznikl jednoduchý a přehledný vstupní a výstupní prostor. Stačilo by ale zavést další vstup nebo výstup (například úhel natočení vozíku a jeho rychlost) a už by vznikl nepřehledný čtyřdimenzionální prostor, který by bylo mnohem složitější interpretovat.

Pro jednodušší interpretaci je možné prostor rozdělit na několik podprostorů a zobrazovat každý zvlášť. Zde ale hrozí zanedbání směrovosti systému, protože každý podprostor o ní nese jen částečnou informaci.

2.2 Vstupní a výstupní signály jako vektory

Vstupy a výstupy MIMO systémů jsou vyjadřovány jako vektory popisující kombinaci jednotlivých vstupů systému, tedy kombinaci bázových vektorů vstupního a výstupního prostoru.

Přestože se většinou v teorii řízení bavíme o vstupu jako o funkci času,

$$u(t); \quad t \in \langle 0; \infty \rangle \quad (2.1)$$

budeme se zde bavit o vstupu zpravidla jako o působení sinusového signálu konkrétní úhlové frekvence ω

$$\vec{u}(\omega) = [u_1(\omega) \ u_2(\omega) \ \dots \ u_m(\omega)]^T \quad (2.2)$$

Dosažením žádané úhlové frekvence dostaneme reálné číslo vyjadřující velikost amplitudy sinusoidy.

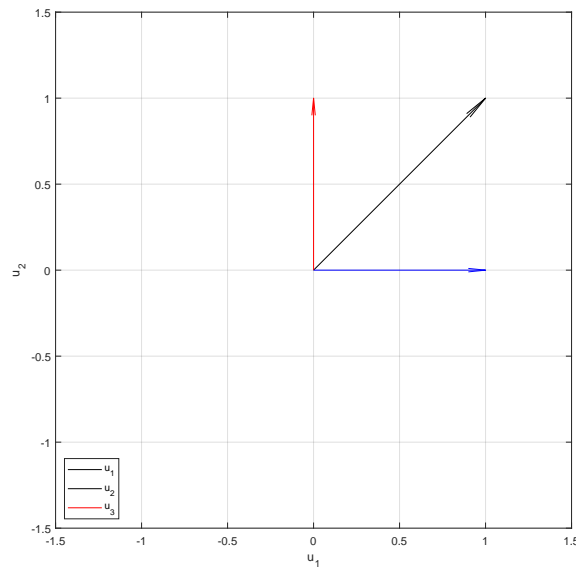
$$\vec{u}(\omega) \Big|_{\omega=\omega_u} = [u_1(\omega_u) \ u_2(\omega_u) \ \dots \ u_m(\omega_u)]^T \equiv [|u_1| \cdot \sin(\omega_u) \ |u_2| \cdot \sin(\omega_u) \ \dots \ |u_m| \cdot \sin(\omega_u)]^T \quad (2.3)$$

2.2.1 Směr vstupního signálu MIMO systémů

Všimněme si nyní, že vstupní signál je vektor reálných čísel pro každé ω .

$$\vec{u}(\omega) \Big|_{\omega=\omega_u} = [u_1(\omega_u) \ u_2(\omega_u) \ \dots \ u_m(\omega_u)]^T \quad (2.4)$$

Tento vektor má v prostoru (vstupním prostoru pro vstupní signál, výstupním prostoru pro výstupní signál a stavovém prostoru pro stavový signál) určitou velikost a určitý směr.



Obr. 2.2: Příklad signálu $\vec{u}(\omega)$ pro různé ω

2.3 Matice operátorových přenosů jako transformační matice

Rovnice $\mathbf{A}\vec{x} = \vec{y}$ vyjadřuje vektor \vec{y} jako vektor \vec{x} transformovaný transformační maticí \mathbf{A} . V teorii řízení můžeme tuto rovnici používat následovně

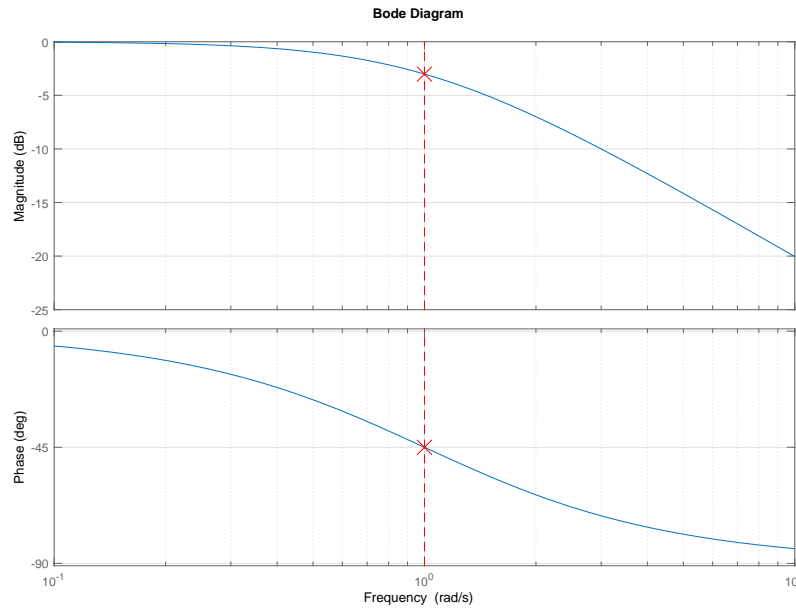
$$\mathbf{A}\vec{x} = \vec{y} \quad \rightarrow \quad \mathbf{G}_c(j\omega)\vec{u}(\omega) = \vec{y}_c(j\omega) \quad (2.5)$$

Rovnici 2.5 můžeme v teorii řízení interpretovat tak, že výstupní vektor $\vec{y}_c(j\omega)$ pro konkrétní ω odpovídá vstupnímu vektoru $\vec{u}(j\omega)$, pro stejné ω transformovanému maticí operátorových přenosů $\mathbf{G}_c(j\omega)$

Pro konkrétní ω je tedy možné rovnici 2.5 zapsat ve zjednodušeném tvaru

$$\mathbf{G}_c \vec{u} = \vec{y}_c \quad ; \omega = \omega_{selected} \quad (2.6)$$

Všimněme si, že jelikož platí $\mathbf{G}_c \in \mathbb{C}$, platí také $\vec{y}_c \in \mathbb{C}$. Je tomu tak proto, že $|\frac{\vec{y}}{\vec{x}}|$ je velikost zesílení pro dané ω , a $\arg\{\frac{\vec{y}}{\vec{x}}\}$ je fázový posun pro dané ω . Vykreslením zesílení a fázových posunů pro všechny ω do logaritmických měřítek dostaneme známou Bodeho frekvenční charakteristiku.



Obr. 2.3: Zobrazení $\mathbf{G}\vec{u}$ v Bodeho charakteristice pro první vstup a první výstup MIMO systému

2.4 Jednotkový vstup

Jednotkový vstup se používá například pro zjištění zesílení systému, protože zesílení systému je u SISO systémů definováno jako

$$k = \left| \frac{G(s)u}{u} \right| \quad ; s \rightarrow 0, \quad (2.7)$$

kde k je zesílení systému. Z rovnice 2.7 je jasně vidět, že pokud $u = 1$, tedy vstup je jednotkový, výraz se zjednoduší na

$$k = |G(s)| \quad ; s \rightarrow 0 ; |u| = 1 \quad (2.8)$$

Zesílení pro konkrétní ω je potom definováno jako

$$k = |G| \quad ; \quad u = 1, \quad (2.9)$$

tedy jako velikost výstupu při jednotkovém vstupu.

U MIMO systémů není možné zvolit $\vec{u} = 1$, protože vstup není skalár. Zvolit $\vec{u} = [1, 1, \dots, 1]$ by bylo pro demonstraci nedostatečné, protože tento vstup nedemonstruje vliv různých kombinací vstupů.

Je-li potřeba demonstrovat vliv více kombinací vstupů, je nutné zvolit více vstupních vektorů. Vzniká tak množina vstupních vektorů, pomocí kterých je možné vizualizovat vlastnosti MIMO systémů, jako je například směrovost systému.

Proto je zavedena množina jednotkových vstupů podle rovnice 2.10.

$$1 = \sqrt{u_1(\omega)^2 + u_2(\omega)^2 + \dots + u_m(\omega)^2}; \quad \forall \omega \quad (2.10)$$

V euklidovském prostoru tedy vznikl, v závislosti na počtu vstupů, bod², kruh, sféra nebo hypersféra vymezující množinu jednotkových vstupů, tedy vstupů jejichž velikost je rovna jedné, jak je vidět z rovnice 2.11.

$$|\vec{u}(\omega)| = \left| \sqrt{u_1(\omega)^2 + u_2(\omega)^2 + \dots + u_m(\omega)^2} \right| = 1; \quad \forall \omega \quad (2.11)$$

2.5 Směrovost systému

Nyní máme dostatek informací k tomu, abychom objevili význam směrovosti systému. Jedná se o fenomén, který přímo plyne z 2.12.

$$\begin{aligned} \vec{u}_1 &\neq \vec{u}_2 \\ |\vec{u}_1| &= |\vec{u}_2| \\ |G\vec{u}_1| &\neq |G\vec{u}_2|, \end{aligned} \quad (2.12)$$

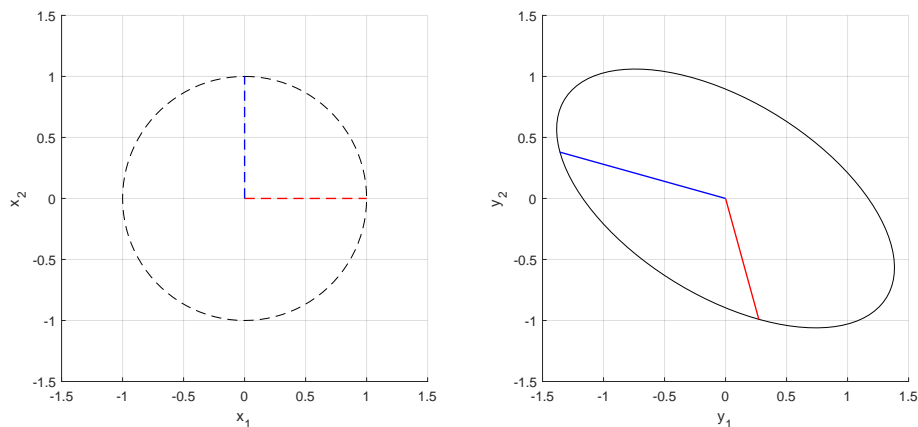
kde \vec{u}_1 a \vec{u}_2 jsou různé kombinace vstupů, tedy různé vstupní vektory.

Slovy by se rovnice 2.12 dala interpretovat jako tvrzení, že velikost vstupu, za znalosti matice přenosových funkcí, samotná nestačí pro určení velikosti výstupu. Je potřeba znát i směr vstupu.

Jinými slovy, na směru vstupu nezávisí pouze směr výstupu, ale i jeho velikost.

Toto je přímo viditelné v obrázku 2.4, který vznikl tak, že se postupně dosazovaly všechny jednotkové vstupy z jednotkové množiny vstupů (\vec{u} , pro které platí vztah 2.10), tedy vstupy s velikostí rovnou 1.

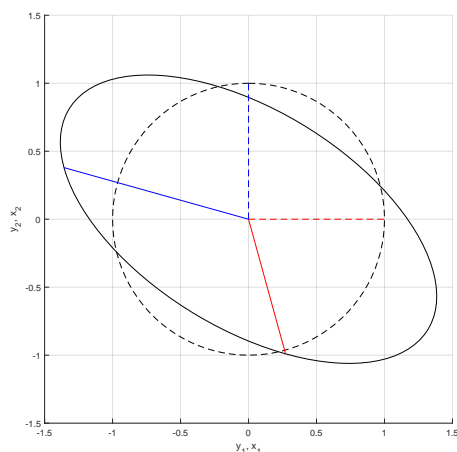
²Pro SISO a SIMO systémy tak vzniká jednotkový vstup $\vec{u} = 1$.



Obr. 2.4: Zobrazení vstupního a výstupního prostoru se zvýrazněným zesílením jako odpovědí na množinu jednotkových vstupů.

Pro demonstraci byly zvýrazněny dva vektory ve vstupním prostoru a odpovídající dva ve výstupním prostoru. Je vidět, že se změnil jak jejich směr, tak i velikost.

Vektory na obrázku 2.4 se dají zobrazit kompaktněji sloučením vstupního a výstupního prostoru. Vznikne obrázek 2.5.



Obr. 2.5: Kompaktní zobrazení zesílení systému pro všechny jednotkové vstupy u MIMO systému se dvěma vstupy a dvěma výstupy.

Na obrázcích 2.4 a 2.5 je jasně vidět, že velikost výstupu se mění pro různé jednotkové vstupy, takže zesílení MIMO systémů není konstantní.

Směrovost systému je tedy vlastnost MIMO systémů, která způsobuje, že zesílení MIMO systému není konstantní, ale je závislé na směru vstupu. Později bude ukázáno, že tento interval je definován pomocí 2.13.

$$k \in \langle \sigma_{min}; \sigma_{max} \rangle \quad (2.13)$$

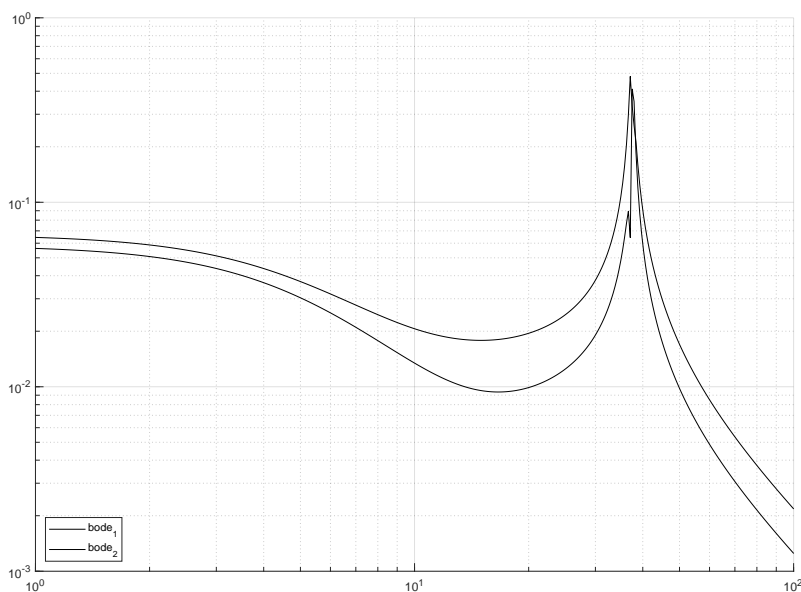
3 Frekvenční charakteristika MIMO systémů

Rovnice bývají často nepřehledné a proto je vhodné zobrazit chování systému jiným způsobem. Často se používají Bodeho nebo Nyquistovy charakteristiky, tedy frekvenční charakteristika v logaritmických měřítkách nebo v komplexní rovině. Každé zobrazení má své výhody a nevýhody.

Jednu vlastnost mají ovšem společnou. Pro MIMO systémy nejsou vhodné, protože zobrazují vždy vztah mezi jedním vstupem a jedním výstupem. To by bylo u složitějších MIMO systémů velice nepřehledné, nehledě na skutečnost, že není brána v potaz směrnost systému. Z tohoto důvodu je potřeba najít jiné, vhodnější zobrazení.

3.1 Bodeho frekvenční charakteristika MIMO systémů

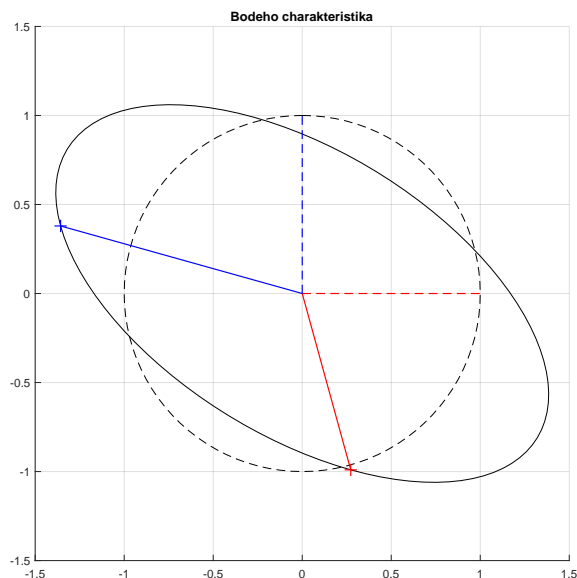
Bodeho frekvenční charakteristika zobrazuje frekvenční charakteristiku jednotlivých prvků matice operátorových přenosů. Vznikne tedy $m \times k$ frekvenčních charakteristik.



Obr. 3.1: Bodeho frekvenční charakteristika pro MIMO systém se dvěma vstupy a dvěma výstupy.

Každá charakteristika udává velikost jednoho výstupu, jako odpověď na konkrétní vstup roven jedné a ostatní rovny nule.

Pro konkrétní ω si můžeme zobrazit zesílení signálu a body zobrazené v obrázku 3.1.



Obr. 3.2: Zobrazení zesílení systému pro $\omega_{selected}$ a bodů zobrazených v Bodeho frekvenční charakteristice.

Jak je vidět z obrázků 3.1 a 3.2, Bodeho frekvenční charakteristika je pro MIMO systémy nevhodná hned ze dvou důvodů. Za prvé nerepresentují směrovost systému a za druhé vznikne nepřehledné množství frekvenčních charakteristik. Druhý lze vyřešit pomocí frekvenční charakteristiky vlastních čísel matice operátorových přenosů.

3.2 Frekvenční charakteristika vlastních čísel

Problém s nepřehledným počtem charakteristik, by bylo možné vyřešit pomocí následující myšlenky: Nalezneme-li jednotkový vstupní vektor, pro který platí, že směr vstupu a výstupu je stejný, stačí zobrazit pouze jednu charakteristiku. Později bude dokázáno, že tuto podmínku splňuje více různých vstupních vektorů a je potřeba vykreslit charakteristiku pro každý z nich.

Lineární algebra [1] popisuje matici, pro kterou platí následující rovnice, kterou můžeme aplikovat v teorii řízení. Tuto matici nazýváme maticí vlastních čísel λ .

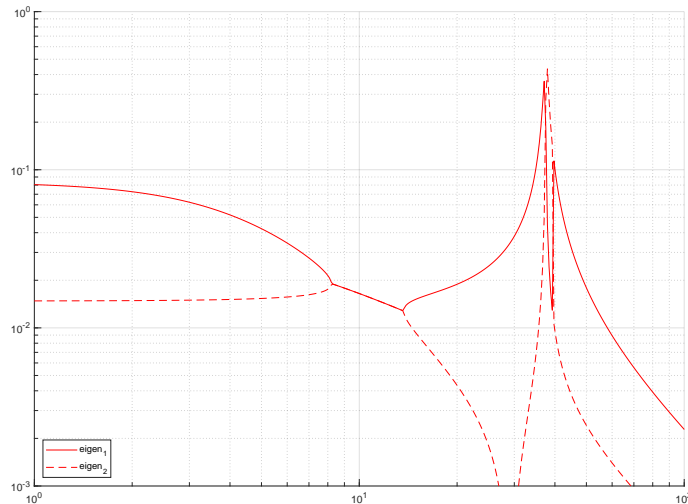
$$\mathbf{A}\vec{x} = \lambda\vec{x} \quad \rightarrow \quad \mathbf{G}\vec{u} = \lambda\vec{u} = \vec{y} \quad (3.1)$$

$$\mathbf{G}(j\omega)\vec{u}(\omega) = \lambda(j\omega)\vec{u}(\omega) = \vec{y}(j\omega), \quad (3.2)$$

kde \mathbf{A} je obecná matice, \vec{x} je obecný vektor a λ je matice vlastních čísel $\lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_{k=m}\}$.

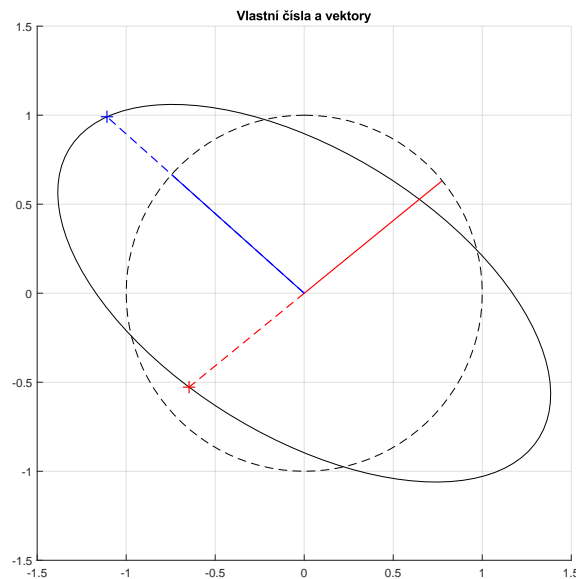
Z definice matice vlastních čísel plyne, že frekvenční charakteristiku vlastních čísel je možné získat pouze pro invertovatelné matice operátorových přenosů, tedy pro systémy, které mají stejný počet vstupů a výstupů.

Problém nepřehledného množství charakteristik, který vznikal v případě Bodeho frekvenční charakteristiky, je vyřešen. Jak ukazuje rovnice 3.1, počet vzniklých charakteristik bude menší nebo rovna $m = k$. Menší než počet vstupů bude tehdy, budou-li vlastní čísla komplexně sdružená nebo stejná. V jiném případě bude počet vlastních čísel roven počtu vstupů, potažmo výstupů systému.



Obr. 3.3: Frekvenční charakteristika vlastních čísel matice operátorových přenosů

Podívejme se nyní, co získáme vykreslením frekvenční charakteristiky vlastních čísel matice operátorových přenosů pro konkrétní ω .



Obr. 3.4: Vlastní čísla a vlastní vektory pro $\mathbf{G}(j\omega_{selected}) = \mathbf{G}$

Z obrázku 3.4 je vidět, že se opravdu povedlo najít takový vstupní vektor, jehož směr se transformací maticí \mathbf{G} nezmění. Je-li jeho vstupní velikost rovna jedné, jeho výstupní velikost je rovna λ .

Bohužel, jak je také vidět z obrázku 3.4, je možné najít takový vstupní vektor, pro který je velikost výstupního vektoru větší než velikost výstupního vektoru charakteristiky vlastních čísel.

Frekvenční charakteristika vlastních čísel tedy řeší problém velkého počtu charakteristik, ale neřeší problém směrovosti systému.

3.3 Frekvenční charakteristika singulárních hodnot

Problém s počtem charakteristik i problém se směrovostí systému je možné vyřešit charakteristikou, která pro každé ω zobrazí minimální a maximální hodnotu zesílení, tedy hodnoty

$$\max \{ \mathbf{G}(j\omega)\vec{u} \} \quad \forall \omega \quad ; \quad \|\vec{u}\|_2 = 1 \quad (3.3)$$

$$\min \{ \mathbf{G}(j\omega)\vec{u} \} \quad \forall \omega \quad ; \quad \|\vec{u}\|_2 = 1 \quad (3.4)$$

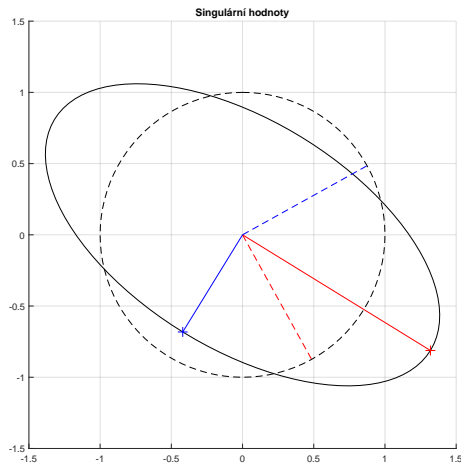
Tuto charakteristiku je možné získat pomocí singulárních hodnot matice [1] operátorových přenosů pro každé ω .

$$\mathbf{G} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \quad (3.5)$$

$$\mathbf{G} = \begin{bmatrix} \vec{u}_1 & \vec{u}_2 & \dots & \vec{u}_m \end{bmatrix} \begin{bmatrix} \sigma_1 & & & 0 \\ & \sigma_2 & & \\ & & \ddots & \\ 0 & & & \sigma_m \end{bmatrix} \begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \dots & \vec{v}_m \end{bmatrix}^T, \quad (3.6)$$

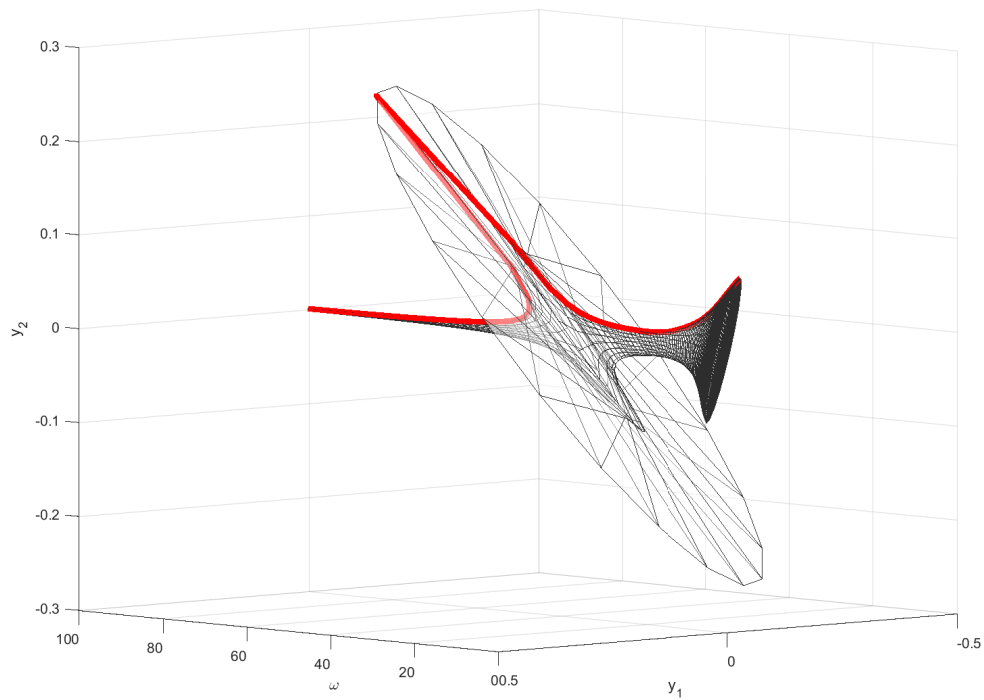
kde $\mathbf{\Sigma}$ je diagonální matice singulárních hodnot seřazených podle velikosti. Tedy $\max_{ij} \{ \mathbf{\Sigma} \} = \sigma_1 = \bar{\sigma}$ a $\min_{ij} \{ \mathbf{\Sigma} \} = \sigma_m = \underline{\sigma}$. \vec{v}_1 je vstupní vektor, pro který je zesílení maximální, a \vec{v}_m je vstupní vektor, pro který je zesílení minimální. \vec{u}_1 je výstupní vektor s maximální velikostí a \vec{u}_m je výstupní vektor s minimální velikostí.

Pomocí singulárního rozkladu je tedy možné získat hodnoty minimálního a maximálního zesílení v podobě hodnot $\bar{\sigma}$ a $\underline{\sigma}$.



Obr. 3.5: Singulární rozklad \mathbf{G} pro konkrétní ω

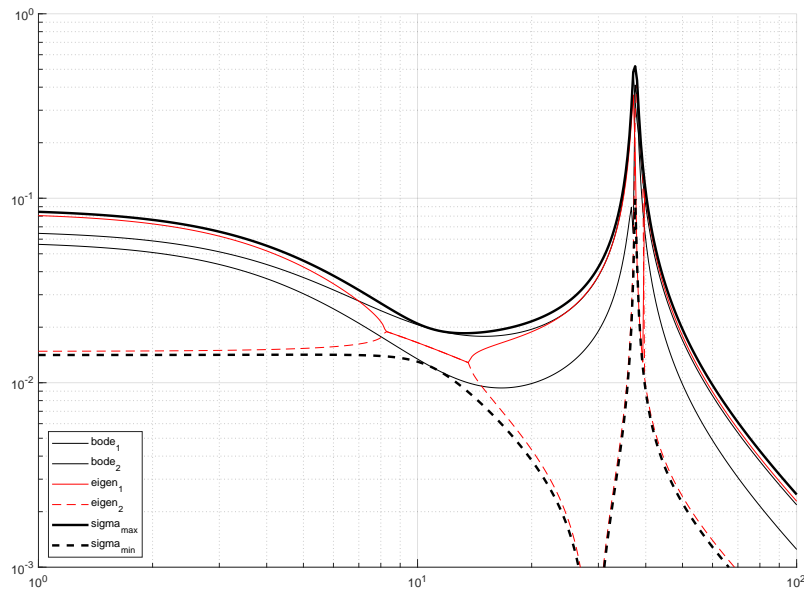
Vykreslením frekvenční charakteristiky singulárních hodnot tedy získáme, mimo jiné, přehled o maximálním a minimálním zesílení systému pro každé ω .¹



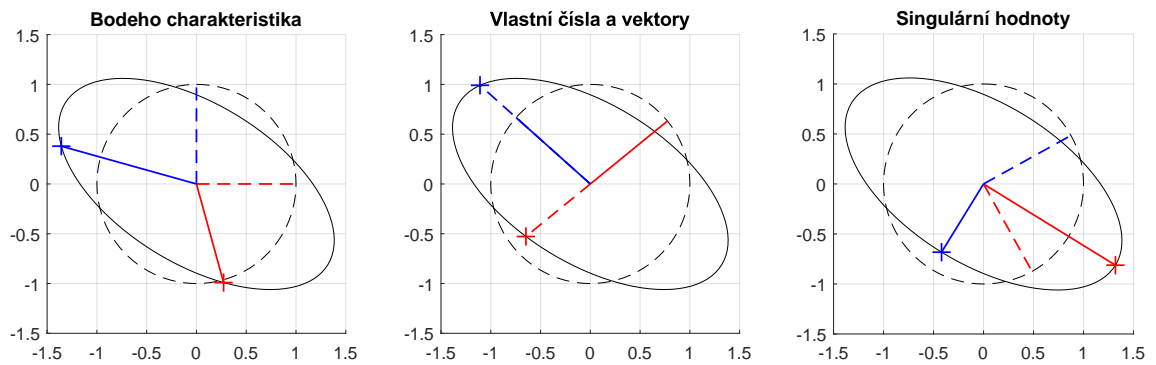
Obr. 3.6: Hodnoty singulárního rozkladu \mathbf{G} pro všechny ω

¹Často se do frekvenční charakteristiky singulárních hodnot vykreslují také ostatní singulární hodnoty, mimo minimální a maximální. Tyto ostatní charakteristiky ale nemají větší význam, než frekvenční charakteristiky vlastních čísel nebo Bodeho frekvenční charakteristiky. Je tedy možné tyto průběhy ignorovat a soustředit se pouze na minimum a maximum.

Získáváme tedy interval, ve kterém se může zesílení systému měnit v závislosti na směru vstupu. Toto je vhodné zejména pro určování stability systému, protože jsme schopni najít maximální možný výstup pro množinu všech jedničkových vstupů.



Obr. 3.7: Porovnání všech frekvenčních charakteristik pro MIMO systém 18. řádu se dvěma vstupy a dvěma výstupy



Obr. 3.8: Porovnání všech frekvenčních charakteristik pro konkrétní ω

Na obrázku 3.7 je vidět, že všechny průběhy leží v intervalu omezeném maximální a minimální singulární charakteristikou. Proto je frekvenční charakteristika singulárních hodnot nejlépe vypovídající charakteristikou pro MIMO systémy.

4 Normy v teorii řízení

V teorii řízení je často praktické mít možnost vyjádřit nějakou vlastnost signálu nebo systému pouze jedním číslem. Je-li vlastnost vyjádřena pouze jedním číslem, je možné vzniklá čísla mezi sebou porovnávat, nebo například použít toto číslo jako minimalizační parametr. Pro tyto účely se často používají normy [1].

4.1 Normy signálu

V kapitole 2.2 bylo popsáno, že signál je možné vyjádřit jako funkci

$$\vec{u}(\omega) = [u_1(\omega) \ u_2(\omega) \ \dots \ u_m(\omega)]^T, \quad (4.1)$$

kde dosazením ω dostaneme vektor popisující amplitudy sinusových signálů, jejichž úhlová rychlost odpovídá dosazenému ω . Takový signál lze použít jako vstup MIMO systému, kde každý prvek vektoru odpovídá signálu na jednom vstupu. Takový vstupní vektor má určitý směr, daný jednotlivými prvky vektoru, a velikost definovanou rovnicí 4.2.

$$|\vec{u}| = \sqrt{u_1^2 + u_2^2 + \dots + u_m^2} \quad (4.2)$$

4.1.1 \mathcal{H}_2 norma

p norma je definována jako [1]:

$$\|\vec{x}\|_p = \left(\sum_i |x_i|^p \right)^{1/p} \quad (4.3)$$

Nyní dosazením 2 za p získáme \mathcal{H}_2 normu. Použitím \mathcal{H}_2 normy je dosaženo ekvivalence mezi rovnicemi 4.2 a 4.3.

$$|\vec{u}| = \sqrt{u_1^2 + u_2^2 + \dots + u_m^2} = \|\vec{u}\|_2 = \left(\sum_i |u_i|^2 \right)^{1/2} \quad (4.4)$$

\mathcal{H}_2 norma signálu tedy udává jeho velikost pro konkrétní ω .

Ze znalosti \mathcal{H}_2 normy signálu je možné definovat množinu jednotkových vstupů jako množinu všech vstupů splňujících $\|\vec{u}(\omega)\|_2 = 1; \forall \omega$.

Pokud se $\vec{u}(\omega)$ mění v závislosti na ω , můžeme odvodit, jak bude \mathcal{H}_2 norma vypadat.

$$\|\vec{u}(\omega)\|_2 = \sqrt{\sum_i \int_0^\infty |u_i(\omega)|^2 d\omega} \quad (4.5)$$

Výsledkem \mathcal{H}_2 normy signálu je průměrné zesílení přes všechny ω . Amplituda signálu se musí pro vysoké úhlové rychlosti blížit k nule, jinak bude norma rovna nekonečnu. To ale není překážkou, protože všechny reálné signály tento požadavek splňují.

\mathcal{H}_2 normu signálu je také možné definovat v časové oblasti [2, 3]

$$\|\vec{u}(\tau)\|_2 = \sqrt{\int_0^\infty \vec{u}^T(\tau)\vec{u}(\tau)d\tau} \quad (4.6)$$

4.1.2 \mathcal{H}_∞ norma

Další normou používanou pro signály je \mathcal{H}_∞ norma signálu.

$$\|\vec{u}(\omega)\|_\infty = \sup_\omega \sqrt{u_1(\omega)^2 + u_2(\omega)^2 + \dots + u_m(\omega)^2} = \sup_\omega |\vec{u}(\omega)| \quad (4.7)$$

Touto normou je možno získat maximální velikost signálu přes všechny ω .

V teorii řízení se samozřejmě používají i další normy jak pro signály, tak pro systémy. Zpravidla z důvodu, že signál je definován jinak, například jako série jednotkových impulsů. Potom se samozřejmě mění interpretace jednotlivých norem. Těmito interpretacemi a normami se ale tato práce nezabývá.

4.2 Normy systému

Protože systémy popsané maticí operátorových přenosů jsou matice funkcí, lze na ně aplikovat veškeré maticové normy a každá bude mít význam. Pro naše účely ale není potřeba popisovat všechny interpretace všech maticových norem. Podíváme se pouze na dvě nejčastěji používané – \mathcal{H}_2 normu a \mathcal{H}_∞ normu.

4.2.1 \mathcal{H}_2 norma

\mathcal{H}_2 norma je definována rovnicí:

$$\|\mathbf{G}(s)\|_2 = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{tr} [G(j\omega)^H G(j\omega)] d\omega} \quad (4.8)$$

Rovnici je nutné nejdříve dekomponovat. Využijeme-li skutečnosti, že z matice operátorových přenosů lze zpětnou Laplaceovou transformací získat matici impulzních charakteristik systému a že obě tyto matice vyjadřují stejnou myšlenku.

$$\|\mathbf{G}(s)\|_2 = \|\mathbf{g}(\tau)\|_2 = \sqrt{\int_0^\infty \text{tr} [\mathbf{g}(\tau)^H \mathbf{g}(\tau)] d\tau}, \quad (4.9)$$

kde \mathbf{g} je matice impulzních charakteristik a τ je čas.

Dále využijeme znalosti, že funkce tr , tedy trace, je suma diagonálních prvků matice, a že je možné vyměnit pořadí sumace a integrace.

$$\|\mathbf{G}(s)\|_2 = \|\mathbf{g}(\tau)\|_2 = \sqrt{\sum_{ij} \int_0^\infty |g_{ij}(\tau)|^2 d\tau} \quad (4.10)$$

Nakonec využijeme toho, že impulsní charakteristika je funkce času, vyjadřující odezvu systému na jednotkový impuls na i -tém vstupu a znalosti \mathcal{H}_2 normy pro vektor.

$$\|\mathbf{G}(s)\|_2 = \sqrt{\sum_{i=1}^m \|\vec{y}_i(t)\|_2^2}, \quad (4.11)$$

kde $\vec{y}_i(t)$ je vektor výstupních funkcí času systému $\mathbf{G}(s)$ při působení jednotkového impulsu na i -tý vstup a m je počet vstupů systému.

Z rovnice 4.11 plyne, že \mathcal{H}_2 norma je průměrná hodnota výstupu systému při působení jednotkového impulsu na jeden vstup.

Všimněme si, že \mathcal{H}_2 norma je nekonečná pro systémy, které jsou nestabilní nebo nereálné, tedy jejich zesílení pro $\omega \rightarrow \infty$ je nenulové.

4.2.2 \mathcal{H}_2 norma pomocí singulárních hodnot

\mathcal{H}_2 normu je také možné definovat pomocí singulárních hodnot.

$$\|\mathbf{G}(s)\|_2 = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} \sum_i \sigma_i^2(\mathbf{G}(j\omega)) d\omega}, \quad (4.12)$$

tedy jako sumu kvadrátů singulárních hodnot. Minimalizace \mathcal{H}_2 normy bude tedy „tlačit dolů“ celou charakteristiku singulárních hodnot systému.¹

4.2.3 \mathcal{H}_∞ norma

\mathcal{H}_∞ norma je definována rovnicí

$$\|\mathbf{G}(s)\|_\infty = \max_{\vec{u}(t) \neq 0} \frac{\|\vec{y}(t)\|_2}{\|\vec{u}(t)\|_2} \quad (4.13)$$

¹Jsou samozřejmě možné i další interpretace a definice, těmi se ale tato práce nezabývá.

a vyjadřuje maximální zesílení systému. Ze znalosti definice zesílení je možné rovnici zjednodušit na:

$$\|\mathbf{G}(s)\|_{\infty} = \max_{\vec{u}(t)} \|\vec{y}(t)\|_2; \quad \|\vec{u}(t)\|_2 = 1 \quad (4.14)$$

Opět platí, že \mathcal{H}_{∞} norma je rovna nekonečnu pro nestabilní systémy a systémy s integrátorem.

\mathcal{H}_{∞} norma pomocí singulárních hodnot

\mathcal{H}_{∞} normu je také možné definovat pomocí singulárních hodnot.

$$\|\mathbf{G}(s)\| = \sup_{\omega} \bar{\sigma}(\mathbf{G}(j\omega)), \quad (4.15)$$

tedy jako supremum charakteristiky singulárních hodnot.

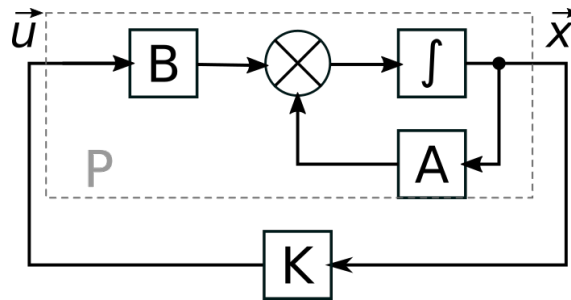
Minimalizace \mathcal{H}_{∞} normy bude tedy „tlačit dolů“ vrchol největší singulární hodnoty systému, což je velice výhodné z toho důvodu, že singulární hodnota odpovídá převrácené hodnotě zásoby stability v modulu. Z toho plyne, že minimalizace \mathcal{H}_{∞} normy zajišťuje stabilizaci a následnou maximalizaci zásoby stability v modulu, tedy jak ve fázi, tak v amplitudě.

5 Stavová regulace

Velmi známým způsobem regulace MIMO systémů je takzvaná stavová regulace. Jde o způsob regulace statickým regulátorem soustavy, která má měřitelné všechny stavy. Potom může být stavový popis 1.3 redukován na 5.1¹. [4]

$$\dot{\vec{x}} = \mathbf{A}\vec{x} + \mathbf{B}\vec{u} \quad (5.1)$$

Je-li systém dosažitelný, lze najít statický regulátor \mathbf{K} , který zapojením do zpětné vazby, z vektoru stavů \vec{x} na vektor vstupů \vec{u} , zaručí požadované chování regulované soustavy. Vhodnou volbou hodnot matice \mathbf{K} , lze dosáhnout libovolných komplexně sdružených vlastních čísel matice $\mathbf{A} + \mathbf{BK}$. Této metodě se říká metoda umístění pólů.



Obr. 5.1: Schematické zapojení stavového regulátoru

Regulovanou soustavu lze tedy popsat jako²

$$\dot{\vec{x}} = (\mathbf{A} + \mathbf{BK})\vec{x} \quad (5.2)$$

Cílem tohoto typu řízení je dostat všechny stavy z počáteční hodnoty \vec{x}_0 do ustálené hodnoty $\vec{x} = 0$. Proto je vhodné regulovanou soustavu vždy linearizovat okolo bodu, do kterého bude soustava regulována.

Pro systémy s jedním vstupem vždy platí, že řešení metody umístění pólů je právě jedno. Této skutečnosti bude dále využíváno pro porovnání regulátorů, navržených různými metodami. To ale neplatí pro systémy s více vstupy, u nichž je vždy možné nalézt více různých řešení, která všechna vedou na identické přenosy řízení. [4, 2]

5.1 Optimální stavová regulace

Největším problémem při řešení úlohy stavové regulace, pomocí metody umístění pólů, je zvolení vhodných vlastních čísel.

¹Za předpokladu, že matice stavového popisu $\mathbf{D} = 0$, což splňuje většina reálných systémů.

²V literatuře je možné narazit také na $\dot{\vec{x}} = (\mathbf{A} - \mathbf{BK})\vec{x}$. Oba zápisy jsou ekvivalentní a používané. Proto je třeba dodržet znaménka při všech krocích návrhu a následné aplikace regulátoru.

Jednou z metod, pomocí které lze navrhnout optimální matici regulátoru \mathbf{K} , je návrh pomocí lineárně kvadraticky optimálního kritéria $J(\vec{x}, \vec{u})$. Vzniká tak lineárně optimální regulátor, v literatuře [4, 2] je často popisován jako LQ regulátor nebo zkráceně LQR.

$$J(\vec{x}, \vec{u}) = \int_0^{\infty} (\vec{x}^T \mathbf{Q} \vec{x} + \vec{u}^T \mathbf{R} \vec{u}) d\tau, \quad (5.3)$$

kde $\mathbf{Q} = \mathbf{Q}^T \geq 0$ a $\mathbf{R} = \mathbf{R}^T > 0$.

Jak je vidět z rovnice 5.3, při výpočtu kritéria je možné pomocí váhovací matice³ \mathbf{Q} dávat určitou váhu době, za kterou se soustava ustálí, a tak si volit důležitost rychlosti regulace při návrhu. Stejně tak je možné penalizovat velikost akčního zásahu pomocí váhovací matice \mathbf{R} . Samozřejmě vždy platí, že j -tý prvek diagonály váhovací matice váhuje j -tý vstup nebo stav.

Z rovnice jasně plyne, že není důležitá konkrétní hodnota matic, ale jejich poměr.

Za předpokladu, že dvojice \mathbf{A} a \mathbf{B} je stabilizovatelná a dvojice \mathbf{A} a $\mathbf{Q}^{1/2}$ je pozorovatelná, je výsledný regulátor \mathbf{K} vypočten jako

$$\mathbf{K} = -\mathbf{R}^{-1} \mathbf{B}^T \mathbf{P}, \quad (5.4)$$

kde \mathbf{P} je řešením algebraické Ricattiho rovnice [4, 2]

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} - \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} + \mathbf{Q} = 0 \quad (5.5)$$

³Záměrně zde byl použit výraz „váhovací matice“, přestože standardní označení stochastické interpretace LQR je „kovarianční matice“, protože pojem „váhovací matice“ se bude dále používat u interpretace LQR pomocí \mathcal{H}_2 optimalizace.

6 Rekonstrukce stavů

Častou překážkou pro stavovou regulaci představuje nemožnost měření všech stavů v dostatečné kvalitě. Měřené stavy bývají zatíženy šumem, nebo nejsou vůbec měřitelné. Zároveň je často kladen důraz na minimální počet měřicích zařízení.

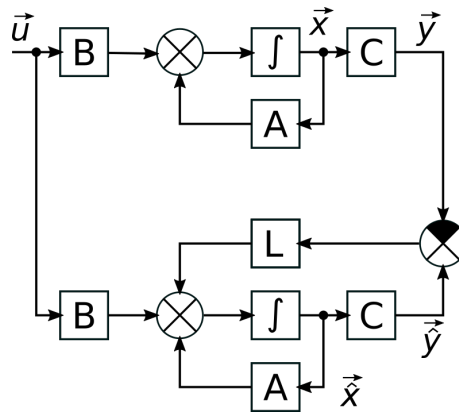
Řešením tohoto problému jsou stavové rekonstruktory,¹ díky kterým je možné rekonstruovat neměřené stavy a zároveň vyfiltrovat šum ze stavů měřených. To samozřejmě za předpokladu, že jsou rekonstruované stavy pozorovatelné. Problematika rekonstrukce stavů je obecně dobře známá, proto bude v této práci popsána pouze zběžně. Pro bližší informace je možné nahlédnout například do [4], [2] nebo [5].

Pozorovatel tedy řeší problém, kdy pro danou soustavu²

$$\begin{aligned}\dot{\vec{x}} &= \mathbf{A}\vec{x} + \mathbf{B}\vec{u} \\ \vec{y} &= \mathbf{C}\vec{x}\end{aligned}\quad (6.1)$$

platí, že diagonální matice \mathbf{C} , systému definovaného rovnicí 5.1, má některý řádek nulový, z čehož plyne, že nejsou měřitelné všechny stavy. Ty se rekonstruují pomocí kopie původního systému, jak je vidět na obrázku 6.1, se zavedenou vazbou z rozdílu výstupu soustavy \vec{y} a výstupu pozorovatele $\vec{\hat{y}}$. [4, 2]

$$\vec{\hat{y}} - \vec{y} = \vec{\tilde{y}} \quad (6.2)$$



Obr. 6.1: Blokové schéma stavového rekonstrukturu

Cílem je tedy navrhnout takový statický regulátor \mathbf{L} , který zařídí stabilitu pozorovatele a rychlé³ vyregulování $\vec{\tilde{y}}$ na nulu.

¹Pojmy „stavový rekonstruktor“, „pozorovatel“, nebo i „stavový filtr“ (Kalmanův filtr, robustní filtr) jsou, pokud není v literatuře definováno jinak, volně zaměnitelné. V některé literatuře [2] se rozlišuje pozorovatel od filtru tím, že na vstup filtru působí šum, zatímco na vstup pozorovatele nikoliv.

²Předpokladem je, že matice stavového popisu $\mathbf{D} = 0$, což splňuje většina reálných systémů.

³Většinou se pro rekonstruktor volí desetkrát rychlejší póly v porovnání s regulovanou soustavou.

Z poslední věty je patrná jistá podobnost se stavovou regulací. Přestože problémy stavové regulace a rekonstrukce nejsou obecně duální [5], v tomto případě jsou všechny podmínky duality splněny. Je tedy možné s dualitou počítat i nadále.

Díky dualitě problémů není třeba dále rozebírat návrh stavového rekonstruktoru, protože se jedná pouze o speciální případ návrhu stavového regulátoru.

Například v prostředí MATLAB je možné navrhnout stavový rekonstruktor pomocí příkazu `place`.

```
1 K~ = -place(A, B, [-3, -2, -1, -0.5]);
2 L = -(place(A', C', [-0.7+0.5i -0.7-0.5i -2 -2.5]))';
```

Výpis 6.1: Příklad návrhu stavového regulátoru \mathbf{K} a rekonstruktoru \mathbf{L} stejným příkazem v prostředí MATLAB

Stavový popis výsledného rekonstruktoru v maticovém zápisu, jak je blíže objasněno v příloze B, má tvar

$$\vec{\hat{x}} = (\mathbf{A} + \mathbf{LC}) \vec{\hat{x}} + \begin{bmatrix} -\mathbf{L} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \vec{y} \\ \vec{u} \end{bmatrix}, \quad (6.3)$$

kde \vec{y} je výstup rekonstruované soustavy, \vec{u} je vstup rekonstruované soustavy a $\vec{\hat{x}}$ je vektor rekonstruovaných stavů.

6.1 Optimální stavová rekonstrukce

Návrh hodnot matice \mathbf{L} provází stejná problematika jako návrh matice \mathbf{K} pro stavovou regulaci. Stejně jako v případě stavové regulace, jedná se i v tomto případě o duální problém. Je tedy možné definovat optimalizační kritérium, díky kterému je možné definovat optimální stavový rekonstruktor.

Optimalizační kritérium může být libovolné. V tomto případě bude nejpraktičtější udržet dualitu problémů a použít lineárně kvadratické kritérium 5.3, což znamená použít rekonstruktor typu Kalmanovo zesílení⁴. Díky volbě tohoto kritéria je možné přistupovat k optimálnímu stavovému rekonstruktoru jako ke speciálnímu případu návrhu optimálního stavového regulátoru. To může být demonstrováno například návrhem optimálního stavového rekonstruktoru a regulátoru stejným příkazem v prostředí MATLAB. [4, 2]

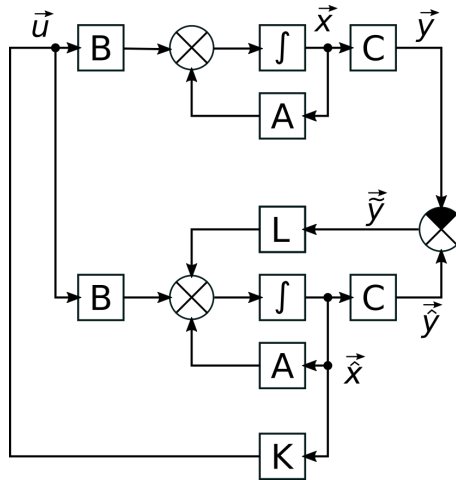
```
1 K_LQR = -lqr(A, B, Q, R);
2 L_LQR = (-lqr(A', C', Vd, Vn))';
```

Výpis 6.2: Příklad návrhu stavového optimálního regulátoru \mathbf{K} a rekonstruktoru \mathbf{L} stejným příkazem v prostředí MATLAB

⁴Kalmanovo zesílení se od Kalmanova filtru liší tím, že zesílení není přepočítáno po každém cyklu.[4]

7 LQG

V předchozí kapitole bylo zhruba vysvětleno, jak stavově regulovat soustavu optimálním stavovým regulátorem, a to i v případě, kdy nejsou všechny stavy měřitelné nebo jsou zatíženy šumem, za pomoci optimálního stavového rekonstruktoru typu Kalmanovo zesílení. Kaskádním zapojením těchto dvou optimálních zařízení vzniká struktura zvaná LQG.



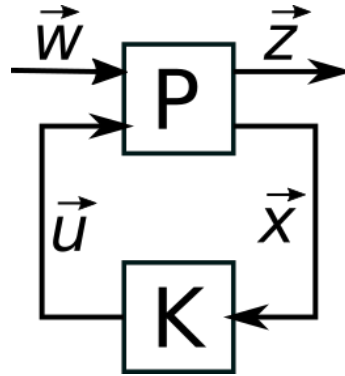
Obr. 7.1: Schematické zapojení LQG

Tato struktura je obecně známá a je v praxi stále používána [6, 7, 8], přestože bylo již v roce 1978 dokázáno, že není možné zaručit žádnou zásobu stability [9] při návrhu struktury.

Hlavním cílem této práce je srovnat tuto regulační strukturu s regulátorem optimalizovaným na \mathcal{H}_∞ normu, která přímo souvisí se zásobou stability v modulu.

8 \mathcal{H}_∞ a \mathcal{H}_2 regulace

Ve své podstatě není mezi optimalizací \mathcal{H}_∞ a \mathcal{H}_2 norem velký rozdíl. V obou případech se regulátor syntetizuje tak, že se úloha nejprve formuluje jako obecný regulační problém, a následně se hledá takový regulátor, který minimalizuje vybranou normu na systému se vstupem \vec{w} a výstupem \vec{z}



Obr. 8.1: Obecný regulační problém

V literatuře, například [3] nebo [10], je často obecný regulační problém definován stavovým popisem

$$\begin{bmatrix} \vec{x} \\ \vec{z} \\ \vec{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_1 & \mathbf{B}_2 \\ \mathbf{C}_1 & \mathbf{D}_{11} & \mathbf{D}_{12} \\ \mathbf{C}_2 & \mathbf{D}_{21} & \mathbf{D}_{22} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix} = \mathbf{P} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix}, \quad (8.1)$$

zatímco v jiné literatuře, například [2] a [11], je vyměněno pořadí vstupů \vec{w} a \vec{u} . To znamená, že v jedné literatuře vyjadřuje \mathbf{D}_{11} vztah mezi vstupem \vec{w} a výstupem \vec{z} , v jiné literatuře vyjadřuje stejně pojmenovaná matice vztah mezi \vec{u} a \vec{z} . Tato kolize ve značení může způsobit nedorozumění. Proto zde bude dále používáno označení z rovnice 8.2

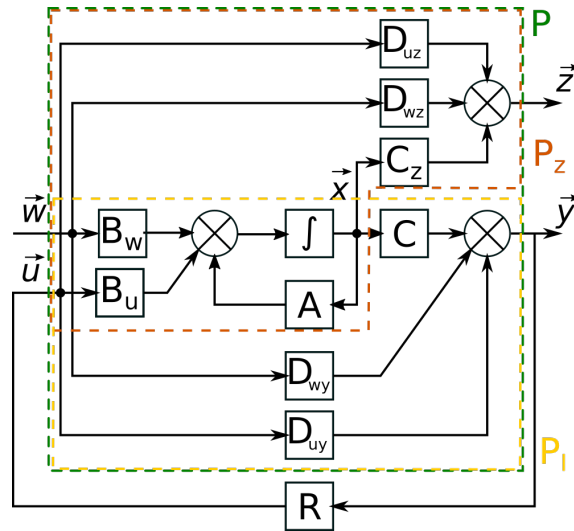
$$\begin{bmatrix} \vec{x} \\ \vec{z} \\ \vec{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_w & \mathbf{B}_u \\ \mathbf{C}_z & \mathbf{D}_{wz} & \mathbf{D}_{uz} \\ \mathbf{C}_y & \mathbf{D}_{wy} & \mathbf{D}_{uy} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix} = \mathbf{P} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix}, \quad (8.2)$$

kde \vec{z} je optimalizovaný výstup, \vec{w} je vnější vstup, často plnicí funkci vstupu žádané hodnoty, a \vec{u} je vstup do původní soustavy, tedy výstup z regulátoru. Vnitřní matice \mathbf{P} popisují zesílení mezi jednotlivými vstupy a výstupy.

Rovnice 8.2 je zbytečně složitá jak pro optimalizaci, tak pro implementaci. Pro optimalizaci je možné ignorovat výstup \vec{y} , tedy i celý jeho řádek. Vzniká tak optimalizační zapojení 8.3. Pro implementaci není důležitý výstup \vec{z} a jemu příslušící řádek. Vzniká tak implementační zapojení 8.4.

$$\begin{bmatrix} \vec{\dot{x}} \\ \vec{z} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_w & \mathbf{B}_u \\ \mathbf{C}_z & \mathbf{D}_{wz} & \mathbf{D}_{uz} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix} = \mathbf{P}_o \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix} \quad (8.3)$$

$$\begin{bmatrix} \vec{\dot{x}} \\ \vec{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_w & \mathbf{B}_u \\ \mathbf{C}_y & \mathbf{D}_{wy} & \mathbf{D}_{uy} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix} = \mathbf{P}_i \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix} \quad (8.4)$$



Obr. 8.2: Obecný regulační problém, detail

Norma se počítá pro zapojení včetně regulátoru. Stavový popis lze získat dolní¹ zlomkovou transformací

$$\vec{z} = F_l(\mathbf{P}_o, \mathbf{R}) \vec{w} \quad (8.5)$$

kde \mathbf{R} je regulátor a $F_l()$ je operátor dolní zlomkové transformace. Tím se získá stavový popis s jedním vstupním a jedním výstupním vektorem. Cílem návrhu je najít takové \mathbf{R} , které minimalizuje normu tohoto přenosu.

Aby byla úloha řešitelná, musí být splněny následující podmínky:

- $(\mathbf{A}, \mathbf{B}_u, \mathbf{C}_y)$ je říditelné,
- \mathbf{D}_{uz} a \mathbf{D}_{wy} mají plnou hodnost,
- $\begin{bmatrix} \mathbf{A} - j\omega\mathbf{I} & \mathbf{B}_u \\ \mathbf{C}_z & \mathbf{D}_{uz} \end{bmatrix}$ má plnou sloupcovou hodnost pro všechna ω ,

¹Je možné provést i horní zlomkovou transformaci, je-li to výhodnější.

- $\begin{bmatrix} \mathbf{A} - j\omega\mathbf{I} & \mathbf{B}_w \\ \mathbf{C}_y & \mathbf{D}_{wy} \end{bmatrix}$ má plnou řádkovou hodnotu pro všechna ω ,
- $\mathbf{D}_{wz} = 0$ a $\mathbf{D}_{uz} = 0$. [3]

Jsou-li podmínky splněny, je možné problém řešit různými způsoby, například pomocí lineárních maticových nerovností. Speciální případ návrhu pomocí LMI bude popsán níže. Dalším způsobem je řešení algebraickými Ricattiho rovnicemi, které není v této práci blíže zkoumáno.²

8.1 \mathcal{H}_2 syntéza regulátoru

V předchozí kapitole bylo vysvětleno, jak je možné formulovat problém tak, aby pro něj bylo možné navrhnout (syntetizovat) regulátor minimalizací normy.³ V této kapitole bude vysvětleno, jakým způsobem lze syntetizovat \mathcal{H}_2 regulátor.

Je zjištěno, že minimalizace \mathcal{H}_2 normy výše popsaného problému je problém konvexní. Nyní je třeba zvolit matice \mathbf{C}_z , \mathbf{D}_{wz} a \mathbf{D}_{uz} . Pokud se vezme v potaz, jaká je definice a význam \mathcal{H}_2 normy 4.12, bude jasné, že minimalizací \mathcal{H}_2 normy bude minimalizován integrál singulárních hodnot přenosu.

Zaměříme se nyní na význam jednotlivých matic \mathbf{C}_z , \mathbf{D}_{wz} a \mathbf{D}_{uz} . Pomocí nichž můžeme volit (dosazením nulového nebo nenulového čísla), která část obvodu bude optimalizována, a která ne. Například zvolením nenulové matice \mathbf{C}_z bude optimalizována frekvenční charakteristika signálu \vec{x} , tedy stavů. Čím větší bude váha, tím větší bude snaha o zmenšení zesílení singulární charakteristiky signálu, protože vynásobením signálu větším číslem bude mít signál větší zesílení ve frekvenční charakteristice a tím větší bude jeho příspěvek k \mathcal{H}_2 normě systému.

Stejným způsobem je možné ukládat váhu na minimalizaci frekvenčních charakteristik signálů \vec{u} a \vec{w} pomocí matic \mathbf{D}_{wu} a \mathbf{D}_{wz} . Často se volí $\mathbf{D}_{wu} = \mathbf{0}$ [3].

Stejnou myšlenku je možné vyjádřit i pomocí optimalizace na bílý šum. Bude-li zmenšeno zesílení singulární charakteristiky přes všechny frekvence, pak bude potlačena citlivost systému na bílý šum.

8.1.1 LQR pomocí \mathcal{H}_2 optimalizace

Mezi \mathcal{H}_2 syntézou a LQR je možné najít několik podobností. V obou případech se jedná o návrh optimálního regulátoru, v obou případech se minimalizuje integrál

²Pro více informací o řešení pomocí algebraických Ricattiho rovnic je možné využít například [2], [3], [12] nebo [13].

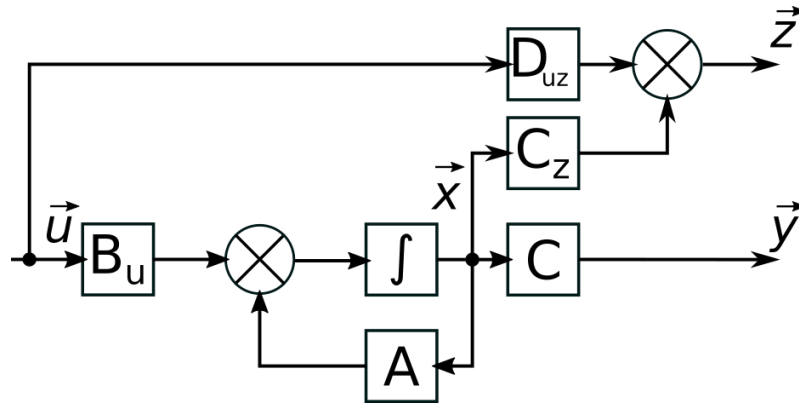
³Ve skutečnosti se pro takto formulovaný problém dají syntetizovat regulátory i na základě jiných optimalizačních kritérií, ty ale nejsou v této práci dále rozebírány. [3]

a v obou případech je možné nastavovat váhy jednotlivým signálům. Je tomu tak proto, že LQR je speciální případ \mathcal{H}_2 regulátoru.

Předpokládejme soustavu popsanou rovnicí 8.3, kde $\mathbf{D}_{uy} = \mathbf{0}^4$ a $\mathbf{C}_y = \mathbf{I}$. Systém bude regulován na nulu, proto $\vec{w} = 0$. Tento vstup je tedy možné úplně ignorovat. Vznikne následující systém

$$\begin{bmatrix} \vec{x} \\ \vec{z} \\ \vec{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_u \\ \mathbf{C}_z & \mathbf{D}_{uz} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{u} \end{bmatrix} = \mathbf{P}_{LQR} \begin{bmatrix} \vec{x} \\ \vec{u} \end{bmatrix}, \quad (8.6)$$

který má měřitelné všechny stavy a pomocí \mathbf{C}_z a \mathbf{D}_{uz} je možné váhovat frekvenční charakteristiky signálů \vec{x} a \vec{u} .



Obr. 8.3: LQR definováno jako \mathcal{H}_2 problém

Je-li n_x počet stavů a n_u počet vstupů systému, zvolíme si počet optimalizovaných výstupů $n_z = n_x + n_u$. Nyní definujeme jednotlivé signály \vec{z} tak, aby každý signál byl pouze jeden ze signálů \vec{x} nebo \vec{u} vynásobený konkrétní, skalární vahou.

$$\begin{aligned} \vec{x} &= [x_1, x_2, \dots, x_{n_x}]^T \\ \vec{u} &= [u_1, u_2, \dots, u_{n_u}]^T \\ \vec{z} &= [W_{x_1}x_1, W_{x_2}x_2, \dots, W_{x_{n_x}}x_{n_x}, W_{u_1}u_1, W_{u_2}u_2, \dots, W_{u_{n_u}}u_{n_u}]^T \end{aligned} \quad (8.7)$$

⁴Jedná se o stejný předpoklad jako ten, který byl použit v kapitole 5.

Vytvoříme-li nyní vektory, potažmo diagonální matice vah

$$\mathbf{I}\vec{W}_x = \mathbf{I} [W_{x_1}, W_{x_2}, \dots, W_{x_{n_x}}]^T = \begin{bmatrix} W_{x_1} & 0 & \dots & 0 \\ 0 & W_{x_2} & & \\ \vdots & & \ddots & \\ 0 & & & W_{x_{n_x}} \end{bmatrix} = \mathbf{W}_x \quad (8.8)$$

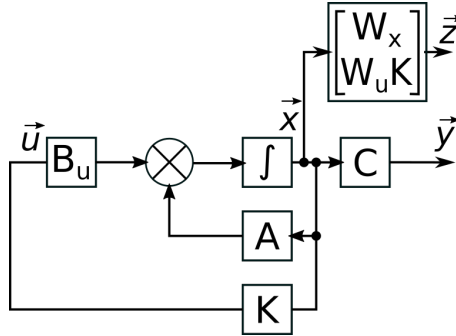
$$\mathbf{I}\vec{W}_u = \mathbf{I} [W_{u_1}, W_{u_2}, \dots, W_{u_{n_u}}]^T = \begin{bmatrix} W_{u_1} & 0 & \dots & 0 \\ 0 & W_{u_2} & & \\ \vdots & & \ddots & \\ 0 & & & W_{u_{n_u}} \end{bmatrix} = \mathbf{W}_u, \quad (8.9)$$

je jednoduché určit \mathbf{C}_z a \mathbf{D}_{uz}

$$\vec{z} = \begin{bmatrix} \mathbf{W}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_u \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{u} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_z & \mathbf{D}_{uz} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{u} \end{bmatrix} \quad (8.10)$$

$$\begin{bmatrix} \mathbf{W}_x \\ \mathbf{0} \end{bmatrix} = \mathbf{C}_z, \quad \begin{bmatrix} \mathbf{0} \\ \mathbf{W}_u \end{bmatrix} = \mathbf{D}_{uz}, \quad (8.11)$$

což je patrné z obrázku 8.3. Nyní, za předpokladu, že $\vec{u} = \mathbf{K}_{H_2}\vec{x}$, můžeme použít pouze jednu váhu $\mathbf{W} = \begin{bmatrix} \mathbf{W}_x \\ \mathbf{W}_u\mathbf{K}_{H_2} \end{bmatrix}$.



Obr. 8.4: LQR definováno jako \mathcal{H}_2 problém s jednou váhou

Nyní je možné z definic \mathcal{H}_2 normy 4.6 a LQR optimalizačního kritéria 5.3 určit

vztah mezi vahami \mathbf{Q} a \mathbf{R} určenými pro LQR návrh a vahami \mathbf{W}_x a \mathbf{W}_u .⁵ [2]

$$\begin{aligned}
\|\vec{z}\|_2^2 &= \left(\sqrt{\int_0^\infty \vec{z}^T \vec{z} d\tau} \right)^2 \doteq \\
&= \int_0^\infty \left(\vec{x}^T \mathbf{W}^T \mathbf{W} \vec{x} \right) d\tau = \\
&= \int_0^\infty \left(\vec{x}^T \begin{bmatrix} \mathbf{W}_x & \mathbf{W}_u \mathbf{K}_{H2} \end{bmatrix} \begin{bmatrix} \mathbf{W}_x \\ \mathbf{W}_u \mathbf{K}_{H2} \end{bmatrix} \vec{x} \right) d\tau = \\
&= \int_0^\infty \left(\vec{x}^T \mathbf{C}_z^T \mathbf{C}_z \vec{x} + \vec{u}^T \mathbf{D}_{uz}^T \mathbf{D}_{uz} \vec{u} \right) d\tau = \\
&= \int_0^\infty \left(\vec{x}^T \begin{bmatrix} \mathbf{Q}^{1/2} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{Q}^{1/2} \\ \mathbf{0} \end{bmatrix} \vec{x} + \vec{u}^T \begin{bmatrix} \mathbf{0} & \mathbf{R}^{1/2} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{R}^{1/2} \end{bmatrix} \vec{u} \right) d\tau = \\
&= \int_0^\infty \left(\vec{x}^T \mathbf{Q} \vec{x} + \vec{u}^T \mathbf{R} \vec{u} \right) d\tau = \\
&= J(\vec{x}, \vec{u})
\end{aligned} \tag{8.12}$$

Z rovnice 8.12 je patrný vztah mezi vahami

$$\begin{bmatrix} \mathbf{Q}^{1/2} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{W}_x \\ \mathbf{0} \end{bmatrix} = \mathbf{C}_z, \quad \begin{bmatrix} \mathbf{0} \\ \mathbf{R}^{1/2} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{W}_u \end{bmatrix} = \mathbf{D}_{uz} \tag{8.13}$$

Nyní, když byl vysvětlen způsob návrhu LQR regulátoru pomocí \mathcal{H}_2 syntézy, je možné díky dualitě problémů navrhnout také rekonstruktor stavů typu Kalmanovo zesílení. Postupnou syntézou regulátoru \mathbf{K}_{H2} a rekonstruktoru \mathbf{L}_{H2} a následným zapojením do LQG struktury bude navržen LQG pomocí \mathcal{H}_2 syntézy.

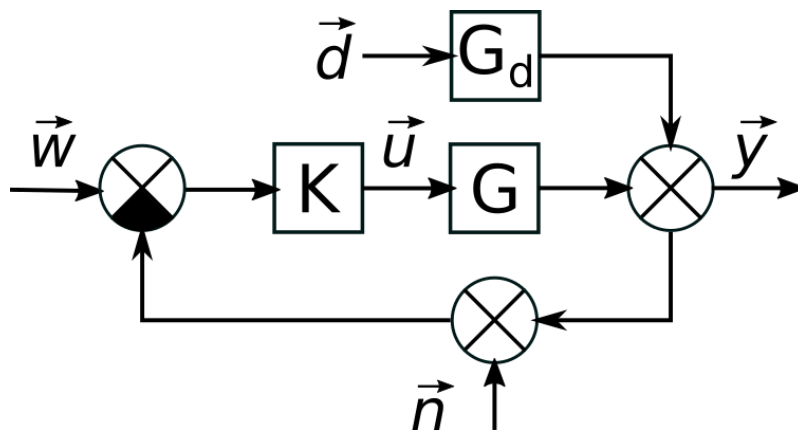
8.2 \mathcal{H}_∞ syntéza regulátoru

\mathcal{H}_∞ regulátor teoreticky lze navrhnout stejně jako \mathcal{H}_2 regulátor s tím rozdílem, že bude minimalizována jiná norma. Bohužel by vzhledem k definici normy 4.15 bylo se statickými vahami potlačeno zesílení charakteristiky singulárních hodnot v nižších frekvencích, což by vedlo ke špatné, nebo dokonce nestabilní regulaci. [3]

Z toho důvodu se používají dynamické váhy, tedy váhy, které nemají konstantní charakteristiku singulárních hodnot přes všechny frekvence.

Častým způsobem návrhu \mathcal{H}_∞ regulátoru je tvarováním citlivostní a komplementární citlivostní funkce \mathbf{S} a \mathbf{T} .

⁵V rovnici 8.12 se předpokládá, že všechny signály jsou závislé na čase (např. $\vec{z}(\tau)$), označení ale bylo pro větší přehlednost vynecháno.



Obr. 8.5: Schéma obecného \mathcal{H}_∞ problému

Jak je patrné z definice obecného \mathcal{H}_∞ problému zobrazeného na obrázku 8.5, je možné výstup vyjádřit jako

$$\begin{aligned} \vec{y} &= (\mathbf{I} + \mathbf{G}\mathbf{K})^{-1}\mathbf{G}\mathbf{K}\vec{w} + (\mathbf{I} + \mathbf{G}\mathbf{K})^{-1}\mathbf{G}_d\vec{d} - (\mathbf{I} + \mathbf{G}\mathbf{K})^{-1}\mathbf{G}\mathbf{K}\vec{n} = \\ &= \mathbf{T}\vec{w} + \mathbf{S}\mathbf{G}_d\vec{d} - \mathbf{T}\vec{n}, \end{aligned} \quad (8.14)$$

kde \vec{w} je vektor žádaných hodnot, \vec{d} je vektor poruch, \vec{n} je vektor šumu, \mathbf{S} je citlivostní funkce, také známá jako přenos odchylky, a \mathbf{T} je komplementární citlivostní funkce, neboli přenos řízení.

Podobně jako se v LQR penalizovaly signály \vec{x} , \vec{y} a \vec{u} , v \mathcal{H}_∞ optimalizaci se vážují funkce \mathbf{S} , \mathbf{T} a $\mathbf{K}\mathbf{S}$ ⁶. Ve skutečnosti se ale v uzavřené smyčce jedná o stejnou myšlenku, protože je-li větší váha na citlivostní funkci \mathbf{S} , je penalizováno sledování žádané hodnoty a potlačení poruchy. Dynamickým váhováním komplementární citlivostní funkce \mathbf{T} je zase pro různé ω penalizováno potlačení šumu. Váhováním funkce $\mathbf{K}\mathbf{S}$ je možné penalizovat velikost akčního zásahu.

Problematika volby vah pro citlivostní a komplementární citlivostní funkce je velmi obsáhlá, proto zde nebude detailněji rozebírána. Více podrobností například v literatuře [3] a [13].

S navrženými vahami, ať už metodou tvarování citlivostní a komplementární citlivostní funkce nebo libovolnou jinou, je možné pokročit k dalšímu kroku návrhu, a tím je samotná syntetizace.

Nejčastěji bývá navržen dynamický \mathcal{H}_∞ optimální regulátor. Tento regulátor je často velmi vysokého řádu, což nemusí být vždy vhodné.

Je také možné navrhnout regulátor ve tvaru $\vec{u} = \mathbf{K}\vec{y}$, čímž vzniká problém \mathcal{H}_∞ optimalizace stavového regulátoru. Je dokázáno, že i tento problém je stejně jako dynamický \mathcal{H}_∞ regulátor konvexní. [2, 10, 3]

⁶Dále v textu bude používáno zkrácené značení S , T a KS .

9 Implementace syntézy \mathcal{H}_2 a \mathcal{H}_∞ regulátoru

Problém syntézy regulátoru pomocí normy lze řešit více způsoby. Nejpoužívanějším způsobem je řešení pomocí algebraických Ricattiho rovnic. Další cestou je řešení lineárních maticových nerovností, dále zkráceně LMI.

V této práci bude rozebrán přístup návrhu stavových regulátorů pomocí LMI.

9.1 Suboptimální regulátor

Většina výpočetních algoritmů je ve skutečnosti schopna vypočítat pouze suboptimální regulátor. Je tomu tak zejména proto, že s optimálním regulátorem se v počítači nedá počítat.¹ V praxi se však hodnoty před implementací zaokrouhlují, suboptimalita řešení se tedy neprojeví.

Nejčastějším iteračním způsobem řešení syntézy \mathcal{H}_2 a \mathcal{H}_∞ regulátorů je použití γ -iterace, kdy se místo řešení s minimální normou hledá řešení, pro které platí

$$\|\mathbf{G}(s)\| < \gamma, \quad (9.1)$$

kde γ je libovolný pozitivní skalár.

Je-li řešení nalezeno, je nalezen suboptimální regulátor. Zmenšováním parametru γ , většinou metodou půlení intervalu, se hledá minimální γ , pro které je možné najít řešení a tedy i suboptimální řešení, které se co nejvíce blíží řešení optimálnímu.

9.2 \mathcal{H}_2 stavový regulátor pomocí LMI

\mathcal{H}_2 stavový regulátor pomocí LMI řeší problém popsany rovnicí 9.2².

$$\begin{bmatrix} \vec{x} \\ \vec{z} \\ \vec{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_w & \mathbf{B}_u \\ \mathbf{C}_z & \mathbf{0} & \mathbf{D}_{uz} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix} \quad (9.2)$$

Předpokladem je řešení ve tvaru $\vec{u} = \mathbf{K}\vec{x}$ takové, že \mathcal{H}_2 norma systému

$$\begin{aligned} \vec{x} &= (\mathbf{A} + \mathbf{B}_u\mathbf{K})\vec{x} + \mathbf{B}_w\vec{w} \\ \vec{z} &= (\mathbf{C} + \mathbf{D}_{uz}\mathbf{K})\vec{x} \end{aligned} \quad (9.3)$$

je minimální.

¹Například kvůli možnému nekonečnému desetinnému rozvoji.

²Všimněme si, že tento problém je obecnější, než problém vysvětlený v kapitole 8.1.1, což znamená, že LQR pomocí \mathcal{H}_2 optimalizace je speciálním případem popsáního problému.

\mathcal{H}_2 suboptimální regulátor je možné navrhnout pomocí

$$\min \gamma \left\{ \begin{array}{l} \text{s.t.} \\ \left[\begin{array}{cc} -\mathbf{Z} & \mathbf{C}\mathbf{X} + \mathbf{D}\mathbf{W} \\ (\mathbf{C}\mathbf{X} + \mathbf{D}\mathbf{W})^T & -\mathbf{X} \end{array} \right] < 0 \\ \mathbf{A}\mathbf{X} + \mathbf{B}_u\mathbf{W} + (\mathbf{A}\mathbf{X} + \mathbf{B}_u\mathbf{W})^T + \mathbf{B}_w\mathbf{B}_w^T < 0 \\ \text{trace}(\mathbf{Z}) < \gamma^2 \end{array} \right. , \quad (9.4)$$

kde je optimalizace prováděna pomocí $\mathbf{Z} = \mathbf{Z}^T$, $\mathbf{X} = \mathbf{X}^T$ a \mathbf{W} .

Výsledný regulátor se poté vyjádří jako

$$\mathbf{K} = \mathbf{W}\mathbf{X}^{-1} \quad (9.5)$$

Odvození příslušné LMI není objektem této práce, proto bylo použito odvození z [2]³.

9.3 \mathcal{H}_∞ stavový regulátor pomocí LMI

Předpokládejme problém popsany rovnicí 9.6.

$$\begin{bmatrix} \vec{x} \\ \vec{z} \\ \vec{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_w & \mathbf{B}_u \\ \mathbf{C}_z & \mathbf{D}_{wz} & \mathbf{D}_{uz} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u} \end{bmatrix} \quad (9.6)$$

Předpokladem je řešení ve tvaru $\vec{u} = \mathbf{K}\vec{x}$ takové, že \mathcal{H}_∞ norma systému

$$\begin{aligned} \vec{x} &= (\mathbf{A} + \mathbf{B}_u\mathbf{K})\vec{x} + \mathbf{B}_w\vec{w} \\ \vec{z} &= (\mathbf{C} + \mathbf{D}_{uz}\mathbf{K})\vec{x} + \mathbf{D}_{wz}\vec{w} \end{aligned} \quad (9.7)$$

je minimální.

Pro takto definovaný problém je možné najít řešení pomocí

$$\min \gamma \left\{ \begin{array}{l} \text{s.t.} \\ \mathbf{X} > 0 \\ \left[\begin{array}{ccc} (\mathbf{A}\mathbf{X} + \mathbf{B}_u\mathbf{W})^T + \mathbf{A}\mathbf{X} + \mathbf{B}_u\mathbf{W} & \mathbf{B}_w & (\mathbf{C}\mathbf{X} + \mathbf{D}_{uz}\mathbf{W})^T \\ \mathbf{B}_w^T & -\gamma\mathbf{I} & \mathbf{D}_{wz}^T \\ \mathbf{C}\mathbf{X} + \mathbf{D}_{uz}\mathbf{W} & \mathbf{D}_{wz} & -\gamma\mathbf{I} \end{array} \right] < 0 \end{array} \right. , \quad (9.8)$$

kde je optimalizace prováděna pomocí $\mathbf{X} = \mathbf{X}^T$ a \mathbf{W} . [2]

Výsledný regulátor se poté vyjádří jako

$$\mathbf{K} = \mathbf{W}\mathbf{X}^{-1} \quad (9.9)$$

³Další možná odvození jsou například v [2], [10] nebo [14].

9.4 Implementace syntézy v prostředí MATLAB

V předchozích kapitolách bylo vysvětleno, jak využít LMI pro výpočet suboptimálního regulátoru. Nyní je potřeba tento výpočet implementovat. Pro implementaci bylo zvoleno prostředí MATLAB, které umožňuje nejrůznější způsoby řešení LMI. Mezi nimi vestavěné funkce [15], toolbox `cvx` nebo toolbox `YALMIP`. [16]

Z důvodu syntaktické přehlednosti byl zvolen toolbox `YALMIP` [17, 18, 19] a solver `csdp` [20]. Tento solver je v dokumentaci toolboxu `YALMIP` uveden na prvním místě podporovaných solverů s otevřeným kódem.

9.4.1 Syntéza \mathcal{H}_2 regulátoru pomocí LMI v prostředí MATLAB

Jelikož teorie byla vysvětlena v předchozích kapitolách, je možné demonstrovat návrh přímo úryvkem kódu.

```
1 %% Definice systemu
2 A = [0 1; -1 1];
3 B = [0; 1];
4 C = eye(2);
5 D = [0; 0];
6 X0 = [1; 0];
7 %% Definice vahovacich matic
8 Q = eye(2);
9 R = 1;
10 %% Definice obecného optimalizačního schématu
11 [numStateVars, ~] = size(B);
12 Bu = B;
13 Bw = B;
14 Cz = [Q^(1/2); 0 0];
15 Duz = zeros(numStateVars+length(R), length(R));
16 Duz(end) = R^(1/2);
17 %% LQR
18 LQR = -lqr(A, B, Q, R);
19 %% LMI
20 gamma = 1.639;
21 [numStateVars, numInputs] = size(Bw);
22 Z = sdpvar(numStateVars+1, numStateVars+1, 'symmetric');
23 X = sdpvar(numStateVars, numStateVars, 'symmetric');
24 W = sdpvar(numInputs, numStateVars, 'full');
25
26 constraintMatrix1 = [ -Z      Cz*X+Duz*W
27                      (Cz*X+Duz*W)' -X ];
28 constraintMatrix2 = A*X + Bu*W + (A*X+Bu*W)' + Bw*Bw';
29 F = trace(Z) <= gamma^2;
30 F = [F, constraintMatrix1 <= eye(size(constraintMatrix1))*eps];
31 F = [F, constraintMatrix2 <= eye(size(constraintMatrix2))*eps];
32 optimize(F);
33 K_H2 = value(W)*inv(value(X));
```

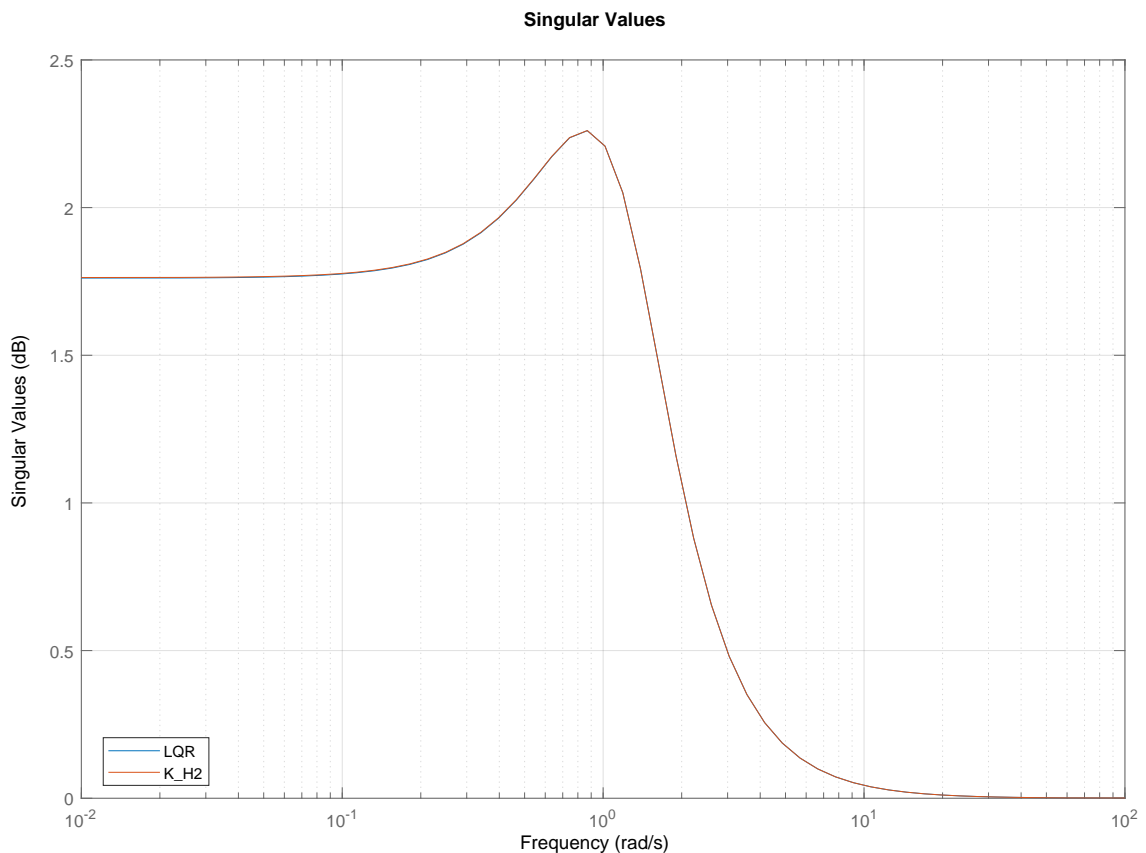
Výpis 9.1: Příklad výpočtu LQR regulátoru a ekvivalentního \mathcal{H}_2 stavového regulátoru v prostředí MATLAB.

Pro zjednodušení není v kódu uveden algoritmus půlení intervalu, ale je již zvoleno minimální γ . Výsledky výpočtu pomocí příkazu `lqr` a pomocí řešení LMI jsou velice podobné, jak je vidět na konzoli po spuštění kódu

1	names	gamma	result	
2	-----	-----	-----	
3				
4	'LQR '	NaN	-9.0499	-11.913
5	'K_H2 '	1.092	-9.0364	-11.907

Výpis 9.2: Výsledky zobrazené po spuštění kódu 9.1

nebo na obrázku 9.1



Obr. 9.1: Porovnání singulárních hodnot systémů s regulátorem navrženým příkazem `lqr` a vypočítaným pomocí LMI

Jak je vidět, není velký rozdíl mezi regulátorem vypočítaným pomocí LMI a regulátorem navrženým příkazem `lqr`. Proto bude dále v práci používán příkaz `lqr` pro jednoduchost jeho použití.

Další příklad, který dokazuje, že se v tomto případě nejedná o náhodnou shodu, je v příloze C.

9.4.2 Syntéza \mathcal{H}_∞ regulátoru pomocí LMI v prostředí MATLAB

Dříve, než bude popsán návrh pomocí LMI, je potřeba zvolit srovnání, aby bylo možné určit, zda byl návrh úspěšný. Prostředí MATLAB umožňuje návrh statického \mathcal{H}_∞ optimálního stavového regulátoru dvěma způsoby. Prvním způsobem je použití příkazu `hinfffc`, který řeší problém v obecném minimalizačním tvaru 9.10

$$\begin{bmatrix} \vec{\dot{x}} \\ \vec{z} \\ \vec{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}_w & \mathbf{B}_{u_1} & \mathbf{0} \\ \mathbf{C}_z & \mathbf{D}_{wz} & \mathbf{0} & \mathbf{I} \\ \mathbf{C}_y & \mathbf{D}_{wy} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{w} \\ \vec{u}_1 \\ \vec{u}_2 \end{bmatrix} \quad (9.10)$$

$$\vec{u} = \begin{bmatrix} \vec{u}_1 \\ \vec{u}_2 \end{bmatrix} = \mathbf{K}\vec{y}, \quad (9.11)$$

kde \vec{u}_2 je vstup ovlivňující optimalizovaný výstup. Použití příkazu je patrné z 9.3.

Dalším způsobem návrhu statického regulátoru je využití příkazu `hinfstruct`. Tento příkaz umožňuje nalezení suboptimálního \mathcal{H}_∞ regulátoru libovolné struktury, přestože tento problém není obecně konvexní.⁴ Jeho použití je podobné jako pro většinu ostatních příkazů v `robust constol toolbox`,⁵ proto nebude v této práci blíže popisováno. Příklad použití je dobře patrný z výpisu 9.3.

```

1 %% Definice systemu
2 A = [2 1 -2
3       1 -1 -3
4       4 0 -1];
5 Bu = [1; 0; 3];
6 Bw = [1; 0.2; -0.5];
7 %% Definice vahovacich matic
8 Cz = [2 1 -0.5];
9 Duz = 1;
10 Dwz = 0.05;
11 %% hinffi
12 [numStateVars, numInputs] = size(Bu);
13 P = ss(A, [Bw Bu], Cz, [Dwz Duz]);
14 [K_hinffi,~,gamma_hinffi] = hinffi(P,numInputs,[0.6865 Inf]);
15 K_hinffi = K_hinffi(1:numInputs,1:numStateVars);
16 %% hinfstruct
17 Plant = ss(A, [Bw Bu], [Cz; eye(3)], [Dwz Duz; zeros(3,2)]);
18 Plant.InputName = {'w', 'u'};
19 Plant.OutputName = {'z', 'x1', 'x2', 'x3'};
20 Plant.name = 'P';
21
22 K_hinfstruct_tunable = tunableGain('K_hinfstruct_tunable', 1, 4);
23 K_hinfstruct_tunable.InputName = {'x1', 'x2', 'x3', 'w'};
24 K_hinfstruct_tunable.OutputName = {'u'};
25
26 system = connect(Plant, K_hinfstruct_tunable,...
```

⁴Více informací o fungování příkazu je možné dohledat v [21].

⁵Například `hifsyn`, `h2syn`, `h2hifsyn`, `musyn` nebo `mixsyn`.

```

27     {'w'},...
28     {'z'});
29
30 rng('default')
31 opt = hinfstructOptions('Display','final','RandomStart',5);
32 [sysOpt, gamma_hinfstruct] = hinfstruct(system,opt);
33
34 K_hinfstruct = getBlockValue(sysOpt,'K_hinfstruct_tunable');
35 K_hinfstruct = K_hinfstruct.D;
36 K_hinfstruct = K_hinfstruct(1:numInputs,1:numStateVars);
37 %% LMI
38 gamma = 0.73269;
39
40 [numZoutputs, ~] = size(Duz);
41 [numStateVars, numInputs] = size(Bu);
42 X = sdpvar(numStateVars,numStateVars,'symmetric');
43 W = sdpvar(numStateVars,numStateVars,'symmetric');
44 V = sdpvar(numInputs,numStateVars,'full');
45 ops = sdpsettings('verbose',0);
46
47 constraintMatrix = [(A*X+Bu*W)' + A*X+Bu*W Bw (Cz*X+Duz*W)'
48                   Bw' -gamma*eye(1,1) Dwz'
49                   (Cz*X+Duz*W) Dwz -gamma*eye(numZoutputs)
50                   ];
51 F = [X >= eye(numStateVars)*eps];
52 F = [F, constraintMatrix <= eye(size(constraintMatrix))*eps];
53 optimize(F,[ ],ops);
54 K_Hinf = value(W)*inv(value(X));

```

Výpis 9.3: Příklad výpočtu statického \mathcal{H}_∞ optimálního regulátoru pomocí řešení LMI problému a příkazů `hinffi` a `hinfstruct`.

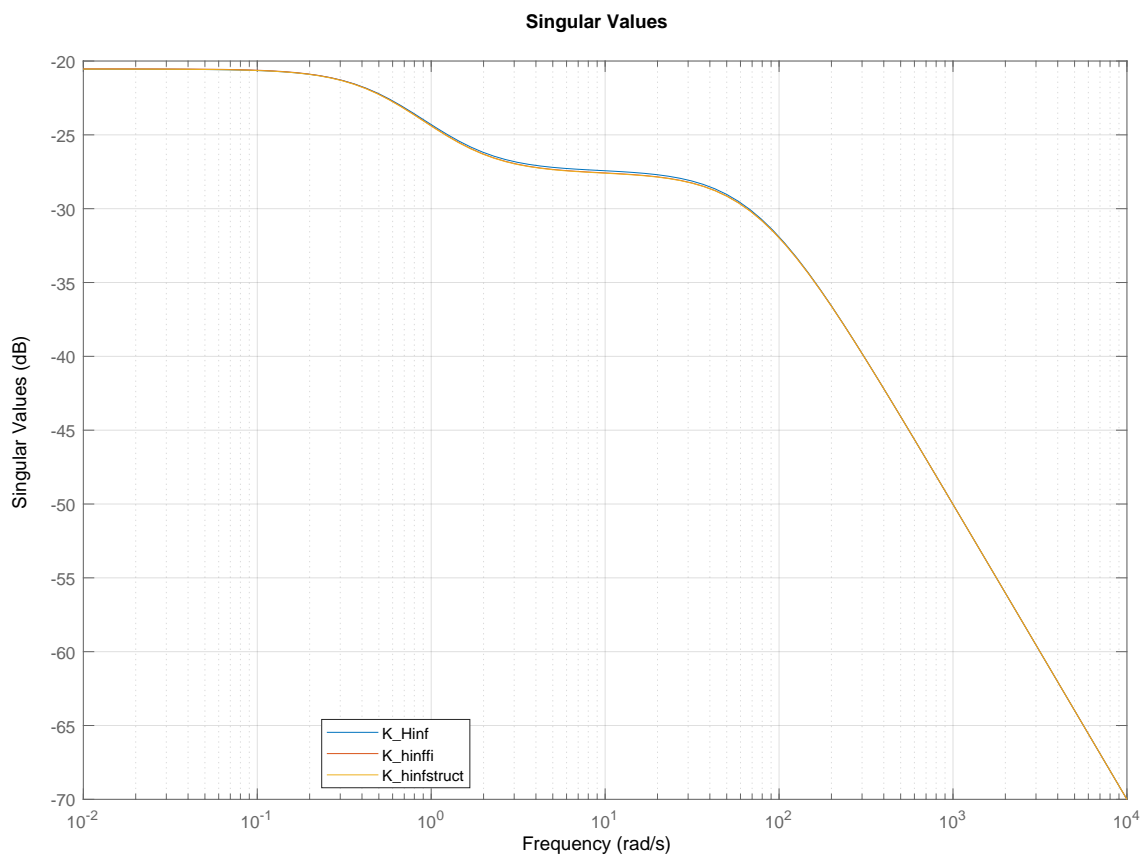
Jak je vidět z výsledků skriptu 9.4, všechny tři způsoby výpočtu mají takřka shodný výsledek. Rozdíly jsou pravděpodobně způsobeny použitím jiné numerické metody.

Z toho důvodu bude pro návrh statického stavového \mathcal{H}_∞ optimálního regulátoru v dalších částech práce používán příkaz `hinfstruct`.

	names	gamma	result		
1	-----	-----	-----	-----	-----
2					
3					
4	'K_Hinf'	0.73269	23.838	21.668	-34.055
5	'K_hinffi'	0.68923	24.003	21.963	-34.546
6	'K_hinfstruct'	0.65806	24.061	22.014	-34.624

Výpis 9.4: Výsledky zobrazené po spuštění kódu 9.3

Další příklad, který dokazuje, že se v tomto případě nejedná o náhodnou shodu, je v příloze D.



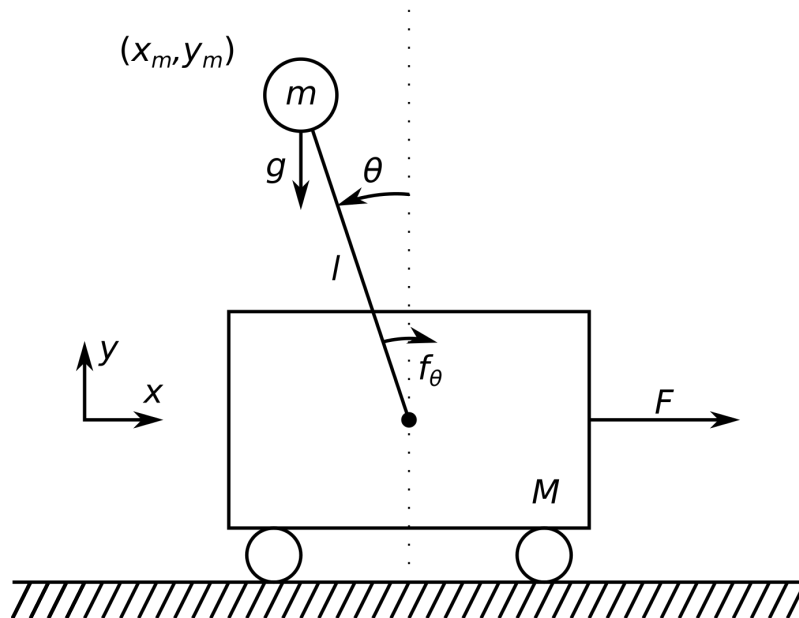
Obr. 9.2: Porovnání singulárních hodnot systémů s regulátorem navrženým třemi různými způsoby

10 Regulovaná soustava

Pro demonstraci LQG se často používá inverzní kyvadlo na vozíku.[13, 22] Tato úloha je dobře známá a splňuje všechny požadavky, které byly uvedeny v předchozích kapitolách. Proto je tato soustava vhodným modelem pro demonstraci LQG a srovnání s jinými typy řízení.

10.1 Model inverzního kyvadla na vozíku

Jelikož je model inverzního kyvadla na vozíku často používaný a známý, jeho odvození a popis jsou v příloze E. Odvozený linearizovaný model vozíku, který bude dále uvažován, je na obrázku 10.1 a popsán stavovým popisem 10.1.



Obr. 10.1: Schematický náčrt modelu inverzního kyvadla

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{gm}{M} & -\frac{F_{\theta}m}{M} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g(M+m)}{Ml} & -\frac{f_{\theta}(M+m)}{Ml} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{Ml} \end{bmatrix} F \quad (10.1)$$

Kde stavový vektor \vec{x} se skládá ze čtyř stavů. x je poloha vozíku, \dot{x} je rychlost vozíku, θ je natočení osy kyvadla ($\theta = 0$ je poloha, kdy je kyvadlo přímo nad vozíkem) a $\dot{\theta}$ je úhlová rychlost osy kyvadla. F je síla působící na vozík, vstup systému \vec{u} .

Další parametry byly zvoleny tak, aby bylo teoreticky možné takový vozík skutečně sestavit.

- $g = 10 \text{ ms}^{-2}$ - gravitační zrychlení,
- $m = 1 \text{ kg}$ - váha závaží kyvadla,
- $M = 5 \text{ kg}$ - váha vozíku,
- $l = 2 \text{ m}$ - délka kyvadla,
- $f_\theta = 1 [-]$ - třecí síla osy kyvadla

Výstupy soustavy nejsou uvedeny, protože je potřeba je nejprve zvolit. Lze předpokládat, že je možné měřit polohu vozíku s malou chybou a malým šumem. Je také možné spolehlivě měřit úhel natočení kyvadla, ale v rámci zmenšení počtu senzorů může být tento výstup rekonstruován, nebo úplně ignorován. Dále je možné předpokládat, že měření $\dot{\theta}$ a \dot{x} by bylo natolik nepřesné, že by mohlo kvalitu regulace zhoršit. [13]

Po provedení analýzy pozorovatelnosti je patrné, že všechny stavy jsou rekonstruovatelné i za předpokladu, že bude měřena pouze poloha vozíku x .

$$\mathbf{C} = [1 \ 0 \ 0 \ 0] \quad (10.2)$$

$$\mathbf{V} = \begin{bmatrix} \mathbf{A} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \mathbf{CA}^3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{gm}{M} & -\frac{f_T m}{M} \\ 0 & 0 & -\frac{f_T gm(M+m)}{M^2 l} & \frac{gm}{M} + \frac{f_T^2 m(M+m)}{M^2 l} \end{bmatrix} \quad (10.3)$$

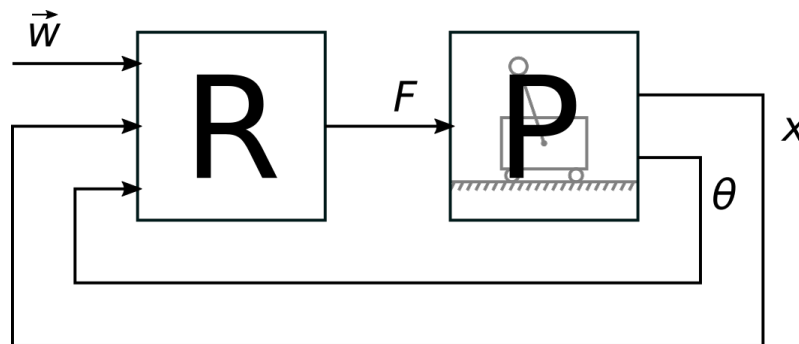
$$\vec{x} \in \mathbb{R}^4 = \mathbb{R}^n \quad (10.4)$$

$$\text{rank}(\mathbf{V}) = 4 = n \quad (10.5)$$

Kde \mathbf{C} je výstupní matice při měření pouze výchylky vozíku x , \mathbf{V} je matice pozorovatelnosti a n je počet stavů.

10.2 Cíl úlohy

Cílem úlohy je vytvořit regulátor R , který je schopen splnit požadavky na regulaci uvedené níže. Schematický náčrt požadované struktury je na obrázku 10.2.



Obr. 10.2: Schéma požadovaného řešení úlohy

Měření stavu θ je pro řešení problému možné, ale absence tohoto senzoru bude považována za výhodu.

Prvním a nejdůležitějším požadavkem je samozřejmě udržet kyvadlo ve vzpřímené poloze, a to i za předpokladu nenulové počáteční podmínky. Dalším regulačním požadavkem je možnost nastavit polohu vozíku pomocí vstupu $\vec{w} = w_x$. Posledním požadavkem je akční zásah menší než 30 N.

Splnění požadavků bude testováno simulovaným pokusem sestávajícím ze dvou částí. V první části bude testována schopnost udržení kyvadla ve vzpřímené poloze použitím nenulové počáteční podmínky natočení osy kyvadla. Požadovaná hodnota polohy vozíku odpovídá počátečnímu stavu jeho polohy. Ostatní počáteční stavy regulované soustavy jsou nulové. Všechny počáteční podmínky regulátoru budou nulové. Bude-li regulátor obsahovat rekonstruktor stavů, budou jeho počáteční podmínky také nulové. Počáteční podmínky jsou shrnuty rovnicemi 10.6 až 10.8.

$$\vec{w} = w_x = x_0 = 0 \quad (10.6)$$

$$\vec{x}_0 = [x_0 \ \dot{x}_0 \ \theta_0 \ \dot{\theta}_0]^T = [0 \ 0 \ 0, 15 \ 0]^T \quad (10.7)$$

$$\vec{\hat{x}}_0 = [\hat{x}_0 \ \hat{\dot{x}}_0 \ \hat{\theta}_0 \ \hat{\dot{\theta}}_0]^T = [0 \ 0 \ 0 \ 0]^T \quad (10.8)$$

Druhá část navazuje přímo na první část tím, že se změni žádaná poloha vozíku $w_x = 10$.

Simulace každé části poběží 20 s. Je-li regulace úspěšná, bude po této době poloha vozíku ustálena na poloze $w_x = x \pm 1$ m a kyvadlo bude ve vzpřímené poloze $\theta = 0 \pm 0, 1$ rad. Akční zásah nesmí během simulace překročit limit $|F| \leq 30$ N. Dřívější dosažení podmínek je považováno za výhodu.

11 Řešení úlohy

V následujících kapitolách budou navrženy tři různé způsoby řešení výše popsané úlohy.

11.1 \mathcal{H}_2 LQG řešení úlohy

Jelikož definice úlohy neposkytuje žádné informace o předpokládaném šumu, jsou matice \mathbf{Q} , \mathbf{R} , \mathbf{V}_d a \mathbf{V}_n nastaveny experimentálně tak, aby výsledný LQG regulátor splnil požadavky na řešení úlohy.

$$\begin{aligned}\mathbf{Q} &= \text{diag}\{1, 10, 1, 1\}, & \mathbf{R} &= 0.1 \\ \mathbf{V}_d &= \text{diag}\{1, 10, 1, 1\}, & \mathbf{V}_n &= 10\end{aligned}\quad (11.1)$$

Jak je patrné z rozměrů matic 11.1, je použito pouze měření polohy x . Výstup θ použit není, a to nejen proto, že použití menšího počtu senzorů je považováno za výhodu. Dalším důvodem je, že je-li pouze jeden měřitelný stav, lze převést stavový rekonstruktor na problém stavové regulace s jedním vstupem. Pro stavově regulované systémy s jedním vstupem vzniká regulátor (popsáno v kapitole 9.3), jehož rozměry jsou $n \times 1$, kde n je počet stavů. Lze tedy říci, že místo regulátoru v maticovém tvaru vznikne regulátor ve tvaru vektoru.¹ Velká výhoda je, že problém tohoto tvaru má pouze jedno řešení [4], bude tedy možné porovnávat regulátor s ostatními.

Regulátor i stavový rekonstruktor byly navrženy v prostředí MATLAB pomocí příkazu `lqr`.

```
1 K_LQR = lqr(A,B,Q,R);  
2 L_Kalman = (-lqr(A',C',Vd,Vn))';
```

Výpis 11.1: Návrh stavového regulátoru a rekonstruktoru

Výsledné hodnoty jsou následující:

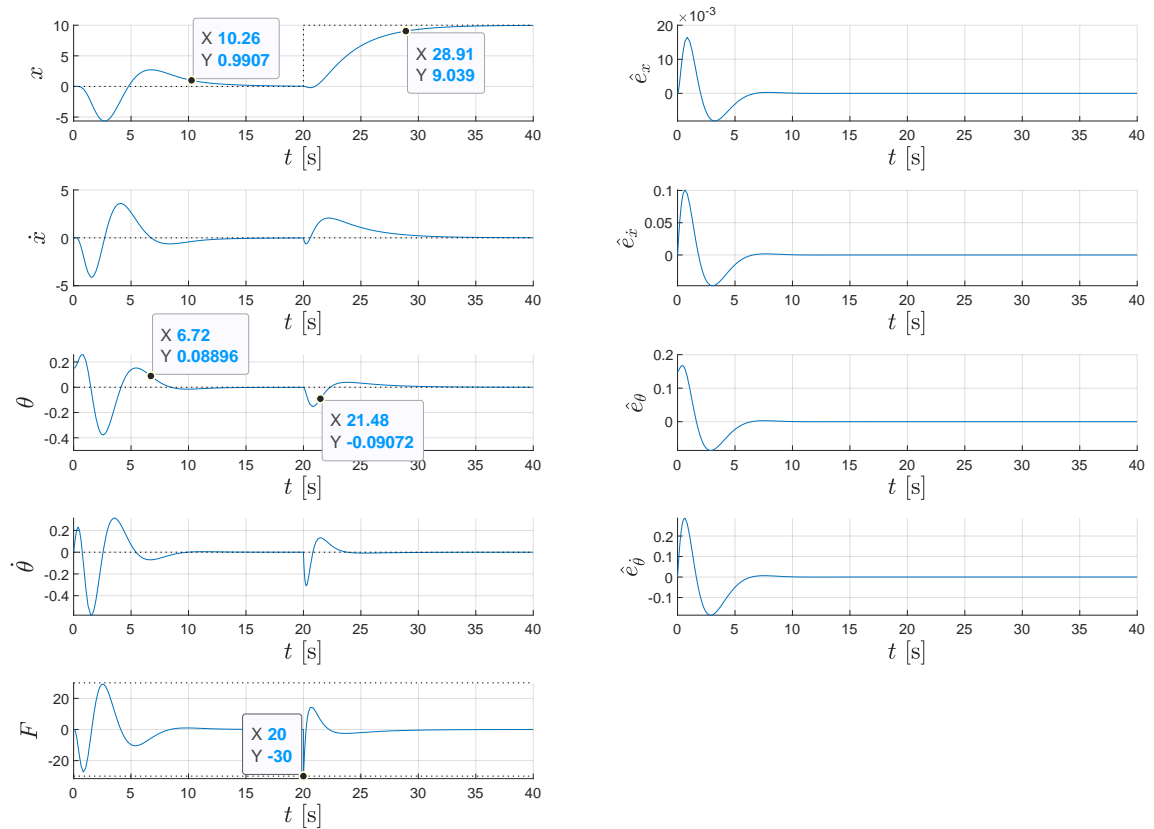
$$\mathbf{K}_{LQR} = [3, 1623 \ 14, 771 \ -246, 87 \ -93, 98] \quad (11.2)$$

$$\mathbf{L}_{Kalman} = [-5, 7895 \ -16, 7092 \ -24, 5167 \ -53, 1441]^T \quad (11.3)$$

Na obrázku 11.1 je vidět výsledek simulovaného pokusu. Grafy v pravé části ukazují, že chyba odhadovaných stavů \vec{e} je po 7s velmi malá, takže rozdíl mezi $\vec{\hat{x}}$ a \vec{x} je zanedbatelný. Graf v levém dolním rohu ukazuje, že během celé simulace nepřekročil akční zásah $F = 30$ N. Z grafu výchylky osy θ je vidět, že úvodní výchylka

¹V textu bude dále používáno maticové označení pro udržení jednotného značení v celé práci.

je vyregulována na hodnotu menší než 0,1 rad za 6,72 s. Při změně polohy vozíku je výchylka vyregulována za 1,48 s. Regulace polohy vozíku také splňuje požadavky. První část simulace je vyregulována za 10,26 s, druhá část za 8,91 s.



Obr. 11.1: Výsledek simulace navrženého \mathcal{H}_2 LQG regulátoru

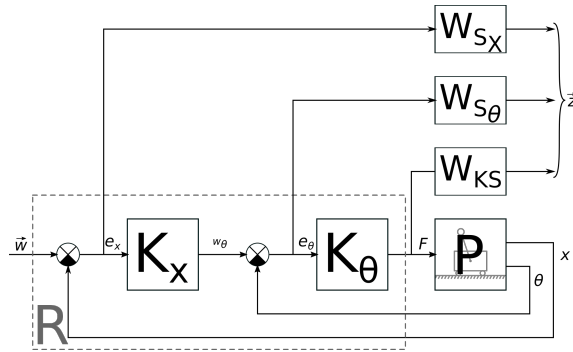
Na základě simulace se tedy dá usoudit, že regulace LQG regulátorem, považmo \mathcal{H}_2 optimálním stavovým regulátorem s \mathcal{H}_2 optimální stavovou rekonstrukcí, je úspěšná.

R	$ F \leq 30 \text{ N}$	První část		Druhá část		Měřeno θ	
		$x = w_x \pm 1 \text{ m}$	$\theta = 0 \pm 0,1 \text{ rad}$	$ F \leq 30 \text{ N}$	$x = w_x \pm 1 \text{ m}$		$\theta = 0 \pm 0,1 \text{ rad}$
\mathcal{H}_{2LQG}	✓	10,26 s	6,72 s	✓	8,91 s	1,48 s	Ne

Tab. 11.1: Tabulka výsledků simulace \mathcal{H}_{2LQG} regulátoru

11.2 \mathcal{H}_∞ řešení úlohy

Pravděpodobně standardním návrhem \mathcal{H}_∞ regulátoru pro tuto úlohu by byla metoda postupného návrhu kaskádní regulace. Kaskádní regulátor se skládá ze dvou, nebo více regulátorů, kaskádně zařazených za sebe, z nichž každý reguluje jednu část obvodu. Výchylka θ bude tedy regulována regulátorem \mathbf{K}_θ a poloha vozíku x regulátorem \mathbf{K}_x . Výsledné zapojení včetně vah je zobrazeno na obrázku 11.2.



Obr. 11.2: Schematické zobrazení kaskádního Hinf optimálního regulátoru

Prvním krokem je návrh vnitřního regulátoru výchylky kyvadla. Pro tento regulátor bude použit výstup θ .

Požadavky na regulátor omezují zejména rychlost regulace a velikost akčního zásahu. Proto byl regulátor syntetizován s vahou \mathbf{W}_{S_θ} na citlivostní funkci $S = \vec{e}_\theta$, vahou \mathbf{W}_{KS} na funkci $KS = F$.

$$W_{S_\theta}(s) = \frac{0,3984s + 0,05315}{s + 0,03348} \quad (11.4)$$

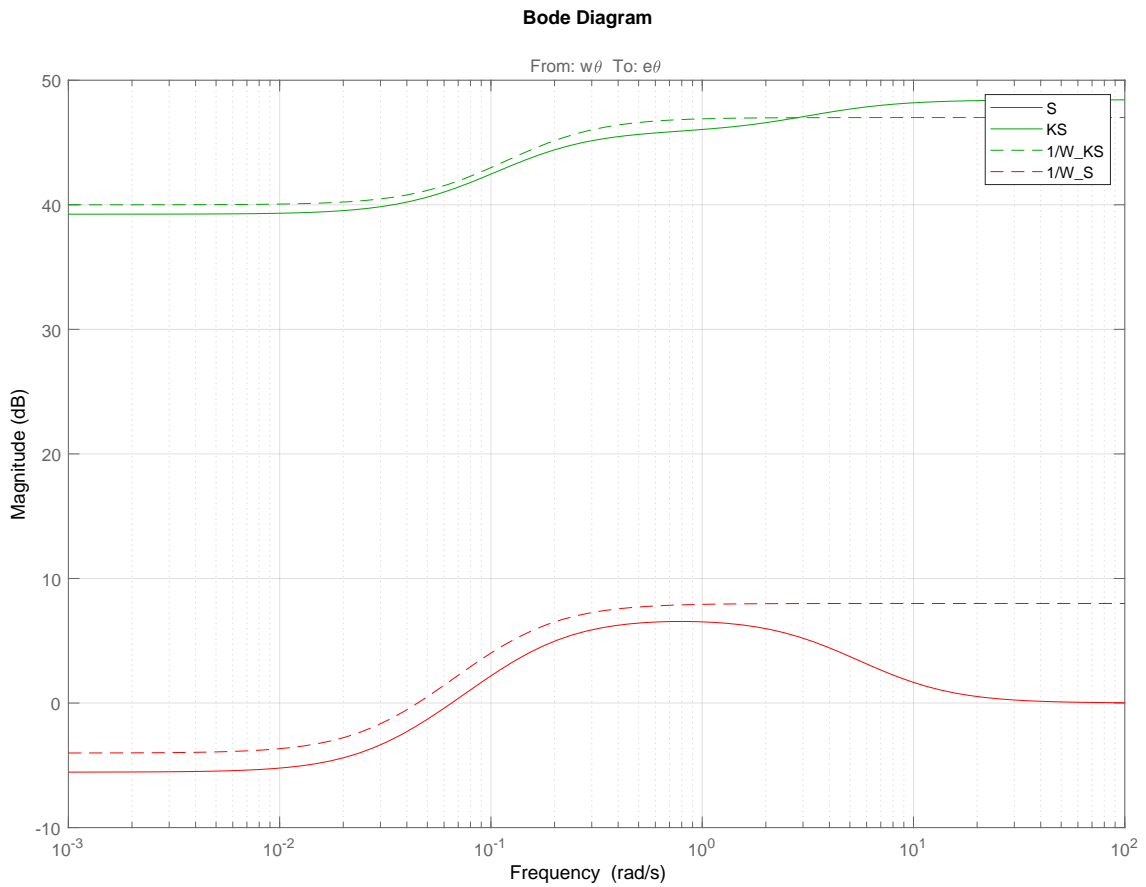
$$W_{KS}(s) = \frac{0,004464s + 0,0007817}{s + 0,07817} \quad (11.5)$$

Regulátor třetího řádu \mathbf{K}_θ byl syntetizován příkazem `hinfsyn`.²

$$\mathbf{K}_\theta = \begin{cases} \vec{x}_{K_\theta} = \begin{bmatrix} 0,1153 & -1,01 & 0,4804 \\ 0,984 & -6,076 & 3,494 \\ -0,6313 & 3,852 & -2,287 \end{bmatrix} \vec{x}_{K_\theta} + \begin{bmatrix} 2,909 \\ 9,687 \\ -6,165 \end{bmatrix} e_\theta \\ F = \begin{bmatrix} 20,4 & -113,8 & 59,8 \end{bmatrix} \vec{x}_{K_\theta} + 263,8 e_\theta \end{cases} \quad (11.6)$$

²Kompletní zdrojový kód návrhu je k nahlédnutí v příloze F.

Výsledná $\gamma = 1,2434$ ukazuje, že se požadavkům nepodařilo vyhovět, jak je také vidět na obrázku 11.3. Singulární hodnoty ale vykazují předpokládané vlastnosti, proto je návrh považován za úspěšný.



Obr. 11.3: Singulární charakteristiky signálů S a KS v porovnání s vahami $W_{S_\theta}(s)$ a $W_{KS}(s)$ při návrhu regulátoru odchyly \mathbf{K}_θ

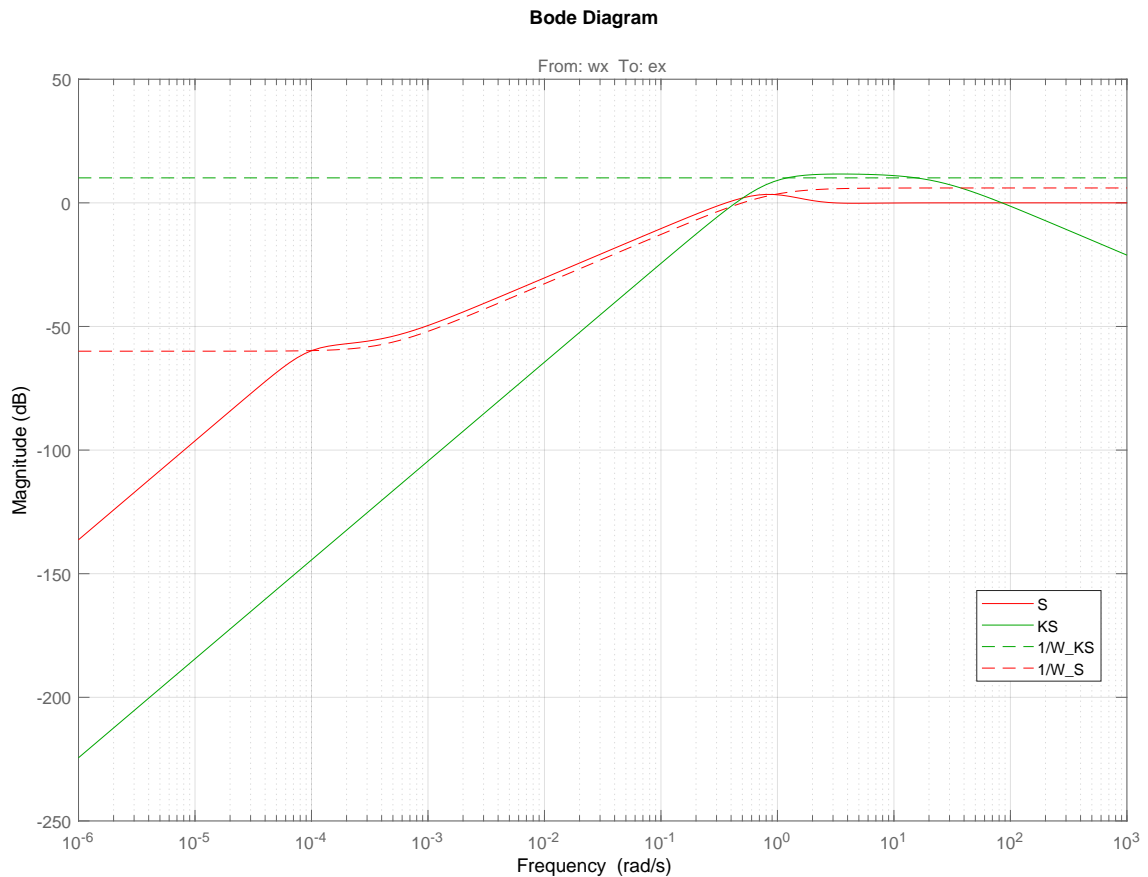
Dalším krokem je návrh regulátoru polohy, který se navrhuje na zapojení včetně regulátoru výchyly. Regulátor polohy \mathbf{K}_x byl optimalizován přes váhy \mathbf{W}_{S_x} a tentokrát statickou \mathbf{W}_{KS} .

$$W_{S_\theta}(s) = \frac{0,5s + 0,433}{s + 0,000433} \quad (11.7)$$

$$W_{KS}(s) = 0,3125 \quad (11.8)$$

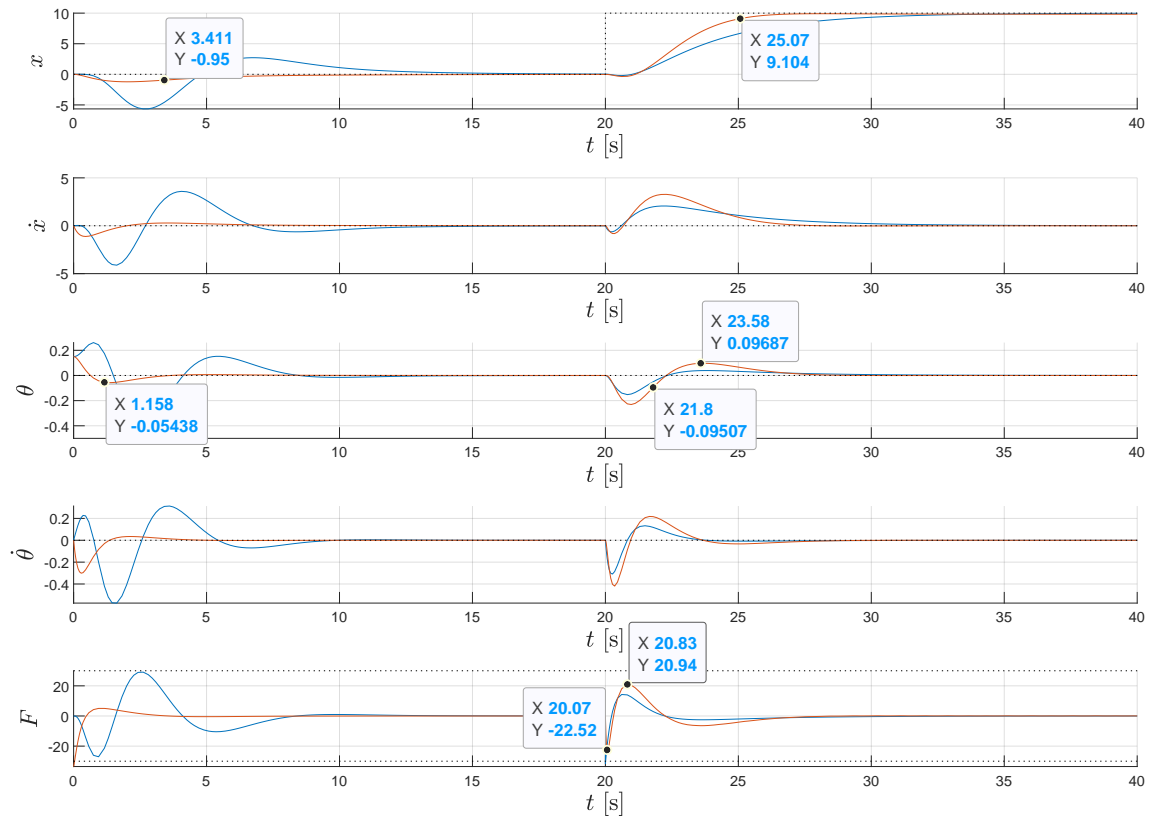
Regulátor osmého řádu \mathbf{K}_x byl také syntetizován příkazem `hinfsyn`. Výsledná hodnota $\gamma = 1,3131$ opět vykazuje, že podmínkám stanoveným vahami nebylo vyhověno.

$$\mathbf{K}_\theta = \left\{ \begin{array}{l}
\mathbf{A}_{K_{\theta 1}} = \begin{bmatrix} -0,512 & 0,3509 & 4,34 & 1,732 \\ 52 & 68,24 & -480,6 & -191,9 \\ 9,399 \cdot 10^{-14} & 1,191 \cdot 10^{-13} & -7,966 \cdot 10^{-13} & 1 \\ 26 & 34,12 & -235,3 & -96,46 \\ 7,356 \cdot 10^{-13} & 9,324 \cdot 10^{-13} & -6,234 \cdot 10^{-12} & -2,488 \cdot 10^{-12} \\ 2,867 & 3,762 & -26,61 & -10,57 \\ 9,548 & 12,53 & -88,61 & -35,2 \\ -6,076 & -7,974 & 56,4 & 22,4 \end{bmatrix} \\
\mathbf{A}_{K_{\theta 2}} = \begin{bmatrix} -0,512 & 0,3509 & 4,34 & 1,732 \\ 52 & 68,24 & -480,6 & -191,9 \\ 9,399 \cdot 10^{-14} & 1,191 \cdot 10^{-13} & -7,966 \cdot 10^{-13} & 1 \\ 26 & 34,12 & -235,3 & -96,46 \\ 7,356 \cdot 10^{-13} & 9,324 \cdot 10^{-13} & -6,234 \cdot 10^{-12} & -2,488 \cdot 10^{-12} \\ 2,867 & 3,762 & -26,61 & -10,57 \\ 9,548 & 12,53 & -88,61 & -35,2 \\ -6,076 & -7,974 & 56,4 & 22,4 \end{bmatrix} \\
\mathbf{B}_{K_{\theta 1}} = \begin{bmatrix} -0,007651 & -5,002 \cdot 10^{-07} & 1,405 \cdot 10^{-15} & -1,093 \cdot 10^{-16} \end{bmatrix}^T \\
\mathbf{B}_{K_{\theta 2}} = \begin{bmatrix} 0,5 & -3,119 \cdot 10^{-14} & -1,525 \cdot 10^{-14} & -1,36 \cdot 10^{-14} \end{bmatrix}^T \\
\mathbf{C}_{K_{\theta 2}} = \begin{bmatrix} 0,9856 & 1,293 & -8,147 & -3,634 & -0,6492 & -0,07734 & 0,4314 & -0,2267 \end{bmatrix} \\
\vec{x}_{K_x} = \begin{bmatrix} \mathbf{A}_{K_{\theta 1}} & \mathbf{A}_{K_{\theta 2}} \end{bmatrix} \vec{x}_{K_x} + \begin{bmatrix} \mathbf{B}_{K_{\theta 1}} \\ \mathbf{B}_{K_{\theta 2}} \end{bmatrix} e_\theta \\
\vec{w}_\theta = \mathbf{C}_{K_\theta} \vec{x}_{K_x}
\end{array} \right. \quad (11.9)$$



Obr. 11.4: Charakteristiky singulárních hodnot signálů S a KS v porovnání s vahami $W_{S_x}(s)$ a $W_{KS}(s)$ při návrhu regulátoru polohy \mathbf{K}_x

Jak je vidět z obrázku 11.5, \mathcal{H}_∞ kaskádní regulátor vykazuje o poznání lepší vlastnosti než \mathcal{H}_2 optimální LQG regulátor. Maximální akční zásah je takřka o třetinu menší, výchylka kyvadla je při změně žádané hodnoty polohy w_x větší, ale rychlost regulace je srovnatelná. Větší výchylka kyvadla umožnila výrazně rychlejší regulaci polohy. Navíc na začátku experimentu není problém s kmitáním z důvodu ustalování stavového rekonstruktoru. Velkou výhodou \mathcal{H}_∞ návrhu je, že lze poměrně jednoduše měnit konkrétní vlastnosti regulátoru jednoduchou úpravou vah. Výsledky jsou shrnuty v tabulce 11.2.



Obr. 11.5: Srovnání \mathcal{H}_2 optimálního LQG regulátoru (modře) a \mathcal{H}_∞ optimálního kaskádního regulátoru (červeně)

\mathbf{R}	$ F \leq 30 \text{ N}$	První část		Druhá část		Měřeno θ	
		$x = w_x \pm 1 \text{ m}$	$\theta = 0 \pm 0, 1 \text{ rad}$	$ F \leq 30 \text{ N}$	$x = w_x \pm 1 \text{ m}$		$\theta = 0 \pm 0, 1 \text{ rad}$
\mathcal{H}_{2LQG}	✓	10,26 s	6,72 s	✓	8,91 s	1,48 s	Ne
\mathcal{H}_∞	✓	3,42 s	1,16 s	✓	5,07 s	1,8 s	Ano

Tab. 11.2: Tabulka výsledků simulace kaskádního \mathcal{H}_∞ optimálního regulátoru v porovnání s \mathcal{H}_{2LQG}

Tento návrh má samozřejmě i své nevýhody, zejména velikost regulátoru a s ní spojenou výpočetní náročnost. Další nevýhodou je potřeba měření výstupu θ .³

Je jasné, že není možné regulátory mezi sebou jednoznačně porovnávat. Výsledky simulace \mathcal{H}_∞ regulátoru vykazují výrazně lepší chování ve srovnání s LQG, ale nevýhody tohoto regulátoru v podobě výpočetní náročnosti jsou značné.

Obecné srovnání vlastností tedy není možné. Z důvodu velké odlišnosti i způsobu návrhu obou řešení, nebudou mezi sebou dále porovnávány.

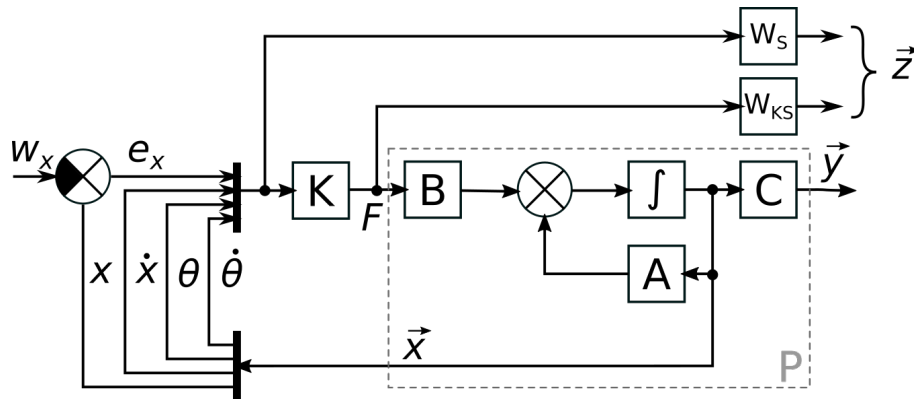
³Výstup θ by samozřejmě bylo možné rekonstruovat a regulátor by mohl použít signál $\hat{\theta}$. Výpočetní náročnost by tím ale ještě více vzrostla, tato možnost tedy není uvažována jako praktická.

11.3 \mathcal{H}_∞ LQG řešení úlohy

Jelikož předchozí pokus o srovnání \mathcal{H}_2 a \mathcal{H}_∞ regulátorů selhal z důvodu úplné odlišnosti, bude nyní navržen \mathcal{H}_∞ regulátor struktury LQG. Na rozdíl od minulého řešení bude navržen statický, strukturovaný regulátor, jehož vlastnosti se budou, díky stejné struktuře, dobře srovnávat s \mathcal{H}_2 optimálním LQG regulátorem. Tomu nic nebrání, jelikož, jak bylo uvedeno v kapitole 9.3, optimalizací \mathcal{H}_∞ normy lze navrhnout stavový regulátor a tedy, ze znalosti duality problémů, i stavový rekonstruktor. Z identické struktury obou řešení plyne, že oba regulátory budou mít stejnou výpočetní náročnost. Zároveň budou oba využívat pouze jeden výstup ze soustavy. Z těchto důvodů bude možné objektivně zhodnotit rozdíly, které vzniknou jinou metodou optimalizace regulátoru.

11.3.1 Návrh stavového \mathcal{H}_∞ optimálního regulátoru

Nejdříve je nutné zvolit zapojení, jehož \mathcal{H}_∞ norma bude optimalizována. Lze vycházet přímo ze zapojení popsáném na obrázku 8.3. Protože ale víme, že nebudou všechny stavy regulovány na nulu, byl přidán vstup $\vec{w} = w_x$. Vzniklo tak zapojení zobrazené na obrázku 11.6.⁴



Obr. 11.6: Schematické znázornění návrhu stavového regulátoru pomocí \mathcal{H}_∞ normy

Z obrázku je patrné, že optimalizované proměnné jsou citlivostní funkce S a funkce KS váhované pomocí \mathbf{W}_S a \mathbf{W}_{KS} .

Jsou-li obě dvě váhy blokově diagonální, bude váhován každý signál funkce S a KS zvlášť, analogicky vzhledem k \mathcal{H}_2 LQR návrhu, kde se diagonální maticí \mathbf{W}_x penalizuje \mathcal{H}_2 norma stavů a diagonální maticí \mathbf{W}_u norma akčního zásahu.

⁴Mohlo by se zdát zvláštní, že vstup je přičítán se záporným znaménkem. To proto, že byl zvolen tvar regulátoru $\vec{x} = (\mathbf{A} - \mathbf{BK})\vec{x}$. Znaménko bude tedy v samotném regulátoru změněno podruhé.

Z důvodu zachování analogie mezi \mathcal{H}_2 LQG a \mathcal{H}_∞ LQG návrhem, bude každý signál citlivostní funkce S váhován zvlášť, proto lze váhu $W_S(s)$ rozdělit na více váhovacích funkcí

$$\begin{aligned} W_S(s) &= \text{diag} \left\{ W_{S_x}(s), W_{S_{\dot{x}}}(s), W_{S_\theta}(s), W_{S_{\dot{\theta}}}(s) \right\} = \\ &= \text{diag} \left\{ \frac{0,5s + 0,1992}{s + 0,001992}, \frac{0,9s + 0,08718}{s + 0,0008718}, \frac{0,9s + 26,15}{s + 0,2615}, \frac{0,9s + 8,718}{s + 0,08718} \right\} \end{aligned} \quad (11.10)$$

$$W_{KS}(s) = 0,3125, \quad (11.11)$$

kde $W_{S_x}(s)$ značí přenos mezi signálem x a váhovaným signálem z_x . Obdobně to platí i pro ostatní signály.

Pro syntézu regulátoru $\mathbf{K}_{\mathcal{H}_\infty}$ byl použit příkaz `hinfstruct`.⁵ Z výsledné hodnoty $\gamma = 0,99333$ plyne, že návrh byl úspěšný.

$$\mathbf{K}_{LQR} = [3,1623 \ 14,771 \ -246,87 \ -93,98] \quad (11.12)$$

$$\mathbf{K}_{\mathcal{H}_\infty} = [2,5416 \ 12,679 \ -225,99 \ -79,42], \quad (11.13)$$

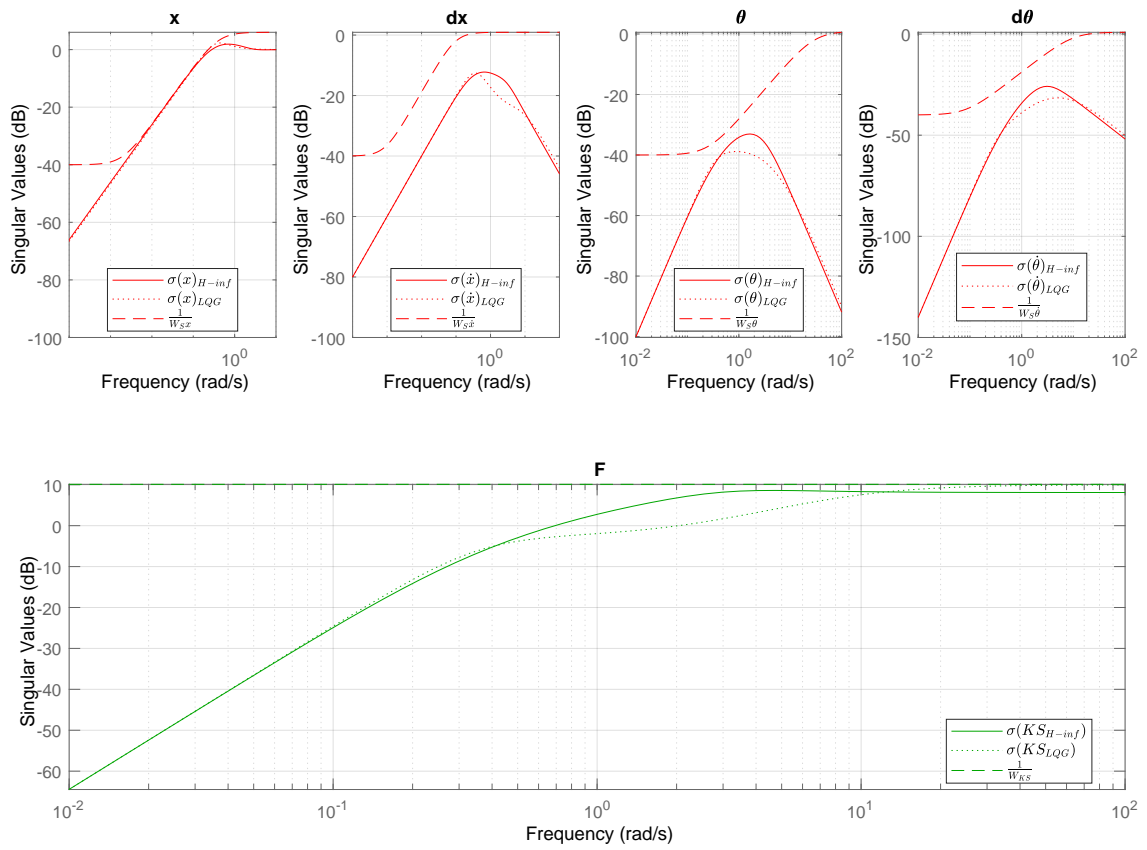
Při porovnání výsledného regulátoru $\mathbf{K}_{\mathcal{H}_\infty}$ s \mathcal{H}_2 optimálním LQG regulátorem \mathbf{K}_{LQR} , jehož výsledek je možné vidět v rovnici 11.2, je vidět pouze malý rozdíl, který může být způsoben jinou výpočetní metodou.⁶

Podobnost regulátorů není náhodná. Váhy byly totiž navrženy tak, aby byl výsledek co nejpodobnější, což je vidět na obrázku 11.7. Je vidět, že tam, kde se singulární hodnoty \mathcal{H}_2 LQR a \mathcal{H}_∞ LQR liší, to umožňují váhy. Teoreticky by bylo možné zvolit váhy podobnější singulárním hodnotám \mathcal{H}_2 LQR a tak dosáhnout ještě větší podobnosti.

Jediný případ, kde váha neodpovídá singulárním hodnotám \mathcal{H}_2 LQR, je u \mathbf{W}_{KS} , kde je váha nastavena přísněji za účelem zmenšení akčního zásahu, který je u \mathcal{H}_2 LQR na hraně přijatelnosti. Akční zásah je tímto opravdu zmenšen, jak je vidět na obrázku 11.10.

⁵Kompletní skript s návrhem stavového regulátoru je možné najít v příloze G

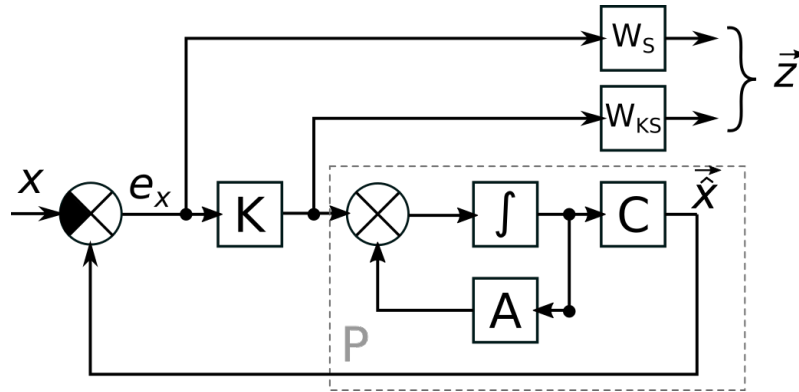
⁶Příkaz `lqg` používá řešení pomocí algebraických Ricattiho rovnic, zatímco příkaz `hinfstruct` řeší problém metodou popsanou v [21, 23].



Obr. 11.7: Charakteristiky singulárních hodnot signálů S a KS v porovnání s příslušnými vahami a charakteristikami singulárních hodnot ekvivalentních signálů \mathcal{H}_2 optimálního LQR.

11.3.2 Návrh stavového \mathcal{H}_∞ optimálního rekonstruktoru

Vzhledem k tomu, že stavový rekonstruktor je speciálním případem stavového regulátoru,⁷ musí být možné návrh stavového rekonstruktoru převést na návrh stavového regulátoru také v případě \mathcal{H}_∞ optimalizace. Podrobné odvození je možné najít v příloze I. Vzniká tak schematické zapojení, ne nepodobné schematickému zapojení pro návrh stavového regulátoru 11.6.



Obr. 11.8: Schematické znázornění návrhu stavového rekonstruktoru pomocí \mathcal{H}_∞ normy

Opět vznikají analogické váhy k \mathcal{H}_2 návrhu. Váha \mathbf{W}_S penalizuje regulační odchylku analogicky k \mathbf{V}_n a váha \mathbf{W}_{KS} penalizuje akční zásah analogicky k \mathbf{V}_d .

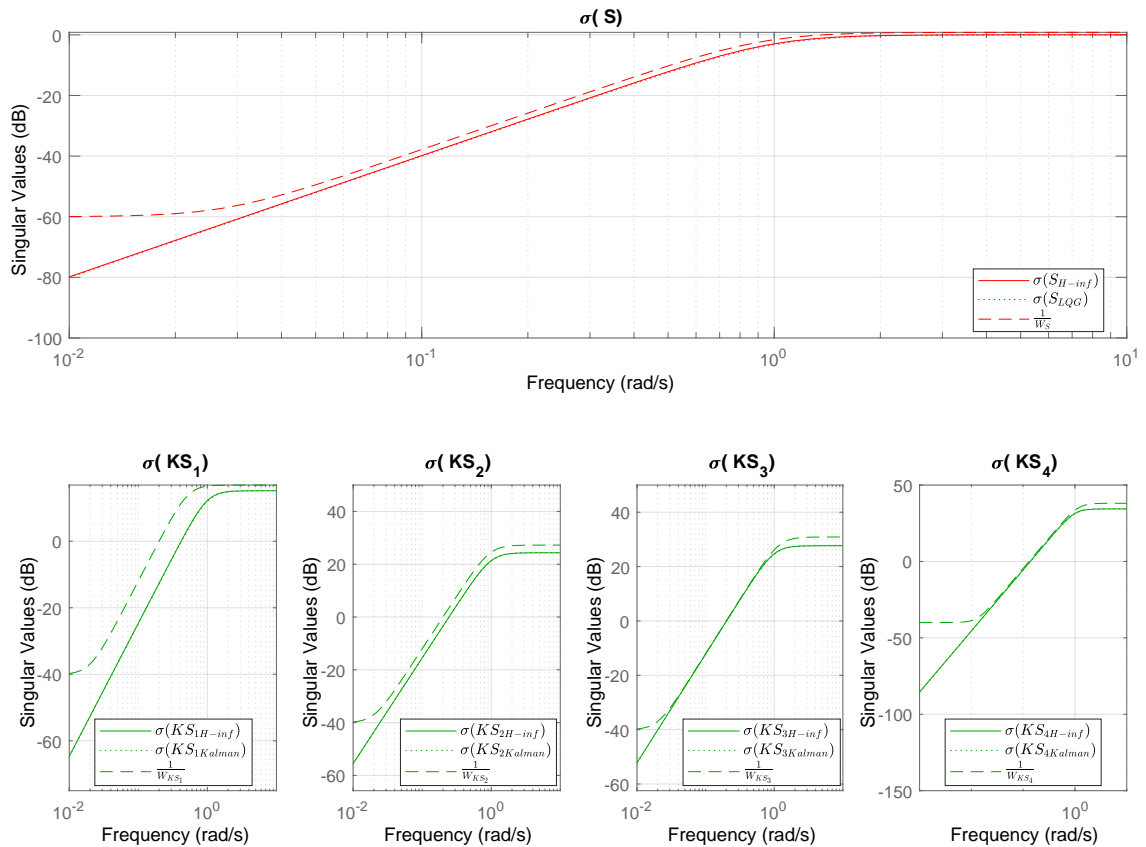
Váhy opět tvoří diagonální matici operátorových přenosů.

$$W_{KS}(s) = \text{diag} \left\{ \frac{0,1429s^2 + 0,1064s + 0,03959}{s^2 + 0,02814s + 0,0003959}, \frac{0,04348s^2 + 0,05895s + 0,03996}{s^2 + 0,02827s + 0,0003996}, \right. \\ \left. \frac{0,02857s^2 + 0,0478s + 0,03999}{s^2 + 0,02828s + 0,0003999}, \frac{0,0125s^2 + 0,02055s + 0,0169}{s^2 + 0,01838s + 0,000169} \right\} \quad (11.14)$$

$$W_S(s) = \frac{0,9091s^2 + 1,191s + 0,7808}{s^2 + 0,03952s + 0,0007808} \quad (11.15)$$

Regulátor byl opět syntetizován pomocí příkazu `hinfstruct`.

⁷Toto tvrzení neplatí obecně, ale v tomto případě jsou splněny všechny potřebné podmínky [4].



Obr. 11.9: Charakteristiky singulárních hodnot signálů S a KS v porovnání s příslušnými vahami a charakteristikami singulárních hodnot ekvivalentních signálů \mathcal{H}_2 optimálního stavového rekonstruktoru.

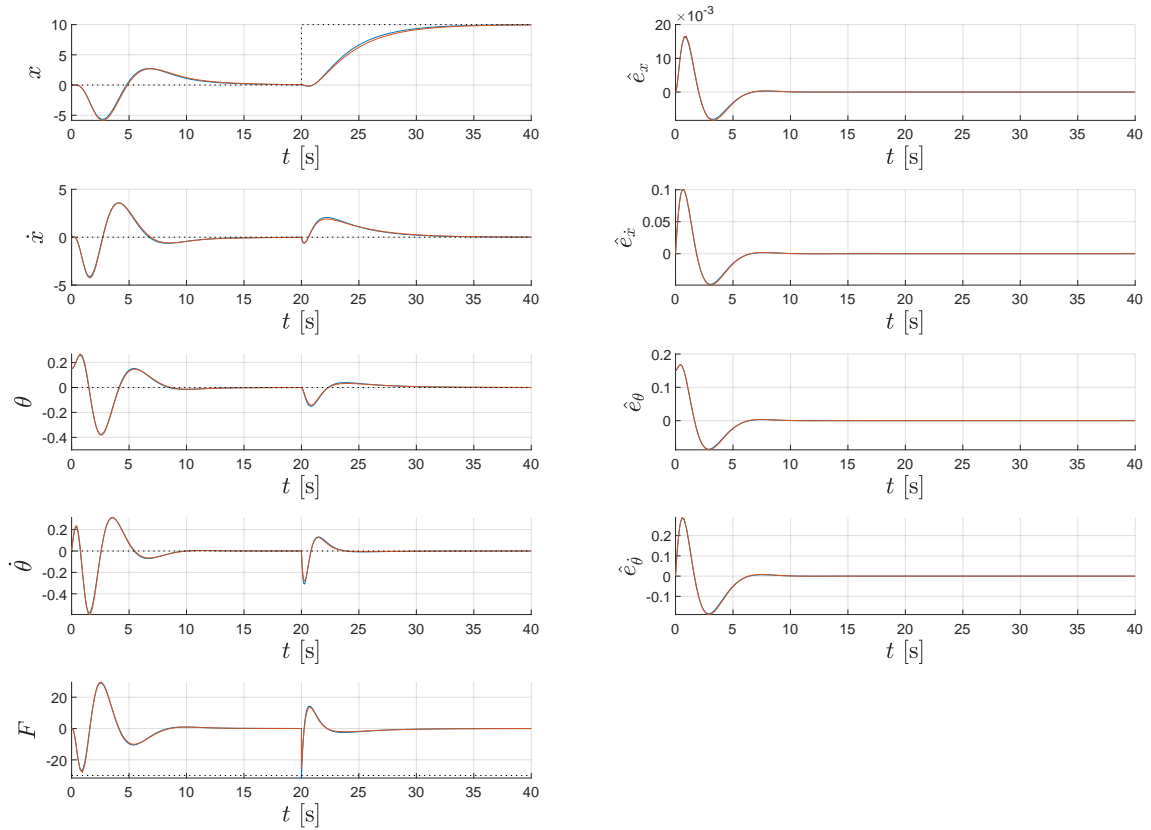
Z obrázku 11.9 je patrné, že váhy byly opět nastaveny tak, aby byl výsledek podobný \mathcal{H}_2 optimálnímu stavovému regulátoru. Tentokrát je rozdíl mezi výslednými maticemi ještě menší než v případě stavové regulace.

$$\mathbf{L}_{Kalman} = [-5, 7895 \quad -16, 7092 \quad -24, 5167 \quad -53, 1441]^T \quad (11.16)$$

$$\mathbf{K}_{\mathcal{H}_\infty} = [-5, 7367 \quad -16, 4547 \quad -24, 1299 \quad -52, 3085]^T \quad (11.17)$$

11.3.3 \mathcal{H}_∞ LQG

Zapojením regulátorů navržených v předchozích dvou kapitolách vzniká \mathcal{H}_∞ optimální LQG regulátor. Jelikož mezi výslednými maticemi \mathcal{H}_2 a \mathcal{H}_∞ optimálních regulátorů nejsou velké rozdíly, není velkým překvapením, že rozdíly v simulaci jsou zanedbatelné, jak je patrné z obrázku 11.10.



Obr. 11.10: Srovnání \mathcal{H}_2 optimálního LQG regulátoru (modře) a \mathcal{H}_∞ optimálního LQG regulátoru (červeně)

Úloha regulace inverzního kyvadla byla tedy úspěšně vyřešena i pomocí \mathcal{H}_∞ optimálního regulátoru.

12 Srovnání \mathcal{H}_2 a \mathcal{H}_∞ stavové regulace

Jelikož srovnávání stavové \mathcal{H}_2 optimální regulace s dynamickým \mathcal{H}_∞ regulátorem je prakticky nemožné kvůli velké odlišnosti, jak bylo ukázáno v kapitole 11, bude srovnávána pouze \mathcal{H}_2 a \mathcal{H}_∞ stavová regulace.

V předchozích kapitolách byly popsány rozdíly mezi definicí \mathcal{H}_2 a \mathcal{H}_∞ norem, ze kterých plyne rozdílný postup při návrhu vah, kterými jsou optimalizované signály penalizovány. Na příkladu bylo demonstrováno, že \mathcal{H}_∞ optimalizací je možné, navrhnout stejný, nebo velmi podobný regulátor jako \mathcal{H}_2 optimalizací.

Tato skutečnost vychází z následující teorie. Pakliže lze \mathcal{H}_2 optimalizací navrhnout stavový regulátor, potom je možné nastavit váhy pro \mathcal{H}_∞ optimalizaci tak, aby minimalizace \mathcal{H}_∞ normy stejné struktury vedla na identický výsledek.

Na příkladu bylo dokázáno, že stačí nastavit váhy podobné singulárním charakteristikám \mathcal{H}_2 optimálního zapojení. Teoreticky by bylo možné jako váhy použít přímo konkrétní přenosové funkce \mathcal{H}_2 optimálního zapojení. Jak plyne z definice \mathcal{H}_∞ optimalizace, singulární charakteristiky \mathcal{H}_∞ optimálního řešení jsou pro všechny hodnoty ω menší nebo rovny svým vahám. Jsou-li váhy identické s optimálním řešením, pak je řešení rovno vahám a \mathcal{H}_∞ norma optimalizovaného zapojení je rovna 1. Vzniknou tak dva identické regulátory, z nichž jeden je \mathcal{H}_∞ optimální a druhý \mathcal{H}_2 optimální.

Tato teorie není prokázána, pouze otestována experimentálně v kapitole 11.3. Její bližší prozkoumání by bylo vhodným tématem pro další práci.

Je-li teorie správná, není možné hodnotit rozdíl v kvalitě regulace, výsledné robustnosti, ani náročnosti výpočtu akčního zásahu, jelikož výsledné regulátory jsou identické, což lze říci i o výsledcích příkladu uvedeného v kapitole 11.

Bude-li teorie prokázána, pak vystává otázka, platí-li vztah v obou směrech. Tedy je-li možné navrhnout ekvivalentní \mathcal{H}_2 optimální statický regulátor k dříve navrženému \mathcal{H}_∞ optimálnímu statickému regulátoru. Pravděpodobně to možné nebude. \mathcal{H}_∞ návrh totiž použitím dynamických vah umožňuje více stupňů volnosti. Je tedy pravděpodobné, že bude možné vytvořit takový \mathcal{H}_∞ optimální stavový regulátor, ke kterému nebude možné navrhnout identický \mathcal{H}_2 optimální regulátor, čímž se teorie vyvrátí.

Náročnost výpočtu regulátoru ale rozdílná je. V kapitole 9 byl představen rozdíl v syntéze při použití LMI. Je ale nutné nezanedbat, že použité váhy při \mathcal{H}_∞ návrhu nejsou statické, proto rozměr počítaných LMI matic vzrostl a s ním i výpočetní náročnost syntézy. Náročnost syntézy \mathcal{H}_∞ regulátoru tedy bude vždy větší, než výpočetní náročnost syntézy \mathcal{H}_2 regulátoru. Výpočetní náročnost regulace ale zůstává neovlivněna a je identická pro oba přístupy.

Další nevýhodou \mathcal{H}_∞ optimalizace je právě volba dynamických vah. Zatímco při

optimalizaci \mathcal{H}_2 normy stačí navrhnout pouze $n_z = n_x + n_u$ ¹ statických vah (skalárů), při \mathcal{H}_∞ návrhu je třeba navrhnout stejný počet vah dynamických. To znamená, že pro \mathcal{H}_∞ návrh je potřeba lépe chápat vnitřní dynamiku soustavy a její požadované chování.

Použití dynamických vah má také svoje výhody. Má-li uživatel konkrétní představu o chování regulované soustavy, není problém tuto skutečnost zahrnout do návrhu. Obdobně lze do návrhu zahrnout i předpokládaný šum a nejistoty použitím robustního \mathcal{H}_∞ návrhu.

Obecně lze tedy říci, že návrh \mathcal{H}_2 optimalizací je jednodušší, zatímco \mathcal{H}_∞ optimalizace poskytuje uživateli více stupňů volnosti. Nejrozumnějším postupem je ale použití \mathcal{H}_2 optimalizace k prvotnímu návrhu regulátoru a \mathcal{H}_∞ optimalizace pro úpravu nevyhovujících vlastností \mathcal{H}_2 návrhu. Tento postup byl použit v návrhu \mathcal{H}_∞ optimálního stavového regulátoru v kapitole 11, kde byl více penalizován signál F za účelem zmenšení akčního zásahu, což se podařilo, jak je vidět na obrázku 11.10.

¹ n_z je počet optimalizovaných výstupů, n_x je počet stavů a n_u je počet vstupů soustavy.

Závěr

Tato práce se v první části věnovala popisu a analýze MIMO systémů pomocí lineární algebry a následně návrhu strukturovaných robustních regulátorů. Nejdříve byl popsán vztah mezi lineární algebrou a teorií řízení MIMO systémů. Následně byly popsány způsoby, které je možno využít pro analýzu MIMO systémů. Problematika směrovosti systémů byla rozebrána v kapitole 2.

Dále byl vysvětlen význam frekvenční charakteristiky singulárních hodnot pro popis MIMO systémů a návrh MIMO regulátorů a byly zdůvodněny jeho výhody oproti jiným charakteristikám, například v SISO systémech hojně používané Bodeho charakteristice.

V kapitole 4 byly popsány některé normy používané pro popis a analýzu MIMO systémů a jejich význam při návrhu MIMO regulátorů. Větší důraz byl kladen zejména na normy \mathcal{H}_2 a \mathcal{H}_∞ , které byly hojně používány v dalších kapitolách.

Další kapitoly byly věnovány stavové regulaci MIMO systémů (kapitola 5) a s ní spojenou stavovou rekonstrukcí (kapitola 6). Bylo vysvětleno, jak se navrhuje optimální stavová regulace a rekonstrukce a jak vzniká regulační struktura LQG.

Následně byla vysvětlena syntéza \mathcal{H}_2 optimálních regulátorů, její význam a použití. Bylo dokázáno, že LQG je speciálním případem \mathcal{H}_2 optimálního regulátoru, a vysvětleno, jak pomocí \mathcal{H}_2 optimalizace navrhnout optimální stavový regulátor, rekonstruktor, a tedy i LQG.

Dalším krokem bylo vysvětlení rozdílů mezi \mathcal{H}_2 a \mathcal{H}_∞ syntézou, které ukázalo, že návrh vah pro \mathcal{H}_∞ optimalizaci je obtížnější. Bylo nastíněno, jak funguje tvarování citlivostní a komplementární citlivostní funkce a možnost návrhu statického regulátoru.

Ve všech těchto kapitolách bylo zejména definováno značení a byly poskytnuty teoretické základy, kterých využívá část praktická.

Praktická část se nejprve věnuje implementaci syntézy \mathcal{H}_2 a \mathcal{H}_∞ optimálních stavových regulátorů pomocí lineárních maticových nerovností. Výsledky syntézy jsou srovnány s ostatními způsoby návrhu těchto regulátorů v prostředí MATLAB. Je ukázáno, že implementované metody dosahují stejných výsledků jako metody srovnávací.

Pro srovnání \mathcal{H}_2 a \mathcal{H}_∞ regulace, která byla obsahem další části, byla definována úloha regulace inverzního kyvadla. Z požadavků na řešení vyplynula omezení pro regulaci a také výhody a nevýhody při porovnávání navržených regulátorů.

Úloha byla nejdříve vyřešena pomocí LQG regulace, tedy \mathcal{H}_2 optimálního stavového regulátoru a rekonstruktoru. Následně byl navržen dynamický \mathcal{H}_∞ regulátor. Při pokusu o srovnání těchto dvou řešení se ukázalo, že přes dobrou definici úlohy není možné regulátory mezi sebou srovnávat pro úplnou rozdílnost jak kvality řízení

jednotlivých stavů a dynamiky, tak návrhové i výpočetní složitosti.

Z toho důvodu bylo navrženo druhé \mathcal{H}_∞ optimální řešení, kterým je statický, \mathcal{H}_∞ optimální regulátor struktury LQG. Záměrem bylo srovnávat regulátory stejné struktury, tedy stejné výpočetní náročnosti. Ukázalo se, že pro systém, kterým se úloha zabývá, lze \mathcal{H}_∞ optimalizací navrhnout stejný regulátor jako při použití \mathcal{H}_2 syntézy. Srovnávání jakýchkoliv rozdílů ve výsledku, ať už ohledně kvality řízení, splnění podmínek, nebo robustnosti, postrádá význam.

Proto byly srovnány pouze rozdíly ve složitosti a výpočetní náročnosti návrhu, kde se \mathcal{H}_2 norma ukázala jako méně složitá a návrhově méně výpočetně náročná. Větší složitost volby vah je ale pro \mathcal{H}_∞ návrh jeho velkou výhodou, je-li uživatel dobře seznámen s požadovaným chováním soustavy, protože může lépe a přesněji definovat svoje požadavky na výsledek.

Závěrem srovnání těchto dvou metod návrhu LQG tedy je, že zanedbáme-li výpočetní náročnost návrhu, která hraje ve většině případů malou roli, lze považovat návrh pomocí \mathcal{H}_∞ normy za lepší, než návrh pomocí \mathcal{H}_2 normy. Optimalizací \mathcal{H}_∞ normy lze totiž dosáhnout přinejmenším stejně dobrých výsledků jako při optimalizaci \mathcal{H}_2 normy, jelikož lze zvolit takové váhy, že výsledky budou totožné.

Dalším vhodným postupem v této práci by mohlo být teoretické dokázání experimentálně demonstrovaných výsledků a případné zobecnění na další struktury vyjímaje LQG pro soustavu s jedním vstupem a jedním výstupem.

Literatura

- [1] K. Kuttler. *Linear Algebra, Theory And Applications*. <https://saylor.org>, 2012.
- [2] G. Duan. *LMIs in control systems : analysis, design and applications*. CRC Press, Boca Raton, FL, 2013.
- [3] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. John Wiley & Sons, Inc., Hoboken, NJ, USA, 2005.
- [4] E. Ostertag. *Mono- and multivariable control and estimation : linear, quadratic and LMI methods*. Springer, Heidelberg New York, 2011.
- [5] J. Trump. A note on duality of control and observation. pages 7–11, 2014.
- [6] D. Van Thach and S. Lee. Lqg control design for a coupled ballbot dynamical system. In *2018 18th International Conference on Control, Automation and Systems (ICCAS)*, pages 666–670, 2018.
- [7] F. Tabassum and M. S. Rana. An optimal lqg-dc control for compensating voltage fluctuation in a single area power system. In *2019 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2)*, pages 1–4, 2019.
- [8] R. Banerjee and A. Pal. Stabilization of inverted pendulum on cart based on lqg optimal control. In *2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET)*, pages 1–4, 2018.
- [9] J. Doyle. Guaranteed margins for lqg regulators. *IEEE Transactions on Automatic Control*, 23(4):756–757, 1978.
- [10] L. Fortuna and M. Frasca. *Optimal and robust control: Advanced topics with MATLAB®*. 01 2012. doi:10.1201/b11660.
- [11] Robust control toolbox release notes. URL: <https://www.mathworks.com/help/robust/release-notes.html>.
- [12] H. T. Toivonen. *ROBUST CONTROL METHODS*. 2020-02-22. URL: <http://users.abo.fi/htoivone/courses/robust/>.
- [13] G. Dullerud and F. Paganini. *A Course in Robust Control Theory a convex approach*. 01 2000. doi:10.1007/978-1-4757-3290-0.

- [14] M. V. Khlebnikov, P. S. Shcherbakov, and V. N. Chestnov. Linear-quadratic regulator. i. a new solution. *Automation and Remote Control*, 76(12):2143–2155, December 2015. doi:10.1134/s0005117915120048.
- [15] Lmi solvers, 5.3.2020. URL: <https://www.mathworks.com/help/robust/lmis.html>.
- [16] M. Daniel. Optimalizace v řízení dynamických systémů, 2017.
- [17] J. Löfberg. Yalmip : A toolbox for modeling and optimization in matlab. In *In Proceedings of the CACSD Conference*, Taipei, T., 2004.
- [18] J. Löfberg. Pre- and post-processing sum-of-squares programs in practice. *IEEE Transactions on Automatic Control*, 54(5):1007–1011, 2009.
- [19] J. Löfberg. Automatic robust convex programming. *Optimization methods and software*, 27(1):115–129, 2012.
- [20] B. Borchers. Csdp, a c library for semidefinite programming. *Optimization Methods and Software*, 11(1-4):613–623, 1999. doi:10.1080/10556789908805765.
- [21] P. Apkarian and D. Noll. Nonsmooth h inf synthesis. *IEEE Transactions on Automatic Control*, 51(1):71–86, Jan 2006. doi:10.1109/TAC.2005.860290.
- [22] E. S. Varghese, A. K. Vincent, and V. Bagyaveereswaran. Optimal control of inverted pendulum system using PID controller, LQR and MPC. *IOP Conference Series: Materials Science and Engineering*, 263:052007, nov 2017. URL: <https://doi.org/10.1088%2F1757-899x%2F263%2F5%2F052007>, doi:10.1088/1757-899x/263/5/052007.
- [23] H-infinity tuning of fixed-structure controllers, 17.12.2019. URL: <https://www.mathworks.com/help/robust/ref/hinfstruct.html>.

Seznam příloh

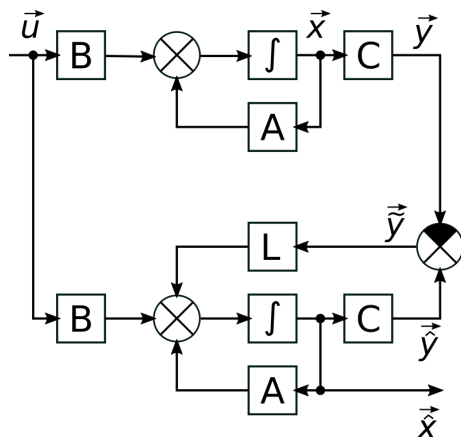
A	Obsah přiloženého CD	76
B	Odvození stavového popisu stavového rekonstruktoru	77
C	Další příklady výpočtu statického \mathcal{H}_2 optimálního regulátoru	78
D	Další příklady výpočtu statického \mathcal{H}_∞ optimálního regulátoru	80
E	Odvození matematického modelu inverzního kyvadla na vozíku	83
F	Návrh \mathcal{H}_∞ optimálního kaskádního regulátoru	88
G	Návrh \mathcal{H}_∞ optimálního stavového regulátoru	90
H	Návrh \mathcal{H}_∞ optimálního stavového rekonstruktoru	92
I	Převodění stavového rekonstruktoru na \mathcal{H}_∞ problém	94

A Obsah přiloženého CD

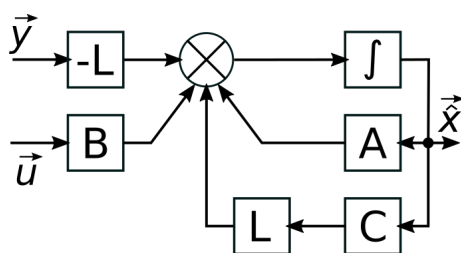
/	Kořenový adresář přiloženého CD
—	LMI Zdrojové soubory a příklady pro LMI
—	pendulumStateController Zdrojové a simulační soubory řešené úlohy
—	thesis Zdrojové soubory textu práce
—	thesis.pdf Práce v elektronickém formátu

B Odvození stavového popisu stavového rekonstruktoru

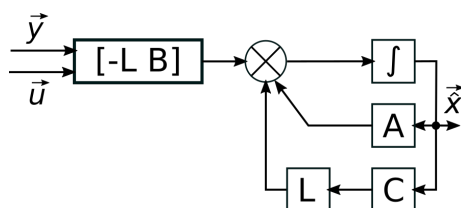
Odvození vychází z blokového diagramu stavového rekonstruktoru.



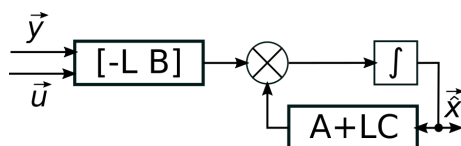
Odstraněním rekonstruované soustavy a rozpojením pravého sumačního bloku vzniká následující blokový diagram.



Nyní už stačí pouze sjednotit vstupy \vec{y} a \vec{u} do jednoho vektoru.



Následně sloučit bloky A , L a C .



Blokový diagram je možné popsat rovnicí

$$\dot{\hat{x}} = (A + LC) \hat{x} + [-L \ B] \begin{bmatrix} \vec{y} \\ \vec{u} \end{bmatrix}$$

C Další příklady výpočtu statického \mathcal{H}_2 optimálního regulátoru různými metodami

```
1 close all
2 clear all
3 format shortG
4 %% Definice systemu
5 A = [0 1; -1 1];
6 B = [0; 1];
7 C = eye(2);
8 D = [0; 0];
9 X0 = [1;0];
10 %% Definice vahovacich matic
11 Q = eye(2);
12 R = 1;
13 %% Definice obecného optimalizacního schématu
14 [numStateVars, ~] = size(B);
15 Bu = B;
16 Bw = B;
17 Cz = [Q^(1/2); 0 0];
18 Duz = zeros(numStateVars+length(R), length(R));
19 Duz(end) = R^(1/2);
20 %% LQR
21 LQR = -lqr(A,B,Q,R);
22 %% LMI
23 gamma = 1.639;
24 [numStateVars, numInputs] = size(Bw);
25 Z = sdpvar(numStateVars+1, numStateVars+1, 'symmetric');
26 X = sdpvar(numStateVars, numStateVars, 'symmetric');
27 W = sdpvar(numInputs, numStateVars, 'full');
28
29 constraintMatrix1 = [ -Z      Cz*X+Duz*W
30                      (Cz*X+Duz*W)' -X      ];
31 constraintMatrix2 = A*X + Bu*W + (A*X+Bu*W)' + Bw*Bw';
32 F = trace(Z) <= gamma^2;
33 F = [F, constraintMatrix1 <= eye(size(constraintMatrix1))*eps];
34 F = [F, constraintMatrix2 <= eye(size(constraintMatrix2))*eps];
35 optimize(F);
36 K_H2 = value(W)*inv(value(X));
37 %% Vysledky
38 figure
39 names = {'LQR'; 'K_H2'};
40 gamma = [NaN; gamma];
41 result = [LQR; K_H2];
42 T1 = table(names, gamma, result)
43 systems = [];
44 for i = 1:size(result)
45     sys = ss(A+Bu*result(i,:), Bu, Cz, Duz);
46     systems{i} = sys;
47     hold on
48     sigma(sys)
49 end
50 legend('LQR', 'K_H2', 'Location', 'Best')
51 grid on
52 %% Příklad 2
53 Q = 10*eye(2);
54 R = 0.1;
55 %% Definice obecného optimalizacního schématu
56 [numStateVars, ~] = size(B);
```

```

57 Bu = B;
58 Bw = B;
59 Cz = [Q^(1/2); 0 0];
60 Duz = zeros(numStateVars+length(R),length(R));
61 Duz(end) = R^(1/2);
62 %% LQR
63 LQR = -lqr(A,B,Q,R);
64 %% LMI
65 gamma = 1.092;
66 [numStateVars, numInputs] = size(Bw);
67 Z = sdpvar(numStateVars+1,numStateVars+1,'symmetric');
68 X = sdpvar(numStateVars,numStateVars,'symmetric');
69 W = sdpvar(numInputs,numStateVars,'full');
70
71 constraintMatrix1 = [ -Z      Cz*X+Duz*W
72                      (Cz*X+Duz*W)' -X      ];
73 constraintMatrix2 = A*X + Bu*W + (A*X+Bu*W)' + Bw*Bw';
74 F = trace(Z) <= gamma^2;
75 F = [F, constraintMatrix1 <= eye(size(constraintMatrix1))*eps];
76 F = [F, constraintMatrix2 <= eye(size(constraintMatrix2))*eps];
77 optimize(F);
78 K_H2 = value(W)*inv(value(X));
79 %% Vysledky
80 figure
81 names = {'LQR'; 'K_H2'};
82 gamma = [NaN; gamma];
83 result = [LQR; K_H2];
84 T1 = table(names, gamma, result)
85 systems = [];
86 for i = 1:size(result)
87     sys = ss(A+Bu*result(i,:),Bu,Cz,Duz);
88     systems{i} = sys;
89     hold on
90     sigma(sys)
91 end
92 legend('LQR', 'K_H2', 'Location', 'Best')
93 grid on

```

Výpis C.1: MATLAB skript s příklady výpočtu statického \mathcal{H}_2 optimálního regulátoru různými metodami

D Další příklady výpočtu statického \mathcal{H}_∞ optimálního regulátoru různými metodami

```

1  close all
2  clear all
3  format shortG
4  %% Definice systemu
5  A = [2 1 -2
6       1 -1 -3
7       4 0 -1];
8  Bu = [1; 0; 3];
9  Bw = [1; 0.2; -0.5];
10 %% Definice vahovacich matic
11 Cz = [2 1 -0.5];
12 Duz = 1;
13 Dwz = 0.05;
14 %% hinffi
15 [numStateVars, numInputs] = size(Bu);
16 P = ss(A, [Bw Bu], Cz, [Dwz Duz]);
17 [K_hinffi,~, gamma_hinffi] = hinffi(P,numInputs,[0.6865 Inf]);
18 K_hinffi = K_hinffi(1:numInputs,1:numStateVars);
19 %% hinfstruct
20 Plant = ss(A, [Bw Bu], [Cz; eye(3)], [Dwz Duz; zeros(3,2)]);
21 Plant.InputName = {'w', 'u'};
22 Plant.OutputName = {'z', 'x1', 'x2', 'x3'};
23 Plant.name = 'P';
24
25 K_hinfstruct_tunable = tunableGain('K_hinfstruct_tunable', 1, 4);
26 K_hinfstruct_tunable.InputName = {'x1', 'x2', 'x3', 'w'};
27 K_hinfstruct_tunable.OutputName = {'u'};
28
29 system = connect(Plant, K_hinfstruct_tunable,...
30                 {'w'},...
31                 {'z'});
32
33 rng('default')
34 opt = hinfstructOptions('Display','final','RandomStart',5);
35 [sysOpt, gamma_hinfstruct] = hinfstruct(system,opt);
36
37 K_hinfstruct = getBlockValue(sysOpt,'K_hinfstruct_tunable');
38 K_hinfstruct = K_hinfstruct.D;
39 K_hinfstruct = K_hinfstruct(1:numInputs,1:numStateVars);
40 %% LMI
41 gamma = 0.73269;
42
43 [numZoutputs, ~] = size(Duz);
44 [numStateVars, numInputs] = size(Bu);
45 X = sdpvar(numStateVars, numStateVars, 'symmetric');
46 V = sdpvar(numStateVars, numStateVars, 'symmetric');
47 W = sdpvar(numInputs, numStateVars, 'full');
48 ops = sdpsettings('verbose',0);
49
50 constraintMatrix = [(A*X+Bu*W)' + A*X+Bu*W Bw (Cz*X+Duz*W)'
51                    Bw' -gamma*eye(1,1) Dwz'
52                    (Cz*X+Duz*W) Dwz -gamma*eye(numZoutputs)
53                    ];
54 F = [X >= eye(numStateVars)*eps];
55 F = [F, constraintMatrix <= eye(size(constraintMatrix))*eps];
56 optimize(F,[ ],ops);

```

```

56 K_Hinf = value(W)*inv(value(X));
57 %% Vysledky
58 figure
59 names = {'K_Hinf'; 'K_hinffi'; 'K_hinfstruct'};
60 gamma = [gamma; gamma_hinffi; gamma_hinfstruct];
61 result = [K_Hinf; K_hinffi; K_hinfstruct];
62 T1 = table(names, gamma, result)
63 systems = [];
64 for i = 1:size(result)
65     sys = ss(A+Bu*result(i,:),Bu,eye(3),zeros(3,1));
66     systems{i} = sys;
67     hold on
68     sigma(sys)
69 end
70 legend('K_Hinf', 'K_hinffi', 'K_hinfstruct', 'Location', 'Best')
71 grid on
72 %% Příklad 2
73 Cz = [1 2 -0.5];
74 Duz = 2;
75 Dwz = 0.1;
76 %% hinffi
77 P = ss(A, [Bw Bu], Cz, [Dwz Duz]);
78 [K_hinffi,CL,gamma_hinffi] = hinffi(P,numInputs,[2.1296 Inf]);
79 K_hinffi = K_hinffi(1:numInputs,1:numStateVars);
80 %% hinfstruct
81 Plant = ss(A, [Bw Bu], [Cz; eye(3)], [Dwz Duz; zeros(3,2)]);
82 Plant.InputName = {'w', 'u'};
83 Plant.OutputName = {'z', 'x1', 'x2', 'x3'};
84 Plant.name = 'P';
85
86 K_hinfstruct_tunable = tunableGain('K_hinfstruct_tunable', 1, 4);
87 K_hinfstruct_tunable.InputName = {'x1', 'x2', 'x3', 'w'};
88 K_hinfstruct_tunable.OutputName = {'u'};
89
90 system = connect(Plant, K_hinfstruct_tunable,...
91     {'w'},...
92     {'z'});
93
94 rng('default')
95 opt = hinfstructOptions('Display','final','RandomStart',20);
96 [sysOpt, gamma_hinfstruct] = hinfstruct(system,opt);
97
98 K_hinfstruct = getBlockValue(sysOpt,'K_hinfstruct_tunable');
99 K_hinfstruct = K_hinfstruct.D;
100 K_hinfstruct = K_hinfstruct(1:numInputs,1:numStateVars);
101 %% LMI
102 gamma = 2.2255;
103
104 [numZoutputs, ~] = size(Duz);
105 [numStateVars, numInputs] = size(Bu);
106 X = sdpvar(numStateVars, numStateVars, 'symmetric');
107 V = sdpvar(numStateVars, numStateVars, 'symmetric');
108 W = sdpvar(numInputs, numStateVars, 'full');
109 ops = sdpsettings('verbose',0);
110
111 constraintMatrix = [(A*X+Bu*W)' + A*X+Bu*W Bw (Cz*X+Duz*W)'
112     Bw' -gamma*eye(1,1) Dwz'
113     (Cz*X+Duz*W) Dwz -gamma*eye(numZoutputs)
114     ];
115 F = [X >= eye(numStateVars)*eps];
116 F = [F, constraintMatrix <= eye(size(constraintMatrix))*eps];
117 optimize(F,[ ],ops);

```

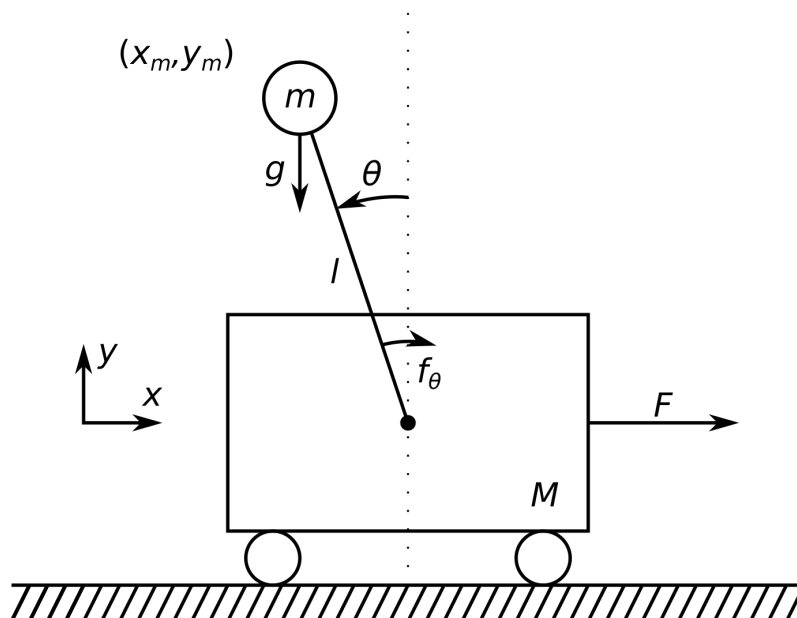
```

117 K_Hinf = value(W)*inv(value(X));
118 %% Vysledky
119 figure
120 names = {'K_Hinf'; 'K_hinffi'; 'K_hinfstruct'};
121 gamma = [gamma; gamma_hinffi; gamma_hinfstruct];
122 result = [K_Hinf; K_hinffi; K_hinfstruct];
123 T1 = table(names, gamma, result)
124 systems = [];
125 for i = 1:size(result)
126     sys = ss(A+Bu*result(i,:),Bu,eye(3),zeros(3,1));
127     systems{i} = sys;
128     hold on
129     sigma(sys)
130 end
131 legend('K_Hinf', 'K_hinffi', 'K_hinfstruct', 'Location', 'Best')
132 grid on

```

Výpis D.1: MATLAB skript s příklady výpočtu statického \mathcal{H}_∞ optimálního regulátoru různými metodami

E Odvození matematického modelu inverzního kyvadla na vozíku Lagrangeovou metodou



Obr. E.1: Schematický náčrt modelované soustavy

$$\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = Q_i \quad (\text{E.1})$$

$$L = T - V \quad (\text{E.2})$$

Kde L je Lagrangeova funkce, T je kinetická energie a V je potenciální energie, q_i je i -tá stavová proměnná a Q_i je i -tá vnější síla.

Informace o vztahu polohy jednotlivých konců tyče.

$$x_m = x - l \sin \theta \quad (\text{E.3})$$

$$y_m = l \cos \theta \quad (\text{E.4})$$

$$\dot{x}_m = \dot{x} - l \dot{\theta} \cos \theta \quad (\text{E.5})$$

$$\dot{y}_m = -l \dot{\theta} \sin \theta \quad (\text{E.6})$$

Vyjádření potenciální energie

$$V = mgy_m = mgl \cos \theta \quad (\text{E.7})$$

Vyjádření kinetické energie

$$\begin{aligned}
T &= \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(\dot{x}_m^2 + \dot{y}_m^2) = \\
&= \frac{1}{2}M\dot{x}^2 - \frac{1}{2}m[\dot{x}^2 + 2l\dot{\theta}\dot{x}\cos\theta + l^2\dot{\theta}^2(\cos^2\theta + \sin^2\theta)] = \\
&= \frac{1}{2}M\dot{x}^2 - \frac{1}{2}m[\dot{x}^2 + 2l\dot{\theta}\dot{x}\cos\theta + l^2\dot{\theta}^2] = \\
&= \frac{1}{2}(M+m)\dot{x}^2 + \frac{1}{2}ml^2\dot{\theta}^2 - ml\dot{\theta}\dot{x}\cos\theta
\end{aligned} \tag{E.8}$$

Dozením E.8 a E.7 do rovnice E.2 získáme E.9

$$\begin{aligned}
L &= T - V = \\
&= \frac{1}{2}(M+m)\dot{x}^2 + \frac{1}{2}ml^2\dot{\theta}^2 - ml\dot{\theta}\dot{x}\cos\theta - mgl\cos\theta
\end{aligned} \tag{E.9}$$

Dosadíme do rovnice E.1, x za q_i a vyjádříme jednotlivé členy.

$$\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{x}} \right) - \frac{\partial L}{\partial x} = Q_x \tag{E.10}$$

$$\frac{\partial L}{\partial x} = 0 \tag{E.11}$$

$$\begin{aligned}
\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{x}} \right) &= \frac{\partial}{\partial t} [(M+m)\dot{x} - ml\dot{\theta}\cos\theta] \\
&= (M+m)\ddot{x} - ml\ddot{\theta}\cos\theta + ml\dot{\theta}^2\sin\theta
\end{aligned} \tag{E.12}$$

Nyní můžeme dosadit.

$$(M+m)\ddot{x} - ml\ddot{\theta}\cos\theta + ml\dot{\theta}^2\sin\theta = F \tag{E.13}$$

Dosazením do rovnice E.1, θ za q_i a vyjádříme jednotlivé členy.

$$\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = Q_\theta \tag{E.14}$$

$$\frac{\partial L}{\partial \theta} = ml\dot{\theta}\dot{x}\sin\theta + mgl\sin\theta \tag{E.15}$$

$$\begin{aligned}
\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{\theta}} \right) &= \frac{\partial}{\partial t} (ml^2\dot{\theta} - ml\dot{x}\cos\theta) \\
&= ml^2\ddot{\theta} - ml\ddot{x}\cos\theta + ml\dot{x}\dot{\theta}\sin\theta
\end{aligned} \tag{E.16}$$

Díky tomu je možné dosadit.

$$\begin{aligned}
ml^2\ddot{\theta} - ml\ddot{x}\cos\theta + \cancel{ml\dot{x}\dot{\theta}\sin\theta} - \cancel{ml\dot{x}\dot{\theta}\sin\theta} - mgl\sin\theta &= mlf_\theta\dot{\theta} \\
l\ddot{\theta} - \ddot{x}\cos\theta - g\sin\theta &= f_\theta\dot{\theta} \quad ; m \neq 0, l \neq 0
\end{aligned} \tag{E.17}$$

Získáváme tak pohybové rovnice E.13 a E.17. Z nich můžeme vyjádřit rovnice stavové vyjádřením \ddot{x} a $\ddot{\theta}$.

$$\ddot{x} = \frac{ml(-\dot{\theta}^2 \sin \theta + \ddot{\theta} \cos \theta) + F}{M + m} \quad (\text{E.18})$$

$$\ddot{\theta} = \frac{1}{l} (\ddot{x} \cos \theta + g \sin \theta + f_{\theta} \dot{\theta}) \quad (\text{E.19})$$

Vidíme, že ve stavových rovnicích vznikla algebraická smyčka. Rovnice E.18 je závislá na $\ddot{\theta}$ a rovnice E.19 je závislá na \ddot{x} . Smyčku vyřešíme tak, že dosadíme za $\ddot{\theta}$ rovnici E.19 a za \ddot{x} rovnici E.18.

$$\ddot{x} = \frac{ml(-\dot{\theta}^2 \sin \theta + \frac{1}{l} (\ddot{x} \cos \theta + g \sin \theta + f_{\theta} \dot{\theta}) \cos \theta) + F}{M + m} \quad (\text{E.20})$$

$$\ddot{\theta} = \frac{1}{l} \left(\frac{ml(-\dot{\theta}^2 \sin \theta + \ddot{\theta} \cos \theta) + F}{M + m} \cos \theta + g \sin \theta + f_{\theta} \dot{\theta} \right) \quad (\text{E.21})$$

Nyní z rovnic vyjádříme \ddot{x} a $\ddot{\theta}$. Tím získáme stavový model systému pro stavový vektor $\vec{x} = [x, \dot{x}, \theta, \dot{\theta}]^T$

$$\dot{x} = \dot{x} \quad (\text{E.22})$$

$$\ddot{x} = \frac{-l m \sin(\theta) \dot{\theta}^2 - f_{\theta} m \cos(\theta) \dot{\theta} + F + g m \cos(\theta) \sin(\theta)}{-m \cos(\theta)^2 + M + m} \quad (\text{E.23})$$

$$\dot{\theta} = \dot{\theta} \quad (\text{E.24})$$

$$\ddot{\theta} = \frac{(M + m) \left(g \sin(\theta) - f_{\theta} \dot{\theta} + \frac{\cos(\theta) (F - l m \dot{\theta}^2 \sin(\theta))}{M + m} \right)}{l (-m \cos(\theta)^2 + M + m)} \quad (\text{E.25})$$

Linearizace

Soustavu má význam linearizovat ve dvou bodech. První je globálně nestabilní bod $\theta = 0$, tedy stav, kdy je závaží nahoře. Systém bude linearizován pomocí Jacobiho matice.

$$\mathbf{J}_A|_{\theta=0} = \begin{bmatrix} \frac{\partial \dot{x}}{\partial x} & \frac{\partial \dot{x}}{\partial \dot{x}} & \frac{\partial \dot{x}}{\partial \theta} & \frac{\partial \dot{x}}{\partial \dot{\theta}} \\ \frac{\partial \ddot{x}}{\partial x} & \frac{\partial \ddot{x}}{\partial \dot{x}} & \frac{\partial \ddot{x}}{\partial \theta} & \frac{\partial \ddot{x}}{\partial \dot{\theta}} \\ \frac{\partial \dot{\theta}}{\partial x} & \frac{\partial \dot{\theta}}{\partial \dot{x}} & \frac{\partial \dot{\theta}}{\partial \theta} & \frac{\partial \dot{\theta}}{\partial \dot{\theta}} \\ \frac{\partial \ddot{\theta}}{\partial x} & \frac{\partial \ddot{\theta}}{\partial \dot{x}} & \frac{\partial \ddot{\theta}}{\partial \theta} & \frac{\partial \ddot{\theta}}{\partial \dot{\theta}} \end{bmatrix} \Bigg|_{\theta=0} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{gm}{M} & -\frac{F_{\theta}m}{M} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g(M+m)}{Ml} & -\frac{f_{\theta}(M+m)}{Ml} \end{bmatrix} = \mathbf{A}_0 \quad (\text{E.26})$$

$$\mathbf{J}_B|_{\theta=0} = \left. \begin{bmatrix} \frac{\partial \dot{x}}{\partial F} \\ \frac{\partial \ddot{x}}{\partial F} \\ \frac{\partial \dot{\theta}}{\partial F} \\ \frac{\partial \ddot{\theta}}{\partial F} \end{bmatrix} \right|_{\theta=0} = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{Ml} \end{bmatrix} = \mathbf{B}_0 \quad (\text{E.27})$$

Z jakobiánů \mathbf{J}_A a \mathbf{J}_B získáme po dosazení $\theta = 0$ lineární matice \mathbf{A} a \mathbf{B} pro stavový popis lineárního systému linearizovaného kolem $\theta = 0$.

$$\begin{bmatrix} \dot{x}_0 \\ \ddot{x}_0 \\ \dot{\theta}_0 \\ \ddot{\theta}_0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{gm}{M} & -\frac{F_{\theta}m}{M} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g(M+m)}{Ml} & -\frac{f_{\theta}(M+m)}{Ml} \end{bmatrix} \begin{bmatrix} x_0 \\ \dot{x}_0 \\ \theta_0 \\ \dot{\theta}_0 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{Ml} \end{bmatrix} F \quad (\text{E.28})$$

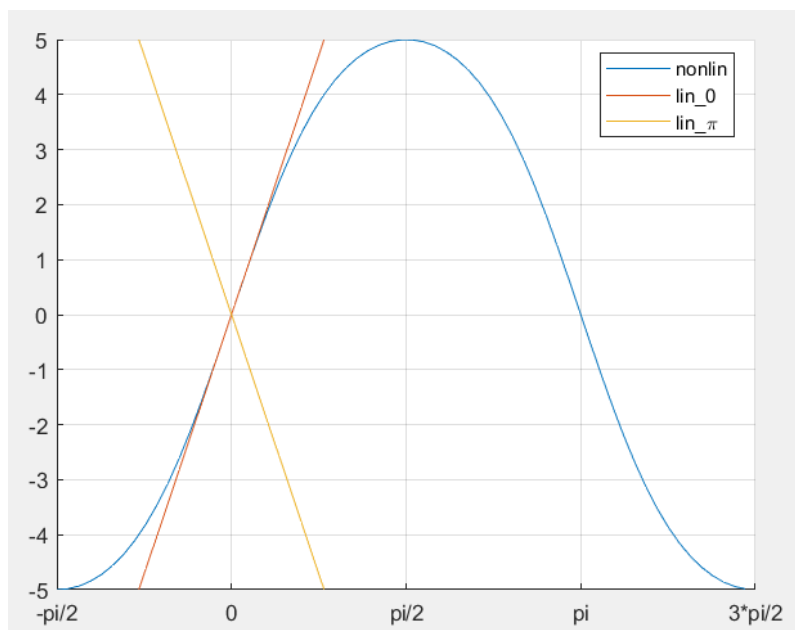
Stejně lze postupovat pro bod $\theta = \pi$, tedy pro globálně stabilní stav, kdy je kyvadlo přímo pod středem vozíku.

$$\mathbf{J}_A|_{\theta=\pi} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{gm}{M} & \frac{f_{\theta}m}{M} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{g(M+m)}{Ml} & -\frac{f_{\theta}(M+m)}{Ml} \end{bmatrix} = \mathbf{A}_{\pi} \quad (\text{E.29})$$

$$(\text{E.30})$$

$$\mathbf{J}_B|_{\theta=0} = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ -\frac{1}{Ml} \end{bmatrix} = \mathbf{B}_{\pi} \quad (\text{E.31})$$

$$\begin{bmatrix} \dot{x}_{\pi} \\ \ddot{x}_{\pi} \\ \dot{\theta}_{\pi} \\ \ddot{\theta}_{\pi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{gm}{M} & \frac{F_{\theta}m}{M} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{g(M+m)}{Ml} & -\frac{f_{\theta}(M+m)}{Ml} \end{bmatrix} \begin{bmatrix} x_{\pi} \\ \dot{x}_{\pi} \\ \theta_{\pi} \\ \dot{\theta}_{\pi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ -\frac{1}{Ml} \end{bmatrix} F \quad (\text{E.32})$$



Obr. E.2: Srovnání $\ddot{\theta}$ pro nelineární a linearizované průběhy.

F Návrh \mathcal{H}_∞ optimálního kaskádního regulátoru

```
1 m = 1;
2 M = 5;
3 l = 2;
4 g = 10;
5 f_T = 1;
6 F = 0;
7 %% Define State Space
8 A = [ 0, 1, 0, 0,
9       0, 0, (g*m)/M, -(f_T*m)/M,
10      0, 0, 0, 1,
11      0, 0, (g*(M + m))/(M*l), -(f_T*(M + m))/(M*l)];
12
13 B = [0; 1/M; 0; 1/(M*l)];
14 C = [0 0 1 0];
15 D = 0;
16
17 P = ss(A,B,C,D);
18 P.InputName = 'F';
19 P.OutputName = {'\theta'};
20 P.StateName = {'x', 'dx', '\theta', 'd\theta'};
21 P.name = 'P';
22
23 %% Stabilizující regulátor
24 sum1 = sumblk('e\theta = w\theta - \theta');
25 sum1.name = 'sum1';
26 % vaha citlivostní funkce pro stab.reg.
27 W_KS = ss(1/3.2);
28 W_KS = makeweight(1/100, [10e-2 1/141], 1/224);
29 W_KS.InputName = 'F';
30 W_KS.OutputName = 'zks';
31 W_KS.name = 'W_KS';
32 % vaha kompl. citlivostní funkce pro stab.reg.
33 W_S = makeweight(1/0.63, 0.045, 1/2.51);
34 W_S.InputName = {'e\theta'};
35 W_S.OutputName = 'zs';
36 W_S.name = 'W_S';
37
38 nmeas = 1; % pocet merenych vystupu
39 ncont = 1; % pocet akcnich velicin
40
41 % Zapojeni pro syntezu stab. reg.
42 system = connect(sum1, P, W_KS, W_S, ...
43                 {'w\theta', 'F'}, ...
44                 {'zks', 'zs', 'e\theta'});
45 [Ktheta,~,gam_Ktheta] = hinfyn(system,nmeas,ncont, 'Method', '
46                               LMI');%, 'Display', 'On');
47 Ktheta.InputName = {'e\theta'};
48 Ktheta.OutputName = 'F';
49 Ktheta.name = 'K_\theta';
50
51 systemS = connect(sum1, P, Ktheta, {'w\theta'}, {'e\theta'});
52 systemKS = connect(sum1, P, Ktheta, {'w\theta'}, {'F'});
53
54 figure
55 bodemag(systemS, 'r')
56 hold on
```

```

56 bodemag(systemKS, 'g')
57 bodemag(1/W_KS, 'g--')
58 bodemag(1/W_S, 'r--')
59 legend('S', 'KS', '1/W_KS', '1/W_S')
60 grid on
61
62 %% Polohovy regulator
63 sum2 = sumblk('ex = wx-x');
64 sum2.name = 'sumx';
65
66 % pridani vystupu z polohy do systemu
67 C = [1 0 0 0; 0 0 1 0];
68 P = ss(A,B,C,D);
69 P.InputName = 'F';
70 P.OutputName = {'x', '\theta'};
71 P.StateName = {'x', 'dx', '\theta', 'd\theta'};
72 P.name = 'P';
73
74 W_KS = ss(1/3.2);
75 W_KS.InputName = 'F';
76 W_KS.OutputName = 'zks';
77 W_KS.name = 'W_KS';
78
79 W_S = makeweight(1000,0.5,0.5);
80 W_S.InputName = 'ex';
81 W_S.OutputName = 'zs';
82 W_S.name = 'W_S';
83
84 system = connect(sum2, sum1, P, W_KS, W_S, Ktheta, {'wx', 'w\
      theta'}, {'zs', 'zks', 'ex'});
85
86 [Kx,CL,gam_Kx] = hinfsyn(system,1,1);
87 Kx.InputName = {'ex'};
88 Kx.OutputName = 'w\theta';
89 Kx.name = 'K_x';
90
91 systemKS = connect(sum2, sum1, P, Kx, Ktheta, {'wx'}, {'F'});
92 systemS = connect(sum2, sum1, P, Kx, Ktheta, {'wx'}, {'ex'});
93 systemT = connect(sum2, sum1, P, Kx, Ktheta, {'wx'}, {'x', '\
      theta', 'F'});
94 %%
95 figure
96 bodemag(systemS, 'r')
97 hold on
98 bodemag(systemKS, 'g')
99 bodemag(1/W_KS, 'g--')
100 bodemag(1/W_S, 'r--')
101 legend('S', 'KS', '1/W_KS', '1/W_S', 'Location', 'Best')
102 grid on
103
104 figure
105 step(systemT)

```

Výpis F.1: MATLAB skript pro návrh \mathcal{H}_∞ optimálního kaskádního regulátoru

G Návrh \mathcal{H}_∞ optimálního stavového regulátoru

```
1 %% Definice konstant
2 thetaNoisePower = 0.0001;
3 thetaNoiseSampleTime = 0.01;
4 FNoisePower = 0;
5 FNoiseSampleTime = 0.01;
6 wx = 0;
7 wdx = 0;
8 wTheta = 0;
9 wdTheta = 0;
10
11 m = 1;
12 M = 5;
13 l = 2;
14 g = 10;
15 f_T = 1;
16 F = 0;
17 %% Definice systemu
18 A_0 = [ 0, 1, 0, 0,
19         0, 0, (g*m)/M, -(f_T*m)/M,
20         0, 0, 0, 1,
21         0, 0, (g*(M + m))/(M*1), -(f_T*(M + m))/(M*1)];
22
23 B_0 = [0; 1/M; 0; 1/(M*1)];
24
25 C = eye(4);
26 D = zeros(4, 1);
27
28 P = ss(A_0, B_0, C, D);
29 P.name = 'P';
30 P.InputName = {'F'};
31 P.OutputName = {'x', 'dx', '\theta', 'd\theta'};
32 P.StateName = {'x', 'dx', '\theta', 'd\theta'};
33 %% Navrh LQR
34 Q = [1 0 0 0;
35      0 10 0 0;
36      0 0 1 0;
37      0 0 0 1];
38 R = 0.1;
39
40 LQR = lqr(A_0, B_0, Q, R);
41
42 Tunable_LQR = ss(realp('LQR', LQR));
43 Tunable_LQR.InputName = {'ex', 'dx', '\theta', 'd\theta'};
44 Tunable_LQR.OutputName = 'F';
45 %% Sumacni blok
46 sum1 = sumblk('%e = %r + %y', {'ex'}, {'wx'}, {'x'});
47 sum1.name = 'sum1';
48 %% Definice prenosu
49 LQR_T = connect(P, sum1, Tunable_LQR, {'wx'}, {'x', 'dx', '\theta',
50                                         'd\theta'});
51 LQR_S = connect(P, sum1, Tunable_LQR, {'wx'}, {'ex', 'dx', '\theta',
52                                         'd\theta'});
53 LQR_KS = connect(P, sum1, Tunable_LQR, {'wx'}, {'F'});
54 %% Navrh vah
55 W1_S = makeweight(100, 0.23, 1/2);
56 W2_S = makeweight(100, 0.2, 0.9);
```

```

55 W3_S = makeweight(100,60,0.9);
56 W4_S = makeweight(100,20,0.9);
57 W_S = [W1_S ss(0) ss(0) ss(0)
58         ss(0) W2_S ss(0) ss(0)
59         ss(0) ss(0) W3_S ss(0)
60         ss(0) ss(0) ss(0) W4_S];
61 W_S.name = 'W_S';
62 W_S.InputName = {'ex', 'dx', '\theta', 'd\theta'};
63 W_S.OutputName = {'zsx', 'zsdx', 'zs\theta', 'zsd\theta'};
64
65 W_KS = ss(1/3.2);
66 W_KS.name = 'W_KS';
67 W_KS.InputName = {'F'};
68 W_KS.OutputName = {'zksF'};
69 %% Synteza regulatoru
70 controlsys = connect(P, sum1, W_KS, W_S, Tunable_LQR,...
71                     {'wx'},...
72                     {'zksF'},...
73                     'zsx', 'zsdx', 'zs\theta', 'zsd\theta'});
74
75 rng('default')
76 opt = hinfstructOptions('Display','final','RandomStart',20);
77 [K_Hinf, gamma] = hinfstruct(controlsys,opt);
78 K_Hinf = K_Hinf.Blocks.LQR.Value;
79 K_Hinf_SS = ss(K_Hinf);
80 K_Hinf_SS.InputName = {'ex', 'dx', '\theta', 'd\theta'};
81 K_Hinf_SS.OutputName = 'F';
82 K_Hinf_SS.name = 'K_Hinf';
83 %% Definice prenosu vysledku
84 ROB_T = connect(P, sum1, K_Hinf_SS, {'wx'}, {'x', 'dx', '\theta',
85                                         'd\theta'});
86 ROB_S = connect(P, sum1, K_Hinf_SS, {'wx'}, {'ex', 'dx', '\theta',
87                                         'd\theta'});
88 ROB_KS = connect(P, sum1, K_Hinf_SS, {'wx'}, {'F'});
89 %% Vysledky
90 for i = 1:4
91     subplot(2,4,i)
92     sigmaplot(ROB_S(i,:), 'r', LQR_S(i,:), 'r:', 1/W_S(i,i), 'r--')
93     grid on
94     title(strcat({'\sigma( S_{'} , P.OutputName(i), {'}')}))
95     a = strcat({'$\sigma( S_{'} , P.OutputName(i), {'})_{H-inf}$'});
96     b = strcat({'$\sigma( S_{'} , P.OutputName(i), {'})_{LQG}$'});
97     c = strcat({'$\frac{1}{W_S('} , P.OutputName(i), {'})$'});
98     legend(a{1},b{1},c{1}, 'Location', 'Best', 'Interpreter', '
99         latex');
100    legend(a{1},b{1},c{1}, 'Location', 'Best', 'Interpreter', '
101        latex');
102 end
103 subplot(2,4,5:8)
104 sigmaplot(ROB_KS, 'g', LQR_KS, 'g:', 1/W_KS, 'g--')
105 grid on
106 title(strcat({'\sigma( KS_{'} , P.InputName, {'}')}))
107 a = strcat({'$\sigma( KS_{'} , P.InputName, {'})_{H-inf}$'});
108 b = strcat({'$\sigma( KS_{'} , P.InputName, {'})_{LQG}$'});
109 c = strcat({'$\frac{1}{W_KS('} , P.InputName, {'})$'});
110 legend(a{1},b{1},c{1}, 'Location', 'Best', 'Interpreter', 'latex')
111 legend(a{1},b{1},c{1}, 'Location', 'Best', 'Interpreter', 'latex')

```

Výpis G.1: MATLAB skript pro návrh \mathcal{H}_∞ optimálního stavového regulátoru

H Návrh \mathcal{H}_∞ optimálního stavového rekonstruktoru

```

1 %% Definice konstant
2 thetaNoisePower = 0.0001;
3 thetaNoiseSampleTime = 0.01;
4 FNoisePower = 0;
5 FNoiseSampleTime = 0.01;
6 wx = 0;
7 wdx = 0;
8 wTheta = 0;
9 wdTheta = 0;
10
11 m = 1;
12 M = 5;
13 l = 2;
14 g = 10;
15 f_T = 1;
16 F = 0;
17 %% Definice systemu
18 A_0 = [ 0, 1, 0, 0;
19         0, 0, (g*m)/M, -(f_T*m)/M;
20         0, 0, 0, 1;
21         0, 0, (g*(M + m))/(M*l), -(f_T*(M + m))/(M*l)];
22
23 B_0 = [0; 1/M; 0; 1/(M*l)];
24
25 C = [1 0 0 0];
26 D = 0;
27
28 P = ss(A_0, eye(4), C, D);
29 P.name = 'P';
30 P.InputName = {'v'};
31 P.OutputName = {'x_h'};
32 P.StateName = {'x_h', 'dx_h', '\theta_h', 'd\theta_h'};
33 %% Navrh Kalmanova zesileni
34 Vd = [1 0 0 0;
35       0 10 0 0;
36       0 0 1 0;
37       0 0 0 1];
38 Vn = 10;
39
40 L_Kalman = (-lqr(A_0',C',Vd,Vn))';
41
42 Tunable_L = ss(realp('L',L_Kalman));
43 Tunable_L.InputName = 'ex';
44 Tunable_L.OutputName = 'v';
45 %% Sumacni blok
46 sum2 = sumblk('%e = %r - %y', {'ex'},{'x_h'},{'x'});
47 sum2.name = 'sum2';
48 %% Definice prenosu
49 Kalman_T = connect(P, Tunable_L, sum2,{'x'},{'x_h'});
50 Kalman_S = connect(P, Tunable_L, sum2,{'x'},{'ex'});
51 Kalman_KS = connect(P, Tunable_L, sum2,{'x'},{'v(1)', 'v(2)', 'v(3)', 'v(4)'});
52 %% Navrh vah
53 W_S = makeweight(1000,[0.28 1/0.1],1/1.1,0,2);
54 W_S.name = 'W_S';
55 W_S.InputName = 'ex';

```

```

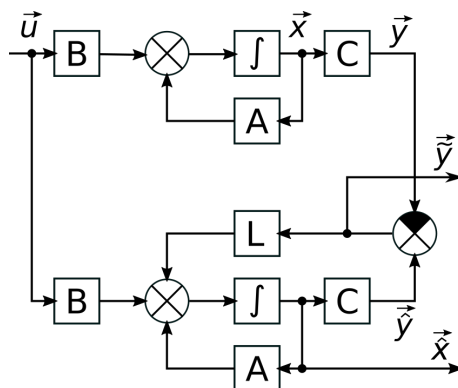
56 W_S.OutputName = 'Z_S';
57
58 W1_KS = makeweight(100,0.2,1/7 ,0,2);
59 W2_KS = makeweight(100,0.2,1/23 ,0,2);
60 W3_KS = makeweight(100,0.2,1/35 ,0,2);
61 W4_KS = makeweight(100,0.13,1/80 ,0,2);
62 W_KS = [W1_KS ss(0) ss(0) ss(0)
63         ss(0) W2_KS ss(0) ss(0)
64         ss(0) ss(0) W3_KS ss(0)
65         ss(0) ss(0) ss(0) W4_KS];
66 W_KS.name = 'W_KS';
67 W_KS.InputName = {'v(1)', 'v(2)', 'v(3)', 'v(4)'};
68 W_KS.OutputName = {'Z_KS(1)', 'Z_KS(2)', 'Z_KS(3)', 'Z_KS(4)'};
69 %% Synteza regulatoru
70 controlsys = connect(P, sum2, W_KS, W_S, Tunable_L,...
71                     {'x'},...
72                     {'Z_S'},...
73                     'Z_KS(1)', 'Z_KS(2)', 'Z_KS(3)', 'Z_KS(4)'});
74
75 rng('default')
76 opt = hinfstructOptions('Display','final','RandomStart',20);
77 [L_Hinf,gamma] = hinfstruct(controlsys,opt);
78 L_Hinf = L_Hinf.Blocks.L.Value;
79 L_Hinf_SS = ss(L_Hinf);
80 L_Hinf_SS.InputName = 'ex';
81 L_Hinf_SS.OutputName = 'v';
82 L_Hinf_SS.name = 'L_Hinf';
83 %% Definice prenosu vysledku
84 ROB_T = connect(P, L_Hinf_SS, sum2,{'x'},{'x_h'});
85 ROB_S = connect(P, L_Hinf_SS, sum2,{'x'},{'ex'});
86 ROB_KS = connect(P, L_Hinf_SS, sum2,{'x'},{'v(1)', 'v(2)', 'v(3)',
87         'v(4)'});
88 %% Vysledky
89 subplot(2,4,1:4)
90 sigmaplot(ROB_S,'r', Kalman_S,'r:', 1/W_S, 'r--')
91 grid on
92 title(strcat({'\sigma( S)'}))
93 a = strcat({'$\sigma( S_{H-inf})$'});
94 b = strcat({'$\sigma( S_{LQG})$'});
95 c = strcat({'$\frac{1}{W_{S}}$'});
96 legend(a{1},b{1},c{1}, 'Location', 'Best', 'Interpreter','
97         latex');
98 legend(a{1},b{1},c{1}, 'Location', 'Best', 'Interpreter','
99         latex');
100 for i = 5:8
101     subplot(2,4,i)
102     sigmaplot(ROB_KS(i-4,:), 'g', Kalman_KS(i-4,:), 'g:', 1/W_KS(i
103         -4,i-4), 'g--')
104     grid on
105     title(strcat({'\sigma( KS_{'} , num2str(i-4), {'}')'}))
106     a = strcat({'$\sigma( KS_{'} , num2str(i-4), {'H-inf})$'});
107     b = strcat({'$\sigma( KS_{'} , num2str(i-4), {'Kalman})$'});
108     ;
109     c = strcat({'$\frac{1}{W_{KS_{'} , num2str(i-4), {'}}$'});
110     legend(a{1},b{1},c{1}, 'Location', 'Best', 'Interpreter','
111         latex');
112     legend(a{1},b{1},c{1}, 'Location', 'Best', 'Interpreter','
113         latex');
114 end

```

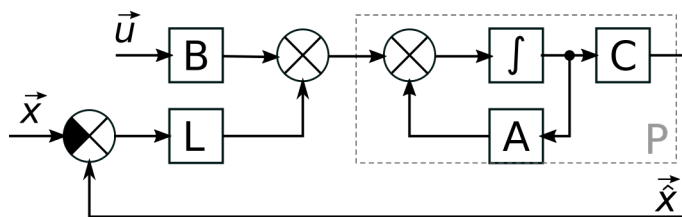
Výpis H.1: MATLAB skript pro návrh \mathcal{H}_∞ optimálního stavového rekonstruktoru

I Převedení stavového rekonstruktoru na \mathcal{H}_∞ problém

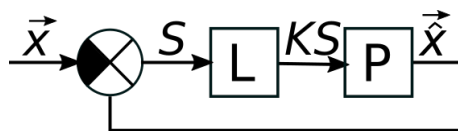
Odvození vychází z blokového diagramu stavového rekonstruktoru.



Z obrázku je patrné, že vstupem do rekonstruktoru jsou vektory \vec{y} a \vec{u} . Regulovaným výstupem je poté $\hat{\vec{x}}$. Prvním krokem je tedy odstranění rekonstruované soustavy a pozměnění schématu tak, aby více připomínalo běžnou regulační smyčku.



Předpokládáme-li, že signál \vec{u} zde funguje jako porucha, je možné signál pro návrh ignorovat.



Vzniká tak schematické zapojení, pro které jsou definovány funkce S , T a KS .