



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Disertační práce

Prediktivní řízení procesů s rychlou dynamikou

Predictive Control of Processes with Fast Dynamics

Autor: **Ing. Tomáš Barot**

Studijní program: P3902 Inženýrská informatika

Studijní obor: 3902V037 Automatické řízení a informatika

Školitel: doc. Ing. Marek Kubalčík, Ph.D.

Zlín, 2016

ABSTRAKT

Prediktivní řízení procesů je řídicí metodou, jež je vhodná pro řízení mnoha typů procesů. V případě řízení procesů s rychlou dynamikou nemusí být řídicí algoritmus vždy realizovatelný v rámci periody vzorkování. Tato situace může nastat v případech náročnějších požadavků na řízení, a to pro vyšší hodnoty horizontů, vyšší počty omezení kladených na veličiny řízení či v případě mnohazměrového řízení. V rámci disertační práce je zmapován současný stav problematiky prediktivního řízení procesů s rychlou dynamikou. Zavedené přístupy jsou zaměřeny na způsoby řešení úlohy kvadratického programování a zkoumají její vliv na prediktivní řízení. Přičemž tato řešení spočívají na teorii duality, Kuhn–Tuckerových podmínkách a modifikují algoritmus prediktivního řízení zahrnující operace s podmínkami omezení. Cílem práce je inovace těchto přístupů s ohledem na snížení jejich výpočetní náročnosti. Prezentované návrhy jsou založeny na šetrnějších eliminacích omezení v kvadratickém programování. Je navržena nová modifikace duální optimalizační metody. Implementace těchto návrhů je realizována v softwaru pro MATLAB, a to pro SISO a TITO procesy. Hlavní výsledky jsou porovnány po stránce schopnosti realizovat algoritmus prediktivního řízení ve stanovené periodě vzorkování, a to navrženými přístupy oproti zavedeným metodám.

ABSTRACT

Predictive control is a control method, which is appropriate for control of various kinds of processes. In certain cases of predictive control of fast–dynamics processes, a predictive control algorithm may not be feasible within the sampling–period time. These situations occur when requirements on control are more complex. For higher horizons and many constraints on control variables or in multivariable control, the overloading of the sampling period can occur. In the thesis, the current state of art is researched in the area of predictive control of fast–dynamics processes. The established approaches are focused primarily on the quadratic programming and its influence on the predictive control. These solutions are based on the duality theory, Kuhn–Tucker conditions and operations with constraints in the algorithm of the predictive control. Aim of this thesis is an innovation of these approaches with respect to a decreasing of their computational complexity. Presented approaches are based on the more detailed elimination of constraints in the quadratic programming. The new modified dual optimization method is designed. The proposed approaches are implemented for the SISO and TITO processes in the software for MATLAB. The main results are verified and compared to the established methods with respect to ability of predictive control of fast–dynamics processes in the defined sampling–period time.

KLÍČOVÁ SLOVA

Prediktivní řízení, MPC, diskrétní řízení, lineární diskrétní dynamické systémy, procesy s rychlou dynamikou, optimalizace, Kuhn–Tuckerovy podmínky, teorie duality, kvadratické programování, omezení.

KEYWORDS

Predictive Control, MPC, Discrete Control, Linear Discrete Dynamical Systems, Processes with Fast Dynamics, Optimization, Kuhn–Tucker Conditions, Duality Theory, Quadratic Programming, Constraints.

PODĚKOVÁNÍ

Je mi ctí poděkovat mému školiteli panu doc. Ing. Marku Kubalčíkovi, Ph.D. za odborné vedení disertační práce. Chtěl bych vyjádřit velké poděkování mé rodině za podporu při mé snaze dosáhnout kvalifikace Ph.D. Poděkování patří též Váženým pánům docentům a profesorům z Fakulty aplikované informatiky za cenné předání odborných vědomostí z vědního oboru teorie automatického řízení v rámci doktorského studijního programu.

Prostřednictvím finanční podpory Interní grantové soutěže Univerzity Tomáše Bati ve Zlíně bylo možno pořídit renomované monografie a publikovat výsledky tvůrčí činnosti na hodnocených konferencích, a to v souladu s hodnocením tvůrčích aktivit dle metodiky Rady vlády pro výzkum, vývoj a inovace. Jednalo se jmenovitě o následující projekty: IGA/FAI/2012/004, IGA/FAI/2014/002 a IGA/FAI/2015/006.

OBSAH

1.	ÚVOD	8
2.	SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY	10
3.	CÍLE DISERTAČNÍ PRÁCE	12
4.	TEORIE ZAVEDENÝCH PŘÍSTUPŮ A JEJICH MODIFIKACÍ.....	14
4.1	Základy optimalizačních technik.....	14
4.1.1	Základní terminologie optimalizace.....	14
4.1.2	Klasifikace úloh podle způsobu zadání množiny \mathcal{K}	15
4.1.3	Úloha volného extrému	15
4.1.4	Klasická úloha na vázaný extrém.....	16
4.1.5	Neklasická úloha na vázaný extrém.....	19
4.1.6	Kvadratické programování.....	20
4.1.7	Teorie duality v kvadratickém programování.....	21
4.2	Přístupy řešení neklasické úlohy na vázaný extrém.....	23
4.2.1	Analytický převod na klasickou úlohu na vázaný extrém	23
4.2.2	Existence jevu převodu na úlohu volného extrému	24
4.2.3	Využití teorie duality v kvadratickém programování	25
4.3	Prediktivní řízení procesů s rychlou dynamikou.....	26
4.3.1	Prediktivní řízení procesů	26
4.3.2	Procesy s rychlou dynamikou	27
4.3.3	Jednorozměrové prediktivní řízení.....	29
4.3.4	Dvourozměrové prediktivní řízení	33
4.3.5	Zavedená modifikace v online prediktivním řízení	38
5.	HLAVNÍ VÝSLEDKY PRÁCE.....	39
5.1	Algoritmus analytického řešení kvadratického programování... 39	
5.1.1	Rozbor analytického řešení kvadratického programování.....	39

5.1.2	Sestavení algoritmu pro účely verifikace	48
5.2	Důkaz jevu výskytu volného extrému v řešení neklasické úlohy na vázaný extrém	52
5.3	Návrh optimalizační metody kvadratického programování.....	55
5.4	Návrh modifikací v prediktivním řízení.....	58
5.4.1	Princip částečných eliminací v definici omezení veličin MPC..	58
5.4.2	Možnosti začlenění navrhované optimalizační metody	63
6.	EXPERIMENTÁLNÍ ČÁST PRÁCE	65
6.1	Prediktivní řízení procesů s rychlou dynamikou.....	65
6.1.1	Realizace prediktivního řízení	65
6.1.2	Volba modelů procesů s rychlou dynamikou	68
6.1.3	Verifikace implementovaných optimalizačních metod.....	72
6.1.4	Simulace prediktivního řízení zvolených modelů procesů.....	72
6.2	Aparát pro stanovení výpočetní náročnosti algoritmu a pro určení kvality řízení.....	83
6.2.1	Počet aritmeticko logických operací algoritmu.....	83
6.2.2	Měření doby provedení operací v prostředí MATLAB.....	84
6.2.3	Definice kritérií kvality řízení	85
6.3	Posouzení navrhované optimalizační metody.....	86
6.4	Analýza výpočetní náročnosti navrhovaných řešení v MPC.....	88
6.4.1	Stanovení výchozích podmínek pro měření	88
6.4.2	Jednorozměrové prediktivní řízení	90
6.4.3	Dvourozměrové prediktivní řízení.....	98
7.	PŘÍNOS PRÁCE PRO VĚDU A PRAXI	106
8.	ZÁVĚR.....	108
	SEZNAM POUŽITÉ LITERATURY	110
	SEZNAM OBRÁZKŮ.....	114

SEZNAM TABULEK.....	116
SEZNAM POUŽITÝCH SYMBOLŮ	117
SEZNAM ZKRATEK.....	125
PUBLIKAČNÍ AKTIVITY AUTORA.....	126
ODBORNÝ ŽIVOTOPIS	128
PŘÍLOHA A.....	131
A.1 VERIFIKAČNÍ ALGORITMUS PRO OPT. METODY	131
A.2 OVĚŘENÍ SPRÁVNOSTI VERIFIKAČNÍHO ALGORITMU	132
A.3 IMPLEMENTACE HILDRETHOVY METODY.....	133
A.4 KÓD WANGOVY MODIFIKACE.....	134
A.5 KÓD NAVRHOVANÉ OPTIMALIZAČNÍ METODY	135
A.6 KÓD VERIFIKACE OPTIMALIZAČNÍCH METOD	136
A.7 KÓD POSOUZENÍ EFEKTIVITY OPT. METOD.....	137
PŘÍLOHA B – STRUKTURA PŘILOŽENÉHO CD.....	138

1. ÚVOD

Prediktivní řízení (MPC, Model Predictive Control) [1]–[6] je moderní metodou řízení procesů [7]–[10]. Prakticky je realizováno zejména v diskrétní variantě. Jeho algoritmy jsou mnohostranně použitelné, a to pro různé charakteristiky řízených procesů. Lze jimi úspěšně řídit i neminimálně fázové procesy [11] a procesy s dopravním zpožděním [12]. Kromě jednorozměrových procesů (SISO, Single Input Single Output) [7] je metoda prediktivního řízení vhodná pro mnoharozměrové procesy (MIMO, Multi Input Multi Output) [9]. V disertační práci bude diskutována mnoharozměrová varianta se dvěma vstupy a dvěma výstupy (TITO, Two Input Two Output).

Předností prediktivního řízení je možnost zakomponovat požadavek na omezení veličin [4] již při samotném návrhu regulátoru. Je možné takto omezit řídicí, stavové a regulované veličiny. Z pohledu struktury zahrnuje prediktivní regulátor dvě důležité součásti, kterými jsou prediktor [1] a optimalizátor [4]. Zahrnutí modelu řízeného procesu do prediktoru je pro prediktivní řízení charakteristické. Je možné pracovat se vstupně-výstupním [7] nebo stavovým modelem procesu [10].

Prediktivní regulátor zahrnuje řadu nastavitelných parametrů, které souvisí s principem pohyblivého horizontu [2], který je jednou ze strategií prediktivního řízení. Jsou stanoveny predikce [13] budoucích průběhů veličin vzhledem ke stávajícímu okamžiku řízení. A to na základě znalosti minulých a současných informací z řízení. Nastavením rozsahů horizontů lze ovlivňovat kvalitu regulace. Horizonty se definují jakožto minimální, maximální a řídicí [2]. Za vstupní informace pro výpočet lze považovat průběh žádané veličiny, minulé přírůstky akčních zásahů, hodnoty stavů a regulované veličiny. Predikované veličiny mají platnost pouze pro aktuální okamžik vzorkování a poté je celý výpočet opakován.

Zákon řízení [4] je řešen v optimalizační části regulátoru. Je zde proveden výpočet přírůstků budoucích akčních zásahů, přičemž pro následný krok řízení je aplikován vždy první člen této posloupnosti. Pro případ nezahrnutí omezení veličin řízení se řeší optimalizační úloha volného vícerozměrného extrému [14]. Při existenci požadavků na omezení je potřeba řešit neklasickou úlohu na vázaný extrém [14]. Požadavky na MPC řízení jsou promítnuty do podoby výběru účelové funkce. Většinou bývá účelová funkce kvadratická [4] a při omezení tvaru nerovností je řešena úloha kvadratického programování [15]–[17], jež je speciálním tvarem problému nelineárního programování [14]–[21].

Přestože jsou v prediktivním řízení uplatňovány moderní přístupy, je zde stále prostor pro výzkum vedoucí ke zlepšení jejich výpočetní efektivity. A to zejména u řízení specifických typů procesů jako jsou procesy s rychlou dynamikou [8]. Potřebné požadavky na rychlý běh algoritmu řízení nastává též při kombinaci složitějšího nastavení (vyšší horizonty), výskytu vyššího počtu omezujících podmínek při úloze řízení a mnohazměrovosti řízeného procesu.

V rámci disertační práce bude zmapován současný stav problematiky prediktivního řízení procesů s rychlou dynamikou. Zejména bude za zavedení přístup považována modifikace [4, s. 67, krok 2] aplikující eliminace v úloze kvadratického programování v optimalizátoru s nasazením duální Hildrethovy metody [4].

Budou navrženy takové přístupy k řešení MPC procesů s rychlou dynamikou, které umožní zvládat realizaci tohoto algoritmu v rámci stanovené periody vzorkování, a to v případech, kdy jsou zavedené přístupy neúčinné.

Navrhované postupy budou teoreticky navazovat na dosavadní stav poznání v problematice prediktivního řízení. Snaha efektivněji prediktivně řídit procesy s rychlou dynamikou bude především založena na teorii duality [22], na práci s podmínkami Kuhn–Tuckerovy věty [14] a na šetrnějším principu eliminací podmínek v rámci definice omezení veličin MPC, a to ve srovnání se zavedenou modifikací dle [4, s. 67, krok 2].

2. SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY

Prediktivní řízení [1]–[6] bývá většinou řešeno v diskrétní formě [9]. A to buď ve vstupně-výstupní oblasti nebo ve stavovém prostoru [10]. Řešení ve stavové oblasti je bráno v úvahu v monografii [3] nebo v publikacích [23]–[24]. Z pohledu analogického odvození obecných tvarů predikčních rovnic pro jednorozměrové i mnoharozměrové procesy je matematický aparát stavového prostoru přehledný, avšak maticové výpočty s ním spojené vykazují vyšší paměťovou náročnost.

Zavedené přístupy prediktivního řízení je potřeba modifikovat pro konkrétní potřeby řízení specifických typů procesů. Dále bude brán v potaz typ lineárních dynamických systémů [10] s rychlou dynamikou [8], kdy je nutné řešit zákon řízení [7] v čase definovaném periodou vzorkování v řádu setin až tisícín sekundy. Výpočetní náročnost algoritmu řízení se přitom zvyšuje při uvažování vyšších hodnot horizontů v rámci MPC, při definici více omezujících podmínek veličin řízení či při mnoharozměrovosti řízení.

Výzkum urychlení výpočtů v rámci MPC bývá často zaměřen na subsystém prediktoru [13]. Je v něm kladen důraz na efektivitu výpočtů při stanovení rekurentních formulí pro určení predikcí budoucích výstupů řízeného procesu. Jedná se především o maticový přístup, postup získání řešení prostřednictvím diofantických rovnic nebo skrze CARIMA model [25].

Předmětem výzkumu bývá zejména optimalizační část MPC řízení, a to např. v publikaci [26]. Specifickými přístupy lze začlenit činnost tohoto modulu i mimo výpočetní algoritmus v každém kroku diskrétního řízení. Tato strategie je označována jako offline nebo též jako explicitní optimalizace [27]–[29]. Na rozdíl od průběžné implicitní resp. online optimalizace [4] spočívá v prvotním vyřešení optimalizačního problému před zahájením samotného řízení. V každém kroku řízení je posléze vyhledána požadovaná informace podle kritérií, a to v databázi vytvořené před spuštěním řízení. Jako nástroj pro řešení offline optimalizace při syntéze prediktivního regulátoru může sloužit software [29].

Mnoho vědeckých prací inovuje postupy širokého spektra optimalizačních metod a jejich vzájemných kombinací, anebo diskutuje volbu nastavení startovního bodu numerických metod. Příkladem může být postup zmiňovaný v práci [23], jež upřednostňuje online optimalizaci oproti její explicitní variantě. Tato publikace sice aplikuje kombinaci několika optimalizačních metod, avšak nezaobírá se natolik mechanismy optimalizace jako práce [4], v níž je implementována rychlá duální Hildrethova metoda [14].

V publikaci [4] je brán zřetel na použití teorie duality pro urychlení výpočtů optimalizační úlohy kvadratického programování dle [4, s. 67–68]. Jelikož duální úloha kvadratického programování má pouze jedno omezení, a to podmínku nezápornosti duální vektorové proměnné [14], je výpočetně méně náročná oproti principu primárních metod. Použití simplexových metod (např. Wolfeho [18], Shetty–Lemkeho [14]) není též pro prediktivním řízení vhodné, neboť se předpokládá podmínka nezápornosti pro primární proměnnou dle [22, s. 17]. Bez ní by simplexové metody nebylo možné matematicky dokázat. Ani metody vnitřního bodu [19], nahrazující historicky simplexové metody, nemají obecně zakomponovány principy duality. Aspekty snížení výpočetní náročnosti metod v oblasti optimalizace a priori podporuje právě využití teorie duality.

Publikace [4] navíc kromě teorie duality [22] využívá úsporné modifikace v podobě eliminace omezení [4, s. 67, krok 2], jež lze zpětně dokázat odvozením speciálního tvaru podmínek Kuhn–Tuckerovy věty [14]. Tento jev možného převodu neklasické úlohy na vázaný extrém na úlohu volného extrému není zaručen v každém okamžiku diskrétního MPC, i když zajišťuje redukci operací v rámci algoritmu MPC. Matematický princip této modifikace lze též vyjádřit jako eliminace všech podmínek z definice omezení veličin MPC a test, zda-li výsledek této modifikované úlohy spadá pod omezení původního problému. Zde je pole působnosti pro navázání vývoje modifikací v rámci disertační práce. Budou navrženy šetrnější eliminace v definici omezení veličin MPC spolu s vlastním návrhem optimalizační metody, a to s cílem snížit výpočetní náročnost algoritmu MPC procesů s rychlou dynamikou, aby je bylo možné řídit i při náročnějších požadavcích kladených na řízení. Přičemž nutnou podmínkou je, aby bylo navrhované řešení aplikovatelné v každém okamžiku diskrétního MPC.

3. CÍLE DISERTAČNÍ PRÁCE

V návaznosti na současný stav poznání v oblasti prediktivního řízení procesů s rychlou dynamikou (kap. 2) jsou stanoveny následné cíle disertační práce.

Cílem disertační práce je snížit výpočetní náročnost algoritmu online MPC při řízení procesů s rychlou dynamikou s uvažováním vstupně–výstupních modelů. Předpokládá se navíc zvýšení náročnosti zadání tohoto problému, a to zejména snížením periody vzorkování, navýšením počtu omezujících podmínek v MPC, navyšování hodnoty horizontů a navíc také uvažování dvourozměrnosti řízení. Schopnost prediktivně řídit procesy s rychlou dynamikou při takto stanovených podmínkách bude analyzována měřením času provedení algoritmu v rámci periody vzorkování diskrétního MPC. Dále bude proveden rozbor výpočetní náročnosti navrhovaných řešení pro MPC ve srovnání se zavedenými přístupy v rámci současného stavu řešené problematiky.

Budou navrženy takové přístupy k řešení prediktivního řízení procesů s rychlou dynamikou, jež navazují na myšlenku eliminace podmínek v definici omezení veličin MPC [4, s. 67, krok 2]. Přičemž zavedený přístup předpokládá speciální případ tvaru podmínek Kuhn-Tuckerovy věty a celkově eliminuje všechna omezení v prediktivním řízení. Nevýhodou zavedeného přístupu je možnost jeho částečné aplikace v MPC řízení, neboť tento uvedený jev týkající se podmínek Kuhn–Tuckerovy věty nenastává vždy. Budou proto navrženy šetrnější eliminace omezení, aby byl návrh účinný po celou dobu řízení MPC procesů s rychlou dynamikou a bylo tím zajištěno provedení algoritmu řízení v rámci stanovené periody vzorkování.

Součástí navrhovaných řešení bude vlastní návrh optimalizační metody. Tato metoda bude řešit úlohu kvadratického programování a bude navržena na základě poznatků teorie optimalizace opírající se o teorii duality a souvislosti v rámci podmínek Kuhn–Tuckerovy věty. Předložený návrh bude verifikován a porovnán se zavedenou Hildrethovou metodou [4],[14].

Pro účely verifikace metod v oblasti optimalizace bude sestaven algoritmus analytického řešení úlohy kvadratického programování. Tento postup bude založen na zavedení doplňkových proměnných a bude prakticky uplatnitelný oproti obecnému řešení původní úlohy – nalezení sedlového bodu Lagrangeovy funkce [14]. Ověření správnosti optimalizačních metod tak bude alternativou vůči verifikaci interním příkazem *quadprog* programu MATLAB.

Algoritmem analytického řešení úlohy kvadratického programování bude též ověřen a tím i dokázán výskyt jevu možného převodu řešení neklasické úlohy na vázaný extrém na řešení úlohy volného extrému, což je zapříčiněno zmíněným speciálním tvarem podmínek Kuhn–Tuckerovy věty. Navrženým algoritmem bude tedy navíc matematicky prokázán základ zavedeného přístupu k řešení MPC dle [4, s. 67, krok 2].

Navrhované i zavedené přístupy k řešení problematiky MPC procesů s rychlou dynamikou budou implementovány do podoby skriptů a softwarové aplikace s GUI rozhraním v prostředí MATLAB 6.5. Tato softwarová řešení budou umožňovat provedení simulací a analýzy výpočetní časové náročnosti daných přístupů k řešení, a to s ohledem na kontrolu kvality řízení. Cílem je ověřit, zda-li navrhované přístupy k řešení MPC snižují výpočetní náročnost algoritmu MPC a zda-li jsou účinné v případech, kdy zavedenými přístupy nelze realizovat prediktivní řízení v rámci stanovené periody vzorkování.

Inovace prediktivního řízení procesů s rychlou dynamikou se opírají o výzkum v rámci doktorského studia autora. Dosud neprezentované návrhy metod mohou být dále předmětem publikací v odborných časopisech.

4. TEORIE ZAVEDENÝCH PŘÍSTUPŮ A JEJICH MODIFIKACÍ

4.1 Základy optimalizačních technik

4.1.1 Základní terminologie optimalizace

Bod \mathbf{x} n -rozměrného prostoru je definován jako sloupcový vektor (4.1) s n prvky (souřadnicemi) v Euklidovském n -rozměrném prostoru. [14]

$$\mathbf{x} = [x_1 \quad x_2 \quad \cdots \quad x_n]^T; \mathbf{x} \in \mathcal{R}^n \quad (4.1)$$

Reálná funkce více (n) reálných proměnných $f(\mathbf{x})$, dále též funkce $f(\mathbf{x})$, je definována na množině \mathcal{K} , $\mathcal{K} \subseteq \mathcal{R}^n$ v n -rozměrném vektorovém Euklidovském prostoru \mathcal{R}^n . Funkce $f(\mathbf{x})$ je přiřazením čísla z množiny reálných čísel \mathcal{R} každému bodu $\mathbf{x} \in \mathcal{K}$. Dále je předpokládáno, že je funkce $f(\mathbf{x})$ na množině \mathcal{K} spojitá a diferencovatelná a množina \mathcal{K} je neprázdná a současně uzavřená. [14]

Bodem $\mathbf{x}^* \in \mathcal{K}$ je označen *globální extrém funkce $f(\mathbf{x})$* na množině \mathcal{K} , pokud platí (4.2). Pak je tento bod minimem funkce $f(\mathbf{x})$ na množině \mathcal{K} . Dále bude uvažován případ *unimodální funkce $f(\mathbf{x})$* , což znamená, že má funkce $f(\mathbf{x})$ lokální extrém totožný s globálním extrémem. [14]

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad (4.2)$$

Optimalizační úloha hledání extrému, dále uvažovaná ve formě úlohy hledání minima funkce $f(\mathbf{x})$ na množině \mathcal{K} , je definována dle (4.3). [14]

$$\mathbf{x}^* = \arg \min \{f(\mathbf{x}) \mid \mathbf{x} \in \mathcal{K}\} \quad (4.3)$$

Stacionární bod $\hat{\mathbf{x}}$, též označován jako bod podezřelý z extrému, je takovým řešením v rámci kroků řešení úlohy optimalizace, který po dalším ověření může být klasifikován minimem, maximem či sedlovým bodem funkce. Až dalšími matematickými vztahy může být typ bodu blíže určen. U úlohy hledání minima funkce se u stacionárních bodů ověřuje, zda-li jsou typu minima. [14]

$$\hat{\mathbf{x}} = [\hat{x}_1 \quad \cdots \quad \hat{x}_n]^T \quad (4.4)$$

4.1.2 Klasifikace úloh podle způsobu zadání množiny \mathcal{K}

Bod extrému \mathbf{x}^* funkce $f(\mathbf{x})$ je *volným extrémem*, pokud existuje takové okolí $\mathcal{K}' \in \mathcal{R}^n$ tohoto bodu, jež celé leží na množině \mathcal{K} . V opačném případě je bod \mathbf{x}^* hraničním bodem množiny \mathcal{K} a je nazýván *vázaným extrémem*. [14]

Pokud množina \mathcal{K} není blíže specifikována, je předpokládána její totožnost s definičním oborem funkce $f(\mathbf{x})$, tj. s množinou \mathcal{R}^n . Pro $\mathcal{K} = \mathcal{R}^n$ nazýváme optimalizační úlohu *úlohou volného extrému* (4.5). [14]-[15]

$$\mathbf{x}^* = \arg \min \{ f(\mathbf{x}) \mid \mathbf{x} \in \mathcal{R}^n \} \quad (4.5)$$

Klasická úloha na vázaný extrém je spjata s případem určení množiny \mathcal{K} soustavou m rovnic omezujících podmínek (4.6). Při maticovém zápisu má tato soustava rovnic tvar $\mathbf{M}\mathbf{x} = \boldsymbol{\gamma}$ (4.7), přičemž $\mathbf{M} \in \mathcal{R}^{m,n}$, $\mathbf{x} \in \mathcal{R}^n$, $\boldsymbol{\gamma} \in \mathcal{R}^m$. Matice \mathbf{M} a $\boldsymbol{\gamma}$ obsahují koeficienty soustavy rovnic, což je patrné z (4.7). První index prvků v matici \mathbf{M} odpovídá pořadnici omezující podmínky a druhý index vyjadřuje příslušnost koeficientu k dané proměnné vektoru \mathbf{x} . Obecnější zápis klasické úlohy na vázaný extrém je v rovnici (4.8). [14]-[15]

$$M_j(\mathbf{x}) = \gamma_j, j = 1, 2, \dots, m \quad (4.6)$$

$$\underbrace{\begin{bmatrix} M_{11} & M_{12} & \dots & M_{1n} \\ \vdots & \ddots & & \vdots \\ M_{m1} & M_{m2} & \dots & M_{mn} \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_m \end{bmatrix}}_{\boldsymbol{\gamma}} \quad (4.7)$$

$$\mathbf{x}^* = \arg \min \{ f(\mathbf{x}) \mid \mathbf{M}\mathbf{x} = \boldsymbol{\gamma} \} \quad (4.8)$$

Neklasická úloha na vázaný extrém neboli *úloha matematického programování s nerovnostními omezeními* je definována na množině \mathcal{K} určené soustavou nerovnostních podmínek (4.9), resp. maticovou nerovnicí $\mathbf{M}\mathbf{x} \leq \boldsymbol{\gamma}$, přičemž $\mathbf{M} \in \mathcal{R}^{m,n}$, $\mathbf{x} \in \mathcal{R}^n$, $\boldsymbol{\gamma} \in \mathcal{R}^m$. Tvar neklasické úlohy na vázaný lze zapsat rovnicí (4.10). [14]-[15]

$$M_j(\mathbf{x}) \leq \gamma_j, j = 1, 2, \dots, m \quad (4.9)$$

$$\mathbf{x}^* = \arg \min \{ f(\mathbf{x}) \mid \mathbf{M}\mathbf{x} \leq \boldsymbol{\gamma} \} \quad (4.10)$$

4.1.3 Úloha volného extrému

Úloha (4.5) bez omezujících podmínek je řešitelná pomocí diferenciálního počtu více reálných proměnných. Nejprve je nutno stanovit stacionární body funkce $f(\mathbf{x})$, které jsou podezřelé z přítomnosti extrému, a to vztahem (4.11),

resp. řešením soustavy rovnic (4.12). Přičemž *nulový vektor* $\mathbf{0}$ je sloupcový vektor příslušných rozměrů zahrnující nulové prvky. [14]

$$\nabla f(\mathbf{x}) = \left[\frac{\partial f(\mathbf{x})}{\partial x_1} \quad \frac{\partial f(\mathbf{x})}{\partial x_2} \quad \dots \quad \frac{\partial f(\mathbf{x})}{\partial x_n} \right]^T = \mathbf{0} \quad (4.11)$$

$$\frac{\partial f(\mathbf{x})}{\partial x_i} = 0; i = 1, 2, \dots, n \quad (4.12)$$

Úloha (4.5) je řešitelná, pokud existují spojité první a druhé parciální derivace v funkci $f(\mathbf{x})$ v bodě $\hat{\mathbf{x}}$. [14]

Vyšetření jednotlivých stacionárních bodů na existenci a typ extrému je provedeno prostřednictvím stanovení tzv. definitnosti *Hessovy matice* (4.13) funkce $f(\mathbf{x})$, a to až po dosažení příslušného stacionárního bodu $\hat{\mathbf{x}}$. [14]

$$\nabla^2 f(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{bmatrix} \quad (4.13)$$

Při úloze hledání minima je pro stacionární bod podmíněčné, aby byla Hessova matice pro bod $\hat{\mathbf{x}}$, neboli $\nabla^2 f(\hat{\mathbf{x}})$, tzv. *pozitivně definitní*. Což znamená, že všechny subdeterminanty této matice budou mít kladná znaménka. *Subdeterminanty* jsou stanoveny definicí (4.14). Do vztahů je třeba dosadit stacionární bod $\hat{\mathbf{x}}$ pro konkrétní určení definitnosti Hessovy matice, čímž bude provedeno jeho vyšetření na existenci a typ extrému. [14]-[15]

$$\left\{ \left| \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} \right|, \left| \begin{array}{cc} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} \end{array} \right|, \dots, \left| \begin{array}{ccc} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{array} \right| \right\} \quad (4.14)$$

4.1.4 Klasická úloha na vázaný extrém

Optimalizační úloha (4.8) s minimalizací účelové funkce $f(\mathbf{x})$ dle rovnostních omezujících podmínek, daných funkcemi (4.6), je řešitelná zavedením nové funkce L , jež zahrnuje původní účelovou funkci $f(\mathbf{x})$ v součtu s novým členem reprezentující všechny tyto podmínky. Funkce L je poté podrobena úloze hledání volného extrému, a to bez dalších omezujících požadavků. [14]-[15]

Funkce L je nazývána *Lagrangeovou funkcí* a je tedy nápomocna při převodu optimalizační úlohy kategorie klasické úlohy na vázaný extrém na jednodušší úlohu volného extrému. Ve spojitosti s cílem najít řešení úlohy (4.8) je možné funkci L definovat vyjádřením (4.15). [14]-[15]

$$\begin{aligned} L &= L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \lambda_1(M_1(\mathbf{x}) - \gamma_1) + \dots + \lambda_m(M_m(\mathbf{x}) - \gamma_m) = \\ &= f(\mathbf{x}) + \sum_{j=1}^m \lambda_j(M_j(\mathbf{x}) - \gamma_j); \boldsymbol{\lambda} = [\lambda_1 \quad \lambda_2 \quad \dots \quad \lambda_m]^T; L \in \mathcal{R}^{n+m} \end{aligned} \quad (4.15)$$

Koeficienty $\lambda_j, j = 1, 2, \dots, m$ se nazývají *Lagrangeovými multiplikátory* a tvoří prvky vektoru *Lagrangeových multiplikátorů* $\boldsymbol{\lambda}$ příslušejících dané funkci L . [14]

Řešení původní klasické úlohy na vázaný extrém (4.8) nyní spočívá ve vyšetření Lagrangeovy funkce L na stacionární body, tj. $\hat{\mathbf{x}}$ (4.4) k příslušným multiplikátorům $\hat{\boldsymbol{\lambda}}$ (4.16). [14]

$$\hat{\boldsymbol{\lambda}} = \begin{bmatrix} \hat{\lambda}_1 & \dots & \hat{\lambda}_m \end{bmatrix}^T; \hat{\boldsymbol{\lambda}} \in \mathcal{R}^m \quad (4.16)$$

Stanovení množiny stacionárních bodů v uspořádaných dvojicích $(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}})$ lze určit dle (4.17)-(4.19). [14]

$$\nabla L(\mathbf{x}, \boldsymbol{\lambda}) = \begin{bmatrix} \nabla_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) & \nabla_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) \end{bmatrix}^T = \mathbf{0} \quad (4.17)$$

$$\nabla_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) = \begin{bmatrix} \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_1} & \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_2} & \dots & \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_n} \end{bmatrix}^T \quad (4.18)$$

$$\nabla_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) = \begin{bmatrix} \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial \lambda_1} & \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial \lambda_2} & \dots & \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial \lambda_m} \end{bmatrix}^T \quad (4.19)$$

Vyšetření stacionárních bodů $(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}})$ na přítomnost a typ extrému, lze provést za předpokladu, pokud má funkce $f(\mathbf{x})$ v okolí bodu $\hat{\mathbf{x}}$ spojitě druhé parciální derivace. [14]

Výsledné posouzení stacionárních bodů $(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}})$ je spjato s určením definitnosti Hessovy matice $\nabla_{\mathbf{x}}^2 L(\mathbf{x}, \boldsymbol{\lambda})$ (4.20) Lagrangeovy funkce podle vektorové proměnné \mathbf{x} , a to po dosazení stacionárního bodu $(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}})$. [14]

$$\nabla_x^2 L(\mathbf{x}, \boldsymbol{\lambda}) = \begin{bmatrix} \frac{\partial^2 L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_1^2} & \frac{\partial^2 L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_1 \partial x_n} \\ \vdots & \ddots & & \\ \frac{\partial^2 L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_n \partial x_1} & \frac{\partial^2 L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_n^2} \end{bmatrix} \quad (4.20)$$

Nyní je tedy stanovena definitnost matice $\nabla_x^2 L(\mathbf{x}, \boldsymbol{\lambda})$ pro bod $(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}})$, resp. matice $\nabla_x^2 L(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}})$. V případě její pozitivní definitnosti má účelová funkce $f(\mathbf{x})$ v bodě $\hat{\mathbf{x}}$ vázané minimum \mathbf{x}^* vzhledem k podmínkám $\mathbf{M}\mathbf{x} = \boldsymbol{\gamma}$. Čímž je vyřešena klasická úloha na vázaný extrém s omezeními typu rovnost. [14]

Řešení klasické úlohy na vázaný extrém má podmínění v tzv. *podmínkách regularity*. První podmínka regularity se váže na určení hodnoty tzv. *Jacobiho matice* sestavené z prvních parciálních derivací v tomto případě jednotlivých funkcí (4.21) levých stran rovnostních omezení podle n proměnných vektoru \mathbf{x} (4.23). Musí platit, že hodnota této matice je rovna počtu omezujících podmínek m . Jinými slovy jsou gradienty (4.22) lineárně nezávislé. Celá podmínka by se dala též vyjádřit jako (4.24), kdy determinant Jacobiho matice, tzv. *Jacobián* je pro funkce (4.21) nenulový. [14]

$$M_j(\mathbf{x}); j = 1, 2, \dots, m \quad (4.21)$$

$$\nabla M_j(\mathbf{x}) = \left[\frac{\partial M_j(\mathbf{x})}{\partial x_1} \dots \frac{\partial M_j(\mathbf{x})}{\partial x_n} \right]^T; j = 1, 2, \dots, m \quad (4.22)$$

$$\begin{bmatrix} \nabla M_1(\mathbf{x})^T \\ \vdots \\ \nabla M_m(\mathbf{x})^T \end{bmatrix} = \begin{bmatrix} \frac{\partial M_1(\mathbf{x})}{\partial x_1} & \frac{\partial M_1(\mathbf{x})}{\partial x_2} & \dots & \frac{\partial M_1(\mathbf{x})}{\partial x_n} \\ \vdots & \ddots & & \vdots \\ \frac{\partial M_m(\mathbf{x})}{\partial x_1} & \frac{\partial M_m(\mathbf{x})}{\partial x_2} & \dots & \frac{\partial M_m(\mathbf{x})}{\partial x_n} \end{bmatrix} \quad (4.23)$$

$$\begin{vmatrix} \frac{\partial M_1(\mathbf{x})}{\partial x_1} & \frac{\partial M_1(\mathbf{x})}{\partial x_2} & \dots & \frac{\partial M_1(\mathbf{x})}{\partial x_n} \\ \vdots & \ddots & & \vdots \\ \frac{\partial M_m(\mathbf{x})}{\partial x_1} & \frac{\partial M_m(\mathbf{x})}{\partial x_2} & \dots & \frac{\partial M_m(\mathbf{x})}{\partial x_n} \end{vmatrix} \neq 0 \quad (4.24)$$

Druhá z podmínek je existence řešení rovnice (4.25). V opačném případě by neexistovaly příslušné Lagrangeovy multiplikátory a úloha by neměla řešení. Nechť je maticová rovnice (4.25) zapsatelná v ekvivalentním tvaru (4.26)

s maticemi koeficientů Ξ a σ . Pak řešení existuje, je-li hodnota matice Ξ této soustavy rovnic rovna $m+n$. Nebo též, je-li determinant matice Ξ nenulový. [14]

$$\nabla L(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{0} \quad (4.25)$$

$$\Xi \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \end{bmatrix} - \boldsymbol{\sigma} = \mathbf{0} \quad (4.26)$$

Třetí podmínka regularity vyjadřuje, že všechny funkce (4.21) z definice omezení $\mathbf{M}\mathbf{x} = \boldsymbol{\gamma}$ mají mít v stacionárních bodech $\hat{\mathbf{x}}$ spojitě druhé parciální derivace. [14]

4.1.5 Neklasická úloha na vázaný extrém

V terminologii optimalizace se neklasická úloha vázaného extrému označuje též úlohou matematického programování. Je řešena úloha (4.10), kdy je minimalizována účelová funkce $f(\mathbf{x})$ za přítomnosti nerovnostních omezení daných funkcemi (4.9). [14]–[15]

Pro obecné řešení neklasické úlohy na vázaný extrém neexistují tak patrná pravidla jako u klasické úlohy, nýbrž jsou zmapovány podmínky a vlastnosti úlohy, kterými se inspiřují algoritmy numerických metod jejího řešení. [14]

Spíše teoretický než přímý aplikační význam má tzv. *Kuhn–Tuckerova věta*. Jedná se o důležitou větu matematického programování, jež je naplňována numerickými algoritmy ve větší či menší míře. Jedná se o soubor pravidel ve formě *Kuhn–Tuckerových podmínek* (nebo též podmínek Kuhn–Tuckerovy věty), které musí splňovat řešení úlohy (4.10) [14]:

Nechť $\mathbf{x}^* \in \mathcal{K} \subseteq \mathcal{R}^n$; $\mathcal{K}: \mathbf{M}\mathbf{x} \leq \boldsymbol{\gamma}$ a necht' všechny funkce omezujících podmínek (4.21) jsou lineární a platí všechny podmínky regularity uvedené v kapitole 4.1.4. Necht' funkce $f(\mathbf{x})$ má v okolí bodu \mathbf{x}^* spojitě první parciální derivace. Pokud má funkce $f(\mathbf{x})$ v bodu \mathbf{x}^* minimum vzhledem na množinu \mathcal{K} , pak existuje takový vektor Lagrangeových multiplikátorů (4.27), že platí Kuhn–Tuckerovy podmínky (4.28)–(4.31) s indexací $j = 1, \dots, m$. Kde $\mathbf{0}$ je nulový vektor příslušných rozměrů. [14], [20]

$$\boldsymbol{\lambda}^* = [\lambda_1^* \quad \dots \quad \lambda_m^*]^T \quad (4.27)$$

$$\nabla f(\mathbf{x}^*) + \sum_{j=1}^m \lambda_j^* \nabla M_j(\mathbf{x}^*) = \mathbf{0}; \boldsymbol{\lambda}^* = [\lambda_1^* \quad \dots \quad \lambda_m^*]^T \quad (4.28)$$

$$M_j(\mathbf{x}^*) - \gamma_j \leq 0 \quad (4.29)$$

$$\lambda_j^*(M_j(\mathbf{x}^*) - \gamma_j) = 0 \quad (4.30)$$

$$\lambda_j^* \geq 0 \quad (4.31)$$

Vektorově lze Kuhn–Tuckerovy podmínky vyjádřit v podobě (4.32)–(4.35), kde u podmínky (4.34) je sice na levé straně rovnice vektorové vyjádření, avšak na její pravé straně není nulový vektor nýbrž skalár. [14], [20]

$$\nabla f(\mathbf{x}^*) + \dot{\lambda}^T \mathbf{M} = \mathbf{0} \quad (4.32)$$

$$\mathbf{M}\mathbf{x}^* - \boldsymbol{\gamma} \leq \mathbf{0} \quad (4.33)$$

$$\dot{\lambda}^T (\mathbf{M}\mathbf{x}^* - \boldsymbol{\gamma}) = 0 \quad (4.34)$$

$$\dot{\lambda} \geq \mathbf{0} \quad (4.35)$$

Pro optimální řešení neklasické úlohy na vázaný extrém platí ne přímo aplikovatelné pravidlo o sedlovém bodu Lagrangeovy funkce (4.15). [14]

Funkce $f(\mathbf{x})$ má v bodě \mathbf{x}^* při podmínkách omezení (4.9) vázané optimum, pokud je bod $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$; $\boldsymbol{\lambda}^* \geq \mathbf{0}$ sedlovým bodem Lagrangeovy funkce (4.15), tedy platí-li (4.36). [14]

$$L(\mathbf{x}^*, \boldsymbol{\lambda}) \leq L(\mathbf{x}^*, \boldsymbol{\lambda}^*) \leq L(\mathbf{x}, \boldsymbol{\lambda}^*); \mathbf{x} \in \mathcal{R}^{n,1}, \boldsymbol{\lambda} \in \mathcal{R}^{m,1} \quad (4.36)$$

Pozn. U klasické úlohy na vázaný extrém je hledáno minimum Lagrangeovy funkce, kdežto pro neklasickou úlohou vázaného extrému je optimum souvislé se sedlovým bodem Lagrangeovy funkce.

4.1.6 Kvadratické programování

V aplikační oblasti disertační práce bude obecná neklasická úloha na vázaný extrém uvažována ve specifickém tvaru, a to s kvadratickou účelovou funkcí a lineárními omezeními (4.21). Úloha (4.37) je nazývána *úlohou kvadratického programování*. [14], [15], [17]

$$\mathbf{x}^* = \arg \min \left\{ \frac{1}{2} \mathbf{x}^T \mathbf{H}\mathbf{x} + \mathbf{b}^T \mathbf{x} \mid \mathbf{M}\mathbf{x} \leq \boldsymbol{\gamma} \right\}; \mathbf{H} \in \mathcal{R}^{n,n}, \mathbf{b} \in \mathcal{R}^{n,1} \quad (4.37)$$

Čtvercová matice \mathbf{H} (4.38) stupně n odpovídá Hessově matici neklasické úlohy na vázaný extrém. Pro hledání minima účelové funkce je třeba, aby byla pozitivně definitní, jak bylo zmíněno v předchozím obecnějším výkladu. Matice \mathbf{H} je také nazývána maticí kvadratické formy dané v rámci vztahu (4.39). Znalost tohoto poznatku je výhodná při konstrukci zadání úlohy kvadratického programování z polynomiálním tvaru na maticový (4.37). [14]

$$\mathbf{H} = \begin{bmatrix} H_{11} & H_{12} & \cdots & H_{1n} \\ H_{21} & H_{22} & \cdots & H_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ H_{n1} & H_{n2} & \cdots & H_{nn} \end{bmatrix} \quad (4.38)$$

$$\mathbf{x}^T \mathbf{H} \mathbf{x} = \sum_{p=1}^n \sum_{q=1}^n H_{pq} x_p x_q = \left\{ \begin{array}{l} H_{11}x_1^2 + H_{12}x_1x_2 + \cdots + H_{1n}x_1x_n + \\ + H_{21}x_2x_1 + H_{22}x_2^2 + \cdots + H_{2n}x_2x_n + \\ \vdots \\ + H_{n1}x_nx_1 + H_{n2}x_nx_2 + \cdots + H_{nn}x_n^2 \end{array} \right\} \quad (4.39)$$

4.1.7 Teorie duality v kvadratickém programování

Teorie duality definuje pojem *dualita* jako vztah mezi dvěma vektorovými prostory. Tato vlastnost je využívána při tvorbě algoritmů neklasické úlohy na vázaný extrém, jejich zkoumání a matematickém dokazování. [22]

Před uvedením vztahů duality je nutné vyjádření její reflexe vůči primární úloze, jež je definována následovně:

Primární úlohou kvadratického programování se rozumí úloha ve tvaru (4.37). Jedná se o speciální tvar neklasické úlohy na vázaný extrém, jenž je oproti duální úloze výpočetně náročnější. [4], [14]

Duální úloha kvadratického programování je definována jako úloha optimalizace (4.44) s hledáním extrému \mathbf{d}^* (4.45) duální proměnné \mathbf{d} . Souvislosti mezi maticemi a vektory primární úlohy a duální úlohy jsou dány převodními vztahy (4.40), (4.41). V rámci duální úlohy je Hessián \mathbf{N} (4.40)–(4.42) a vektor \mathbf{o} (4.41)–(4.43) závislý na informacích o omezeních. Úloha (4.44) může být výpočetně jednodušší oproti příslušující primární úloze v rámci vztahu duality. [14]

$$\mathbf{N} = \mathbf{M}\mathbf{H}^{-1}\mathbf{M}^T; \mathbf{N} \in \mathcal{R}^{m,m} \quad (4.40)$$

$$\mathbf{o} = \mathbf{M}\mathbf{H}^{-1}\mathbf{b}^T + \boldsymbol{\gamma}; \mathbf{o} \in \mathcal{R}^{m,1} \quad (4.41)$$

$$\mathbf{N} = \begin{bmatrix} N_{11} & N_{12} & \cdots & N_{1m} \\ N_{21} & N_{22} & \cdots & N_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ N_{m1} & N_{m2} & \cdots & N_{mm} \end{bmatrix} \quad (4.42)$$

$$\mathbf{o} = [\mathbf{o}_1 \cdots \mathbf{o}_m]^T \quad (4.43)$$

$$\mathbf{d}^\bullet = \arg \min \left\{ \frac{1}{2} \mathbf{d}^T \mathbf{N} \mathbf{d} + \mathbf{o}^T \mathbf{d} \mid \mathbf{d} \geq \mathbf{0} \right\}; \mathbf{d} \in \mathcal{R}^{m,1} \quad (4.44)$$

$$\mathbf{d}^\bullet = [d_1^\bullet \cdots d_m^\bullet]^T \quad (4.45)$$

Využití duality kvadratického programování je možné pro řešení primární úlohy (4.37), a to nejprve jejím převodem na tvar (4.44) s určením matice (4.40) a vektoru (4.41). Po vyřešení duální úlohy je řešení původní primární úlohy (4.37) dáno převodním vztahem (4.46). [4]

$$\mathbf{x}^\bullet = -\mathbf{H}^{-1}(\mathbf{M}^T \mathbf{d}^\bullet + \mathbf{b}^T) \quad (4.46)$$

Pro řešení samotné duální úlohy (4.44) je třeba, aby metoda řešení zohledňovala zahrnutí podmínky nezápornosti pro hledaný vektor \mathbf{d}^\bullet .

Pozn. Při porovnání obou úloh (4.37) a (4.44) je patrné, že v duální úloze výrazně ubylo omezujících podmínek.

Pozn. Pokud by byla úloha kvadratického programování řešena analyticky (s odvozením v kap. 5.1) jako neklasická úloha na vázaný extrém a byla uvažována Lagrangeova funkce L , byl by vektor Lagrangeových multiplikátorů roven řešení duální úlohy kvadratického programování (4.47). [14]

$$\boldsymbol{\lambda}^\bullet \equiv \mathbf{d}^\bullet \quad (4.47)$$

4.2 Přístupy řešení neklasické úlohy na vázaný extrém

4.2.1 Analytický převod na klasickou úlohu na vázaný extrém

Každé omezení typu nerovnost v rámci maticové nerovnice $\mathbf{M}\mathbf{x} \leq \boldsymbol{\gamma}$ lze převést pomocí *doplňkových proměnných* na podmínky typu rovnost (4.48). Výhodou je převod úlohy (4.10) na modifikovanou (4.49) v souladu se zadáním klasické úlohy na vázaný extrém. Nevýhodou tohoto převodu je nárůst počtu proměnných v optimalizační úloze. [14]

$$M_j(\mathbf{x}) + \bar{y}_j^2 = \gamma_j; j = 1, 2, \dots, m \quad (4.48)$$

$$\mathbf{x}^* = \arg \min \{f(\mathbf{x}) \mid \mathbf{M}\mathbf{x} + \bar{\mathbf{y}} = \boldsymbol{\gamma}\}; \bar{\mathbf{y}} = [\bar{y}_1^2 \cdots \bar{y}_m^2]^T; \bar{\mathbf{y}} \in \mathcal{R}^m \quad (4.49)$$

Novou podobu získává též Lagrangeova funkce značená dále $\tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})$ (4.50). Nalezení stacionárních bodů je podmíněno rovnicí (4.51). [14]

$$\tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \sum_{j=1}^m \lambda_j (M_j(\mathbf{x}) + \bar{y}_j^2 - \gamma_j) \quad (4.50)$$

$$\nabla \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) = [\nabla_{\mathbf{x}} \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) \quad \nabla_{\bar{\mathbf{y}}} \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) \quad \nabla_{\boldsymbol{\lambda}} \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})]^T = \mathbf{0} \quad (4.51)$$

V souvislosti s řešením soustavy rovnic (4.51), konkrétně její části (4.52) s řešením (4.53), nastává mnoho kombinací, v nichž jsou některé proměnné nulové a některé nikoliv. Dále je potřeba vyšetřit takto určené stacionární body na přítomnost extrému, a to jejich dosazením do příslušné Hessovy matice (4.54) a zjištěním definitnosti takto vzniklých matic. [14]

$$\nabla_{\bar{\mathbf{y}}} \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) = \mathbf{0} \quad (4.52)$$

$$\left[\frac{\partial \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial \bar{y}_1} \cdots \frac{\partial \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial \bar{y}_m} \right]^T = [2\lambda_1 \bar{y}_1 \cdots 2\lambda_m \bar{y}_m]^T = [0 \cdots 0]^T = \mathbf{0} \quad (4.53)$$

$$\nabla_{\mathbf{x}}^2 \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) = \begin{bmatrix} \frac{\partial^2 \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial x_1^2} & \frac{\partial^2 \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial x_1 \partial x_n} \\ \vdots & \ddots & & \\ \frac{\partial^2 \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial x_n \partial x_1} & \frac{\partial^2 \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial x_n^2} \end{bmatrix} \quad (4.54)$$

4.2.2 Existence jevu převodu na úlohu volného extrému

Za účelem snížení výpočetní náročnosti řešení neklasické úlohy na vázaný extrém je účelná modifikace úlohy spočívající na matematickém jevu [14, s. 46, 1. bod důkazu]. Tato modifikace je založena na totožnosti řešení (4.5) úlohy volného extrému s řešením (4.10) neklasické úlohy na vázaný extrém, a to v určitých případech úlohy, nikoliv ve všech. Jedná se o situaci, kdy po odstranění všech omezení neklasické úlohy (4.10) náleží řešení úlohy volného extrému (4.5) pod všechna omezení originální úlohy (4.10). [14]

Z toho vyplývá, že je tento jev podmíněn vztahem (4.55). Pokud nenastane, je nutno řešit kompletní originální úlohu (4.10). Začlenění podmínky (4.55) do optimalizační metody tedy znamená řešit nejprve úlohu volného extrému a její řešení testovat na příslušnost k množině omezení neklasické úlohy (4.10).

$$\mathbf{x}^* : (\nabla f(\mathbf{x}^*) = \mathbf{0}) \wedge (\mathbf{M}\mathbf{x}^* \leq \boldsymbol{\gamma}) \quad (4.55)$$

Pro případ kvadratického programování (4.37) by se jednalo o řešení příslušné redukované úlohy volného extrému (4.56) a následné testování tohoto řešení podmínkou (4.57). [4], [14]

$$(\nabla f(\mathbf{x}^*) = \mathbf{0}) \Leftrightarrow (\mathbf{x}^* = -\mathbf{H}^{-1}\mathbf{b}) \quad (4.56)$$

$$\mathbf{x}^* : (\mathbf{M}\mathbf{x}^* \leq \boldsymbol{\gamma}) \Leftrightarrow (-\mathbf{M}\mathbf{H}^{-1}\mathbf{b} \leq \boldsymbol{\gamma}) \quad (4.57)$$

Možnost výskytu popsaného jevu lze ověřit Kuhn–Tuckerovými podmínkami. Čtvrtá podmínka (4.31), resp. (4.35), připouští buď nulové nebo kladné hodnoty Lagrangeových multiplikátorů pro řešení neklasické úlohy na vázaný extrém (4.10). První podmínka (4.28) se stane podmínkou existence volného extrému pouze pro nulové Lagrangeovy multiplikátory tj. platí ekvivalence (4.58), v případě kvadratického programování (4.59). Tato redukce připouští výskyt situace, kdy je řešení úlohy volného extrému současně řešením neklasické úlohy na vázaný extrém. Kdy budou Lagrangeovy multiplikátory nulové, resp. bude-li možné provést popsanou modifikaci záleží pouze na tvarech matic původní úlohy (4.10). Proto je nutné vždy provést testování (4.55), má-li být modifikace aplikována do metod optimalizace. [14]

$$\left(\nabla f(\mathbf{x}^*) + \sum_{j=1}^m \lambda_j^* \nabla M_j(\mathbf{x}^*) = \mathbf{0}, \lambda^* = [0 \dots 0]^T \right) \Leftrightarrow (\nabla f(\mathbf{x}^*) = \mathbf{0}) \quad (4.58)$$

$$\left((\mathbf{x}^*)^T \mathbf{H} + \mathbf{b}^T + (\lambda^*)^T \mathbf{M} = \mathbf{0}, \lambda^* = [0 \dots 0]^T \right) \Leftrightarrow (\mathbf{x}^* = -\mathbf{H}^{-1}\mathbf{b}) \quad (4.59)$$

4.2.3 Využití teorie duality v kvadratickém programování

Numerická duální Hildrethova metoda je iterační optimalizační metodou spočívající v numerickém řešení úlohy kvadratického programování (4.44). Její jádro určuje duální řešení \mathbf{d}^* . Následně je proveden jeho převod, na základě zákonitosti (4.46), na finální primární řešení původního tvaru zadání problému (4.37). [4], [14]

Kromě respektování vlastností teorie duality, metoda garantuje takové řešení, jež splňuje vlastnosti Kuhn–Tuckerových podmínek, obzvláště s důrazem na 4. podmínku o nezápornosti řešení duální úlohy. Oproti primární podobě úlohy je duální úloha menšího výpočetního rozsahu, a to pokud platí $n < m$. [14]

Numerické výsledky $\mathbf{d}(k)$ v jednotlivých iteracích metody konvergují k řešení duální úlohy \mathbf{d}^* . Postup je ukončen buď po provedení maximálního počtu iterací nebo přirozeněji, a to pokud se výsledky aktuálního $\mathbf{d}(k)$ a minulého kroku $\mathbf{d}(k-1)$ velmi nepatrně liší nebo jsou shodné. Pak je výsledné řešení \mathbf{d}^* rovno hodnotě vektoru $\mathbf{d}(k)$. Pro inicializaci počátečního vektoru $\mathbf{d}(k=0)$, resp. $\mathbf{d}(0)$, je uvažován nulový vektor rozměru m . [4], [14]

V konkrétní iteraci jsou vypočteny hodnoty jednotlivých prvků aktuálního vektoru $\mathbf{d}(k)$ (4.60) na základě informací z Hessiánu duální úlohy N (4.42), vektoru \mathbf{o} (4.43) a hodnot vektoru $\mathbf{d}(k-1)$ předchozího kroku. [14]

$$\mathbf{d}(k) = [d_1(k) \cdots d_m(k)]^T; \mathbf{d}(k) \in \mathcal{R}^{m,1} \quad (4.60)$$

Výpočty jednotlivých prvků vektoru $\mathbf{d}(k)$ jsou vázány na jejich pořadí v tomto vektoru. Pro první prvek je použit výpočet (4.61), pro druhý až předposlední (4.62) a pro poslední prvek vektoru (4.63). Podmínkou nasazení Hildrethovy metody jsou nenulové diagonální prvky Hessiánu N . [14]

$$d_1(k) = \max\left(-\frac{1}{N_{11}} \left(\sum_{j=2}^m N_{1j} d_j(k-1) + o_1\right), 0\right) \quad (4.61)$$

$$\left. \begin{aligned} & \forall i, i \in \{2, \dots, (m-1)\}: d_i(k) = \\ & = \max\left(-\frac{1}{N_{ii}} \left(\sum_{j=1}^{i-1} N_{ij} d_j(k) + \sum_{j=i+1}^m N_{ij} d_j(k-1) + o_i\right), 0\right) \end{aligned} \right\} \quad (4.62)$$

$$d_m(k) = \max\left(-\frac{1}{N_{mm}} \left(\sum_{j=1}^{m-1} N_{mj} d_m(k) + o_m\right), 0\right) \quad (4.63)$$

4.3 Prediktivní řízení procesů s rychlou dynamikou

4.3.1 Prediktivní řízení procesů

Prediktivní řízení je moderní problematika řešená v oblasti teorie automatického řízení. Je vhodná pro řízení širokého spektra procesů (např. mnohorozměrové, neminimálně fázové, s dopravním zpožděním, nestabilní). Bude diskutováno neadaptivní prediktivní řízení pro jednorozměrové a dvourozměrové lineární procesy s rychlou dynamikou [8]. Varianta bude zahrnovat působení neměřitelné poruchové veličiny. Bude uvažován vstupně-výstupní model řízeného procesu. [1]–[6]

Z pohledu struktury se prediktivní regulátor dělí na dva subsystémy – *prediktor* a *optimalizátor*, jež vzájemně spolupracují na základě strategie pohyblivého horizontu. Tento princip je aplikován při každém dalším kroku diskrétního řízení. Parametry, jež jsou spjaty s definicí tzv. *okna horizontu*, jsou *minimální, řídicí a maximální horizont* – značené N_1, N_u, N_2 . [1]–[6]

Pokud se nejedná o systém s dopravním zpožděním, bývá z pravidla minimální horizont uvažován s hodnotou $N_1 = 1$. [1]–[6]

Předností prediktivního řízení je možnost zakomponovat požadavek na *omezení veličin*, a to na velikost jejich rozsahů a změn, již při samotném návrhu algoritmu řízení. Lze určovat meze řídicích a regulovaných veličin. [1]–[6]

Budoucí výstupy řízeného procesu jsou stanoveny prostřednictvím rovnic prediktoru, a to prostřednictvím v něm zahrnutého modelu řízeného procesu. Tyto predikce jsou stanoveny vzhledem k aktuálním a minulým informacím z diskrétního řízení. Vše lze definovat jednou maticovou rovnicí. Bude uvažováno odvození zákonitostí pro modely druhého řádu, jimiž lze modelovat procesy vyskytující se v praxi ve větší míře. Jelikož je v těchto predikčních rovnicích více neznámých veličin, jsou dále určovány v druhém subsystému – optimalizátoru, ve kterém je vyčíslen *zákon řízení*. Do dalšího kroku řízení je použit první člen výsledku predikovaného průběhu přírůstků akčních zásahů. [1]–[6]

Tvar zákona řízení není u prediktivního řízení analogický se situací např. u PSD regulátorů [7]. V rámci optimalizátoru je řešena optimalizační úloha, kde jsou budoucí přírůstky akční veličiny minimalizujícím argumentem účelové funkce. Je dán požadavek na minimalizaci regulačních odchylek a změn budoucích přírůstků akčních zásahů na okně horizontu. Pro tento požadavek je společně s uvažováním omezujících podmínek veličin MPC vhodnou metodou řešení optimalizační úloha typu kvadratického programování. [1]–[6]

4.3.2 Procesy s rychlou dynamikou

Proces s rychlou dynamikou je takový proces, jenž má velmi malé časové konstanty τ_i, T_j a je mu přiřazena velmi malá perioda vzorkování T . [7], [8]

Pod označením velmi malá je zde uvažována doba v řádu milisekund. Např. ve výzkumu MPC s rychlou dynamikou [23] se lze setkat s hodnotou periody vzorkování 5 ms. V dalších publikacích např. s T v řádu desítek ms.

Matematický model řízeného procesu druhého řádu je v případě SISO možné popsat spojitou přenosovou funkcí (4.64) a ve variantě TITO přenosovou maticí (4.65) se spojitými dílčími přenosy. Kde K , resp. $K_{ij}; i, j \in \{1,2\}$, vyjadřuje zesílení daného spojitého přenosu. Pro dílčí přenosy budou dále uvažovány shodné jmenovatele dílčích přenosů. [7], [9]

$$G(s) = \frac{s - \mathcal{G}_1}{(s - \pi_1)(s - \pi_2)} = K \frac{(1 + s\tau_1)}{(1 + sT_1)(1 + sT_2)} \quad (4.64)$$

$$\mathbf{G}(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \quad (4.65)$$

$$\mathbf{G}(s) = \left. \begin{array}{l} \begin{bmatrix} \frac{s - \mathcal{G}_1}{(s - \pi_1)(s - \pi_2)} & \frac{s - \mathcal{G}_2}{(s - \pi_3)(s - \pi_4)} \\ \frac{s - \mathcal{G}_3}{(s - \pi_5)(s - \pi_6)} & \frac{s - \mathcal{G}_4}{(s - \pi_7)(s - \pi_8)} \end{bmatrix} \\ \begin{bmatrix} K_{11} \frac{(1 + s\tau_1)}{(1 + sT_1)(1 + sT_2)} & K_{12} \frac{(1 + s\tau_2)}{(1 + sT_3)(1 + sT_4)} \\ K_{21} \frac{(1 + s\tau_3)}{(1 + sT_5)(1 + sT_6)} & K_{22} \frac{(1 + s\tau_4)}{(1 + sT_7)(1 + sT_8)} \end{bmatrix} \end{array} \right\}$$

Časové konstanty čitatele přenosu (4.64), resp. dílčích přenosů v matici (4.65), souvisí s reálnými hodnotami nul (nulových bodů) \mathcal{G}_i modelu procesu dle (4.66). Obdobně časové konstanty jmenovatele jsou ve vztahu (4.67) s reálnými hodnotami pólů π_j . [7]

$$\tau_i = -\frac{1}{\mathcal{G}_i} \quad (4.66)$$

$$T_j = -\frac{1}{\pi_j} \quad (4.67)$$

Jelikož platí mezi jednotlivými časovými konstantami pólů a nul nepřímá úměrnost, jsou procesy s rychlou dynamikou charakteristické vyššími hodnotami pólů a nul v řádu desítek (v absolutní hodnotě). Modely řízených procesů budou dále zadávány prostřednictvím definic pólů a nul. [7]

Stanovení periody vzorkování je uvažováno dle doporučení (4.68), které se zakládá na závěrech Kotelnikov–Shannonovy věty [7, s. 414]. Kde \bar{T} je definována jako doba dosažení 95% ustálené hodnoty přechodové charakteristiky modelu řízeného procesu. Dle tohoto předpisu bude uvažována perioda vzorkování T pro diskretizaci spojitého modelu řízeného procesu do podoby diskrétní přenosové funkce (4.69) pro SISO procesy nebo diskrétní přenosové matice (4.70) pro TITO procesy. [7]–[9], [23]

$$T \approx \left(\frac{1}{6} \bar{T}; \frac{1}{15} \bar{T} \right) \quad (4.68)$$

$$G(z^{-1}) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} = \frac{B(z^{-1})}{A(z^{-1})} \quad (4.69)$$

$$\mathbf{G}(z^{-1}) = \mathbf{A}^{-1}(z^{-1})\mathbf{B}(z^{-1}) \quad (4.70)$$

Diskrétní přenosová matice (4.70) s tzv. maticovými zlomky \mathbf{A} a \mathbf{B} musí být pro účely algoritmu dvourozměrného prediktivního řízení vyjádřena v ekvivalentním popisu pomocí dílčích polynomů dle (4.71). [1]–[4], [9]

$$\left. \begin{aligned} \mathbf{G}(z^{-1}) &= \begin{bmatrix} \alpha_{11}(z^{-1}) & \alpha_{12}(z^{-1}) \\ \alpha_{21}(z^{-1}) & \alpha_{22}(z^{-1}) \end{bmatrix}^{-1} \begin{bmatrix} \beta_{11}(z^{-1}) & \beta_{12}(z^{-1}) \\ \beta_{21}(z^{-1}) & \beta_{22}(z^{-1}) \end{bmatrix} \\ \alpha_{11}(z^{-1}) &= 1 + \alpha_{111}z^{-1} + \alpha_{112}z^{-2}; \alpha_{12}(z^{-1}) = \alpha_{121}z^{-1} + \alpha_{122}z^{-2} \\ \alpha_{21}(z^{-1}) &= \alpha_{211}z^{-1} + \alpha_{212}z^{-2}; \alpha_{22}(z^{-1}) = 1 + \alpha_{221}z^{-1} + \alpha_{222}z^{-2} \\ \beta_{11}(z^{-1}) &= \beta_{111}z^{-1} + \beta_{112}z^{-2}; \beta_{12}(z^{-1}) = \beta_{121}z^{-1} + \beta_{122}z^{-2} \\ \beta_{21}(z^{-1}) &= \beta_{211}z^{-1} + \beta_{212}z^{-2}; \beta_{22}(z^{-1}) = \beta_{221}z^{-1} + \beta_{222}z^{-2} \end{aligned} \right\} \quad (4.71)$$

Přechodová charakteristika je funkcí času $h = h(t)$, jež odpovídá reakci modelu procesu na jednotkovou změnu vstupní veličiny v čase $t = 0$ s. Pro dvourozměrnou variantu procesu se jedná o množinu funkcí odpovídajících všem interakcím vstupních a výstupních veličin. Při uvažování diskrétního modelu procesu lze uvažovat diskrétní variantu přechodové charakteristiky $h = h(k.T) = h(k); k \in Z_0^+$, jež má hodnoty v celočíselných násobcích stanovené periody vzorkování. [7]

4.3.3 Jednorozměrové prediktivní řízení

Základním principem prediktivního řízení je stanovení budoucích hodnot výstupní veličiny a budoucích přírůstků akční veličiny na okně horizontu daném již uvedenými parametry N_1, N_u, N_2 . Budoucí hodnoty těchto veličin jsou určeny na základě minulých a aktuálních hodnot veličin řízení. Je použit matematický model řízeného procesu zahrnutý přímo v prediktivním regulátoru. [1]–[6]

Matematické vztahy v prediktivním řízení zohledňují požadavky kladené na řízení jako celek i na omezení jednotlivých dílčích veličin. Kooperace dvou hlavních subsystémů prediktivního regulátoru spočívá z matematického hlediska v řešení optimalizační úlohy, kde neznámé veličiny jsou vyjádřeny právě predikčními vztahy. [1]–[6]

V této podkapitole bude řešen případ neadaptivního prediktivního řízení jednorozměrného procesu ve vstupně–výstupní oblasti. Predikce na okně horizontu jsou pro diskretní model řízeného procesu (4.69) s uvažovanou periodou vzorkování T přepsány do podoby tzv. *CARIMA modelu* (Controlled Autoregressive Integrated Moving Average) [25] (4.72), jenž místo samotných akčních zásahů $u(k)$ zahrnuje jejich přírůstky $\Delta u(k)$. Je v něm též začleněn model neměřitelné poruchové veličiny. Signál e_s reprezentuje neměřitelný šum s nulovou střední hodnotou. Jednotlivé polynomy A a B odpovídají polynomům v diskretní přenosové funkci (4.69). Pro zjednodušení bude dále uvažováno $C(z^{-1}) = 1$. Rovnici (4.72) lze při uvažování těchto předpokladů vyjádřit ve tvaru (4.73) a dále odvodit vztah pro určení aktuální hodnoty výstupní veličiny $y(k)$ (4.74) v daném okamžiku diskretního řízení. Budoucí hodnoty výstupní veličiny lze rekurentně určit dle (4.74) s výslednou maticovou rovnicí (4.75) již bez uvažování náhodné složky. [1]–[6]

$$A(z^{-1})y(k) = B(z^{-1})u(k) + \frac{C(z^{-1})}{\Delta} e_s(k); \Delta = 1 - z^{-1} \quad (4.72)$$

$$(1 - z^{-1})(1 + a_1 z^{-1} + a_2 z^{-2})y(k) = (b_1 z^{-1} + b_2 z^{-2})\Delta u(k) + e_s(k) \quad (4.73)$$

$$y(k) = (1 - a_1)y(k-1) + (a_1 - a_2)y(k-2) + a_2 y(k-3) + b_1 \Delta u(k-1) + b_2 \Delta u(k-2) \quad (4.74)$$

$$\underbrace{\begin{bmatrix} y(k+N_1) \\ \vdots \\ y(k+N_2) \end{bmatrix}}_{\mathbf{y}} = \mathbf{P} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \\ \Delta u(k-1) \end{bmatrix} + \mathbf{G} \underbrace{\begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_u-1) \end{bmatrix}}_{\Delta \mathbf{u}} \quad (4.75)$$

Pozn. Ve vztahu (4.75) znamená symbol \mathbf{G} matici obsahující hodnoty přechodové funkce a nemá souvislost se značením matic přenosových funkcí pro popis modelu TITO procesu: $\mathbf{G}(s)$ a $\mathbf{G}(z^{-1})$. Odlišnost symbolů zajišťuje závorka s argumenty. [1]–[6]

První člen součtu (volná odezva) v maticové rovnici (4.75) s maticí \mathbf{P} pracuje s prvky odpovídajícími aktuálním nebo minulým informacím z řízení, kdežto druhý člen součtu s maticí \mathbf{G} vyjadřuje neznámé budoucí hodnoty (nucená odezva). Tvary matic \mathbf{P} a \mathbf{G} jsou pro rovnost řídicího a maximálního horizontu vyjádřeny v definicích (4.76) a (4.77), s prvky G_{ij} a P_{ij} . Pokud je maximální horizont $N_2 > N_u$, pak je pouze nutné odstranit $N_2 - N_u$ dolních řádků z matice \mathbf{G} . [1]–[6]

$$\left. \begin{aligned} & \mathbf{G} \in \mathcal{R}^{N_2 - N_1 + 1, N_2}; \\ & G_{11} = G_{22} = G_{33} = b_1; G_{21} = b_1 - a_1 b_1 + b_2; \\ & G_{31} = G_{21} + a_1^2 b_1 - a_1 b_2 - a_2 b_1; G_{32} = G_{21}; \\ & \left(\begin{aligned} & G_{ij} = b_1; G_{i,j-1} = G_{i,j-1} + b_2 \end{aligned} \right) i = j \\ & \left(\begin{aligned} & G_{ij} = (1 - a_1)G_{i-1,j} + \\ & + (a_1 - a_2)G_{i-2,j} + a_2 G_{i-3,j} \end{aligned} \right) i \neq j \end{aligned} \right\}; i = 4 \dots N_2, j = 1 \dots i \quad (4.76)$$

$$\left. \begin{aligned} & \mathbf{P} \in \mathcal{R}^{N_2, 4}; \\ & P_{11} = 1 - a_1; P_{12} = a_1 - a_2; P_{13} = a_2; P_{14} = b_2; \\ & P_{21} = a_1^2 - a_1 - a_2 + 1; P_{22} = a_1 - a_1^2 + a_1 a_2; \\ & P_{23} = a_2 - a_1 a_2; P_{24} = b_2 - a_1 b_2; \\ & P_{31} = a_1^2 - a_1 + 2a_1 a_2 - a_1^3 + 1 - a_2; \\ & P_{32} = a_1 - a_1^2 + a_1^3 - a_1^2 a_2 - a_1 a_2 + a_2^2; \\ & P_{33} = a_2 - a_1 a_2 + a_1^2 a_2 - a_2^2; \\ & P_{34} = b_2 - a_1 b_2 + a_1^2 b_2 - a_2 b_2; \\ & \left(\begin{aligned} & P_{ij} = (1 - a_1)P_{i-1,j} + (a_1 - a_2)P_{i-2,j} + \\ & + a_2 P_{i-3,j} \end{aligned} \right); i = 4 \dots N_2, j = 1 \dots 4 \end{aligned} \right\} \quad (4.77)$$

V druhé části prediktivního regulátoru je provedeno řešení optimalizační úlohy, jejímž argumentem jsou hledané budoucí přírůstky akčních zásahů. Pro

okno horizontu je těchto prvků N_u a jsou uspořádány do vektoru $\Delta \mathbf{u}$. Použit je do následného kroku diskrétního řízení vždy jen první člen $\Delta \mathbf{u}(k + 1)$. [1]–[6]

Optimalizační úloha je realizací zákona řízení s požadavky na minimalizaci kvadrátu regulačních odchylek a kvadrátu velikostí přírůstků akčních zásahů, a to na okně horizontu. Tyto záležitosti jsou promítnuty do podoby účelové funkce J (4.78). Funkce J je při vektorovém zápisu kvadratického tvaru (4.79). Žádaná veličina je dána na příslušném okně horizontu vektorem \mathbf{w} s N_2 hodnotami. Při zařazení podmínek omezující rozsahy a velikosti hodnot veličin řízení, ve tvaru maticové nerovnice (4.80), se bude jednat o neklasickou úlohu na vázaný extrém (kap. 4.1.5), přesněji úlohu kvadratického programování (kap. 4.1.6). Její konkrétní tvar je pro prediktivní řízení v podobě (4.81). Je možné ho získat maticovými operacemi ze vztahu (4.79), a to po dosazení rovnice (4.75). Tím je provedena redukce neznámých veličin na rovnici zahrnující již pouze neznámý vektor $\Delta \mathbf{u}$. [1]–[6]

$$J = \sum_{i=N_1}^{N_2} [y(k+i) - w(k+i)]^2 + \sum_{j=1}^{N_u} [\Delta \mathbf{u}(k+j-1)]^2 \quad (4.78)$$

$$J = (\mathbf{y} - \mathbf{w})^T (\mathbf{y} - \mathbf{w}) + \Delta \mathbf{u}^T \Delta \mathbf{u} \quad (4.79)$$

$$\mathbf{M} \Delta \mathbf{u} \leq \boldsymbol{\gamma} \quad (4.80)$$

$$\Delta \mathbf{u} = \arg \min \{ J \mid \mathbf{M} \Delta \mathbf{u} \leq \boldsymbol{\gamma} \};$$

$$\Delta \mathbf{u} = \arg \min \left\{ \frac{1}{2} \Delta \mathbf{u}^T \mathbf{H} \Delta \mathbf{u} + \mathbf{b}^T \Delta \mathbf{u} \mid \mathbf{M} \Delta \mathbf{u} \leq \boldsymbol{\gamma} \right\}; \mathbf{H} \in \mathcal{R}^{N_u, N_u}, \mathbf{b} \in \mathcal{R}^{N_u, 1} \quad (4.81)$$

Pozn.: Oproti vztahu (4.37) z kapitoly 4.1 je uvažováno následné přeznačení (4.82) definovaných symbolů, a to se zvyklostmi značení v prediktivním řízení dle [1]–[6].

$$\Delta \mathbf{u} \equiv \mathbf{x}^*, \text{ resp. } \Delta \mathbf{u} \equiv \mathbf{x}; J \equiv f(\mathbf{x}); n \equiv N_u \quad (4.82)$$

Souvislost Hessovy matice \mathbf{H} a vektoru \mathbf{b} úlohy kvadratického programování s veličinami prediktivního řízení je dána vztahy (4.83)–(4.84). Kde \mathbf{I} je jednotková matice příslušných rozměrů. Tyto vztahy vzešly z matematických úprav vztahu (4.79) do výsledného tvaru (4.81). [1]–[6]

$$\mathbf{H} = \mathbf{G}^T \mathbf{G} + \mathbf{I} \quad (4.83)$$

$$\mathbf{b}^T = \mathbf{G}^T \left(\mathbf{P} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \\ \Delta u(k-1) \end{bmatrix} - \mathbf{w} \right) \quad (4.84)$$

Pro definici omezujících podmínek pro jednotlivé veličiny řízení platí pravidla v tabulce 1. Každé omezení je specifikováno nastavitelnou konstantou určující limitní hodnotu pro daný typ omezení. Je třeba zařadit příslušnou definici do výsledné matice \mathbf{M} a vektoru $\boldsymbol{\gamma}$ vztahu (4.80), a to při uvažování více omezení přidáním dalších řádků pod sebe. [1]–[6]

Tab. 1 Definice podmínek v jednorozměrném prediktivním řízení

Typ omezení	Definice matice \mathbf{M} a vektoru $\boldsymbol{\gamma}$	Rozměry \mathbf{M} , $\boldsymbol{\gamma}$
Δu_{\min}	$\mathbf{M} = \begin{bmatrix} -1 & 0 & \cdots & 0 \\ 0 & -1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & -1 \end{bmatrix}; \boldsymbol{\gamma} = \begin{bmatrix} -\Delta u_{\min} \\ \vdots \\ -\Delta u_{\min} \end{bmatrix}$	$\mathbf{M} \in \mathcal{R}^{N_u \times N_u};$ $\boldsymbol{\gamma} \in \mathcal{R}^{N_u \times 1}$
Δu_{\max}	$\mathbf{M} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}; \boldsymbol{\gamma} = \begin{bmatrix} \Delta u_{\max} \\ \vdots \\ \Delta u_{\max} \end{bmatrix}$	$\mathbf{M} \in \mathcal{R}^{N_u \times N_u};$ $\boldsymbol{\gamma} \in \mathcal{R}^{N_u \times 1}$
u_{\min}	$\mathbf{M} = \begin{bmatrix} -1 & 0 & \cdots & 0 \\ -1 & -1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ -1 & -1 & \cdots & -1 \end{bmatrix}; \boldsymbol{\gamma} = \begin{bmatrix} u(k-1) - u_{\min} \\ \vdots \\ u(k-1) - u_{\min} \end{bmatrix}$	$\mathbf{M} \in \mathcal{R}^{N_u \times N_u};$ $\boldsymbol{\gamma} \in \mathcal{R}^{N_u \times 1}$
u_{\max}	$\mathbf{M} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}; \boldsymbol{\gamma} = \begin{bmatrix} u_{\max} - u(k-1) \\ \vdots \\ u_{\max} - u(k-1) \end{bmatrix}$	$\mathbf{M} \in \mathcal{R}^{N_u \times N_u};$ $\boldsymbol{\gamma} \in \mathcal{R}^{N_u \times 1}$
y_{\min}	$\mathbf{M} = -\mathbf{G}; \boldsymbol{\gamma} = \begin{bmatrix} -y_{\min} \\ \vdots \\ -y_{\min} \end{bmatrix} + \mathbf{P} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \\ \Delta u(k-1) \end{bmatrix}$	$\mathbf{M} \in \mathcal{R}^{N_2 - N_1 + 1, N_u};$ $\boldsymbol{\gamma} \in \mathcal{R}^{N_2 - N_1, 1}$

Typ omezení	Definice matice \mathbf{M} a vektoru $\boldsymbol{\gamma}$	Rozměry \mathbf{M} , $\boldsymbol{\gamma}$
y_{\max}	$\mathbf{M} = \mathbf{G}; \boldsymbol{\gamma} = \begin{bmatrix} y_{\max} \\ \vdots \\ y_{\max} \end{bmatrix} - \mathbf{P} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \\ \Delta u(k-1) \end{bmatrix}$	$\mathbf{M} \in \mathcal{R}^{N_2 - N_1 + 1, N_u}$; $\boldsymbol{\gamma} \in \mathcal{R}^{N_2 - N_1, 1}$

4.3.4 Dvourozměrové prediktivní řízení

Jelikož je základní princip fungování jednorozměrného prediktivního řízení analogický s dvourozměrným případem, budou uvedeny a priori rozdíly těchto dvou variant. Bude uvažováno shodné okno horizontu s parametry N_1, N_u, N_2 . Je možné si povšimnout, že některé matematické zákonitosti lze obdržet pouhou vektorizací jednorozměrných vztahů, avšak některé tvary rovnic budou odvozeny odlišným způsobem. [1]–[6], [10]

Pro dvourozměrný proces ve vstupně-výstupní oblasti je neadaptivní prediktivní řízení založeno na CARIMA modelu (4.85) s úpravou na (4.86)–(4.87). Jeho polynomiální matice odpovídají maticovým zlomkům \mathbf{A} a \mathbf{B} z diskrétního maticového přenosu (4.71) s uvažováním periody vzorkování T . Vektor výstupních veličin, šumový vektor, vektor budoucích přírůstků řízení a následně používaný vektor žádaných veličin jsou pro dané okno horizontu definovány vztahy (4.88)–(4.91). Polynomiální matice \mathbf{C} bude dále pro zjednodušení uvažována jako jednotková matice \mathbf{I} příslušného rozměru. [1]–[6]

$$\left. \begin{aligned} \mathbf{A}(z^{-1})\mathbf{y}(k) &= \mathbf{B}(z^{-1})\mathbf{u}(k) + \Delta^{-1}(z^{-1})\mathbf{C}(z^{-1})\mathbf{e}_s(k); \\ \Delta(z^{-1}) &= \begin{bmatrix} 1 - z^{-1} & 0 \\ 0 & 1 - z^{-1} \end{bmatrix} \end{aligned} \right\} \quad (4.85)$$

$$\begin{aligned} &\begin{bmatrix} (1 - z^{-1})\alpha_{11}(z^{-1}) & (1 - z^{-1})\alpha_{12}(z^{-1}) \\ (1 - z^{-1})\alpha_{21}(z^{-1}) & (1 - z^{-1})\alpha_{22}(z^{-1}) \end{bmatrix} \begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \\ &= \begin{bmatrix} \beta_{11}(z^{-1}) & \beta_{12}(z^{-1}) \\ \beta_{21}(z^{-1}) & \beta_{22}(z^{-1}) \end{bmatrix} \begin{bmatrix} \Delta u_1(k) \\ \Delta u_2(k) \end{bmatrix} + \mathbf{e}_s(k) \end{aligned} \quad (4.86)$$

$$\left. \begin{aligned}
y_1(k) &= (1 - \alpha_{111}) \cdot y_1(k-1) + (\alpha_{111} - \alpha_{112}) \cdot y_1(k-2) + \alpha_{112} \cdot y_1(k-3) - \\
&\quad - \alpha_{121} \cdot y_2(k-1) + (\alpha_{121} - \alpha_{122}) \cdot y_2(k-2) + \alpha_{122} \cdot y_2(k-3) + \\
&+ \beta_{111} \cdot \Delta u_1(k-1) + \beta_{112} \cdot \Delta u_1(k-2) + \beta_{121} \cdot \Delta u_2(k-1) + \beta_{122} \cdot \Delta u_2(k-2) + \\
&\quad + e_{s_1}(k) \\
y_2(k) &= (1 - \alpha_{221}) \cdot y_2(k-1) + (\alpha_{221} - \alpha_{222}) y_2(k-2) + \alpha_{222} \cdot y_2(k-3) - \\
&\quad - \alpha_{211} \cdot y_1(k-1) + (\alpha_{211} - \alpha_{212}) \cdot y_1(k-2) + \alpha_{212} \cdot y_1(k-3) + \\
&\quad + \beta_{211} \cdot \Delta u_1(k-1) + \beta_{212} \cdot \Delta u_1(k-2) + \beta_{221} \cdot \Delta u_2(k-1) + \\
&\quad + \beta_{222} \cdot \Delta u_2(k-2) + e_{s_2}(k)
\end{aligned} \right\} (4.87)$$

$$\mathbf{y} = [\mathbf{y}(k + N_1), \dots, \mathbf{y}(k + N_2)]^T; \mathbf{y}(k) = [y_1(k), y_2(k)]^T \quad (4.88)$$

$$\mathbf{e}_s = [e_{s_1}(k), e_{s_2}(k)]^T \quad (4.89)$$

$$\Delta \mathbf{u} = [\Delta \mathbf{u}(k+1), \dots, \Delta \mathbf{u}(k + N_u)]^T; \Delta \mathbf{u}(k) = [\Delta u_1(k), \Delta u_2(k)]^T \quad (4.90)$$

$$\mathbf{w} = [\mathbf{w}(k + N_1), \dots, \mathbf{w}(k + N_2)]^T; \mathbf{w}(k) = [w_1(k), w_2(k)]^T \quad (4.91)$$

Stanovení N_2 budoucích výstupů výstupních veličin lze rekurentně vypočítat na základě soustavy diferenčních rovnic (4.87) s výslednou maticovou rovnicí (4.92). [1]–[6]

$$\underbrace{\begin{bmatrix} \mathbf{y}(k + N_1) \\ \vdots \\ \mathbf{y}(k + N_2) \end{bmatrix}}_{\mathbf{y}} = \mathbf{P} \begin{bmatrix} \mathbf{y}(k) \\ \mathbf{y}(k-1) \\ \mathbf{y}(k-2) \\ \Delta \mathbf{u}(k-1) \end{bmatrix} + \mathbf{G} \underbrace{\begin{bmatrix} \Delta \mathbf{u}(k) \\ \Delta \mathbf{u}(k+1) \\ \vdots \\ \Delta \mathbf{u}(k + N_u - 1) \end{bmatrix}}_{\Delta \mathbf{u}} \quad (4.92)$$

Soustavu diferenčních rovnic (4.87) lze přepsat do jednodušší podoby (4.93). Pak lze jednotlivé dílčí polynomiální matice definovat dle koeficientů maticových zlomků \mathbf{A} a \mathbf{B} (4.71). Matice ze vztahu (4.93) budou použity pro obecné určení matic \mathbf{P} a \mathbf{G} ve vztazích (4.95)–(4.96). [1]–[6], [10]

$$\left. \begin{aligned}
& \mathbf{y}(k) = \mathbf{A}_1 \mathbf{y}(k-1) + \mathbf{A}_2 \mathbf{y}(k-2) + \mathbf{A}_3 \mathbf{y}(k-3) + \\
& \quad + \mathbf{B}_1 \Delta \mathbf{u}(k-1) + \mathbf{B}_2 \Delta \mathbf{u}(k-2); \\
& \mathbf{A}_1 = \begin{bmatrix} (1-\alpha_{111}) & -\alpha_{121} \\ -\alpha_{211} & (1-\alpha_{221}) \end{bmatrix}; \mathbf{A}_2 = \begin{bmatrix} (\alpha_{111} - \alpha_{112}) & (\alpha_{121} - \alpha_{122}) \\ (\alpha_{211} - \alpha_{212}) & (\alpha_{221} - \alpha_{222}) \end{bmatrix}; \\
& \mathbf{A}_3 = \begin{bmatrix} \alpha_{112} & \alpha_{122} \\ \alpha_{212} & \alpha_{222} \end{bmatrix}; \mathbf{B}_1 = \begin{bmatrix} \beta_{111} & \beta_{121} \\ \beta_{211} & \beta_{221} \end{bmatrix}; \mathbf{B}_2 = \begin{bmatrix} \beta_{112} & \beta_{122} \\ \beta_{212} & \beta_{222} \end{bmatrix}
\end{aligned} \right\} \quad (4.93)$$

Tvary matic \mathbf{P} a \mathbf{G} (4.94) jsou pro rovnost řídicího a maximálního horizonty vyjádřeny v definicích (4.95) a (4.96) a jsou tvořeny dílčími maticemi \mathbf{G}_{ij} a \mathbf{P}_{ij} . Pokud je maximální horizont $N_2 > N_u$, pak je pouze nutné odstranit $2(N_2 - N_u)$ dolních řádků z matice \mathbf{G} . [1]–[6]

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} & \cdots & \mathbf{P}_{14} \\ \mathbf{P}_{21} & \mathbf{P}_{22} & \cdots & \mathbf{P}_{24} \\ \vdots & & \ddots & \vdots \\ \mathbf{P}_{i1} & \mathbf{P}_{i2} & \cdots & \mathbf{P}_{i4} \end{bmatrix}; \mathbf{G} = \begin{bmatrix} \mathbf{G}_{11} & \mathbf{G}_{12} & \cdots & \mathbf{G}_{1j} \\ \mathbf{G}_{21} & \mathbf{G}_{22} & \cdots & \mathbf{G}_{2j} \\ \vdots & & \ddots & \vdots \\ \mathbf{G}_{i1} & \mathbf{G}_{i2} & \cdots & \mathbf{G}_{ij} \end{bmatrix} \quad (4.94)$$

$$\left. \begin{aligned}
& \mathbf{G} = \mathbf{0}; \mathbf{0} \in \mathcal{R}^{N_2 - N_1, N_2}; \mathbf{G}_{11} = \mathbf{G}_{22} = \mathbf{G}_{33} = \mathbf{B}_1; \\
& \mathbf{G}_{21} = \mathbf{G}_{32} = (\mathbf{A}_1 \mathbf{B}_1 + \mathbf{B}_2); \\
& \mathbf{G}_{31} = (\mathbf{A}_1^2 \mathbf{B}_1 + \mathbf{A}_1 \mathbf{B}_2 + \mathbf{A}_2 \mathbf{B}_1); \\
& \left. \begin{aligned}
& \mathbf{G}_{i1} = \mathbf{A}_1 \mathbf{G}_{(i-1)1} + \mathbf{A}_2 \mathbf{G}_{(i-2)1} + \mathbf{A}_3 \mathbf{G}_{(i-3)1} \\
& \mathbf{G}_{i(j-1)} = \mathbf{A}_1 \mathbf{G}_{(i-1)(j-1)} + \mathbf{A}_2 \mathbf{G}_{(i-2)(j-1)} + \mathbf{A}_3 \mathbf{G}_{(i-3)(j-1)} + \mathbf{B}_2 \\
& \mathbf{G}_{ij} = \mathbf{B}_1
\end{aligned} \right\} \quad (4.95) \\
& i = 4, \dots, N_2; j = 1, \dots, i
\end{aligned} \right\}$$

$$\left. \begin{aligned}
& \mathbf{P}_{11} = \mathbf{A}_1; \mathbf{P}_{12} = \mathbf{A}_2; \mathbf{P}_{13} = \mathbf{A}_3; \mathbf{P}_{14} = \mathbf{B}_2; \\
& \mathbf{P}_{21} = \mathbf{A}_1^2 + \mathbf{A}_2; \mathbf{P}_{22} = \mathbf{A}_1 \mathbf{A}_2 + \mathbf{A}_3; \mathbf{P}_{23} = \mathbf{A}_1 \mathbf{A}_3; \mathbf{P}_{24} = \mathbf{A}_1 \mathbf{B}_2; \\
& \mathbf{P}_{31} = \mathbf{A}_1^3 + \mathbf{A}_1 \mathbf{A}_2 + \mathbf{A}_3 + \mathbf{A}_1 \mathbf{A}_2; \mathbf{P}_{32} = \mathbf{A}_1^2 \mathbf{A}_2 + \mathbf{A}_1 \mathbf{A}_3 + \mathbf{A}_2^2; \\
& \mathbf{P}_{33} = \mathbf{A}_1^2 \mathbf{A}_3 + \mathbf{A}_3 \mathbf{A}_2; \mathbf{P}_{34} = \mathbf{A}_1^2 \mathbf{B}_2 + \mathbf{A}_2 \mathbf{B}_2; \\
& (\mathbf{P}_{ij} = \mathbf{A}_1 \mathbf{P}_{(i-1)j} + \mathbf{A}_2 \mathbf{P}_{(i-2)j} + \mathbf{A}_3 \mathbf{P}_{(i-3)j}) i = 4, \dots, N_2; j = 1, \dots, i
\end{aligned} \right\} \quad (4.96)$$

Řešení zákona řízení probíhá pro dvourozměrné prediktivní řízení rovněž kvadratickým programováním. S uvažováním vektorizace veličin řízení (4.88)–(4.91) zůstává zápis zákona řízení podobný z pohledu symboliky. Na základě

úprav rovnice (4.97) je odvozena definice Hessovy matice \mathbf{H} a vektoru \mathbf{b} (4.99)–(4.100) s uvažováním vektorizace. Hledaný vektor $\Delta \mathbf{u}$, jenž je výsledkem optimalizační úlohy (4.98), má skladbu prvků dle (4.90). [1]–[6]

$$J = (\mathbf{y} - \mathbf{w})^T (\mathbf{y} - \mathbf{w}) + \Delta \mathbf{u}^T \Delta \mathbf{u} \quad (4.97)$$

$$\Delta \mathbf{u} = \arg \min \left\{ \frac{1}{2} \Delta \mathbf{u}^T \mathbf{H} \Delta \mathbf{u} + \mathbf{b}^T \Delta \mathbf{u} \mid \mathbf{M} \Delta \mathbf{u} \leq \boldsymbol{\gamma} \right\}; \quad (4.98)$$

$$\mathbf{H} \in \mathcal{R}^{2N_u, 2N_u}; \mathbf{b} \in \mathcal{R}^{2N_u, 1}$$

$$\mathbf{H} = \mathbf{G}^T \mathbf{G} + \mathbf{I} \quad (4.99)$$

$$\mathbf{b}^T = \mathbf{G}^T \left(\mathbf{P} \begin{bmatrix} \mathbf{y}(k) \\ \mathbf{y}(k-1) \\ \mathbf{y}(k-2) \\ \Delta \mathbf{u}(k-1) \end{bmatrix} - \mathbf{w} \right) \quad (4.100)$$

Pro definici omezujících podmínek platí pravidla z tabulky 2. Matice \mathbf{E} je jedničková (všechny prvky rovny 1) a matice \mathbf{I} jednotková matice příslušných rozměrů. Nastavitelné parametry, jež definují podmínky omezení, jsou pro zjednodušení stanoveny pro oba rozměry veličin shodně. Opět je pro uvažování více omezení současně nutné přidat řádky definic omezení do matic \mathbf{M} , $\boldsymbol{\gamma}$ pod sebe. [1]–[6]

Tab. 2 Definice podmínek v dvourozměrném prediktivním řízení

Typ omezení	Definice matice \mathbf{M}	Definice $\boldsymbol{\gamma}$
$\Delta \mathbf{u}_{\min}$	$\mathbf{M} = \begin{bmatrix} -\mathbf{I}^{2,2} & 0 & \dots & 0 \\ 0 & -\mathbf{I}^{2,2} & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & -\mathbf{I}^{2,2} \end{bmatrix};$ $\mathbf{M} \in \mathcal{R}^{2N_u, 2N_u}$	$\boldsymbol{\gamma} = \begin{bmatrix} -\Delta \mathbf{u}_{\min} \\ \vdots \\ -\Delta \mathbf{u}_{\min} \end{bmatrix};$ $\boldsymbol{\gamma} \in \mathcal{R}^{2N_u, 1}$

Typ omezení	Definice matice M	Definice γ
Δu_{\max}	$M = \begin{bmatrix} \mathbf{I}^{2,2} & 0 & \dots & 0 \\ 0 & \mathbf{I}^{2,2} & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{I}^{2,2} \end{bmatrix};$ $M \in \mathcal{R}^{2N_u, 2N_u}$	$\gamma = \begin{bmatrix} \Delta u_{\max} \\ \vdots \\ \Delta u_{\max} \end{bmatrix};$ $\gamma \in \mathcal{R}^{2N_u, 1}$
u_{\min}	$M = \begin{bmatrix} -\mathbf{E}^{2,2} & 0 & \dots & 0 \\ -\mathbf{E}^{2,2} & -\mathbf{E}^{2,2} & \dots & 0 \\ \vdots & & \ddots & \vdots \\ -\mathbf{E}^{2,2} & -\mathbf{E}^{2,2} & \dots & -\mathbf{E}^{2,2} \end{bmatrix};$ $M \in \mathcal{R}^{2N_u, 2N_u}$	$\gamma = \begin{bmatrix} \mathbf{u}(k-1) + \begin{bmatrix} u_{\min} \\ u_{\min} \end{bmatrix} \\ \vdots \\ \mathbf{u}(k-1) + \begin{bmatrix} u_{\min} \\ u_{\min} \end{bmatrix} \end{bmatrix};$ $\gamma \in \mathcal{R}^{2N_u, 1}$
u_{\max}	$M = \begin{bmatrix} \mathbf{E}^{2,2} & 0 & \dots & 0 \\ \mathbf{E}^{2,2} & \mathbf{E}^{2,2} & \dots & 0 \\ \vdots & & \ddots & \vdots \\ \mathbf{E}^{2,2} & \mathbf{E}^{2,2} & \dots & \mathbf{E}^{2,2} \end{bmatrix};$ $M \in \mathcal{R}^{2N_u, 2N_u}$	$\gamma = \begin{bmatrix} -\mathbf{u}(k-1) + \begin{bmatrix} u_{\max} \\ u_{\max} \end{bmatrix} \\ \vdots \\ -\mathbf{u}(k-1) + \begin{bmatrix} u_{\max} \\ u_{\max} \end{bmatrix} \end{bmatrix};$ $\gamma \in \mathcal{R}^{2N_u, 1}$
y_{\min}	$M = -G;$ $M \in \mathcal{R}^{2(N_2 - N_1 + 1), 2N_u}$	$\gamma = \begin{bmatrix} -\mathbf{E}^{2(N_2 - N_1 + 1), 1} y_{\min} - \\ \mathbf{y}(k) \\ -\mathbf{P} \begin{bmatrix} \mathbf{y}(k-1) \\ \mathbf{y}(k-2) \\ \Delta \mathbf{u}(k-1) \end{bmatrix} \end{bmatrix};$ $\gamma \in \mathcal{R}^{2(N_2 - N_1 + 1), 1}$
y_{\max}	$M = G;$ $M \in \mathcal{R}^{2(N_2 - N_1 + 1), 2N_u}$	$\gamma = \begin{bmatrix} \mathbf{E}^{2(N_2 - N_1 + 1), 1} y_{\max} + \\ +\mathbf{P} \begin{bmatrix} \mathbf{y}(k) \\ \mathbf{y}(k-1) \\ \mathbf{y}(k-2) \\ \Delta \mathbf{u}(k-1) \end{bmatrix} \end{bmatrix};$ $\gamma \in \mathcal{R}^{2(N_2 - N_1 + 1), 1}$

4.3.5 Zavedená modifikace v online prediktivním řízení

V knize [4] je detailně probráno prediktivní řízení procesů z pohledu predikcí a online optimalizace. Je doporučeno užití duální metody optimalizace pro řešení úlohy kvadratického programování. Rerezentantem duální metody je zvolena rychlá Hildrethova metoda [14]. Na teorii duality spočívají, dle autora Wanga, přístupy časově efektivního řešení prediktivního řešení. [4], [14]

V publikaci [4] se navíc autor pokouší eliminovat [4, s. 67, krok 2] z optimalizační úlohy nadeřinovaná omezení tvaru maticové nerovnice (4.80). V tomto případě převádí původní podobu neklasické úlohy na vázaný extrém (kap. 4.1.5) na pouhou úlohu hledání volného extrému (kap. 4.1.3). Přičemž v některých případech může být tento postup úspěšný (existující vzdálenostní rezervy mezi vazebními podmínkami a optimem) a v jiných případech nikoliv.

Jedná se prakticky o test redukovaného vztahu (4.101) pod původní skladbou podmínek omezení (4.102). Tento jev (kap. 4.2.2) existující v teorii optimalizace lze zpětně dokázat (kap. 5.2, praktická část) odvozením speciálního tvaru Kuhn–Tuckerových podmínek [14]. [4], [14]

$$\Delta \mathbf{u} = \arg \min \left\{ \begin{array}{l} \frac{1}{2} \Delta \mathbf{u}^T \mathbf{H} \Delta \mathbf{u} + \mathbf{b}^T \Delta \mathbf{u} \\ \Delta \mathbf{u} = -\mathbf{H}^{-1} \mathbf{b} \end{array} \right\}; \quad (4.101)$$

$$\mathbf{M}(-\mathbf{H}^{-1} \mathbf{b}) \leq \gamma \quad (4.102)$$

I když autor Wang. L. doporučuje rychlý způsob duálního řešení zákona řízení pro MPC, prezentuje zajímavou modifikaci (dále též Wangovu modifikaci) podloženou teorií optimalizace. Tato myšlenka, jež není přímo vázána na konkrétní metodu v optimalizátoru, přináší časové úspory algoritmu, bohužel ne vždy v rámci celého řízení. Proto budou navrženy vlastní modifikace (šetrnější eliminace omezení v kap. 5.4), jež navazují na tento postup způsobu urychlení výpočtů algoritmu MPC z pohledu snížení časové náročnosti daného řešení. [4], [14]

5. HLAVNÍ VÝSLEDKY PRÁCE

5.1 Algoritmus analytického řešení kvadratického programování

5.1.1 Rozbor analytického řešení kvadratického programování

Náplň kapitoly 5 bude zahrnovat teoretické poznatky. Prvním z nich bude odvození algoritmu, kterým bude možno verifikovat získané výsledky vlastních návrhů řešení optimalizace a modifikací v MPC prezentovaných dále. Verifikační algoritmus bude založen na možnosti převodu neklasické úlohy na vázaný extrém na klasickou úlohu na vázaný extrém. Bude využito analytického postupu dle kapitoly 4.2.1, a to zavedení doplňkových proměnných pro převod nerovnostních omezujících podmínek na rovnostní. Tím bude umožněn analytický přístup řešení, jenž není v literatuře o nelineární optimalizaci např. v [14]–[22] řešen na praktickém příkladu. Postup je konkrétně realizován na úloze kvadratického programování (5.1)–(5.2), a to na základě pravidel uvedených v kapitolách 4.1 a 4.2. Zadání příkladu bylo zvoleno experimentálně. Cílem je obdržet konkrétní kroky analytického řešení úlohy kvadratického programování za účelem jejich implementace v prostředí MATLAB.

Zadáním úlohy je minimalizace účelové funkce $f(\mathbf{x})$ dle nerovnostních podmínek $\mathbf{M}\mathbf{x} \leq \boldsymbol{\gamma}$ (5.1)–(5.2).

$$f(\mathbf{x}) = \underbrace{x_1^2 - 2x_1x_2 + 3x_2^2 + 5x_3^2 + 2x_2x_3 + x_3x_1}_{\frac{1}{2} \mathbf{x}^T \underbrace{\begin{bmatrix} 2 & -2 & 1 \\ -2 & 6 & 2 \\ 1 & 2 & 10 \end{bmatrix} \mathbf{x}}_{\mathbf{H}} + \underbrace{2x_1 + 2x_2 + x_3}_{\underbrace{\begin{bmatrix} 2 & 2 & 1 \end{bmatrix} \mathbf{x}}_{\mathbf{b}^T}} = \left. \begin{aligned} &= \frac{1}{2} \mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{b}^T \mathbf{x} \end{aligned} \right\} \quad (5.1)$$

$$\left. \begin{aligned} x_1 + x_2 + 2x_3 &\leq 4 \\ 2x_1 + 2x_2 + 3x_3 &\leq 5 \\ x_1 + 2x_2 + x_3 &\leq 2 \end{aligned} \right\}, \left\{ \begin{aligned} M_1(\mathbf{x}) &\leq \gamma_1 \\ M_2(\mathbf{x}) &\leq \gamma_2 \\ M_3(\mathbf{x}) &\leq \gamma_3 \end{aligned} \right\}, \underbrace{\begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 3 \\ 1 & 2 & 1 \end{bmatrix}}_{\mathbf{M}} \mathbf{x} \leq \underbrace{\begin{bmatrix} 4 \\ 5 \\ 2 \end{bmatrix}}_{\boldsymbol{\gamma}} \quad (5.2)$$

Sestavení Lagrangeovy funkce $L(\mathbf{x}, \boldsymbol{\lambda})$:

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \lambda_1(x_1 + x_2 + 2x_3 - 4) + \lambda_2(2x_1 + 2x_2 + 3x_3 - 5) + \lambda_3(x_1 + 2x_2 + x_3 - 2) \quad (5.3)$$

Určení gradientu $\nabla L(\mathbf{x}, \boldsymbol{\lambda})$ (5.4)–(5.10) Lagrangeovy funkce $L(\mathbf{x}, \boldsymbol{\lambda})$:

$$\nabla L(\mathbf{x}, \boldsymbol{\lambda}) = \left[\frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_1} \quad \dots \quad \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_3} \quad \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial \lambda_1} \quad \dots \quad \frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial \lambda_3} \right]^T \quad (5.4)$$

$$\frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_1} = 2x_1 - 2x_2 + x_3 + 2 + \lambda_1 + 2\lambda_2 + \lambda_3 \quad (5.5)$$

$$\frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_2} = -2x_1 + 6x_2 + 2x_3 + 2 + \lambda_1 + 2\lambda_2 + 2\lambda_3 \quad (5.6)$$

$$\frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial x_3} = x_1 + 2x_2 + 10x_3 + 1 + 2\lambda_1 + 3\lambda_2 + \lambda_3 \quad (5.7)$$

$$\frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial \lambda_1} = x_1 + x_2 + 2x_3 - 4 \quad (5.8)$$

$$\frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial \lambda_2} = 2x_1 + 2x_2 + 3x_3 - 5 \quad (5.9)$$

$$\frac{\partial L(\mathbf{x}, \boldsymbol{\lambda})}{\partial \lambda_3} = x_1 + 2x_2 + x_3 - 2 \quad (5.10)$$

Kontrola splnění podmínek regularity dle pravidel kapitoly 4.1.4:

1. podmínka je splněna, neboť hodnost Jacobiho matice (5.11) je rovna počtu omezujících podmínek úlohy, tj. hodnotě $m = 3$.

$$\begin{bmatrix} \frac{\partial M_1(\mathbf{x})}{\partial x_1} & \frac{\partial M_1(\mathbf{x})}{\partial x_2} & \frac{\partial M_1(\mathbf{x})}{\partial x_3} \\ \frac{\partial M_2(\mathbf{x})}{\partial x_1} & \frac{\partial M_2(\mathbf{x})}{\partial x_2} & \frac{\partial M_2(\mathbf{x})}{\partial x_3} \\ \frac{\partial M_3(\mathbf{x})}{\partial x_1} & \frac{\partial M_3(\mathbf{x})}{\partial x_2} & \frac{\partial M_3(\mathbf{x})}{\partial x_3} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 3 \\ 1 & 2 & 1 \end{bmatrix} \quad (5.11)$$

2. podmínka je splněna, neboť soustava rovnic $\nabla L(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{0}$ je řešitelná a matice koeficientů soustavy $\boldsymbol{\Xi}$ má v rámci soustavy rovnic (5.12) nenulový determinant (5.13). Hodnost matice $\boldsymbol{\Xi}$ odpovídá v pořádku hodnoti $m + n = 6$.

$$\underbrace{\begin{bmatrix} 2 & -2 & 1 & 1 & 2 & 1 \\ -2 & 6 & 2 & 1 & 2 & 2 \\ 1 & 2 & 10 & 2 & 3 & 1 \\ 1 & 1 & 2 & 0 & 0 & 0 \\ 2 & 2 & 3 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{E}} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -2 \\ -2 \\ -1 \\ 4 \\ 5 \\ 2 \end{bmatrix} \quad (5.12)$$

$$|\mathbf{E}| = \begin{vmatrix} 2 & -2 & 1 & 1 & 2 & 1 \\ -2 & 6 & 2 & 1 & 2 & 2 \\ 1 & 2 & 10 & 2 & 3 & 1 \\ 1 & 1 & 2 & 0 & 0 & 0 \\ 2 & 2 & 3 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 \end{vmatrix} = -1 \neq 0 \quad (5.13)$$

3. podmínka je též splněna, neboť všechny funkce (4.21) z definice omezení mají spojité druhé parciální derivace, a to rovné konstantě 0.

Postup analytického řešení zadané úlohy spočívá v převodu úlohy (5.1)–(5.2) na tvar (5.14) dle konkrétního vztahu (4.49):

$$\mathbf{x}^* = \arg \min \{f(\mathbf{x}) \mid \mathbf{M}\mathbf{x} + \bar{\mathbf{y}} = \boldsymbol{\gamma}\}, \bar{\mathbf{y}} = [\bar{y}_1^2 \cdots \bar{y}_3^2]^T, \bar{\mathbf{y}} \in \mathcal{R}^3 \quad (5.14)$$

Potom modifikovaná Lagrangeova funkce (4.50) $\tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})$ má novou podobu (5.15).

$$\begin{aligned} \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) &= f(\mathbf{x}) + \lambda_1(x_1 + x_2 + 2x_3 + \bar{y}_1^2 - 4) + \\ &+ \lambda_2(2x_1 + 2x_2 + 3x_3 + \bar{y}_2^2 - 5) + \lambda_3(x_1 + 2x_2 + x_3 + \bar{y}_3^2 - 2) \end{aligned} \quad (5.15)$$

Určení gradientu (4.51) $\nabla \tilde{L} = \nabla \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})$ pro novou Lagrangeovu funkci $\tilde{L} = \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})$ je vyjádřeno rovnicí (5.16).

$$\nabla \tilde{L} = \nabla \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) = \left[\frac{\partial \tilde{L}}{\partial x_1} \cdots \frac{\partial \tilde{L}}{\partial x_3} \quad \frac{\partial \tilde{L}}{\partial \bar{y}_1} \cdots \frac{\partial \tilde{L}}{\partial \bar{y}_3} \quad \frac{\partial \tilde{L}}{\partial \lambda_1} \cdots \frac{\partial \tilde{L}}{\partial \lambda_3} \right]^T \quad (5.16)$$

Pozn.: Výsledky výpočtu parciálních derivací nové Lagrangeovy funkce $\tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})$ podle prvků vektoru \mathbf{x} jsou totožné se vztahy (5.5)–(5.7).

$$\frac{\partial \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial \bar{y}_1} = 2\lambda_1 \bar{y}_1 \quad (5.17)$$

$$\frac{\partial \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial \bar{y}_2} = 2\lambda_2 \bar{y}_2 \quad (5.18)$$

$$\frac{\partial \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial \bar{y}_3} = 2\lambda_3 \bar{y}_3 \quad (5.19)$$

$$\frac{\partial \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial \lambda_1} = x_1 + x_2 + 2x_3 + \bar{y}_1^2 - 4 \quad (5.20)$$

$$\frac{\partial \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial \lambda_2} = 2x_1 + 2x_2 + 3x_3 + \bar{y}_2^2 - 5 \quad (5.21)$$

$$\frac{\partial \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda})}{\partial \lambda_3} = x_1 + 2x_2 + x_3 + \bar{y}_3^2 - 2 \quad (5.22)$$

Nyní je možné určit stacionární body, a to řešením soustavy rovnic $\nabla \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) = \mathbf{0}$. Tato soustava nabývá po úpravách tvaru (5.23)–(5.25).

$$\left. \begin{aligned} 2x_1 - 2x_2 + x_3 + \lambda_1 + 2\lambda_2 + \lambda_3 &= -2 \\ -2x_1 + 6x_2 + 2x_3 + \lambda_1 + 2\lambda_2 + 2\lambda_3 &= -2 \\ x_1 + 2x_2 + 10x_3 + 2\lambda_1 + 3\lambda_2 + \lambda_3 &= -1 \end{aligned} \right\} \quad (5.23)$$

$$\left. \begin{aligned} x_1 + x_2 + 2x_3 + \bar{y}_1^2 &= 4 \\ 2x_1 + 2x_2 + 3x_3 + \bar{y}_2^2 &= 5 \\ x_1 + 2x_2 + x_3 + \bar{y}_3^2 &= 2 \end{aligned} \right\} \quad (5.24)$$

$$\left. \begin{aligned} 2\lambda_1 \bar{y}_1 &= 0 \\ 2\lambda_2 \bar{y}_2 &= 0 \\ 2\lambda_3 \bar{y}_3 &= 0 \end{aligned} \right\} \rightarrow \left\{ \begin{aligned} \lambda_1 \bar{y}_1^2 &= 0 \\ \lambda_2 \bar{y}_2^2 &= 0 \\ \lambda_3 \bar{y}_3^2 &= 0 \end{aligned} \right\} \quad (5.25)$$

Pozn. Ve vztahu (5.25) je proveden převod na druhé mocniny, jenž nezmění účel rovnic – zajistit nulování proměnných.

Dále je třeba rozlišit případy řešení části soustavy, a to konkrétně u případu (5.25). V souvislosti s různými kombinacemi nulovaných proměnných budou určeny stacionární body $(\hat{\mathbf{x}}, \hat{\bar{\mathbf{y}}}, \hat{\boldsymbol{\lambda}})$ optimalizační úlohy. Tyto stacionární body budou podrobeny šetření na typ extrému dle Hessovy matice (5.26), a to na její definitnost pro tyto stacionární body.

Pozn.: Značení prvků doplňkového vektoru v rámci stacionárního bodu je dle formulace (5.27).

$$\nabla_x^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda) = \begin{bmatrix} \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_1^2} & \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_1 \partial x_2} & \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_1 \partial x_3} \\ \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_2 \partial x_1} & \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_2^2} & \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_2 \partial x_3} \\ \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_3 \partial x_1} & \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_3 \partial x_2} & \frac{\partial^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda)}{\partial x_3^2} \end{bmatrix} \quad (5.26)$$

$$\hat{\mathbf{y}} = [\hat{y}_1^2 \quad \dots \quad \hat{y}_m^2]^T \quad (5.27)$$

Pro zadanou úlohu bude mít Hessián pro dané stacionární body vždy stejný tvar (5.28). Dle relací (5.29) je patrné, že pro všechny stacionární body bude Hessova matice pozitivně definitní.

$$\nabla_x^2 L(\mathbf{x}, \bar{\mathbf{y}}, \lambda) = \begin{bmatrix} 2 & -2 & 1 \\ -2 & 6 & 2 \\ 1 & 2 & 10 \end{bmatrix} \quad (5.28)$$

$$|2| > 0; \begin{vmatrix} 2 & -2 \\ -2 & 6 \end{vmatrix} > 0; \begin{vmatrix} 2 & -2 & 1 \\ -2 & 6 & 2 \\ 1 & 2 & 10 \end{vmatrix} > 0 \quad (5.29)$$

Následuje konkrétní hledání stacionárních bodů pro různé kombinace nulování proměnných ve smyslu vztahu (5.25) – v rámci soustavy rovnic $\nabla \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \lambda) = \mathbf{0}$.

Jelikož v rovnicích (5.25) nemůže nastat varianta (5.30), ale vždy alespoň jedna z proměnných v daném součinu (5.31) nabývá hodnoty nula, je možné přepsat soustavu rovnic (5.23)–(5.25) s nelineární částí (5.25) do jednoduššího tvaru soustavy lineárních rovnic (5.32) s řešením (5.33). Nulování jednotlivých proměnných ovlivní tvar celkové matice koeficientů $\mathbf{\Gamma}$.

$$(\lambda_j \neq 0) \wedge (\bar{y}_j^2 \neq 0), j = 1, 2, \dots, m \quad (5.30)$$

$$2\lambda_j \bar{y}_j = 0; j = 1, 2, \dots, m \quad (5.31)$$

$$\mathbf{\Gamma} \begin{bmatrix} \hat{\mathbf{x}} & \hat{\mathbf{y}} & \hat{\lambda} \end{bmatrix}^T = \boldsymbol{\chi}; \mathbf{\Gamma} \in \mathcal{R}^{n+2m, n+2m}; \boldsymbol{\chi} \in \mathcal{R}^{n+2m, 1} \quad (5.32)$$

$$\begin{bmatrix} \hat{\mathbf{x}} & \hat{\mathbf{y}} & \hat{\boldsymbol{\lambda}} \end{bmatrix}^T = \begin{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_3 \end{bmatrix} & \begin{bmatrix} \hat{y}_1^2 \\ \vdots \\ \hat{y}_3^2 \end{bmatrix} & \begin{bmatrix} \hat{\lambda}_1 \\ \vdots \\ \hat{\lambda}_3 \end{bmatrix} \end{bmatrix}^T = \boldsymbol{\Gamma}^{-1} \boldsymbol{\chi} \quad (5.33)$$

Kombinace 1: $(\bar{y}_1 = \bar{y}_2 = \bar{y}_3 = 0) \wedge (\lambda_1 \neq \lambda_2 \neq \lambda_3 \neq 0)$:

$$\underbrace{\begin{bmatrix} 2 & -2 & 1 & 0 & 0 & 0 & 1 & 2 & 1 \\ -2 & 6 & 2 & 0 & 0 & 0 & 1 & 2 & 2 \\ 1 & 2 & 10 & 0 & 0 & 0 & 2 & 3 & 1 \\ 1 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}}_{\boldsymbol{\Gamma}} \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{y}} \\ \hat{\boldsymbol{\lambda}} \end{bmatrix} = \underbrace{\begin{bmatrix} -2 \\ -2 \\ -1 \\ 4 \\ 5 \\ 2 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\boldsymbol{\chi}}, \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{y}_1^2 \\ \hat{y}_2^2 \\ \hat{y}_3^2 \\ \hat{\lambda}_1 \\ \hat{\lambda}_2 \\ \hat{\lambda}_3 \end{bmatrix} = \begin{bmatrix} -3 \\ 1 \\ 3 \\ 0 \\ 0 \\ 0 \\ -92 \\ 59 \\ -23 \end{bmatrix} \quad (5.34)$$

Jelikož všechny Lagrangeovy multiplikátory ve výsledku (5.34) nespĺňujú čtvrtou podmínku (4.31) Kuhn–Tuckerovy věty, a to požadavek kladné nebo nulové hodnoty, není tento stacionární bod řešením optimalizační úlohy.

Dále bude pro zpřehlednění dalších změn v zápisu soustavy (5.32) zavedena její definice pomocí submatic (5.35), které se v jednotlivých kombinacích požadovaným způsobem mění. Ostatní prvky matice (5.34) mimo tyto submatice zůstávají neměnné.

Budou se tedy měnit pouze čtyři submatice na daných rozsazích souřadnic původní matice $\boldsymbol{\Gamma}$, jež jsou uvedeny v závorce.

$$\left. \begin{aligned} \boldsymbol{\Gamma}_1 &= \boldsymbol{\Gamma}(1..3, 7..9); \boldsymbol{\Gamma}_2 = \boldsymbol{\Gamma}(4..6, 4..6) \\ \boldsymbol{\Gamma}_3 &= \boldsymbol{\Gamma}(7..9, 4..6); \boldsymbol{\Gamma}_4 = \boldsymbol{\Gamma}(7..9, 7..9) \end{aligned} \right\} \quad (5.35)$$

Kombinace 2: $(\bar{y}_3 = \lambda_1 = \lambda_2 = 0) \wedge (\bar{y}_1 \neq \bar{y}_2 \neq \lambda_3 \neq 0)$:

$$\boldsymbol{\Gamma}_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix}; \boldsymbol{\Gamma}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \boldsymbol{\Gamma}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \boldsymbol{\Gamma}_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (5.36)$$

$$\left. \begin{aligned} \hat{\mathbf{x}} &= [0,7717 \quad 0,663 \quad -0,0978]^T \\ \hat{\mathbf{y}} &= [2,7609 \quad 2,4239 \quad 0]^T \\ \hat{\boldsymbol{\lambda}} &= [0 \quad 0 \quad -2,1196]^T \end{aligned} \right\} \quad (5.37)$$

Není opět splněna čtvrtá podmínka (4.31) Kuhn–Tuckerovy věty. Bod (5.37) není řešením úlohy.

Kombinace 3: $(\bar{y}_1 = \lambda_2 = \lambda_3 = 0) \wedge (\bar{y}_2 \neq \bar{y}_3 \neq \lambda_1 \neq 0)$:

$$\mathbf{F}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 2 & 0 & 0 \end{bmatrix}; \mathbf{F}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \mathbf{F}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{F}_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.38)$$

$$\left. \begin{aligned} \hat{\mathbf{x}} &= [2,1494 \quad 1,023 \quad 0,4138]^T \\ \hat{\mathbf{y}} &= [0 \quad -2,5862 \quad -2,6092]^T \\ \hat{\boldsymbol{\lambda}} &= [-4,6667 \quad 0 \quad 0]^T \end{aligned} \right\} \quad (5.39)$$

Není splněna čtvrtá podmínka (4.31) Kuhn–Tuckerovy věty. Bod (5.39) není řešením úlohy.

Kombinace 4: $(\bar{y}_2 = \lambda_1 = \lambda_3 = 0) \wedge (\bar{y}_1 \neq \bar{y}_3 \neq \lambda_2 \neq 0)$:

$$\mathbf{F}_1 = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 2 & 0 \\ 0 & 3 & 0 \end{bmatrix}; \mathbf{F}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \mathbf{F}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{F}_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.40)$$

$$\left. \begin{aligned} \hat{\mathbf{x}} &= [1,5225 \quad 0,7416 \quad 0,1573]^T \\ \hat{\mathbf{y}} &= [1,4213 \quad 0 \quad -1,1629]^T \\ \hat{\boldsymbol{\lambda}} &= [0 \quad -1,8596 \quad 0]^T \end{aligned} \right\} \quad (5.41)$$

Není splněna čtvrtá podmínka (4.31) Kuhn–Tuckerovy věty. Bod (5.41) není řešením úlohy.

Kombinace 5: $(\bar{y}_1 = \bar{y}_2 = \lambda_3 = 0) \wedge (\bar{y}_3 \neq \lambda_1 \neq \lambda_2 \neq 0)$:

$$\mathbf{F}_1 = \begin{bmatrix} 1 & 2 & 0 \\ 1 & 2 & 0 \\ 2 & 3 & 0 \end{bmatrix}; \mathbf{F}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \mathbf{F}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{F}_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.42)$$

$$\left. \begin{aligned} \hat{\mathbf{x}} &= [-1,0833 \quad -0,9167 \quad 3]^T \\ \hat{\mathbf{y}} &= [0 \quad 0 \quad 1,9167]^T \\ \hat{\boldsymbol{\lambda}} &= [-42,1667 \quad 18,75 \quad 0]^T \end{aligned} \right\} \quad (5.43)$$

Není splněna čtvrtá podmínka (4.31) Kuhn–Tuckerovy věty. Bod (5.43) není řešením úlohy.

Kombinace 6: $(\bar{y}_1 = \bar{y}_3 = \lambda_2 = 0) \wedge (\bar{y}_2 \neq \lambda_1 \neq \lambda_3 \neq 0)$:

$$\mathbf{\Gamma}_1 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 2 \\ 2 & 0 & 1 \end{bmatrix}; \mathbf{\Gamma}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{\Gamma}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \mathbf{\Gamma}_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (5.44)$$

$$\left. \begin{aligned} \hat{\mathbf{x}} &= [1,0227 \quad -0,3409 \quad 1,6591]^T \\ \hat{\mathbf{y}} &= [0 \quad -1,3409 \quad 0]^T \\ \hat{\boldsymbol{\lambda}} &= [-11,5455 \quad 0 \quad 5,1591]^T \end{aligned} \right\} \quad (5.45)$$

Není splněna čtvrtá podmínka (4.31) Kuhn–Tuckerovy věty. Bod (5.45) není řešením úlohy.

Kombinace 7: $(\bar{y}_2 = \bar{y}_3 = \lambda_1 = 0) \wedge (\bar{y}_1 \neq \lambda_2 \neq \lambda_3 \neq 0)$:

$$\mathbf{\Gamma}_1 = \begin{bmatrix} 0 & 2 & 1 \\ 0 & 2 & 2 \\ 0 & 3 & 1 \end{bmatrix}; \mathbf{\Gamma}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{\Gamma}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \mathbf{\Gamma}_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (5.46)$$

$$\left. \begin{aligned} \hat{\mathbf{x}} &= [1,2791 \quad -0,0698 \quad 0,8605]^T \\ \hat{\mathbf{y}} &= [1,0698 \quad 0 \quad 0]^T \\ \hat{\boldsymbol{\lambda}} &= [0 \quad -5,186 \quad 4,814]^T \end{aligned} \right\} \quad (5.47)$$

Není splněna čtvrtá podmínka (4.31). Bod (5.47) proto není řešením úlohy.

Kombinace 8: $\lambda_1 = \lambda_2 = \lambda_3 = \bar{y}_1 = \bar{y}_2 = \bar{y}_3 = 0$

$$\mathbf{\Gamma}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{\Gamma}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{\Gamma}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{\Gamma}_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (5.48)$$

Neexistuje řešení soustavy rovnic $\nabla \tilde{L}(\mathbf{x}, \bar{\mathbf{y}}, \boldsymbol{\lambda}) = \mathbf{0}$, neboť je uvažovaná matice $\mathbf{\Gamma}$ singulární.

Kombinace 9: $(\lambda_1 = \lambda_2 = \lambda_3 = 0) \wedge (\bar{y}_1 \neq \bar{y}_2 \neq \bar{y}_3 \neq 0)$:

$$\mathbf{F}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{F}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \mathbf{F}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \mathbf{F}_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.49)$$

$$\left. \begin{aligned} \hat{\mathbf{x}} &= [-2,5172 \quad -1,3103 \quad 0,4138]^T \\ \hat{\mathbf{y}} &= [7 \quad 11,4138 \quad 6,7241]^T \\ \hat{\boldsymbol{\lambda}} &= [0 \quad 0 \quad 0]^T \end{aligned} \right\} \quad (5.50)$$

Pro stacionární bod (5.50) platí všechny Kuhn–Tuckerovy podmínky (4.28)–(4.31):

- $[2x_1 - 2x_2 + x_3 + 2 \quad -2x_1 + 6x_2 + 2x_3 + 2 \quad x_1 + 2x_2 + 10x_3 + 1]\hat{\mathbf{x}} + \boldsymbol{\theta} = \boldsymbol{\theta}$
- Pro $\forall j, j = 1, \dots, m: M_j(\hat{\mathbf{x}}) - \gamma_j \leq 0$ neboť:
 $M_1(\hat{\mathbf{x}}) - \gamma_1 = -7$, $M_2(\hat{\mathbf{x}}) - \gamma_2 = -11,4138$ a $M_3(\hat{\mathbf{x}}) - \gamma_3 = -6,7241$.
- Pro $\forall j, j = 1, \dots, m: 0(M_j(\hat{\mathbf{x}}) - \gamma_j) = 0$
- Pro $\forall j, j = 1, \dots, m: \hat{\lambda}_j = 0$

Hessián (5.26) stacionárního bodu (5.50) je pozitivně definitní dle výpočtu (5.28).

Tento stacionární bod je vázaným minimem (5.51). S hodnotou účelové funkce -3,6207. Výsledek úlohy byl též úspěšně verifikován příkazem *quadprog* programu MATLAB.

$$\mathbf{x}^* = \hat{\mathbf{x}} = [-2,5172 \quad -1,3103 \quad 0,4138]^T \quad (5.51)$$

5.1.2 Sestavení algoritmu pro účely verifikace

V návaznosti na rozbor řešení experimentálního příkladu z kapitoly 5.1.1, je v této kapitole algoritmováno a zobecněno jeho analytické řešení. Vlastní inovace spočívá v zavedení volby nulových a nenulových kombinací doplňkových proměnných a Lagrangeových multiplikátorů k zajištění převodu nelineární soustavy rovnic (5.23)–(5.25) na lineární tvar (5.32). I když je algoritmus ze své podstaty výpočetně náročnější, a to díky stanovení mnoha kombinací submatic (5.35) matice Γ , je přínosný jako ukázka algoritmování analytického řešení kvadratického programování a jako verifikační algoritmus v kapitole 6. Ověření správnosti řešení navrhované optimalizační metody kvadratického programování (5.3) není založeno na interním příkazu MATLABu *quadprog*, nýbrž na odvozeném algoritmu s ukázkou programového kódu v příloze A.1.

Analytický postup pro řešení kvadratického programování lze popsat následujícími kroky, jež vyplývají z kapitoly 5.1.1.

- Úloha (5.1)–(5.2) je zadána podobou účelové funkce a podmínkami omezení následujícími maticemi a vektory: $\mathbf{H}, \mathbf{b}, \mathbf{M}, \boldsymbol{\gamma}$. Což je zadání odpovídající úloze kvadratického programování.
- Nejdříve je provedena kontrola na podmínky regularity (kap. 4.1.4):

V rámci první podmínky je určena hodnota matice \mathbf{M} , jež musí odpovídat počtu omezujících podmínek m originální úlohy.

Druhou podmínkou je splnění vztahu (5.52) se zahrnutou nulovou maticí příslušných rozměrů.

$$\begin{vmatrix} \mathbf{H} & \mathbf{M}^T \\ \mathbf{M} & \mathbf{0} \end{vmatrix} \neq 0 \quad (5.52)$$

V případě kvadratického programování mají všechny lineární funkce (4.21) z definice omezení spojitě druhé partiální derivace, a to rovné konstantě 0.

- Nelineární soustava rovnic (5.23)–(5.25), převedená na lineární soustavu (5.32), je řešena pro konkrétní kombinace nulových a nenulových prvků vektorů Lagrangeových multiplikátorů a vektoru $\hat{\mathbf{y}}$ v souladu s (5.25). Tím jsou hledány stacionární body pro zadanou úlohu.

Pro nulování prvků vektorů nastává $2^m + 1$ kombinací. Nemůže nastat kombinace všech nenulových proměnných. Kombinace pouze nulových prvků zapříčiňuje neřešitelnost soustavy rovnic (5.23)–(5.25). Proto je výsledný počet kombinací bez této varianty roven 2^m .

Nechť jsou uvažovány nové vektory $\mathbf{v}_\lambda \in \mathcal{R}^{m,1}$ a $\mathbf{v}_y \in \mathcal{R}^{m,1}$. Vektor \mathbf{v}_λ bude mít daný prvek nulový, pokud má být prvek na stejné pozici ve vektoru Lagrangeových multiplikátorů nulován. V opačném případě nabude hodnoty 1. Totéž pravidlo platí pro vektor \mathbf{v}_y , který se analogicky řídí nulováním prvků vektoru $\hat{\mathbf{y}}$.

Pro kombinaci obsahů obou vektorů \mathbf{v}_λ a \mathbf{v}_y platí, že pokud by byly prvky vektorů považovány za logické proměnné typu 0 a 1, jsou dané vektory vůči sobě v logické relaci negace. Lze proto uvažovat vztah (5.53).

$$(\mathbf{v}_\lambda = \neg \mathbf{v}_y) \Leftrightarrow (\mathbf{v}_y = \neg \mathbf{v}_\lambda) \quad (5.53)$$

Sekvence různých variant může být tedy popsána prostřednictvím jednoho vektoru \mathbf{v}_y a vektor \mathbf{v}_λ bude vždy znám díky vztahu (5.53).

- Určení soustavy rovnic (5.32) pro danou kombinaci nulovaných či nenulovaných proměnných:

Z řešení experimentálního příkladu (kap. 5.1.1) je zřejmé, že se mění pouze matice resp. submatice $\mathbf{F}_1 \cdots \mathbf{F}_4$ (5.35). Definice matice \mathbf{F} lineární soustavy rovnic lze vyjádřit ve tvaru (5.54), kde čtveřice submatic matice \mathbf{F} nabývá tvarů (5.35), resp. (5.55)–(5.58). Nulová matice resp. nulový vektor $\mathbf{0}$ má odpovídající rozměry dle okolních submatic matice \mathbf{F} či počtu prvků vektoru $\boldsymbol{\chi}$.

$$\mathbf{F} = \begin{bmatrix} \mathbf{H} & \mathbf{0} & \mathbf{F}_1 \\ \mathbf{M} & \mathbf{F}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_3 & \mathbf{F}_4 \end{bmatrix}, \boldsymbol{\chi} = [-\mathbf{b} \quad \boldsymbol{\gamma} \quad \mathbf{0}]^T \quad (5.54)$$

$$\mathbf{F}_1 = \mathbf{M}^T \text{diag}(\mathbf{v}_\lambda) \Leftrightarrow \mathbf{M}^T \text{diag}(\neg \mathbf{v}_y) \quad (5.55)$$

$$\mathbf{F}_2 = \text{diag}(\mathbf{v}_y) \quad (5.56)$$

$$\mathbf{F}_3 = \text{diag}(\neg \mathbf{v}_y) \quad (5.57)$$

$$\Gamma_4 = \text{diag}(-\mathbf{v}_\lambda) \Leftrightarrow \text{diag}(\mathbf{v}_y) \quad (5.58)$$

Pozn. Označení *diag* vyjadřuje diagonální matici, ve které jsou nenulové prvky pouze na její hlavní diagonále a jsou to prvky odpovídající danému vektoru v argumentu.

Pozn. Submatice Γ_3, Γ_4 zahrnují informace, které prvky budou nulovány specifickým způsobem popsáním následujícím příkladem.

Například pro kombinaci $(\lambda_1 = \lambda_2 = \lambda_3 = 0) \wedge (\bar{y}_1 \neq \bar{y}_2 \neq \bar{y}_3 \neq 0)$ je vektor $\mathbf{v}_y = [1, \dots, 1]^T$, což vyjadřuje, že doplňkové proměnné \mathbf{y} nemají být nulovány. Soustava lineárních rovnic však prostřednictvím submatic (5.59) a (5.60) operuje s položením proměnných nule. Proto musí submatice (5.59) zahrnovat na své hlavní diagonále negace hodnot prvků příslušného vektoru a (5.60) zahrnovat přímo prvky vektoru. Do soustavy rovnic se tím pomocí těchto dvou submatic zavádí informace, které proměnné nulovat. Ve výsledku tedy budou Lagrangeovy multiplikátory v soustavě rovnic nulovány.

$$\Gamma_3 = \text{diag}(-\mathbf{v}_y) = \text{diag}([0, \dots, 0]^T) \quad (5.59)$$

$$\Gamma_4 = \text{diag}(\mathbf{v}_y) = \text{diag}([1, \dots, 1]^T) \quad (5.60)$$

- Hledání stacionárních bodů $(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\boldsymbol{\lambda}})$, a to řešením soustavy rovnic $\Gamma^1 \boldsymbol{\chi}$, bude provedeno pro kombinaci nulování od $\mathbf{v}_y = [0, \dots, 0]^T$ po $\mathbf{v}_y = [1, \dots, 1]^T$ při uvažování $\mathbf{v}_\lambda = -\mathbf{v}_y$. Obsah prvků vektoru \mathbf{v}_y může být odvozen jako binární vyjádření pořadnic kombinací od 0 do čísla $2^m - 1$, při uvažování první kombinace s nulovým vektorem \mathbf{v}_y . První až $2^m - 1$ -tá kombinace je prakticky indexována od 0 do $2^m - 1$.
- Pro všechny stacionární body přísluší shodný Hessián odpovídající matici \mathbf{H} . Je třeba provést vyšetření Hessiánu na jeho pozitivní definitnost.
- Každý stacionární bod je prošetřen na splnění všech čtyř podmínek (4.32)–(4.35) Kuhn–Tuckerovy věty. Podmínka (4.32) má pro kvadratické programování tvar (5.61).

$$\hat{\mathbf{x}}^T \mathbf{H} + \mathbf{b}^T + \boldsymbol{\lambda}^T \mathbf{M} = \mathbf{0} \quad (5.61)$$

- Výsledkem algoritmu je optimum \mathbf{x}^* s nejmenší hodnotou účelové funkce $f(\mathbf{x})$.
- Jelikož má algoritmus sloužit k verifikaci v kapitole 6 bez uvažování interních příkazů optimalizačního modulu prostředí MATLAB, je nutné pro tento algoritmus ověřit správnost jeho řešení.

Byl sestaven programový kód (příloha A.2) s experimentálně voleným pozitivně definitním Hessiánem \mathbf{H} (matice rozměru $\mathbf{H} \in \mathcal{R}^{n,n}$ s prvky mimo hlavní diagonálu o hodnotě 1 a s prvky na hlavní diagonály o hodnotách aritmetické posloupnosti $\{2i\}_{i=1}^n$) a vektorem \mathbf{b} (vektoru rozměru $\mathbf{b} \in \mathcal{R}^{1,n}$ s prvky o hodnotách aritmetické posloupnosti $\{i\}_{i=1}^n$). A to pro rozsahy rozměrů proměnné \mathbf{x} : $n \in \langle 3; 20 \rangle$. Rozměr $n = 20$ je stanoven z důvodu řešení $2^{20} - 1 = 1048575$ kombinací při řešení algoritmu. Matice omezení je definována jako jednotková $\mathbf{I} \in \mathcal{R}^{n,n}$ a vektor $\boldsymbol{\gamma} \in \mathcal{R}^{n,1}$ s hodnotami 0,5. Takto definovaná matice omezení \mathbf{M} splňuje nutné podmínky regularity. Pro rozměry v úloze platí rovnost $m = n$. Takto definovaná úloha byla provedena navrženým algoritmem i interním příkazem *quadprog* s kladným výsledkem pro stanovené testování s přesností 0,001. Jelikož má algoritmus čistě analytický princip a nejsou zavedeny numerické aproximace, jsou výsledky dostatečně přesné.

- Jelikož při ověření algoritmu, jenž je založen též na práci s Lagrangeovými multiplikátory (zahrnutý v soustavě lineárních rovnic a podmínkách Kuhn–Tuckerovy věty), byly všechny výsledky \mathbf{x}^* správné, lze konstatovat, že jsou v pořádku též stanovené Lagrangeovy multiplikátory.

5.2 Důkaz jevu výskytu volného extrému v řešení neklasické úlohy na vázaný extrém

Patří-li výsledek úlohy volného extrému pod všechna omezení neklasické úlohy na vázaný extrém, potom je řešením původní neklasické úlohy na vázaný extrém. Výsledek je určen rovnicí (4.11) a potvrzen příslušnou definitností Hessovy matice (4.13).

Prvním důkazem tohoto jevu je test podmínky na řešení volného extrému pro úlohu (5.1)–(5.2). Nejprve je nutno konkretizovat vztahy (4.11) a (4.13) pro kvadratickou účelovou funkci (5.62) resp. (5.1). Podmínka pro zjištění stacionárních bodů je odpovídající upravený vztah (5.63) pro kvadratickou účelovou funkci. Pozitivně definitní maticí (5.64) je zajištěna existence minima ve stacionárním bodě (5.65).

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{b}^T \mathbf{x} \quad (5.62)$$

$$\left. \begin{aligned} \nabla f(\mathbf{x}) = \mathbf{x} \mathbf{H} + \mathbf{b} = \mathbf{0} \\ \mathbf{x} = -\mathbf{H}^{-1} \mathbf{b} \end{aligned} \right\} \quad (5.63)$$

$$\nabla^2 f(\mathbf{x}) = \mathbf{H} = \begin{bmatrix} 2 & -2 & 1 \\ -2 & 6 & 2 \\ 1 & 2 & 10 \end{bmatrix} > 0 \quad (5.64)$$

$$\left. \begin{aligned} \mathbf{x} = - \begin{bmatrix} 2 & -2 & 1 \\ -2 & 6 & 2 \\ 1 & 2 & 10 \end{bmatrix}^{-1} [2 \ 2 \ 1]^T \\ \mathbf{x}^* = \hat{\mathbf{x}} = [-2,5172 \ -1,3103 \ 0,4138]^T \end{aligned} \right\} \quad (5.65)$$

Jelikož platí (5.66), detailněji rozepsané ve vztazích (5.67), pak platí, že řešení (5.65) úlohy volného extrému s absencí podmínek omezení spadá pod všechna omezení původní neklasické úlohy na vázaný extrém (5.1)–(5.2) a je optimem.

$$\forall j, j = 1, \dots, m : M_j(\hat{\mathbf{x}}) - \gamma_j \leq 0 \quad (5.66)$$

$$\left. \begin{aligned} M_1(\hat{\mathbf{x}}) - \gamma_1 &= -7 \\ M_2(\hat{\mathbf{x}}) - \gamma_2 &= -11,4138 \\ M_3(\hat{\mathbf{x}}) - \gamma_3 &= -6,7241 \end{aligned} \right\} \quad (5.67)$$

Pomocí postupu (5.62)–(5.67) bylo ověřeno, že může nastat situace, kdy je volný extrém výsledkem neklasické úlohy na vázaný extrém.

Druhý důkaz jevu, o němž pojednává tato kapitola, bude podpořen kapitolou 4.2.2, jež tento jev připouští pro obecnou úlohu s libovolnou účelovou funkcí. Konkrétně nyní budou tvrzení v kapitole 4.2.2 dokázána v souvislosti s experimentálním příkladem (5.1)–(5.2) kapitoly 5.1.1, a tedy pro případ kvadratické účelové funkce. Především se jedná o práci s Kuhn–Tuckerovými podmínkami, kdy se pro variantu nulových Lagrangeových multiplikátorů naskýtá možnost řešení úlohy stanovením pouhého volného extrému.

Pro nulové Lagrangeovy multiplikátory dle vztahu (5.25), tedy gradientu (4.52) nové Lagrangeovy funkce podle doplňkových proměnných (pro kompenzaci nerovnostních omezení na rovnostní), zbývá pouze jediná varianta kombinace nulování proměnných ve vztahu (5.25) resp. (5.68). Z uvedeného faktu vyplývá jediný tvar submatic (5.69) lineární soustavy rovnic (5.32) v souladu s postupem rozboru v rámci kapitoly 5.1.1.

$$(\lambda_1 = \lambda_2 = \lambda_3 = 0) \wedge (\bar{y}_1 \neq \bar{y}_2 \neq \bar{y}_3 \neq 0) \quad (5.68)$$

$$\mathbf{F}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{F}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{F}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{F}_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.69)$$

Toto určení stacionárního bodu plně odpovídá kombinaci 9 v řešení experimentálního příkladu, a proto je stacionární bod roven (5.70).

$$\hat{\mathbf{x}} = [-2,5172 \quad -1,3103 \quad 0,4138]^T \quad (5.70)$$

Skutečnost, že je stacionární bod (5.70) i optimem bylo v příkladu potvrzeno pozitivní definitností Hessovy matice (5.28) a splněním podmínek Kuhn–Tuckerova věty.

Druhý důkaz jevu spočívá v náležitosti řešení (5.70) pod první Kuhn–Tuckerovou podmínku pro kvadratickou účelovou funkci (5.71). Je patrné, že pro nulový vektor Lagrangeových multiplikátorů se podmínka redukuje na vztah (5.63) z prvního důkazu jevu, což je podmínka pro hledání stacionárních bodů úlohy volného extrému. Tímto je jev též dokázán.

$$\begin{aligned} \nabla f(\mathbf{x}^\bullet) + \dot{\lambda}^T \mathbf{M} &= \mathbf{0} \Leftrightarrow \\ \Leftrightarrow \mathbf{x}^\bullet \mathbf{H} + \mathbf{b} + [0 \quad 0 \quad 0] \mathbf{M} &= \mathbf{0} \Leftrightarrow \\ \Leftrightarrow \mathbf{x}^\bullet \mathbf{H} + \mathbf{b} = \mathbf{0} \Leftrightarrow \hat{\mathbf{x}} \mathbf{H} + \mathbf{b} &= \mathbf{0} \end{aligned} \quad (5.71)$$

Výsledky prvního důkazu jsou přímo aplikovatelné a usnadňují ověření, zda-li nedojde k jevu redukce neklasické úlohy na úlohu volného extrému. Pak by určení optima bylo celkově jednodušší.

Druhý důkaz opět potvrzuje možnost výskytu jevu v neklasické úloze na vázaný extrém, a to aparátem podmínek Kuhn–Tuckerovy věty, konkrétně pro kvadratický tvar účelové funkce. Tímto byl jev umožňující redukci výpočtů opět potvrzen, avšak není přímo aplikovatelný v algoritmech.

5.3 Návrh optimalizační metody kvadratického programování

Náplní kapitoly je představení vlastního návrhu řešení neklasické úlohy na vázaný extrém, a to konkrétně tvaru úlohy kvadratického programování. Návrh je založen na analytických vztazích teorie duality a je porovnán v kapitole 6 po stránce výpočetní náročnosti s numerickou Hildrethovou metodou (kap. 4.2.3). Verifikace správnosti jejího řešení se nachází také v kapitole 6.

Navrhovaná optimalizační metoda vychází z obecného principu teorie duality (kap. 4.1.7), jež sama o sobě zajišťuje snížení výpočetní náročnosti řešení úlohy kvadratického programování (4.37) pro $m < n$, a to oproti primárním metodám pracujících pouze s proměnnou x . Uváděná metoda se sice opírá o stejné teoretické předpoklady jako obecně každá duální metoda, ale převládá v ní oproti např. numerické metodě Hildretha analytický přístup k řešení. Duální princip je dále modifikován.

Myšlenka úpravy přístupu k řešení duální úlohy vzešla z experimentů s podmínkami Kuhn–Tuckerovy věty pro duální úlohu kvadratického programování. Byla experimentálně porušena nutná 4. podmínka této věty a poté hledána náprava alternativním způsobem, který nakonec poskytl správné řešení. Tento způsob zahrnuje převod duální úlohy s podmínkou nezáporné duální proměnné na posloupnost menších úloh jednoduššího typu – úloh volného extrému, ze kterých lze vyčíst pouze některé informace. Posloupnost úloh volného extrému sama o sobě nedává finální výsledek původní úlohy. Tyto úlohy volného extrému operují navíc s variabilními rozměry svých proměnných. Tyto úlohy tedy samy o sobě správné řešení neposkytují, ale zpětnou aplikací vynechané 4. Kuhn–Tuckerovy podmínky na jejich dílčí řešení se lze dobrat ke správnému výsledku původní úlohy, který lze dodržáním dále popsaného postupu obdržet.

Např. v knize [20] je prezentován známý převod úlohy neklasické úlohy na vázaný extrém s nelineární účelovou funkcí na posloupnost úloh kvadratického programování (SQP – úloha sekvenčního programování) nebo převod kvadratického programování na posloupnost úloh lineárního programování (SLP – úloha sekvenčního lineárního programování). V žádné z metod není porušen předpoklad nutné Kuhn–Tuckerovy podmínky a výsledek poslední úlohy v sekvenci je přímo považován za aproximaci výsledku originální úlohy. V navrhované metodě je sice analogické rozdělení složitější úlohy na podproblémy, ale informace z výsledků úloh volného extrému musí být využívány pouze částečně a ne jako poslední člen sekvence numericky se upřesňujících řešení.

Nyní bude vysvětlen popis principu navrhované optimalizační metody, jež byla sestavena experimentální cestou. Snahou je tedy obecně řešit duální úlohu (4.44) nejprve s porušením nutné podmínky nezápornosti (4. Kuhn–Tuckerovy podmínky) vektorové proměnné \mathbf{d} . Druhá část postupu je již snahou provést zpětně nápravu, aby bylo vše v souladu s platností všech Kuhn-Tuckerových podmínek zadané úlohy (4.44). Tento přístup zahrnuje několik po sobě následujících úloh typu volného extrému. Každá další úloha volného extrému je oproti předcházející úloze navíc redukováná co do rozměrů matice N a vektoru \mathbf{o} . Tato skutečnost je výhodným aspektem vedoucím ke snížení výpočetní náročnosti řešení úlohy (4.37), resp. (4.44).

Opakování provádění elementárních úloh typu volného extrému souvisí s výskytem záporných nežádoucích hodnot prvků ve vektoru \mathbf{d} . Cyklus tedy probíhá v případě porušení 4. Kuhn–Tuckerovy podmínky jednotlivými prvky vektoru duální proměnné. Při výskytu kladných znamének v tomto dílčím výsledku se zúží počet úloh na volný extrém a pokračuje se k další nápravě porušení podmínek nezápornosti zapříčiněných ostatními prvky vektoru \mathbf{d} .

Prvním krokem postupu je odebrání 4. podmínky Kuhn–Tuckerovy věty z definice duální úlohy (4.44). Modifikovaná úloha má tvar (5.72). Řešení je pak přeznačeno symbolem křížku značící skutečnost porušení podmínky nezápornosti $\mathbf{d} \geq \mathbf{0}$. Výsledek \mathbf{d}^\times (5.73) upravené úlohy (5.72) je pak volným extrémem, a tedy lze úlohu (5.72) vyhodnotit vztahem (5.74). U jednotlivých prvků vektoru \mathbf{d}^\times se připouští možný výskyt záporných hodnot tedy případ (5.75).

$$\mathbf{d}^\times = \arg \min \left\{ \frac{1}{2} \mathbf{d}^T N \mathbf{d} + \mathbf{o}^T \mathbf{d} \right\}; \mathbf{d} \in \mathcal{R}^{m,1} \quad (5.72)$$

$$\mathbf{d}^\times = [d_1^\times \cdots d_m^\times]^T \quad (5.73)$$

$$\mathbf{d}^\times = -N^{-1} \mathbf{o} \quad (5.74)$$

$$\text{sgn}(d_i^\times) = -1; i \in \{1, \dots, m\} \quad (5.75)$$

V druhém kroku navrhované metody jsou provedeny operace k nápravě porušení 4. Kuhn–Tuckerovy podmínky z prvního kroku.

Pro tyto účely je navržena myšlenka řešit novou redukovanou úlohu se strukturou (5.76) s výsledkem \mathbf{d}° (5.80) též typu volného extrému. Úloha je spjatá se vztahy (5.77)–(5.79) a její zadání se váže na dále vysvětlenou množinu indexů \mathcal{T} (5.77) a konstantu δ . Redukovaná je tato úloha v tom smyslu, že její Hessián (5.78) a vektor (5.79) má menší rozměry než-li měly proměnné N a \mathbf{o} v úloze (5.72).

Tento krok je nutné provádět cyklicky dokud pro finální optimum \mathbf{d}^* (4.44) nebude platit $\mathbf{d}^* \geq \mathbf{0}$. To znamená, že se jedná o provedení sekvence úloh volného extrému. Vektor \mathbf{d}^* je sestaven až z takového výsledku \mathbf{d}^o (5.80), pro který platí $\mathbf{d}^o \geq \mathbf{0}$, jinak se použije \mathbf{d}^o jako \mathbf{d}^\times pro další opakování kroku 2.

$$\mathbf{d}^o = \arg \min \left\{ \frac{1}{2} \mathbf{d}^T \mathbf{N}^o \mathbf{d} + \mathbf{o}^{o^T} \mathbf{d} \right\}; \mathbf{d} \in \mathcal{R}^{\delta,1} \subseteq \mathcal{R}^{m,1} \quad (5.76)$$

Je definována množina \mathcal{T} , jež bude zahrnovat řádkové indexy těch prvků vektoru \mathbf{d}^\times , jež splňují podmínku nezápornosti a nenulovosti.

$$\mathcal{T} = \{\}; \forall i \in \{1, \dots, m\}: (\text{sgn}(d_i^\times) = 1) \rightarrow \mathcal{T} = \mathcal{T} \cup \{i\} \quad (5.77)$$

Potom Hessián \mathbf{N}^o (5.78) je taková submatice původní matice \mathbf{N} , v níž byly ponechány pouze takové řádky a současně takové sloupce s indexy shodnými s prvky množiny \mathcal{T} (zapsané zkráceně $\mathbf{N}(\mathcal{T}, \mathcal{T})$). Vektoru \mathbf{o}^o (5.79) budou oproti \mathbf{o} ponechány pouze ty řádky s indexací taktéž odpovídající svou hodnotou prvkům množiny \mathcal{T} . Nechť konstanta δ vyjadřuje počet prvků množiny \mathcal{T} . Ta pak určuje rozměry nově sestavované matice \mathbf{N}^o a vektorů \mathbf{o}^o , \mathbf{d}^o .

$$\mathbf{N}^o = \mathbf{N}(\mathcal{T}, \mathcal{T}); \mathbf{N}^o \in \mathcal{R}^{\delta, \delta} \quad (5.78)$$

$$\mathbf{o}^o = \mathbf{o}(\mathcal{T}, 1); \mathbf{o}^o \in \mathcal{R}^{\delta, 1} \quad (5.79)$$

$$\mathbf{d}^o = -\mathbf{N}^{o^{-1}} \mathbf{o}^o \quad (5.80)$$

Pokud vektor \mathbf{d}^o zahrnuje pouze kladné prvky, je toto řešení naplněním myšlenky metody (5.82) (pozn.: symbol \rightarrow vyjadřuje obecný sled operací) a je dále sestaveno pravidlem (5.81) do podoby duální optima \mathbf{d}^* (4.45) duální úlohy (4.44). Dle (4.46) pak následuje určení optima \mathbf{x}^* primárního řešení úlohy kvadratického programování.

$$\forall i \in \{1, \dots, m\}: \begin{cases} i \in \mathcal{T}; d_i^* = d_i^o \\ i \notin \mathcal{T}; d_i^* = 0 \end{cases} \quad (5.81)$$

$$\mathbf{d}^\times \rightarrow (\mathbf{d}^o; \mathbf{d}^o \geq \mathbf{0}^{\delta,1}) \rightarrow (\mathbf{d}^*; \mathbf{d}^* \geq \mathbf{0}^{m,1}) \rightarrow (\mathbf{x}^* = \mathbf{x}^*(\mathbf{d}^*)) \quad (5.82)$$

V opačném případě, tj. pokud je v řešení vektoru \mathbf{d}^o alespoň jeden záporný prvek, je postup druhého kroku daný vztahy (5.76)–(5.81) opakován. Nevyhovující vektor \mathbf{d}^o je přeznačen na \mathbf{d}^\times a nový vektor \mathbf{d}^o je stanoven výše uvedeným postupem. Metoda je založena na analytických vztazích, takže není numerické povahy, ale zahrnuje prvky opakování. Lze ji tedy hodnotit jako iterační algoritmus, a to s podmínkou ukončení $\mathbf{d}^o \geq \mathbf{0}$ kroku 2.

5.4 Návrh modifikací v prediktivním řízení

5.4.1 Princip částečných eliminací v definici omezení veličin MPC

Z pohledu stanovených cílů disertační práce lze nové návrhy v rámci prediktivního řízení směřovat do modifikací optimalizátoru nebo do modifikací samotného prediktivního řízení.

V kapitole 5.3 je předložen návrh řešení optimalizační úlohy. Jedná se čistě o řešení, jež souvisí s teorií optimalizace. V této kapitole je konkrétně uvažována souvislost optimalizace s prediktivním řízením procesů. Cílem je snížení výpočetní náročnosti algoritmu prediktivního řízení.

Návrh úpravy řešení prediktivního řízení úzce souvisí s operacemi v rámci definice omezení veličin v prediktivním řízení, které mají analogii s omezeními v optimalizační úloze kvadratického programování. Tato skutečnost se promítá do rovnosti odpovídajících proměnných MPC a úlohy kvadratického programování dle vztahu (4.82).

Hodnota parametru m souvisí s počtem řádků matice omezení \mathbf{M} , resp. počtem prvků vektoru $\boldsymbol{\gamma}$. Matice \mathbf{M} zahrnuje definice omezení dle tabulky 1 a 2 (kap. 4.3.3, 4.3.4). Každý typ omezení veličin z daných tabulek pro požadovaný rozměr řízení je sestaven z více řádků. Údaje tabulek 1 a 2 o počtu řádků jednotlivých definic typů omezení sumarizuje tabulka 3. Z tabulky 3 je patrné, že pro vyšší horizonty se stává optimalizační úloha kvadratického programování, tedy realizace zákona řízení MPC, výpočetně rozsáhlejší. Např. při uvažování omezující podmínky typu u_{\min} pro TITO prediktivní řízení s horizontem $N_u = 10$ je počet podmínek již 20. Vliv nárůstu počtu podmínek m na schopnost uřídit procesy s rychlou dynamikou prediktivním řízením bude prověřen na základě měření v kapitole 6.

Tab. 3 Počet řádků jednotlivých typů omezení v prediktivním řízení

	Jednorozměrové MPC	Dvourozměrové MPC
Typ omezení	Počet omezujících podmínek m	
Δu_{\min}	N_u	$2N_u$
Δu_{\max}	N_u	$2N_u$
u_{\min}	N_u	$2N_u$
u_{\max}	N_u	$2N_u$
y_{\min}	$N_2 - N_1 + 1$	$2N_2 - 2N_1 + 2$
y_{\max}	$N_2 - N_1 + 1$	$2N_2 - 2N_1 + 2$

Návrh modifikace algoritmu prediktivního řízení kontinuálně navazuje na myšlenky autora knihy [4] L. Wanga (kap. 4.3.5). Zavedená metoda je význačná i tím, že považuje optimalizační úlohu za modul a nad ním provádí modifikace, a to úpravou zadání optimalizačního problému. Vše v souladu s prezentovanou teorií převodu neklasické úlohy na vázaný extrém na úlohu volného extrému. Proto i pro navrhovanou metodu modifikace MPC bude přínosnou vlastností modulární a obecnější využití. Pokud by v budoucnu byla vyvinuta rychlejší optimalizační metoda, lze na ni uplatnit navrhovaný modulární postup řešení pro aplikace v prediktivním řízení procesů s rychlou dynamikou.

Lze konstatovat, že jev možného převodu neklasické úlohy na vázaný extrém, a to na úlohu volného extrému, dokazuje vzdálenostní rezervy skutečného optima této úlohy od existujících omezení. Výskyt tohoto jevu je možný (kap. 5.2) avšak není vždy zaručený a nelze z úlohy eliminovat vždy najednou všechna omezení. Proto zavedená modifikace není příliš vhodná pro řízení procesů s rychlou dynamikou. Nastává totiž situace, že jev nenastane vůbec a musí se řešit původní originální optimalizační úloha ve svém plném rozsahu.

Předkládaný vlastní návrh nabízí šetrnější způsob eliminací. Je plynule navázáno na práci autora L. Wanga, avšak jsou redukována pouze některá omezení optimalizační úlohy.

Nechť je počet omezení, jež zůstanou po provedení eliminace roven počtu m_r . Pak zavedená metoda (4.3.5) může v některých případech realizovat redukci $m_r = 0$, ale může nastat i negativní výsledek $m_r = m$ s řešením úplné úlohy kvadratického programování. Návrh modifikace v MPC bude předpokládat výskyt parametru m_r v rozpětí $0 \leq m_r < m$, se snahou vzdalovat svou hodnotou od krajní hodnoty m tohoto intervalu.

Jelikož je počet sloupců n matice omezení \mathbf{M} , dle analogie (4.82), přímo roven řídicímu horizontu N_u , není jeho redukci potřeba řešit speciálním přístupem, neboť ho lze volit přímo. Počet řádků však řádově více narůstá s nastavením horizontů prediktivního řízení, uvažovaným rozměrem řízení a počtem omezení veličin MPC.

Redukce podmínek omezení byla prezentována na konferenci Process Control v publikaci [33]. Myšlenkou publikace bylo eliminovat omezení ve shlcích, a to vždy pro větší množinu omezujících podmínek patřící pod určitý typ omezení. Eliminace byly účinné a byly tak potvrzeny rezervy při redukci neklasické úlohy na vázaný extrém na úlohu stejného typu, avšak menšího rozsahu. Návrh v této kapitole bude eliminovat m_r více a eliminace budou šetrnější oproti přístupu [33].

Návrh šetrnější eliminace spočívá v myšlence zachování rozsahu okna horizontu MPC. Sníží se počet podmínek napříč maticí \mathbf{M} a vektorem $\boldsymbol{\gamma}$. Samotná redukce mění originální zadání optimalizační úlohy na nové, jehož řešení spadá pod neredukovaná omezení dle původního zadání. Což je zapříčiněno existencí rezerv, a to v rámci vzdáleností optima od definovaných omezení. Modifikace 4.3.5 odebrala všechna omezení, ale rezervy dle (4.2.2) existují. Typové eliminace též zahrnovaly rezervy a neklasická úloha na vázaný extrém tak zúžila svůj výpočetní rozsah, ale opět splnila všechny původní omezení. Díky rezervám patří i výsledek redukované úlohy pod původní omezení všech podmínek definovaných maticemi \mathbf{M} a vektorem $\boldsymbol{\gamma}$. A to pro navrhovanou variantu napříč oknem horizontu MPC. Tuto redukci způsobují právě potenciální rezervy, které se snaží metoda lépe využít.

Nechť je uvažována matice $\tilde{\mathbf{M}}$ a vektor $\tilde{\boldsymbol{\gamma}}$. Tyto proměnné přejímají obsahy matice \mathbf{M} resp. vektoru $\boldsymbol{\gamma}$, a to dle strategie jejich eliminací. Rozměry těchto nových proměnných jsou definovány dle (5.83). Příklad zavedené modifikace $m_r = 0$ by uvažoval nulovou matici $\tilde{\mathbf{M}}$ a nulový vektor $\tilde{\boldsymbol{\gamma}}$.

$$\left. \begin{array}{l} \tilde{\mathbf{M}} \in \mathcal{R}^{m_r, n} \\ \tilde{\boldsymbol{\gamma}} \in \mathcal{R}^{m_r, 1} \end{array} \right\}; m_r \in (0; m) \quad (5.83)$$

Samotný návrh modifikací v MPC lze vyjádřit následným pravidlem. Nechť je uvažováno značení jednotlivých prvků matice \mathbf{M} a vektoru $\boldsymbol{\gamma}$ v souladu se vztahem (4.7), a to s opakovaným uvedením v (5.84) z důvodu přehlednosti.

$$\underbrace{\begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1n} \\ \vdots & & \ddots & \vdots \\ M_{m1} & M_{m2} & \cdots & M_{mn} \end{bmatrix}}_{\mathbf{M}}; \underbrace{\begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_m \end{bmatrix}}_{\boldsymbol{\gamma}} \quad (5.84)$$

Nová matice $\tilde{\mathbf{M}}$ a vektor $\tilde{\boldsymbol{\gamma}}$ jsou konstruovány z prvků matice \mathbf{M} a vektoru $\boldsymbol{\gamma}$. V každém kroku prediktivního řízení je nejprve $\tilde{\mathbf{M}}$ prázdná a též je prázdný i vektor $\tilde{\boldsymbol{\gamma}}$. Definice je totiž nutné aktualizovat v každém kroku prediktivního řízení z důvodu změny hodnot některých proměnných řízení (při uvažování typů omezení $u_{\min}, u_{\max}, y_{\min}, y_{\max}$). Poté jsou z definic původní matice \mathbf{M} a vektoru $\boldsymbol{\gamma}$ ponechány jen určité prvky resp. přesunuty pouze vybrané prvky do nové matice $\tilde{\mathbf{M}}$ a vektoru $\tilde{\boldsymbol{\gamma}}$. Nová úloha MPC (5.85) má menší počet podmínek, ale poskytuje stejné výsledky, jako úloha bez eliminací, a to díky existence rezerv.

$$\Delta \mathbf{u} = \arg \min \left\{ \frac{1}{2} \Delta \mathbf{u}^T \mathbf{H} \Delta \mathbf{u} + \mathbf{b}^T \Delta \mathbf{u} \mid \tilde{\mathbf{M}} \Delta \mathbf{u} \leq \tilde{\boldsymbol{\gamma}} \right\} \quad (5.85)$$

Při uvažování jednorozměrového MPC řízení je uvažována nová definice omezení dle tabulky 4. Obdobně je definice aktualizována pro dvourozměrový případ prediktivního řízení v tabulce 5. Při skládání více definic omezení nezáleží na pořadí. Důležité je zařazení úvodní omezující podmínky pro každý typ omezení (SISO 1. omezení, TITO ponechání 2. podmínky). Jedná se o šetrné snížení hustoty omezení v jejich matematické definici.

Tab. 4 Definice modifikovaných podmínek v SISO prediktivním řízení

Typ omezení	Definice matice $\tilde{\mathbf{M}}$ a vektoru $\tilde{\boldsymbol{\gamma}}$	Počet omezení m_r po eliminaci
Δu_{\min} , u_{\min}	$\tilde{\mathbf{M}} = \begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1N_u} \\ \vdots & \vdots & \cdots & \vdots \\ M_{i1} & M_{i2} & \ddots & M_{iN_u} \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}; \tilde{\boldsymbol{\gamma}} = \begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_i \\ \vdots \end{bmatrix};$ $2 \leq \{i = 2j\} \leq N_u; j \in \mathcal{N}$	$m_r \leq \left\{ 1 + \frac{N_u}{2} \right\} <$ $< \{m_r + 1\}$
Δu_{\max} , u_{\max}	$\tilde{\mathbf{M}} = \begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1N_u} \\ \vdots & \vdots & \cdots & \vdots \\ M_{i1} & M_{i2} & \ddots & M_{iN_u} \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}; \tilde{\boldsymbol{\gamma}} = \begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_i \\ \vdots \end{bmatrix};$ $2 \leq \{i = 2j + 1\} \leq N_u; j \in \mathcal{N}$	$m_r \leq \left\{ 1 + \frac{N_u - 1}{2} \right\} <$ $< \{m_r + 1\}$
y_{\min}	$\tilde{\mathbf{M}} = \begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1N_u} \\ \vdots & \vdots & \cdots & \vdots \\ M_{i1} & M_{i2} & \ddots & M_{iN_u} \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}; \tilde{\boldsymbol{\gamma}} = \begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_i \\ \vdots \end{bmatrix};$ $2 \leq \{i = 2j\} \leq N_u; j \in \mathcal{N}$	$m_r \leq \left\{ 1 + \frac{N_2 - N_1 + 1}{2} \right\}$ $< \{m_r + 1\}$
y_{\max}	$\tilde{\mathbf{M}} = \begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1N_u} \\ \vdots & \vdots & \cdots & \vdots \\ M_{i1} & M_{i2} & \ddots & M_{iN_u} \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}; \tilde{\boldsymbol{\gamma}} = \begin{bmatrix} \gamma_1 \\ \vdots \\ \gamma_i \\ \vdots \end{bmatrix};$ $2 \leq \{i = 2j + 1\} \leq N_u; j \in \mathcal{N}$	$m_r \leq \left\{ 1 + \frac{N_2 - N_1}{2} \right\} <$ $< \{m_r + 1\}$

Tab. 5 Definice modifikovaných podmínek v TITO prediktivním řízení

Typ omezení	Definice matice $\tilde{\mathbf{M}}$ a vektoru $\tilde{\boldsymbol{\gamma}}$	Počet omezení m_r po eliminaci
Δu_{\min} , u_{\min}	$\tilde{\mathbf{M}} = \begin{bmatrix} M_{21} & M_{22} & \cdots & M_{2N_u} \\ \vdots & \vdots & \cdots & \vdots \\ M_{i1} & M_{i2} & \ddots & M_{iN_u} \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}; \tilde{\boldsymbol{\gamma}} = \begin{bmatrix} \gamma_2 \\ \vdots \\ \gamma_i \\ \vdots \end{bmatrix};$ $3 \leq \{i = 2j\} \leq N_u; j \in \mathcal{N}$	$m_r \leq \{1 + N_u\} < \\ < \{m_r + 1\}$
Δu_{\max} , u_{\max}	$\tilde{\mathbf{M}} = \begin{bmatrix} M_{21} & M_{22} & \cdots & M_{2N_u} \\ \vdots & \vdots & \cdots & \vdots \\ M_{i1} & M_{i2} & \ddots & M_{iN_u} \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}; \tilde{\boldsymbol{\gamma}} = \begin{bmatrix} \gamma_2 \\ \vdots \\ \gamma_i \\ \vdots \end{bmatrix};$ $3 \leq \{i = 2j + 1\} \leq N_u; j \in \mathcal{N}$	$m_r \leq \left\{1 + N_u - \frac{1}{2}\right\} < \\ < \{m_r + 1\}$
y_{\min}	$\tilde{\mathbf{M}} = \begin{bmatrix} M_{21} & M_{22} & \cdots & M_{2N_u} \\ \vdots & \vdots & \cdots & \vdots \\ M_{i1} & M_{i2} & \ddots & M_{iN_u} \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}; \tilde{\boldsymbol{\gamma}} = \begin{bmatrix} \gamma_2 \\ \vdots \\ \gamma_i \\ \vdots \end{bmatrix};$ $3 \leq \{i = 2j\} \leq N_u; j \in \mathcal{N}$	$m_r \leq \{N_2 - N_1 + 2\} \\ < \{m_r + 1\}$
y_{\max}	$\tilde{\mathbf{M}} = \begin{bmatrix} M_{21} & M_{22} & \cdots & M_{2N_u} \\ \vdots & \vdots & \cdots & \vdots \\ M_{i1} & M_{i2} & \ddots & M_{iN_u} \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}; \tilde{\boldsymbol{\gamma}} = \begin{bmatrix} \gamma_2 \\ \vdots \\ \gamma_i \\ \vdots \end{bmatrix};$ $3 \leq \{i = 2j + 1\} \leq N_u; j \in \mathcal{N}$	$m_r \leq \left\{N_2 - N_1 + \frac{1}{2}\right\} \\ < \{m_r + 1\}$

5.4.2 Možnosti začlenění navrhované optimalizační metody

Problematickým místem navrhované optimalizační metody, souvisejícím pouze s případem regulární matice \mathbf{M} , je operace inverze v rámci výpočtu duálního Hessiánu \mathbf{N} (4.40), resp. (5.74). S tímto tvarem zadání optimalizační úlohy se naopak dokáže numericky vypořádat zavedená Hildrethova metoda. Navíc se operace inverze objevuje také v sekvenci úloh na volný extrém ve vztahu (5.80). Pro neklasickou úlohu na vázaný extrém je v analytickém přístupu k řešení, o což se pokouší i navrhovaná optimalizační metoda, nutná série podmínek regularity (kap. 4.1.4). Podmínky regularity kontroluje i sestavený analytický algoritmus (kap. 5.1). Samotná regularita se testuje pro Jacobiho matici (5.86) s omezeními matice \mathbf{M} . Matice \mathbf{M} má vliv na regularitu duálního Hessiánu a nemohla by být provedena operace jeho inverze. Výše uvedené analogicky platí pro matici (5.87) v TITO prediktivním řízení.

$$\begin{bmatrix} \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+1)} & \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+2)} & \dots & \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+N_u)} \\ \vdots & \vdots & & \vdots \\ \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+1)} & \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+2)} & \dots & \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+N_u)} \end{bmatrix} \quad (5.86)$$

$$\begin{bmatrix} \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+1)} & \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+2)} & \dots & \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+N_u)} \\ \vdots & \vdots & & \vdots \\ \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+1)} & \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+2)} & \dots & \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+N_u)} \end{bmatrix} \quad (5.87)$$

Jev výskytu singulární Jacobiho matice (5.86), resp. (5.87), nastává v MPC při aplikaci definic více typů omezení veličin řízení.

Navrhované modifikace v MPC řízení sice vynechávají některá omezení z původní matice \mathbf{M} , ale jelikož jsou nasazeny vždy první, resp. 2. řádky u TITO, bude k singularitě docházet. Ta je zcela jednoduše zapříčiněna lineárně závislými řádky v matici \mathbf{M} . Dle tabulek 1 a 2 je patrné, že první řádky definic matic u SISO či druhé řádky u TITO jsou pro omezovanou danou veličinu svými lineárními kombinacemi. Např. jsou lineárně závislými matice $-\mathbf{I}, \mathbf{I}$; $-\mathbf{G}, \mathbf{G}$ v tabulce 1 apod. a dále též i v tabulce 2. Samotný vektor γ nemá na regularitu přitom žádný vliv.

Jelikož byla brána navrhovaná duální metoda v kapitole 5.3 pouze z pohledu teorie optimalizace, byl tento krajní případ singularity Jacobiho matice (5.86)

pouze uvažován, a to tak jako u jiných metod optimalizace (připustění výjimek v kap. 4.1.4).

V aplikační oblasti prediktivního řízení je navrhovaná duální metoda použitelná pouze při aplikaci jednoho typu omezení v jednorozměrovém prediktivním řízení, aby nedocházelo k singularitě Jacobiho matice (5.86). Nosná modifikace, jež bude testována v kapitole 6 na svou efektivitu, bude návrh eliminací v MPC. V případě aplikace jednoho typu omezení v SISO MPC bude tento přístup použit společně s navrhovanou optimalizační metodou. Zde již bude platit pravidlo (5.88). V opačném případě bude použita zavedená Hildrethova metoda s navrhovaným principem eliminací.

$$\begin{vmatrix} \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+1)} & \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+2)} & \dots & \frac{\partial M_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+N_u)} \\ \vdots & \ddots & & \\ \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+1)} & \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+2)} & \dots & \frac{\partial M_m(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}(k+N_u)} \end{vmatrix} \neq 0 \quad (5.88)$$

6. EXPERIMENTÁLNÍ ČÁST PRÁCE

6.1 Prediktivní řízení procesů s rychlou dynamikou

6.1.1 Realizace prediktivního řízení

Pro účely simulací bylo implementováno softwarové řešení s GUI rozhraním pro prostředí MATLAB verze 6.5. Tento program je rozšířenou verzí vytvořeného softwaru [34] a nachází na přiloženém disku CD s popisem jednotlivých součástí v příloze B této práce. Software lze spustit skriptem *programPrediktivniRizeni.m*.

Je zde implementována simulace prediktivního řízení procesů založená na vstupně–výstupním modelu dle prezentované teorie v kapitole 4.3. Je realizován zákon řízení pro jednorozměrové i dvourozměrové procesy, jakožto optimalizační úloha kvadratického programování. Přičemž je možné provést rozsáhlejší nastavení všech důležitých parametrů MPC. Tato nastavení, jež mohou být do grafického rozhraní programu zadávána, lze uložit na pevný disk a zpětně načíst prostřednictvím konfiguračních souborů s příponou *gpc*. Hlavním záměrem programu je analýza výpočetní časové náročnosti, určení kvality řízení dle stanovených kritérií a počtu operací v rámci modulu optimalizace. Definice použitých prostředků pro tato měření se nachází v kapitole 6.2. Výsledky měření jsou exportovány automaticky do dvou datových souborů po každém provedeném měření, se strukturou dat oddělených středníky – formátem CSV (vhodném pro další zpracování v tabulkových programech).

Samotné simulace lze provádět pro zadaný matematický model procesu, jenž je dále řízen diskrétně. Výsledné znázornění průběhů veličin v rámci prediktivního řízení je vztaženo k pořadnicím period diskrétního řízení k . Není explicitně zadávána perioda vzorkování, načež jí zadané matematické modely odpovídají.

Zadání matematického modelu odpovídá obecnému tvaru diskrétní přenosové funkce (4.69) pro SISO variantu řízeného procesu a definice jednotlivých prvků maticových zlomků dle (4.71) pro TITO variantu (obr.1). Ukládat lze tato zadání modelů i izolovaně, a to do speciálního konfiguračního souboru s příponami $Gz1R$ (SISO) a $Gz2R$ (TITO).

Rozměr procesu: SISO TITO
 Oblast řešení: vstupně-výstupní

a11:	[1 -0.9448 0.2259]	b11:	[-0.0865 0.0101]
a12:	[0.0996 -0.0087]	b12:	[0.7963 -0.2138]
a21:	[0.1937 0.0860]	b21:	[0.2351 0.0920]
a22:	[1 0.4265 0.2179]	b22:	[0.1520 0.0616]

Lze načíst/uložit zadání procesu pro danou oblast řešení:

Obr. 1 Ukázka zadání diskrétního matematického modelu TITO procesu

Definici průběhu žádané veličiny lze provést určením hodnoty 1. a 2. meze. V rámci miniatury náhledu grafu je patrný výsledný průběh a je možné ovlivnit i délku aktivace jednotlivých mezí v rámci simulace řízení. Pro dvourozměrný případ je u druhé žádané veličiny proveden v její definici mírný posun o určitý úsek dopředu, aby byl v řízení patrný vzájemný vliv jednotlivých veličin.

Definice průběhu žádané veličiny

mez 1.: s počtem kroků:
 mez 2.: s počtem kroků:

Obr. 2 Zadávání průběhu žádané veličiny v rámci programu

Nastavení prediktivního regulátoru zahrnuje definici okna horizontu pomocí minimálního, řídicího a maximálního horizontu. Pro disertaci má největší význam zadání požadované metody pro řešení optimalizace, resp. pro realizaci zákona řízení MPC. Jsou k dispozici izolované metody, ale též varianty v kombinaci se zavedenými i navrhovanými přístupy v rámci disertační práce. Implementace zavedených metod optimalizace a modifikací v MPC jsou uvedeny programovým kódem v přílohách A.3 (Hildrethova metoda), A.4 (modifikace neklasické úlohy na vázaný extrém na úlohu volného extrému). V příloze A.5 je uvedena ukázka programového kódu navrhované optimalizační metody (kap. 5.3), jež je dle kapitoly 5.4.2 doporučena pouze pro řízení SISO procesů s jedním typem omezení veličin v MPC. Návrh eliminací v definicích typů omezení veličin MPC (kap. 5.4.1) zahrnuje kód, jenž je rozšířením algoritmu prediktivního řízení. Tento komplexní programový kód je přístupný v souboru *programPrediktivniRizeni.m* na přiloženém CD.

Omezení jednotlivých veličin mohou být definována pomocí jejich aktivace v odpovídající části GUI softwaru. Po jejich aktivaci zaškrtnutím tlačítka lze nastavit meze dané veličiny a rozsah působení této restriktce v prediktivním řízení. Lze jej nastavit zadáním intervalu pořadnic period vzorkování a projeví se v samotném prediktivním řízení. Z ukázky z programu (obr. 3) je patrné, že lze aktivovat působení poruchové veličiny v MPC, a to v podobě modelované neměřitelné poruchy, jež je přidána k výstupní veličině, resp. veličinám v případě TITO MPC.

Obr. 3 Ukázka nastavení parametrů prediktivního regulátoru

Po spuštění simulace se provede diskrétní prediktivní řízení zadaného modelu procesu. Následně jsou zobrazeny průběhy jednotlivých veličin MPC a případně i průběh poruchové veličiny, a to ve formě grafů programu MATLAB. Pro další způsob zpracování naměřených dat i statistik je proveden již zmíněný automatický export dat na pevný disk (viz. příloha B – výpis souborů disku CD). Zjištěné statistiky tvoří významnou část programu, neboť se podílí na prošetření navrhovaných metod v disertační práci po stránce analýzy jejich výpočetní náročnosti. Jsou blíže definovány a popsány dále, ale celkově se jedná o měření času provedení úseků algoritmu MPC s daným nastavením, zjištění příslušného počtu provedených aritmeticko–logických operací a vyčíslení kritérií kvality řízení.

Vytvořené softwarové prostředí pokrývá oblast vývoje optimalizačního subsystému pro prediktivní řízení. Detaily týkající se měření jsou popsány dále.

6.1.2 Volba modelů procesů s rychlou dynamikou

Princip konstrukce modelů řízených procesů s rychlou dynamikou souvisí s volbou pólů a nul (kap. 4.3.2). Byly zvoleny dva spojité matematické modely, a to pro SISO variantu (6.1) s póly a nulovým bodem dle (6.2) a pro TITO (6.3)–(6.4). Spojité přechodové charakteristiky jsou znázorněny na obrázcích 4–8 dále.

$$G(s) = \frac{s + 230}{(s + 9)(s + 11)} = \frac{s + 230}{s^2 + 20s + 99} \quad (6.1)$$

$$\pi_1 = -9; \pi_2 = -11; \mathcal{G}_1 = -230 \quad (6.2)$$

$$\mathbf{G}(s) = \left[\begin{array}{cc} \frac{s + 200}{(s + 15)(s + 10)} & \frac{s + 520}{(s + 20)(s + 10)} \\ \frac{s + 300}{(s + 20)(s + 15)} & \frac{s + 150}{(s + 15)(s + 10)} \end{array} \right] = \left. \begin{array}{l} \\ \\ \\ \\ \end{array} \right\} \quad (6.3)$$

$$= \left[\begin{array}{cc} \frac{s + 200}{s^2 + 25s + 150} & \frac{s + 520}{s^2 + 30s + 200} \\ \frac{s + 300}{s^2 + 35s + 300} & \frac{s + 150}{s^2 + 25s + 150} \end{array} \right]$$

$$\left. \begin{array}{l} \pi_1 = -15; \pi_2 = -10; \pi_3 = -20; \pi_4 = -10; \\ \pi_5 = -20; \pi_6 = -15; \pi_7 = -15; \pi_8 = -10; \\ \mathcal{G}_1 = -200; \mathcal{G}_2 = -520; \mathcal{G}_3 = -300; \mathcal{G}_4 = -150 \end{array} \right\} \quad (6.4)$$

S ohledem na vztah (4.68) a dobu ustálení jednotlivých spojitých přechodových charakteristik zvolených modelů procesů byla zvolena perioda vzorkování $T = 0,05$ s. Dále byla zvolena menší perioda $T = 0,03$ s, aby bylo porovnání navrhovaných přístupů v MPC se zavedenými přístupy zjevné pro zmenšující se T . Obě periody patří do intervalu dle (4.68) – pro SISO $\{0,03;0,05\} \in (0,029;0,079)$, TITO $\{0,03;0,05\} \in (0,027;0,067)$.

Diskretizace modelů s těmito periodami vzorkování byla provedena příkazy v prostředí MATLAB do podoby odpovídající obecným tvarům modelů (4.69), resp. (4.71). Spojitému SISO modelu (6.1) odpovídá pro periodu 0,05 s diskretní model (6.5), pro periodu 0,03 s model (6.6). Pro spojité TITO model procesu je určen pro periodu 0,05 s diskretizovaný model (6.7)–(6.9), pro periodu 0,03 s model (6.10)–(6.12), a to s potřebným vyjádřením pomocí maticových zlomků.

$$G_1(z^{-1}) = \frac{0,2378z^{-1} + 0,1183z^{-2}}{1 - 1,215z^{-1} + 0,3679z^{-2}} \quad (6.5)$$

$$G_2(z^{-1}) = \frac{0,1072z^{-1} + 0,04733z^{-2}}{1 - 1,482z^{-1} + 0,5488z^{-2}} \quad (6.6)$$

$$\mathbf{G}_1(z^{-1}) = \mathbf{A}^{-1}(z^{-1})\mathbf{B}(z^{-1}) \quad (6.7)$$

$$\mathbf{A}(z^{-1}) = \begin{bmatrix} 1 - 0,9448z^{-1} + 0,2259z^{-2} & 0,0996z^{-1} - 0,0087z^{-2} \\ 0,1937z^{-1} + 0,0860z^{-2} & 1 + 0,4265z^{-1} + 0,2179z^{-2} \end{bmatrix} \quad (6.8)$$

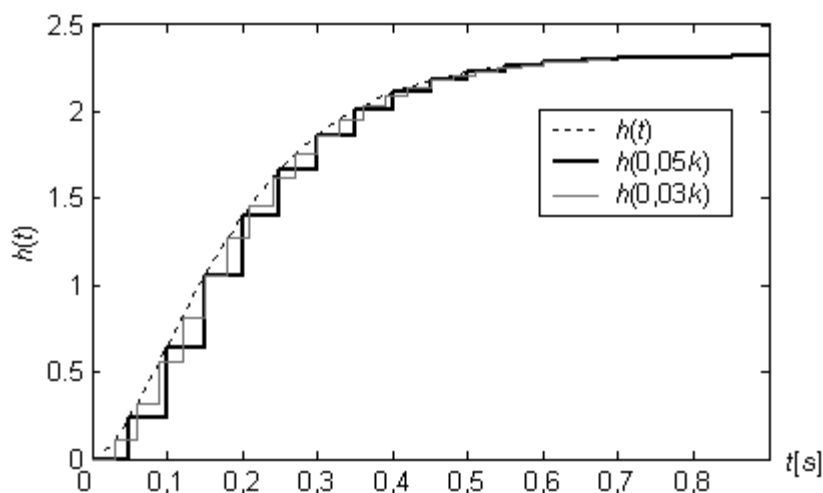
$$\mathbf{B}(z^{-1}) = \begin{bmatrix} -0,0865z^{-1} + 0,0101z^{-2} & 0,7963z^{-1} - 0,2138z^{-2} \\ 0,2351z^{-1} + 0,092z^{-2} & 0,152z^{-1} + 0,0616z^{-2} \end{bmatrix} \quad (6.9)$$

$$\mathbf{G}_2(z^{-1}) = \mathbf{A}^{-1}(z^{-1})\mathbf{B}(z^{-1}) \quad (6.10)$$

$$\mathbf{A}(z^{-1}) = \begin{bmatrix} 1 - 1,2668z^{-1} + 0,4015z^{-2} & 0,0772z^{-1} - 0,0278z^{-2} \\ 0,091z^{-1} + 0,0356z^{-2} & 1 + 0,1939z^{-1} + 0,1089z^{-2} \end{bmatrix} \quad (6.11)$$

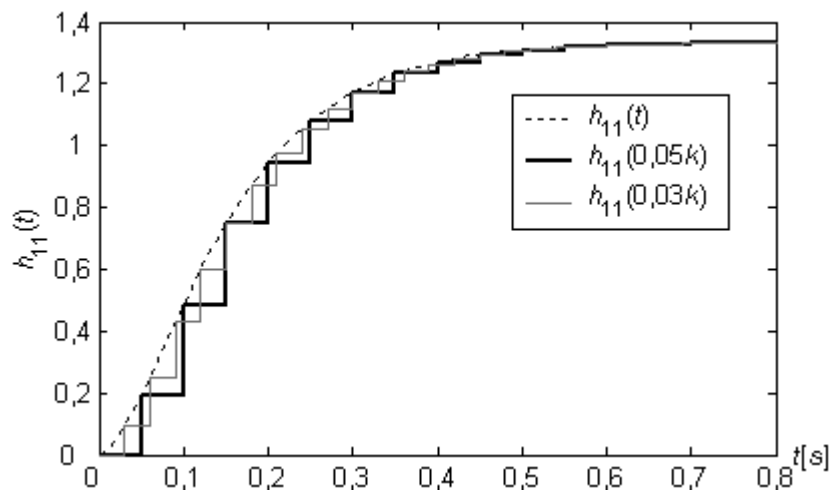
$$\mathbf{B}(z^{-1}) = \begin{bmatrix} -0,0677z^{-1} + 0,0257z^{-2} & 1,1499z^{-1} - 0,36z^{-2} \\ 0,1137z^{-1} + 0,0478z^{-2} & 0,0735z^{-1} + 0,0241z^{-2} \end{bmatrix} \quad (6.12)$$

Pro jednorozměrové modely procesu (6.1), (6.5)–(6.6) jsou vykresleny přechodové charakteristiky $h(t) = h(kT)$ na obrázku 4. Proměnná k (pořadnice vzorkování) nabývá hodnot číselného oboru Z_0^+ .

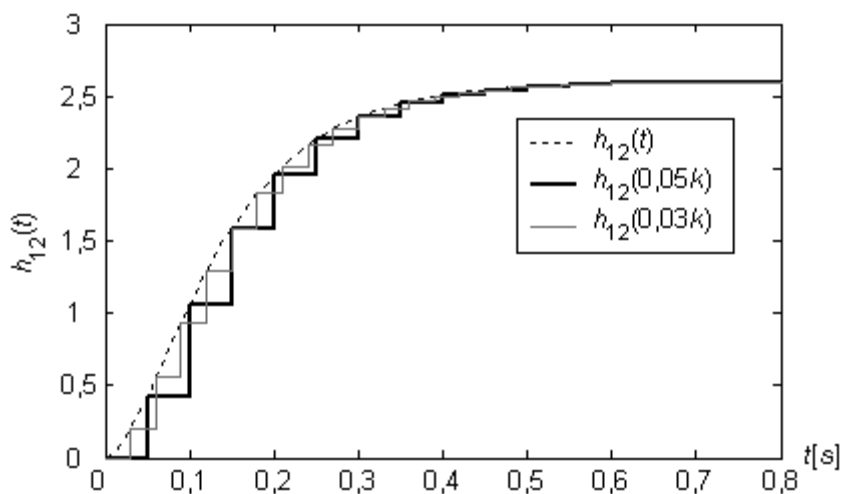


Obr. 4 Přechodové charakteristiky (spojitá a diskrétní) SISO modelu procesu

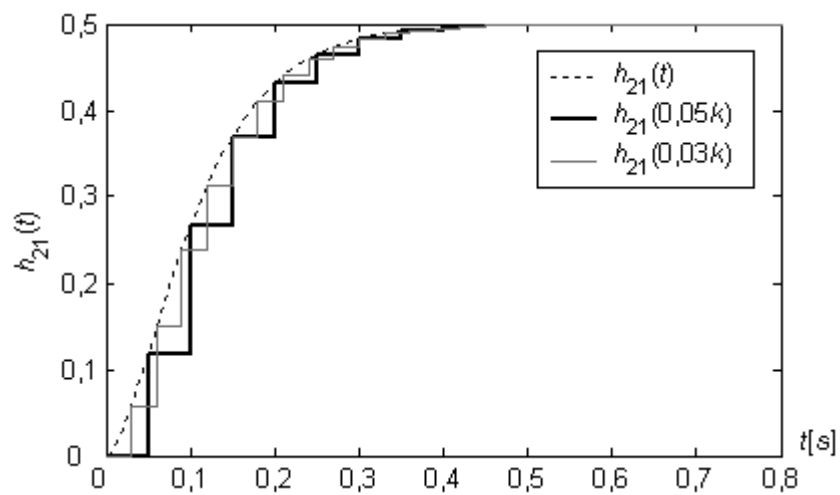
Pro dvourozměrné modely procesu (6.3), (6.7)-(6.12) jsou znázorněny dílčí grafy (obr. 5 – 8) přechodových charakteristik, a to pro jednotlivé dílčí přenosy matematického modelu daného maticí $\mathbf{G}(s)$. Uspořádaná dvojice indexů $\{i, j\}$ dílčí přechodové charakteristiky $h_{ij} = h_{ij}(t) = h_{ij}(kT)$; $k \in \mathbb{Z}_0^+$ odpovídá i -tému výstupu přenosu v reakci na jednotkový skok j -té vstupní veličiny daného přenosu uvnitř přenosové matice.



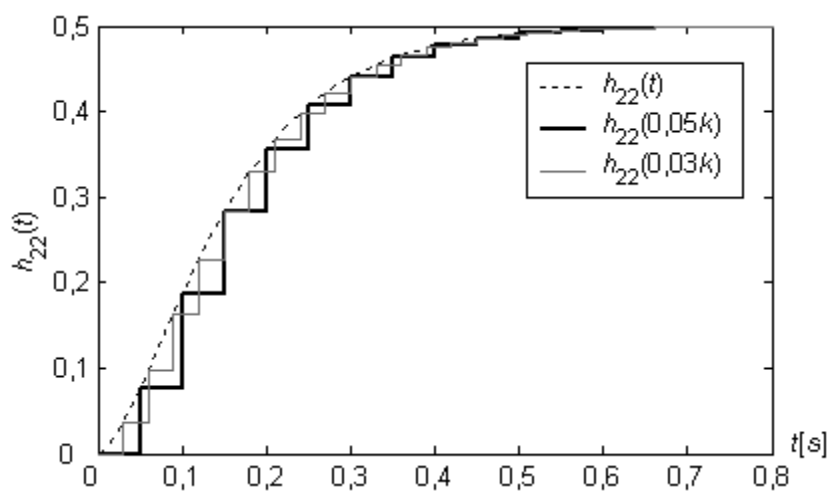
Obr. 5 Přechodové charakteristiky h_{11} (spojitá a diskrétní) TITO modelu procesu



Obr. 6 Přechodové charakteristiky h_{12} (spojitá a diskrétní) TITO modelu procesu



Obr. 7 Přejchodové charakteristiky h_{21} (spojitá a diskrétní) TITO modelu procesu



Obr. 8 Přejchodové charakteristiky h_{22} (spojitá a diskrétní) TITO modelu procesu

6.1.3 Verifikace implementovaných optimalizačních metod

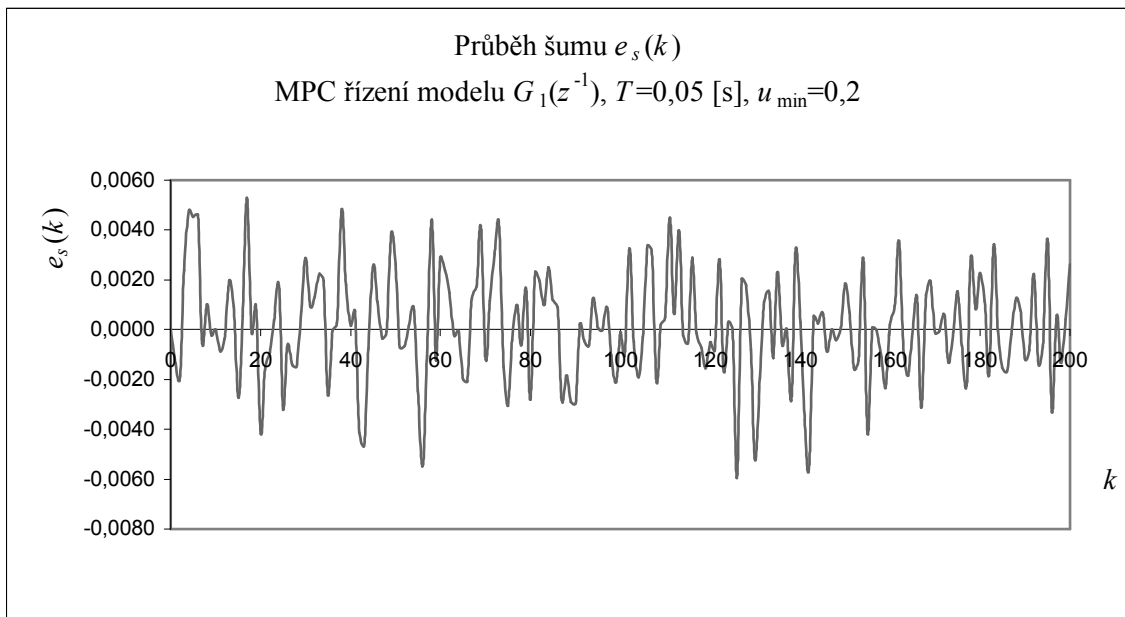
Za účelem ověření správnosti implementace Hildrethovy metody (příloha A.3) a navrhované optimalizační metody (příloha A.5) byl sestaven kontrolní programový kód (příloha A.6), jenž porovnává výsledky obou těchto metod s výsledky verifikačního algoritmu (příloha A.1, kap. 5.1.2). Je experimentálně stanoven pozitivně definitní Hessián \mathbf{H} (matice rozměru $\mathbf{H} \in \mathcal{R}^{n,n}$ s prvky mimo hlavní diagonálu o hodnotě 1 a s prvky na hlavní diagonále o hodnotách aritmetické posloupnosti $\{3i\}_{i=1}^n$) a vektor \mathbf{b} (vektoru rozměru $\mathbf{b} \in \mathcal{R}^{1,n}$ s prvky o hodnotách aritmetické posloupnosti $\{i\}_{i=1}^n$). A to pro rozsahy rozměrů proměnné \mathbf{x} : $n \in \langle 3; 18 \rangle$. Rozměr $n = 18$ je stanoven z důvodu řešení $2^{18} - 1 = 262143$ kombinací při užití verifikačního algoritmu. Matice omezení je definována jako jednotková $\mathbf{I} \in \mathcal{R}^{n,n}$ a vektor $\boldsymbol{\gamma} \in \mathcal{R}^{n,1}$ s hodnotami 0,5. Takto definovaná matice omezení \mathbf{M} splňuje nutné podmínky regularity. Pro rozměry v úloze platí rovnost $m = n$.

Tato verifikace byla provedena s kladným výsledkem pro stanovené testování s přesností 0,001. Samotná verifikace je založena na konfrontaci s odvozeným analytickým algoritmem bez nutnosti využívat interní příkaz *quadprog* prostředí MATLAB.

6.1.4 Simulace prediktivního řízení zvolených modelů procesů

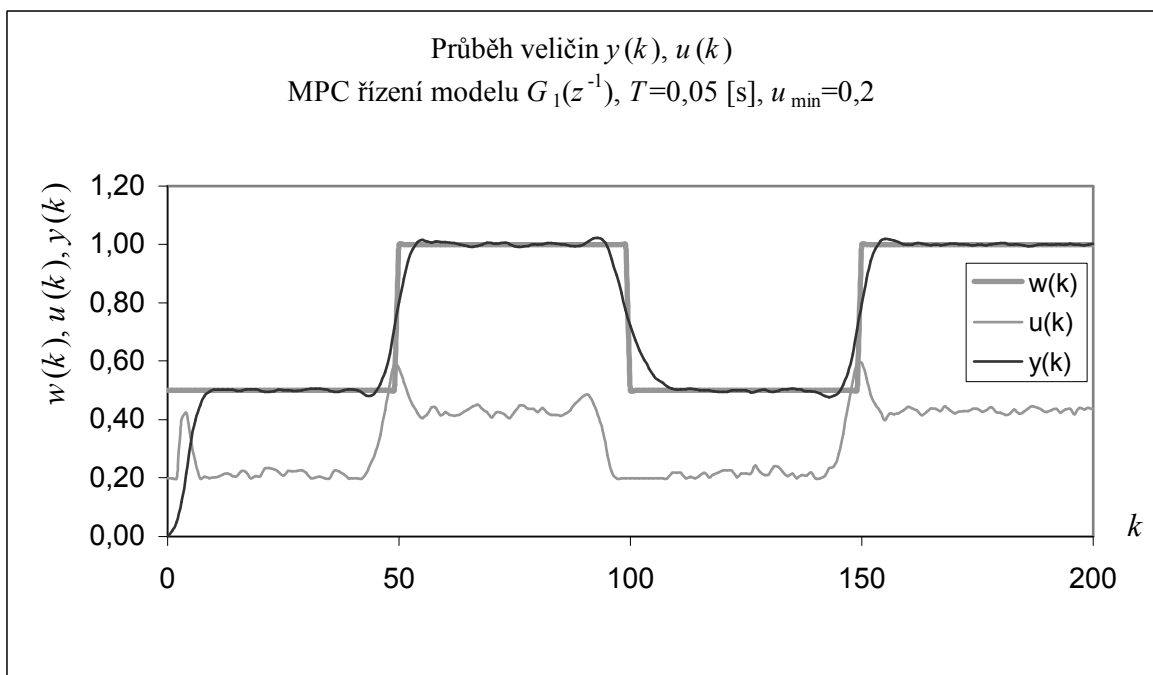
V rámci disertační práce byly v popsaném softwarovém prostředí provedeny simulace prediktivního řízení procesů, a to pro množinu nastavení (kap. 6.4). Pro uvedení příkladu simulace MPC bude zvolena ukázka osmi realizovaných příkladů řízení. Řízení se v rámci měření v kapitole 6.4 liší od těchto příkladů pouze výběrem použité optimalizační metody či hodnotami horizontů.

Jsou řízeny diskrétní modely $G_1(z^{-1})$ (6.5) a $\mathbf{G}_1(z^{-1})$ (6.7)–(6.9) pro uvažovanou periodu vzorkování $T = 0,05$ s, při $T = 0,03$ s pak modely procesů $G_2(z^{-1})$ (6.6) a $\mathbf{G}_2(z^{-1})$ (6.10)–(6.12). Šum $e_s(k)$ (SISO MPC: obr. 9; TITO MPC součástí grafů) byl vygenerován a uložen pro účely stejných podmínek pro všechna řízení dále, aby je bylo možné porovnávat. Platí $e_s(k) = e_{s_1}(k) = e_{s_2}(k)$.



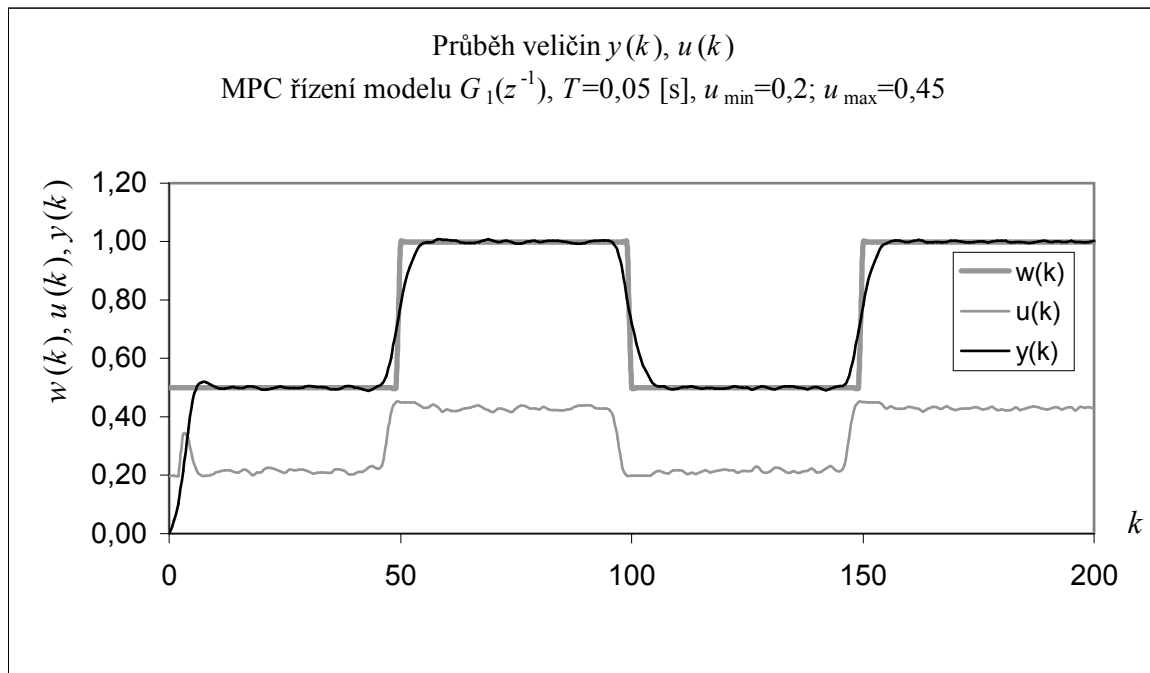
Obr. 9 Průběh vygenerované neměřitelné poruchové veličiny pro SISO řízení

Příklad 1: MPC modelu SISO: $G_1(z^{-1})$, $T = 0,05$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,2$; působení šumu $e_s(k)$, hodnoty horizontů: $N_1 = 1$; $N_u = 5$; $N_2 = 5$, Hildrethova metoda se zavedenou modifikací dle kap. 4.3.5, průběhy veličin MPC (obr. 10), datový soubor *SISO_0.05_1om_N5.csv* (viz. příloha B).



Obr. 10 Průběh veličin prediktivního řízení pro nastavení MPC dle příkladu 1

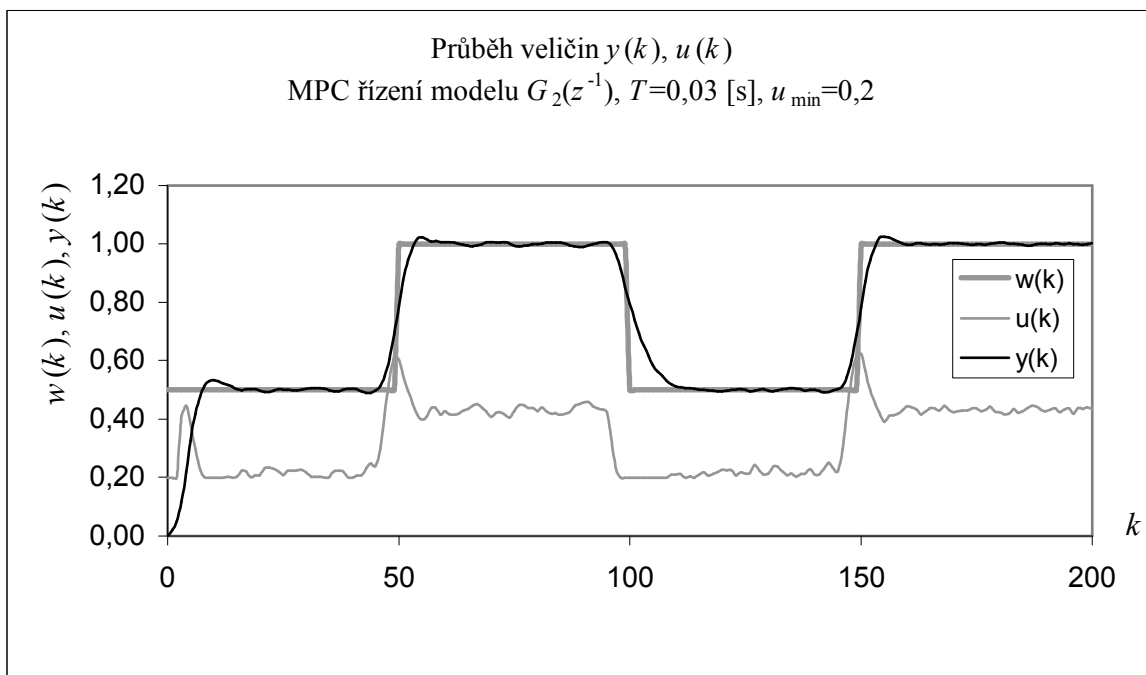
Příklad 2: Prediktivní řízení diskretního modelu SISO: $G_1(z^{-1})$, $T = 0,05$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,2$, $u_{\max} = 0,45$; působení šumu $e_s(k)$, hodnoty horizontů: $N_1 = 1$; $N_u = 5$; $N_2 = 5$, Hildrethova metoda se zavedenou modifikací dle kap. 4.3.5, průběhy veličin MPC (obr. 11), datový soubor *SISO_0.05_2om_N5.csv* (viz příloha B).



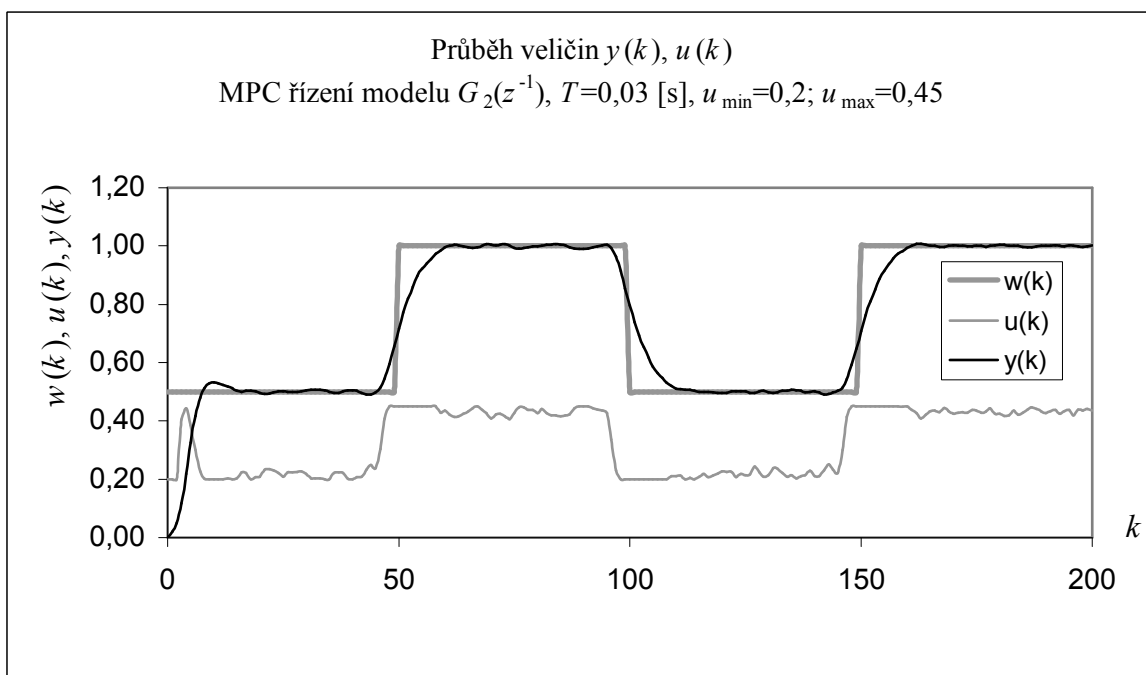
Obr. 11 Průběh veličin prediktivního řízení pro nastavení MPC dle příkladu 2

Příklad 3: Prediktivní řízení diskretního modelu SISO: $G_2(z^{-1})$, $T = 0,03$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,2$; působení šumu $e_s(k)$, hodnoty horizontů: $N_1 = 1$; $N_u = 5$; $N_2 = 5$, Hildrethova metoda se zavedenou modifikací dle kap. 4.3.5, průběhy veličin MPC (obr. 12), datový soubor *SISO_0.03_1om_N5.csv* (viz. příloha B).

Příklad 4: Prediktivní řízení diskretního modelu SISO: $G_2(z^{-1})$, $T = 0,03$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,2$, $u_{\max} = 0,45$; působení šumu $e_s(k)$, hodnoty horizontů: $N_1 = 1$; $N_u = 5$; $N_2 = 5$, Hildrethova metoda se zavedenou modifikací dle kap. 4.3.5, průběhy veličin MPC (obr. 13), datový soubor *SISO_0.03_2om_N5.csv* (viz. příloha B).

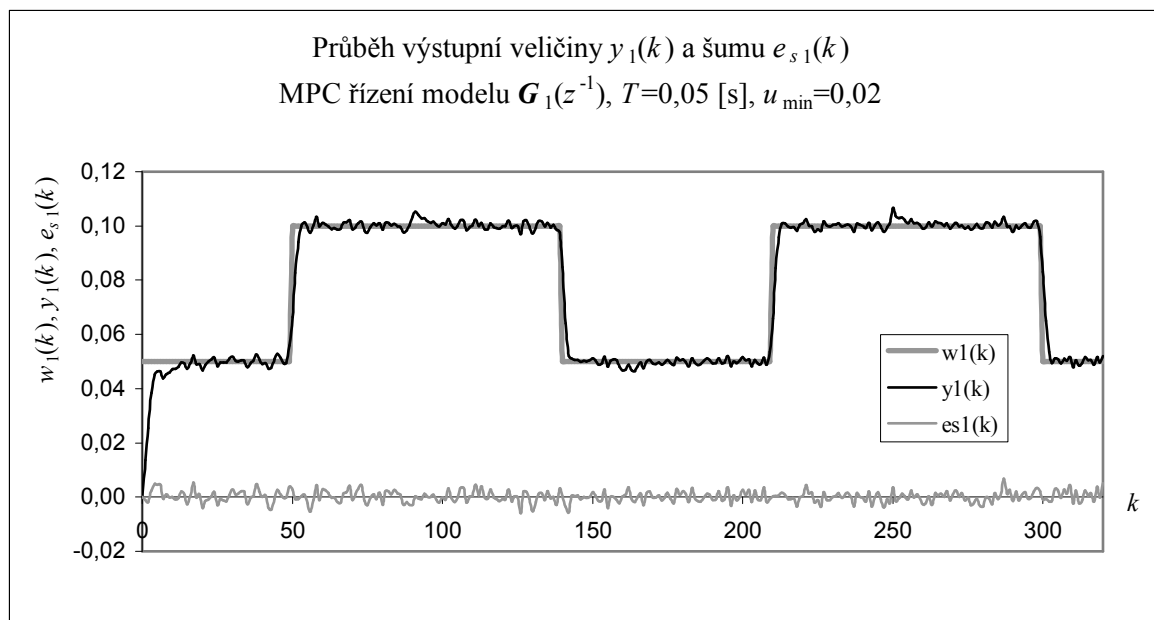


Obr. 12 Průběh veličin prediktivního řízení pro nastavení MPC dle příkladu 3

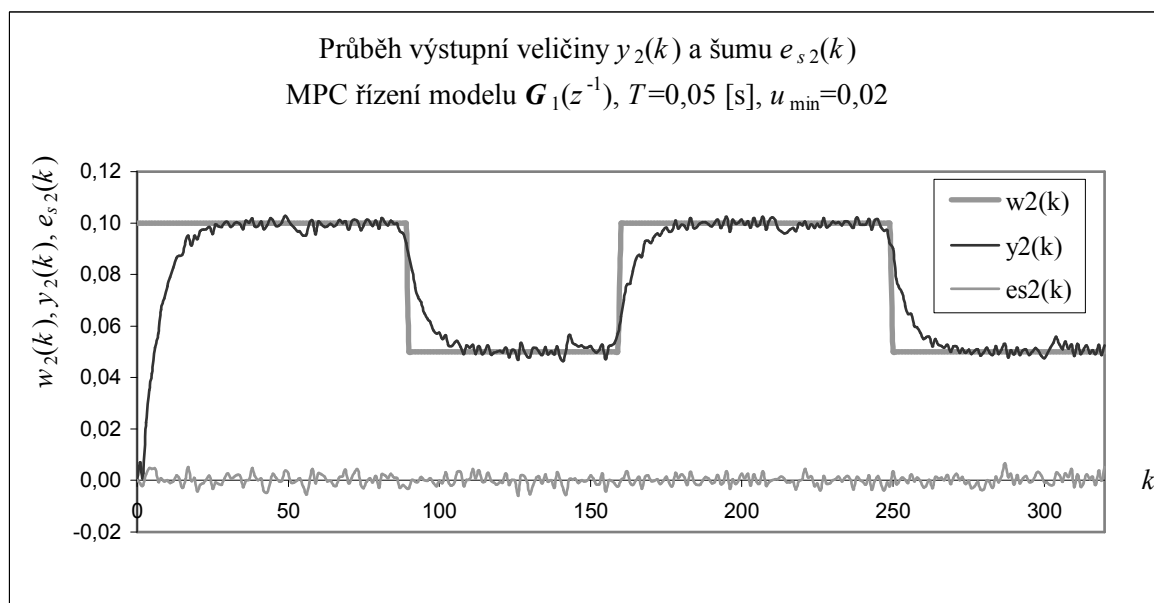


Obr. 13 Průběh veličin prediktivního řízení pro nastavení MPC dle příkladu 4

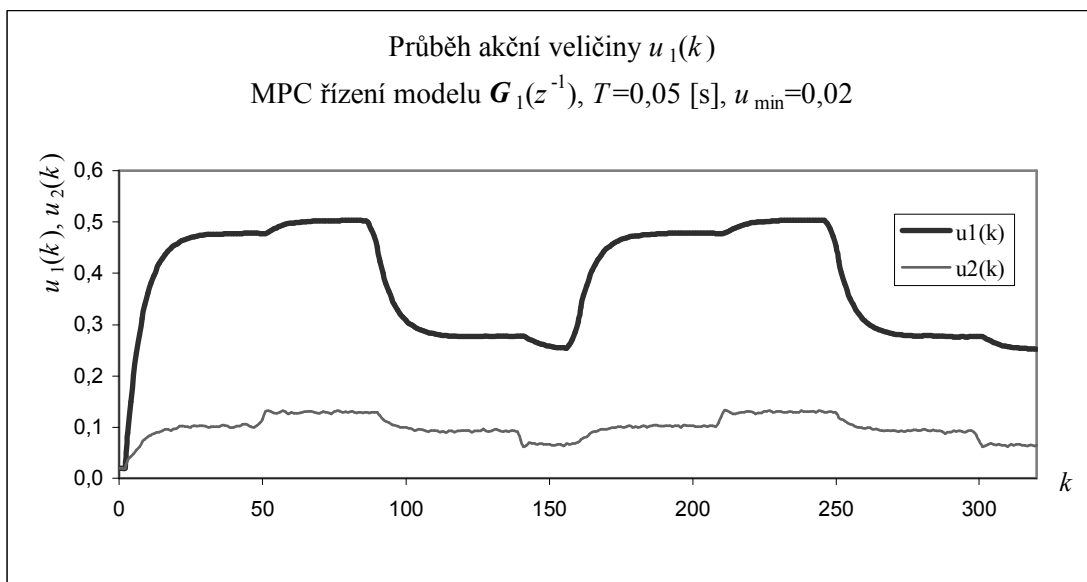
Příklad 5: MPC modelu TITO: $G_1(z^{-1})$, $T = 0,05$ s, meze žádaných veličin $\{0,05;0,1\}$, omezení $u_{\min} = 0,02$; působení šumů $e_s(k)$, hodnoty horizontů: $N_1 = 1; N_u = 5; N_2 = 5$, Hildrethova metoda se zavedenou modifikací dle kap. 4.3.5, průběhy veličin (obr. 14 – 16), datový soubor *TITO_0.05_1om_N5.csv* (viz. příloha B).



Obr. 14 Průběh veličin y_1 a šumu pro nastavení MPC dle příkladu 5

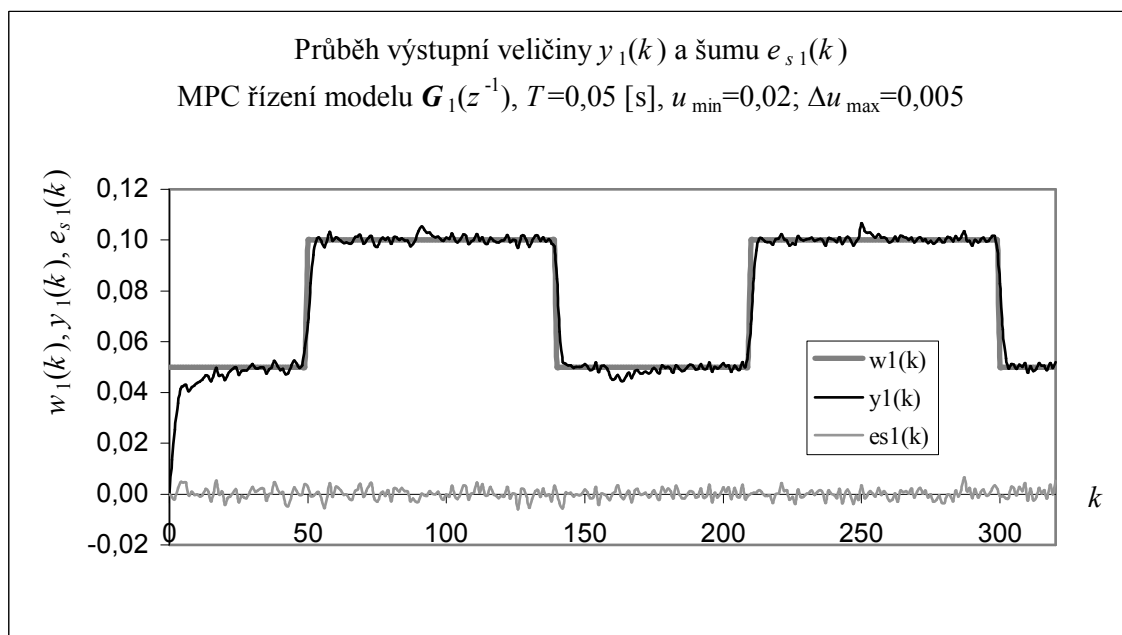


Obr. 15 Průběh veličin y_2 a šumu pro nastavení MPC dle příkladu 5

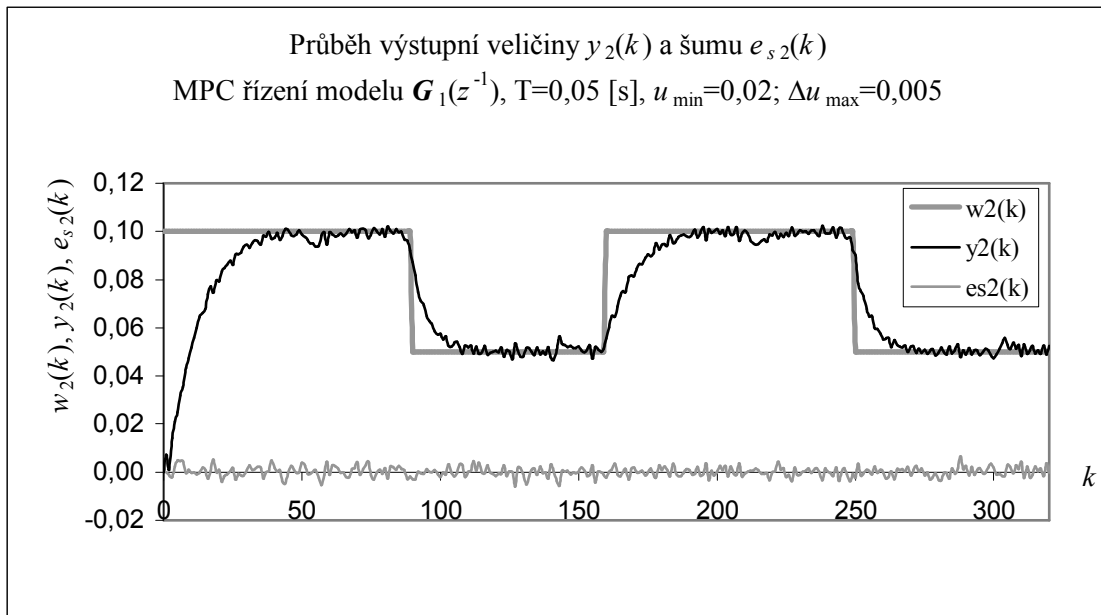


Obr. 16 Průběh řídicích veličin pro nastavení MPC dle příkladu 5

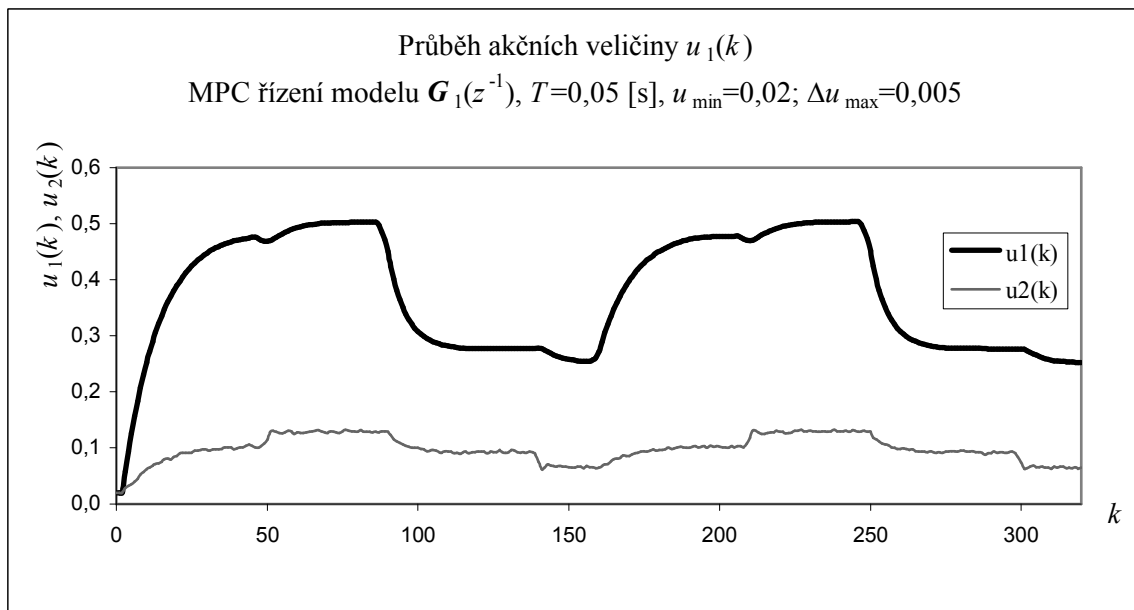
Příklad 6: MPC modelu TITO: $G_1(z^{-1})$, $T = 0,05$ s, meze žádaných veličin $\{0,05;0,1\}$, omezení $u_{\min} = 0,02$; $\Delta u_{\max} = 0,005$; působení šumů $e_s(k)$, hodnoty horizontů: $N_1 = 1$; $N_u = 5$; $N_2 = 5$, Hildrethova metoda se zavedenou modifikací dle kap. 4.3.5, průběhy veličin prediktivního řízení (obr. 17 – 19), datový soubor *TITO_0.05_2om_N5.csv* (viz. příloha B).



Obr. 17 Průběh veličin y_1 a šumu pro nastavení MPC dle příkladu 6

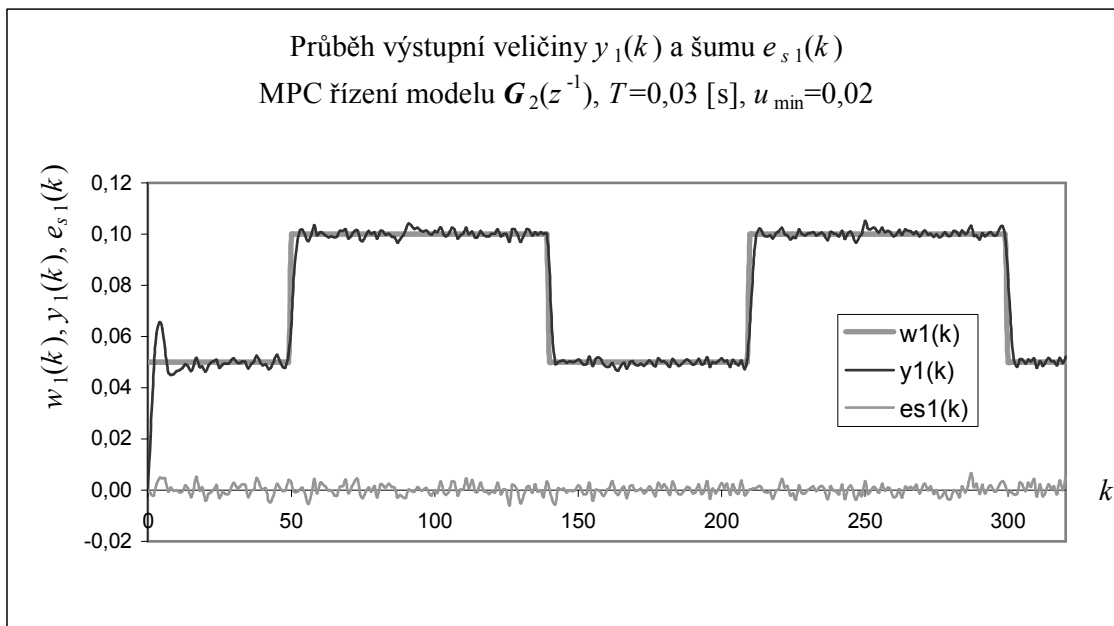


Obr. 18 Průběh veličin y_2 a šumu pro nastavení MPC dle příkladu 6

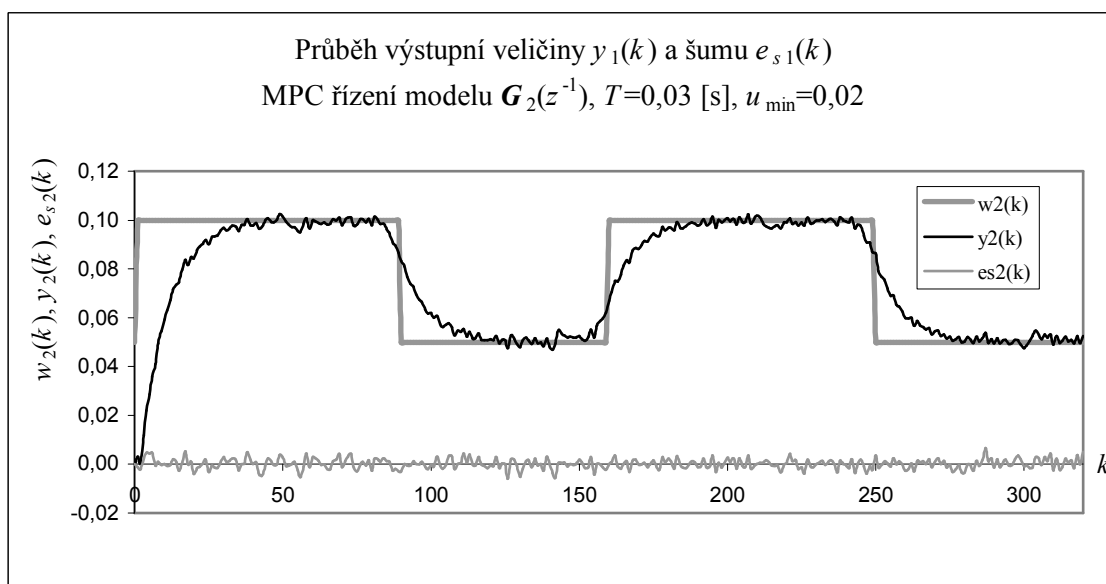


Obr. 19 Průběh řídicích veličin pro nastavení MPC dle příkladu 6

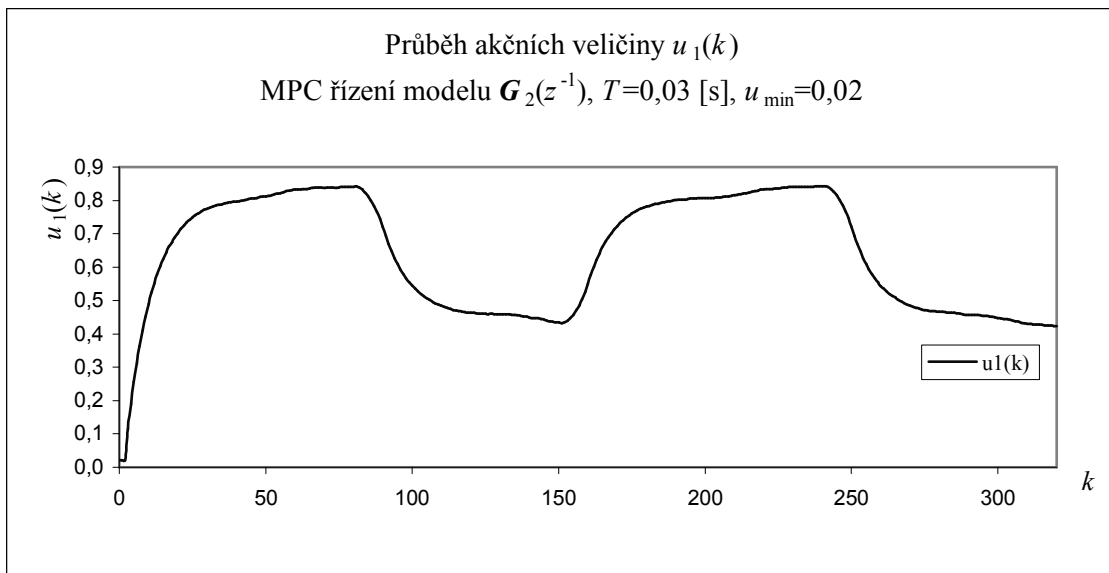
Příklad 7: MPC modelu TITO: $G_2(z^{-1})$, $T = 0,03$ s, meze žádaných veličin $\{0,05;0,1\}$, omezení $u_{\min} = 0,02$; působení šumů $e_s(k)$, hodnoty horizontů: $N_1 = 1$; $N_u = 10$; $N_2 = 10$, Hildrethova metoda se zavedenou modifikací dle kap. 4.3.5, průběhy veličin prediktivního řízení (obr. 20 – 23), datový soubor *TITO_0.03_1om_N10.csv* (viz. příloha B).



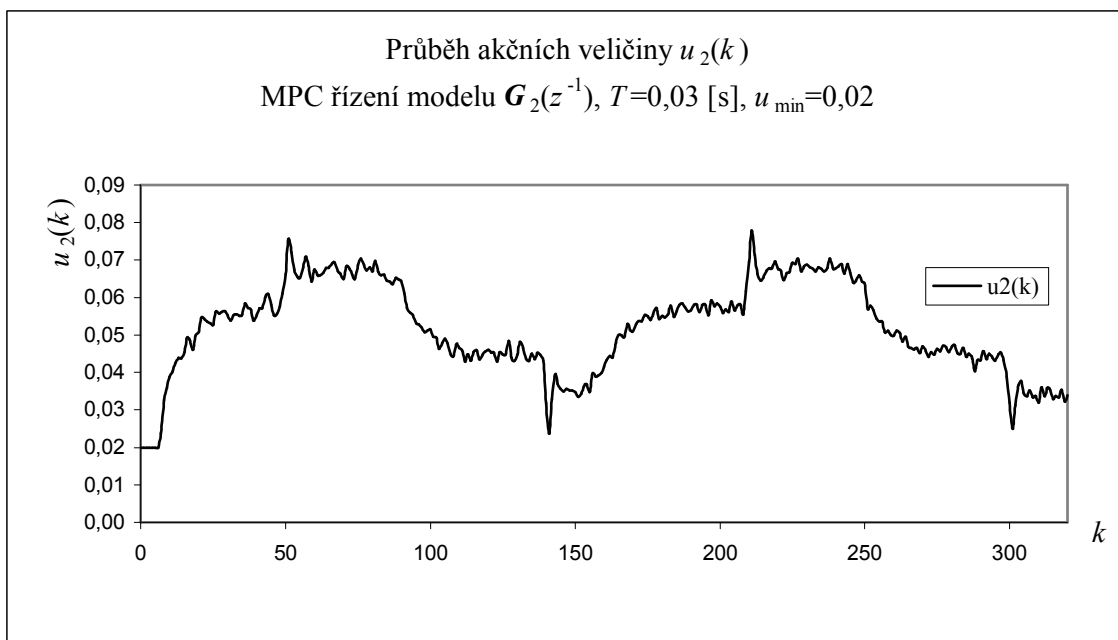
Obr. 20 Průběh veličin y_1 a šumu pro nastavení MPC dle příkladu 7



Obr. 21 Průběh veličin y_2 a šumu pro nastavení MPC dle příkladu 7

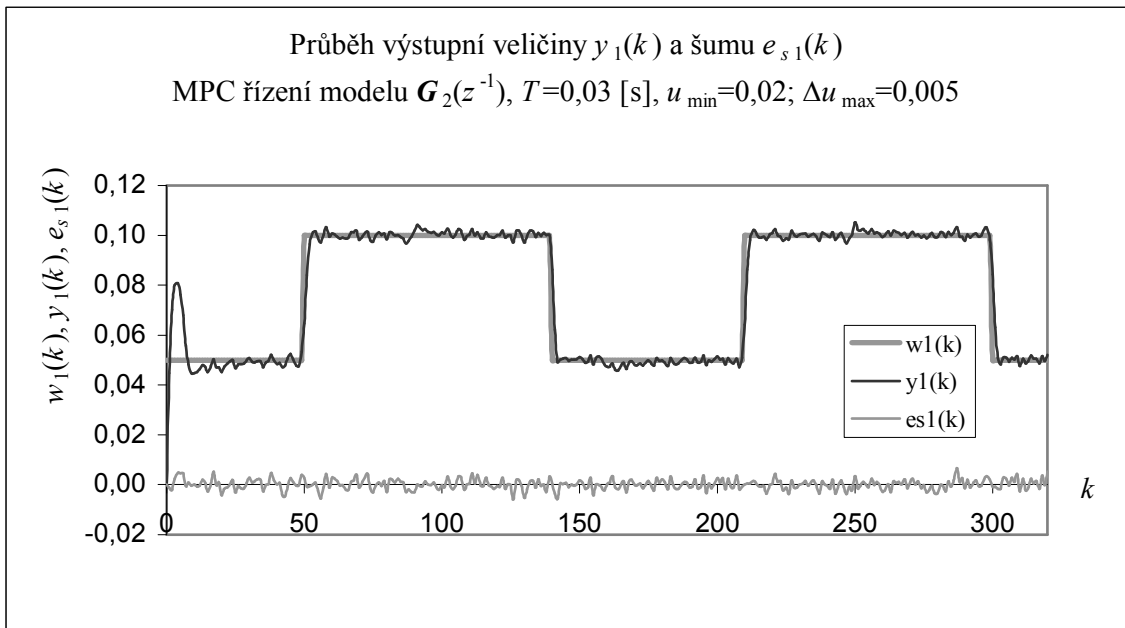


Obr. 22 Průběh řídicí veličiny u_1 pro nastavení MPC dle příkladu 7

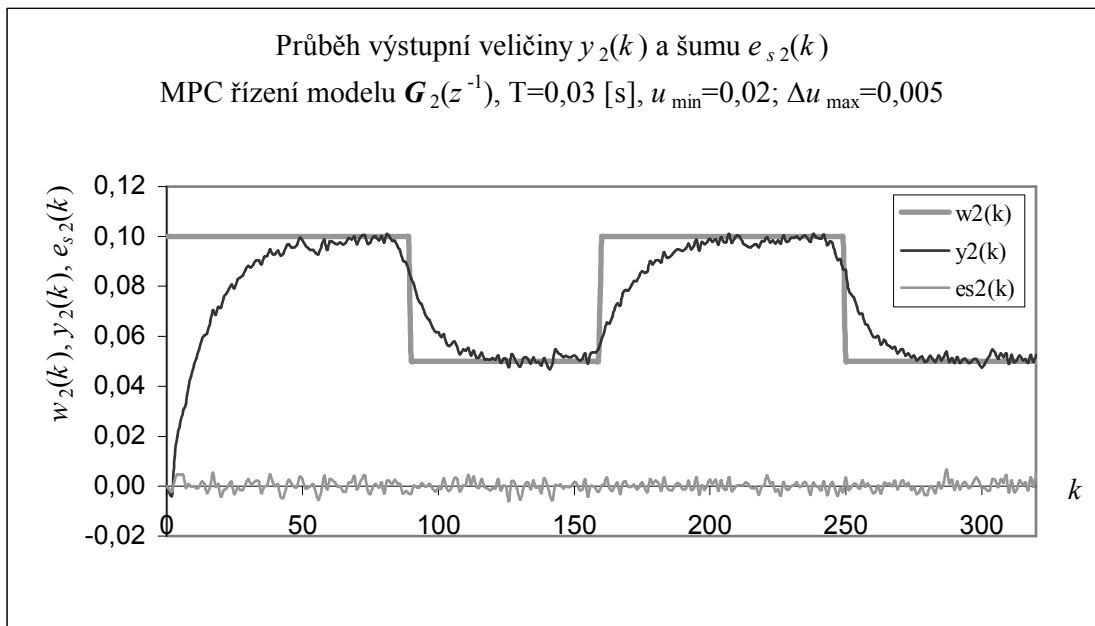


Obr. 23 Průběh řídicí veličiny u_2 pro nastavení MPC dle příkladu 7

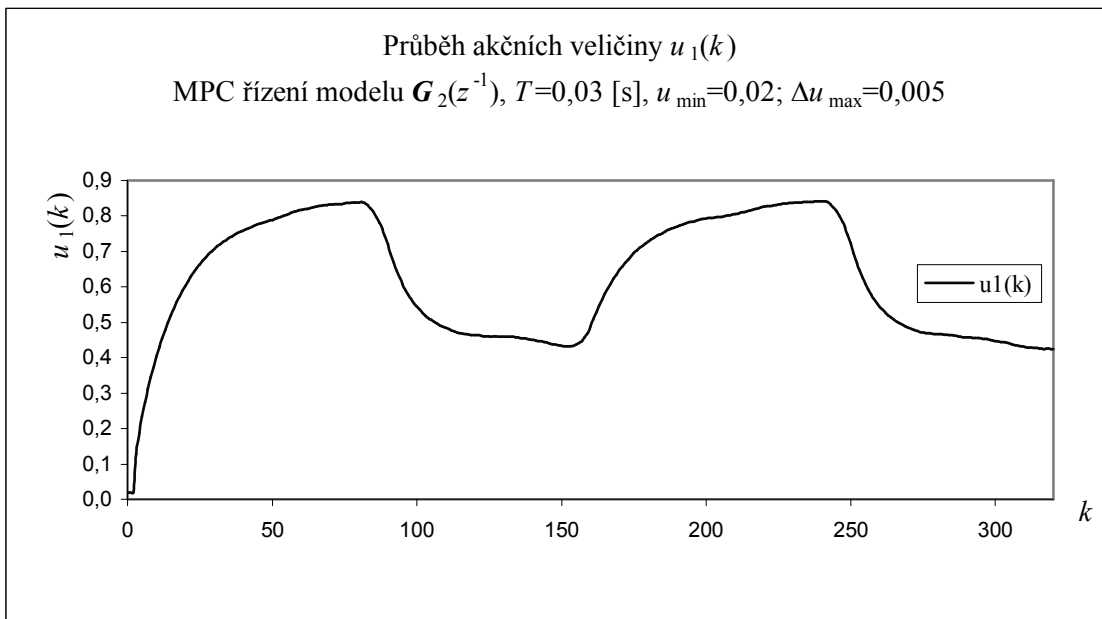
Příklad 8: MPC modelu TITO: $G_2(z^{-1})$, $T = 0,03$ s, meze žádaných veličin $\{0,05;0,1\}$, omezení $u_{\min} = 0,02$; $\Delta u_{\max} = 0,005$; působení šumů $e_s(k)$, hodnoty horizontů: $N_1 = 1; N_u = 10; N_2 = 10$, Hildrethova metoda se zavedenou modifikací dle kap. 4.3.5, průběhy veličin prediktivního řízení (obr. 24 – 27), datový soubor *TITO_0.03_2om_N10.csv* (viz. příloha B).



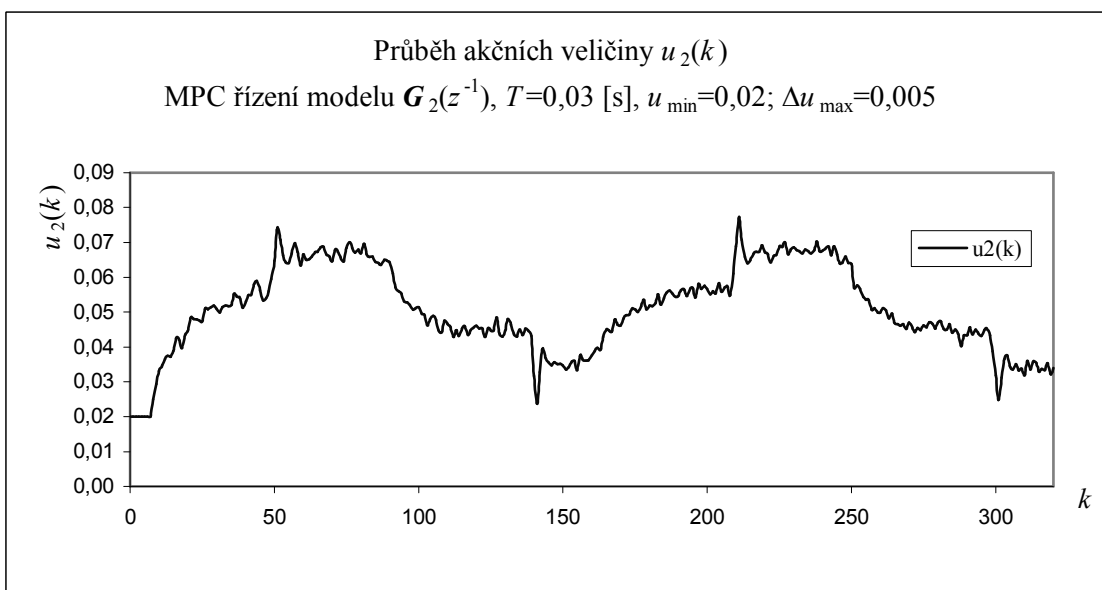
Obr. 24 Průběh veličin y_1 a šumu pro nastavení MPC dle příkladu 8



Obr. 25 Průběh veličin y_2 a šumu pro nastavení MPC dle příkladu 8



Obr. 26 Průběh řídicí veličiny u_1 pro nastavení MPC dle příkladu 8



Obr. 27 Průběh řídicí veličiny u_2 pro nastavení MPC dle příkladu 8

6.2 Aparát pro stanovení výpočetní náročnosti algoritmu a pro určení kvality řízení

6.2.1 Počet aritmeticko logických operací algoritmu

Jedním ze způsobů posouzení efektivity algoritmu je určení počtu elementárních operací programového kódu v jednotkách flopy (flop – floating-point operation). Jedná se o určení výpočetní náročnosti bez ohledu na konkrétní parametry výpočetního stroje (počítače, mikropočítače). [30]

Jeden flop vyjadřuje obecně elementární operaci: součet, násobení či dělení reálného čísla. Přehled počtu nejčastěji použitých operací je uveden v tabulce 6. Nechť jsou matice $\mathbf{W}_1 \in \mathcal{R}^{N_w, M_w}$, $\mathbf{W}_2 \in \mathcal{R}^{M_w, P_w}$ obdélníkové a matice $\mathbf{W}_3 \in \mathcal{R}^{N_w, N_w}$ je čtvercová pozitivně definitní. Kde N_w , M_w a P_w jsou rozměry daných matic. Tyto rozměry vždy ovlivňují počet výsledných operací při práci s maticemi. [30]

Tab. 6 Počty operací frekventovaných matematických vztahů

Matematický vztah	Počet operací [flopy]
$\mathbf{W}_1 + \mathbf{W}_1$	$N_w M_w$
$\varepsilon \mathbf{W}_1; \varepsilon \in \mathcal{R}$	$N_w M_w$
$\mathbf{W}_1 \mathbf{W}_2$	$2N_w M_w P_w - N_w P_w$
\mathbf{W}_3^{-1}	$N_w^3 + N_w^2 + N_w$

Jelikož již nelze využívat v prostředí MATLAB od verze 4.0 funkci pro měření počtu operací, je obtížnější určit složitost vestavěných příkazů MATLABu. Proto bude použito jejich experimentálního odhadu s ohledem na rozměry použitých matic v daných operacích. Pokud se hodnocený příkaz nebude týkat práce s maticemi, nebude při jejich větších rozměrech mít významný vliv na celkovou výpočetní náročnost kódu a lze jej v takovém případě považovat za elementární o hodnotě 1 flopu (např. příkaz přiřazení 1 hodnoty do proměnné). Co se týká příkazů pro řízení toku programu, např. *for*, *while*, pak jejich přítomnost v algoritmu má vliv na násobení počtu operací

v daném cyklu počtem jeho průchodů. Ukázka určení počtu flopů v konkrétním programovém kódu (úryvek přílohy A.5, navrhované optimalizační metody).

```

% Nastavení max.počtu iterací

max_pocet_iteraci=100; flopy=flopy+1;

% Určení matice N a vektoru o duální úlohy

N=M*inv(H)*M';

flopy=flopy+2*m*n*n-m*n+2*m*m*n-m*m+0.3*n*n*n+0.6*n;

o=M*inv(H)*b'+gamma;

flopy=flopy+2*m*n*n-m*n+2*m*n-m+0.3*n*n*n+0.6*n+m;

```

Necht' je definována veličina N jako *rozměr problému* výpočetní náročnosti posuzovaného algoritmu. Pak pro diskrétní funkci vyjadřující závislost mezi počtem flopů a rozměrem problému lze provést regresní analýzu. S tím, že vhodnou regresní křivku (lineární, polynomiální, exponenciální) lze stanovit např. přímo v prostředí MS Excel (s ohledem na koeficient spolehlivosti značený R^2 s cílem dosažení hodnoty 1). Při možnosti zobrazení regresní rovnice aproximující tuto funkční závislost lze následně tuto funkci klasifikovat typem výpočetní náročnosti dle knihy [31], a to pomocí tzv. funkce $O=O(N)$. Je definována řada typů funkcí výpočetní náročnosti. Nejčastěji vyskytující je funkce polynomiální výpočetní náročnosti. Ta může být lineární (např. algoritmus průchodu jednorozměrného pole o N prvcích) nebo složitější polynomiální, kde se bere vždy člen s nejvyšším stupněm polynomu násobený multiplikativní konstantou. Další členy se zanedbávají. Pro dvě takto stanovené funkce pak lze stanovit porovnáním jejich funkčního předpisu, která z nich roste řádově rychleji, resp. pomaleji. Což může být užitečné pro znázornění výpočetní náročnosti ve flopech z pohledu změny uvažovaného rozměru problému. [31]

6.2.2 Měření doby provedení operací v prostředí MATLAB

Jedním z cílů disertační práce je dokázat, že navrhovanými přístupy k řešení MPC bude realizováno prediktivní řízení procesů s rychlou dynamikou v rámci deklarované periody vzorkování. Zavedené přístupy mohou mít problém se zvládnutím algoritmu řízení v rámci tohoto času. Proto je podstatné měřit přímo časy provedení operací MPC v rámci každé periody vzorkování, a to na dostupném hardwarovém zařízení daných vlastností. Počet operací provedení MPC ve flopech není pro tento cíl práce natolik vypovídající, jako měření časů výpočetní náročnosti operací souvisejících s MPC. Přitom se při měření časů předpokládá, že na odlišném zařízení budou naměřené hodnoty jiné pro dané hardwarové vybavení. Proto celková kvalita předkládaných řešení bude

klasifikována počtem provedených operací ve flopech pro subsystém optimalizace, aby bylo zjevné, kde urychlení výpočtů nastalo.

Měření časů operací bude provedeno dvakrát, a to pro operace celého algoritmu MPC v rámci každé periody vzorkování diskretního prediktivního řízení (značeno $T_{mpc}(k)$) a pro provedení optimalizační úlohy (značeno $T_{opt}(k)$). Je předpokládáno, že operace mimo blok optimalizace, budou časově zanedbatelné, neboť se zde nachází nepatrné množství operací (přepočít vektoru \mathbf{b} , výpočet diferenční rovnice, uložení hodnot veličin do paměti, aktualizace některých hodnot v definici omezení, a pod.).

Jelikož v rámci aplikovaných modifikací (zavedených či navrhovaných) může nastat v určitých okamžicích diskretního MPC větší redukce časů prováděných operací oproti jiným okamžikům, je brána v potaz pouze vyšší hodnota měřených časů. Proto budou v kapitole 6.4 pro dané podmínky měření (nastavení regulátoru MPC, definice omezení, a pod.) uvažovány maximální hodnoty časů T_{mpc} (6.13) a T_{opt} (6.14), s jejich měřením v k -té pořadnici diskretního řízení následnými příkazy v prostředí MATLAB:

```
doba=clock;
```

```
% měřený blok kódu
```

```
cas=etime(clock,doba);
```

$$T_{mpc} = \max \{T_{mpc}(k)\} \quad (6.13)$$

$$T_{opt} = \max \{T_{opt}(k)\} \quad (6.14)$$

6.2.3 Definice kritérií kvality řízení

Nechť jsou definována kritéria určení kvality řízení MPC vztahy (6.15)-(6.18). Jejich definice souvisí s podstatou MPC řízení, jež je i analogicky promítnuta do podoby účelové funkce. Kritéria jsou však rozdělena do dvou kategorií, a to na zhodnocení součtu regulačních odchylek a přírůstků akční veličiny (jako je např. v publikaci [32]). Pro jednorozměrové řízení jsou stanovena kritéria kvality (6.15) a (6.16). Pro dvourozměrovou variantu jsou analogicky definována kritéria (6.17) a (6.18). Kde k je pořadnice diskretního MPC nabývající hodnot od 0 do počtu celkových pořadnic realizovaného prediktivního řízení. Pozn.: U symbolu suma není tato horní mez uváděna, ale je explicitně uvažována.

$$J_1 = \sum_k [\Delta u(k)]^2 \quad (6.15)$$

$$J_2 = \sum_k [w(k) - y(k)]^2 \quad (6.16)$$

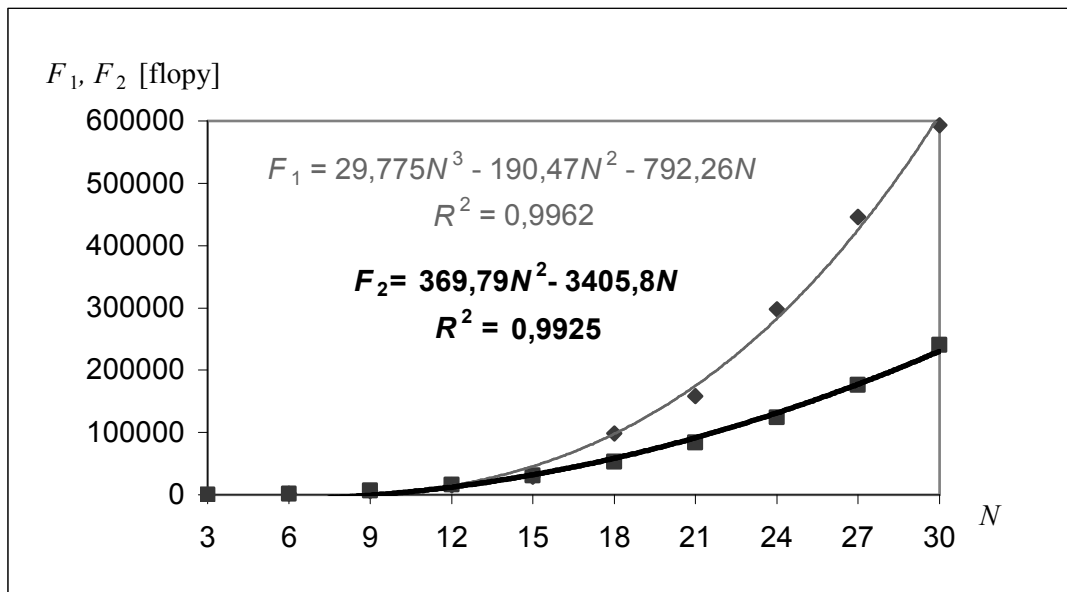
$$J_3 = \sum_k [\Delta u_1(k)]^2 + \sum_k [\Delta u_2(k)]^2 \quad (6.17)$$

$$J_4 = \sum_k [w_1(k) - y_1(k)]^2 + \sum_k [w_2(k) - y_2(k)]^2 \quad (6.18)$$

6.3 Posouzení navrhované optimalizační metody

V této kapitole je porovnána navrhovaná optimalizační metoda řešení úlohy kvadratického programování (příloha A.5) s implementovanou duální Hildrethovou metodou (příloha A.3). Předpokládá se, že navrhovaná metoda bude výpočetně méně náročná ve srovnání s Hildrethovou metodou. Následující experiment (příloha A.7) se snaží tento předpoklad dokázat, a to stanovením počtu flopů operací obou těchto metod pro proměnný rozměr problému N . Tento rozměr problému je roven parametru n experimentálního příkladu dále.

Je experimentálně stanoven pozitivně definitní Hessián \mathbf{H} (matice rozměru $\mathbf{H} \in \mathcal{R}^{n,n}$ s prvky mimo hlavní diagonálu o hodnotě 1 a s prvky na hlavní diagonále o hodnotách aritmetické posloupnosti $\{i\}_{i=1}^n$) a vektor \mathbf{b} (vektoru rozměru $\mathbf{b} \in \mathcal{R}^{1,n}$ s prvky o hodnotách aritmetické posloupnosti $\{i\}_{i=1}^n$). A to pro rozsahy rozměrů proměnné \mathbf{x} : $n = \{3, 6, 9, \dots, 30\}$. Matice omezení je definována jako jednotková $\mathbf{I} \in \mathcal{R}^{n,n}$ a vektor $\boldsymbol{\gamma} \in \mathcal{R}^{n,1}$ s hodnotami 1. Takto definovaná matice omezení \mathbf{M} splňuje nutné podmínky regularity. Pro rozměry v úloze platí rovnost $m = n$. Nechť je značen počet operací Hildrethovy metody F_1 a navrhované metody F_2 , a to se jednotkami ve flopech. Výsledky měření jsou zobrazeny do obrázku 28, a to s proložením vhodným typem regresních křivek.



Obr. 28 Porovnání výpočetní náročnosti Hildrethovy a navrhané opt. metody

Z takto definovaného porovnání výpočetní náročnosti obou optimalizačních metod plyne, že navrhaná optimalizační metoda je za daných podmínek řádově výpočetně rychlejší, neboť pro funkce O obou metod platí: $(O_1 > O_2)$. Kde $O_1 = 29,775N^3$ a $O_2 = 369,79N^2$. Tato navrhaná metoda kvadratického programování bude implementována dle pravidel kapitoly 5.4.2 do řešení, jež reprezentuje celkový navržený přístup k řešení MPC spolu s další aplikací eliminací (kap. 5.4.1). Což bude zhodnoceno vůči celkovému zavedenému přístupu (kap. 4.3.5) v následující kapitole.

6.4 Analýza výpočetní náročnosti navrhovaných řešení v MPC

6.4.1 Stanovení výchozích podmínek pro měření

V rámci disertační práce jsou použity dva způsoby určení výpočetní náročnosti implementovaných navrhovaných řešení ve srovnání s algoritmy zavedených přístupů, a to měření času provedení operací a určení jejich počtu. Dále je zmapován vliv užití přístupů k řešení MPC (navrhované, zavedené) na kvalitu řízení MPC.

Nechť je dle kapitoly 6.2.1 definován maximální počet operací v rámci realizace výpočtů části optimalizace napříč realizací MPC řízení jakožto F_{opt} v jednotkách flopy.

Nechť je dle kapitoly 6.2.2 uvažována maximální doba provedení operací souvisejících s optimalizací T_{opt} (6.14) MPC procesu s rychlou dynamikou. Maximální doba realizace algoritmu MPC je reprezentována časovým údajem T_{mpc} (6.13). Obě veličiny jsou určovány v sekundách.

Nechť jsou dle kapitoly 6.2.3 uvažována kritéria určení kvality řízení MPC s daným nastavením jakožto: J_1 (dle 6.15), J_2 (6.16) pro SISO variantu řízení a J_3 (6.17), J_4 (6.18) pro TITO variantu řízení.

Z důvodu vyhodnocení překročení stanovené periody vzorkování T jsou určovány právě maximální hodnoty měřených časů. Maximální počet F_{opt} je spjat s nejnáročnějším výpočtem optimalizace v rámci realizace daného MPC.

Protože je důležité zhodnotit, zda-li se algoritmus diskrétně realizovaného MPC zvládne časově provést v rámci stanovené periody vzorkování, je nutné tuto skutečnost konfrontovat s měřením času. I když by tyto výsledky byly různé pro jiné hardwarové vybavení, budou uvedeny pro následující definici výchozích podmínek pro měření. Cílem je dokázat výskyt jevu, kdy v určitých konkrétních nastaveních regulátoru MPC zavedené metody selžou a navrhované metody dokáží realizovat algoritmus řízení v dané periodě vzorkování. Navíc bude provedeno diskrétní řízení téhož zvoleného modelu procesu, a to při účelném snížení periody vzorkování. Důkaz bude nosným tvrzením, že lze zvládat prediktivně řídit procesy s rychlou dynamikou i v náročnějších podmínkách řízení (volba T , horizonty, mnohorozměrovost řízení, počty aplikovaných podmínek do řízení), a to pomocí navrhovaných způsobů řešení oproti zavedeným přístupům MPC.

Počty provedených operací v jednotkách flop budou uvedeny pro operace související pouze s optimalizační úlohou. Jelikož jsou výpočty v rámci prediktoru (zejména vyčíslení diferenčních rovnic, uložení hodnot veličin do paměti, aktualizace hodnot některých definic omezení, a pod.), pro zavedené i navrhované přístupy shodné, není proto počet operací prediktoru a celého algoritmu MPC natolik důležitou informací k další analýze výsledků. Avšak právě počty operací subsystému optimalizace jsou nosnou informací vypovídající, kterými výpočetními postupy došlo k urychlení výpočtů.

Dále bude z analýzy výpočetní náročnosti řešení MPC patrný vliv parametru m , resp. m_r v souvislosti s počtem operací F_{opt} a hodnotami času T_{opt} .

Nechť je rozměr problému N pro měření výpočetní náročnosti ve flopech definován jako (6.19).

$$N = N_u = N_2 \quad (6.19)$$

Ideálním případem podmínek pro měření času operací v MATLABu by bylo běh tohoto prostředí pod real-time operačním systémem (OS), nejlépe Linuxem. Jelikož jsou dostupné pro řešení disertace licence pro OS MS Windows, je proto spuštěn pouze program MATLAB a v přehledu procesů (skrze dialog Správce úloh syst. MS Windows) je procesu *MATLAB.exe* nastavena nejvyšší priorita.

Pro výchozí podmínky pro měření je uvažována konfigurace zařízení: OS MS Windows XP, MATLAB 6.5, bez antivirového programu, nastavení priority procesu MATLAB.exe na nejvyšší, procesor Intel Celeron 2,66 GHz.

Celkové měření časů T_{opt} a T_{mpc} je pro sledované nastavení podmínek MPC řízení opakováno desetkrát a v tabulkách uvedeno aritmetickým průměrem.

Pro řízené diskrétní modely a pro T , platí definice dle kapitoly 6.1.2. Kde je definován i šum $e_s(k)$, resp. $e_s(k) = e_{s_1}(k) = e_{s_2}(k)$. Jelikož je dle kapitoly 5.4.2 vhodné aplikovat navrhovanou optimalizační metodu (kap. 5.3) pouze v případě jednoho typu omezení v SISO variantě MPC, bude proto v dalších případech nahrazena Hildrethovou metodou doplněnou Wangovou modifikací (kap. 4.3.5), přičemž nosnou navrhovanou metodou řešení MPC zůstane eliminační přístup (kap. 5.4.1). Metoda (kap. 5.3), jež je řádově výpočetně rychlejší oproti Hildrethově (kap. 6.3), může být navíc vylepšena Wangovou modifikací při existenci jevu převodu optimalizační úlohy na typ volného extrému.

6.4.2 Jednorozměrové prediktivní řízení

Měření 1: MPC modelu SISO: $G_1(z^{-1})$ (6.5), $T = 0,05$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,2$; působení šumu $e_s(k)$, hodnoty horizontů a rozměr problému: $N_1 = 1$; $N_u = N_2 = N = 5, 10, \dots, 35$. Měření veličin T_{opt} , T_{mpc} , F_{opt} , J_1 , J_2 dle definic (kap. 6.2, 6.4.1). Pozn.: Šedé řádky v tabulkách vyjadřují nežádoucí překročení periody vzorkování T časem T_{mpc} .

Zavedený přístup je definován jako MPC s využitím Hildrethovy optimalizační metody (příloha A.3) a Wangovou modifikací (příloha A.4) (kap. 4.3.5). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 7. Data měření *SISO_0.05_1om_HW.csv* jsou uložena na CD, viz. příloha B.

Tab. 7 Analýza zavedeného přístupu k MPC za podmínek měření 1

Hildrethova metoda + Wangova modifikace							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_1	J_2
5	5	5	0,016	0,016	3162	0,1575	0,7861
10	10	10	0,016	0,016	27321	0,1559	0,7816
15	15	15	0,016	0,016	100907	0,1559	0,7820
20	20	20	0,016	0,016	255577	0,1559	0,7820
25	25	25	0,031	0,032	539312	0,1559	0,7820
30	30	30	0,078	0,078	2563355	0,1559	0,7820
35	35	35	0,109	0,109	4137202	0,1559	0,7820

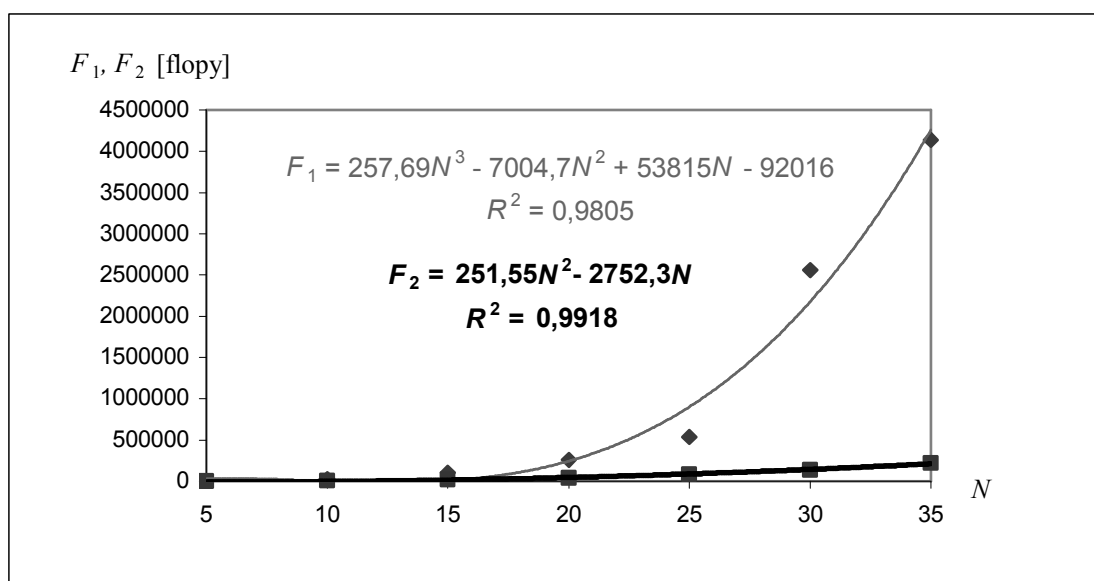
Navrhovaný přístup je definován jakožto MPC s využitím navrhované optimalizační metody (příloha A.5), zařazením eliminací (kap. 5.4.1). Navíc je přístup zkombinován s prospěšnou Wangovou modifikací (příloha A.4). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 8. Data měření *SISO_0.05_1om_NWE.csv* jsou uložena na CD, viz. příloha B.

Tab. 8 Analýza navrhovaného přístupu k MPC definovaného v měření 1

Navrhovaná opt. metoda + Wangova modifikace + návrh eliminací							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_1	J_2
5	5	3	0,016	0,016	822	0,1591	0,7855
10	10	5	0,016	0,016	5349	0,1580	0,7801
15	15	8	0,031	0,031	18318	0,1578	0,7801
20	20	10	0,032	0,032	41099	0,1578	0,7801
25	25	13	0,032	0,032	81655	0,1578	0,7801
30	30	15	0,032	0,032	136880	0,1578	0,7801
35	35	18	0,032	0,032	220485	0,1578	0,7801

Počty operací F_{opt} s přeznačením veličinou F_1 u zavedeného přístupu a veličinou F_2 u navrhovaného přístupu jsou porovnány na obrázku 29.

Z tabulek a grafu měření 1 plyne, že se podařilo navrhovaným řešením (s $m_r < m$) snížit počet operací uvažovaného algoritmu MPC, přičemž se zhoršením kvality řízení maximálně v řádu tisíců u J_1 a J_2 . I v případě $N \geq 30$, kdy nelze algoritmus provést zavedeným přístupem v rámci $T = 0,05$ s. Vliv zařazení navrhovaných přístupů je patrný z funkcí O příslušných regresních křivek aproximující průběhy funkcí F_1 a F_2 ($O_1 > O_2$). Kde $O_1 = 257,69N^3$ a $O_2 = 251,55N^2$. Kdy funkce F_2 je řádově pomalejší oproti funkci F_1 .



Obr. 29 Analýza výpočetní náročnosti přístupů k MPC v měření 1

Měření 2: MPC modelu SISO: $G_2(z^{-1})$ (6.6), $T = 0,03$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,2$; působení šumu $e_s(k)$, hodnoty horizontů a rozměr problému: $N_1 = 1$; $N_u = N_2 = N = 5, 10, \dots, 35$. Měření veličin T_{opt} , T_{mpc} , F_{opt} , J_1 , J_2 dle definic (kap. 6.2, 6.4.1). Pozn.: Šedé řádky v tabulkách vyjadřují nežádoucí překročení periody vzorkování T časem T_{mpc} .

Zavedený přístup je definován jako MPC s využitím Hildrethovy optimalizační metody (příloha A.3) a Wangovou modifikací (příloha A.4) (kap. 4.3.5). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 9. Data měření *SISO_0.03_1om_HW.csv* jsou uložena na CD, viz. příloha B.

Tab. 9 Analýza zavedeného přístupu k MPC za podmínek měření 2

Hildrethova metoda + Wangova modifikace							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flop]	J_1	J_2
5	5	5	0,016	0,016	4758	0,2715	1,1930
10	10	10	0,016	0,016	78719	0,2309	1,1638
15	15	15	0,047	0,047	288616	0,2281	1,1684
20	20	20	0,047	0,047	719237	0,2280	1,1691
25	25	25	0,047	0,047	1450759	0,2280	1,1692
30	30	30	0,047	0,047	1614025	0,2280	1,1692
35	35	35	0,094	0,094	4137202	0,2280	1,1692

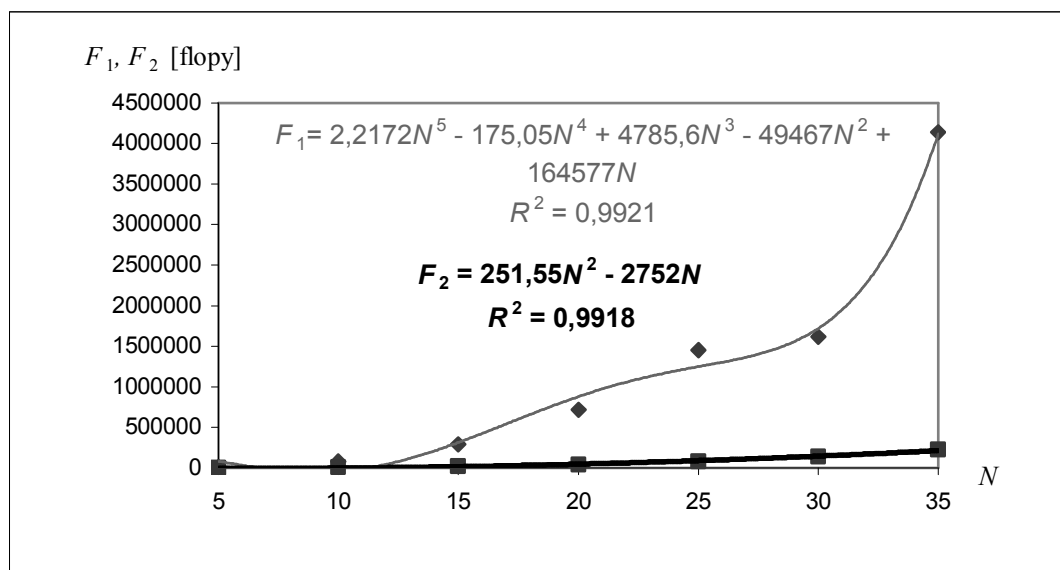
Navrhovaný přístup je definován jakožto MPC s využitím navrhované optimalizační metody (příloha A.5), zařazením eliminací (kap. 5.4.1). Navíc je přístup zkombinován s prospěšnou Wangovou modifikací (příloha A.4). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 10. Data měření *SISO_0.03_1om_NWE.csv* jsou uložena na CD, viz. příloha B.

Tab. 10 Analýza navrhovaného přístupu k MPC definovaného v měření 2

Navrhovaná opt. metoda + Wangova modifikace + návrh eliminací							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_1	J_2
5	5	3	0,016	0,016	822	0,2737	1,1892
10	10	5	0,016	0,016	5349	0,2348	1,1607
15	15	8	0,016	0,016	18318	0,2314	1,1651
20	20	10	0,016	0,016	41099	0,2311	1,1656
25	25	13	0,016	0,016	81674	0,2315	1,1655
30	30	15	0,016	0,016	136905	0,2315	1,1655
35	35	18	0,016	0,016	220484	0,2315	1,1655

Počty operací F_{opt} s přeznačením veličinou F_1 u zavedeného přístupu a veličinou F_2 u navrhovaného přístupu jsou porovnány na obrázku 30.

Z tabulek a grafu měření 2 plyne, že se podařilo navrhovaným řešením (s $m_r < m$) snížit počet operací uvažovaného algoritmu MPC, přičemž se zhoršením kvality řízení maximálně v řádu tisíců u J_1 a J_2 . I v případě $N \geq 15$, kdy nelze algoritmus provést zavedeným přístupem v rámci $T = 0,03$ s. Vliv zařazení navrhovaných přístupů je patrný z funkcí O příslušných regresních křivek aproximující průběhy funkcí F_1 a F_2 ($O_1 > O_2$). Kde $O_1 = 2,2172N^5$ a $O_2 = 251,55N^2$. Kdy funkce F_2 je řádově pomalejší oproti funkci F_1 .



Obr. 30 Analýza výpočetní náročnosti přístupů k MPC v měření 2

Měření 3: MPC modelu SISO: $G_1(z^{-1})$ (6.5), $T = 0,05$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,2$, $u_{\max} = 0,45$; působení šumu $e_s(k)$, hodnoty horizontů a rozměr problému: $N_1 = 1$; $N_u = N_2 = N = 5, 10, \dots, 35$. Měření veličin T_{opt} , T_{mpc} , F_{opt} , J_1 , J_2 dle definic (kap. 6.2, 6.4.1). Pozn.: Šedé řádky v tabulkách vyjadřují nežádoucí překročení periody vzorkování T časem T_{mpc} .

Zavedený přístup je definován jako MPC s využitím Hildrethovy optimalizační metody (příloha A.3) a Wangovou modifikací (příloha A.4) (kap. 4.3.5). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 11. Data měření *SISO_0.05_2om_HW.csv* jsou uložena na CD, viz. příloha B.

Tab. 11 Analýza zavedeného přístupu k MPC za podmínek měření 3

Hildrethova metoda + Wangova modifikace							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_1	J_2
5	10	10	0,016	0,016	32940	0,1295	0,8650
10	20	20	0,016	0,016	189573	0,1253	0,8624
15	30	30	0,047	0,047	1149775	0,1252	0,8623
20	40	40	0,047	0,047	2690833	0,1252	0,8624
25	50	50	0,063	0,063	5582168	0,1252	0,8624
30	60	60	0,172	0,172	20686385	0,1252	0,8624
35	70	70	0,219	0,219	33146987	0,1252	0,8624

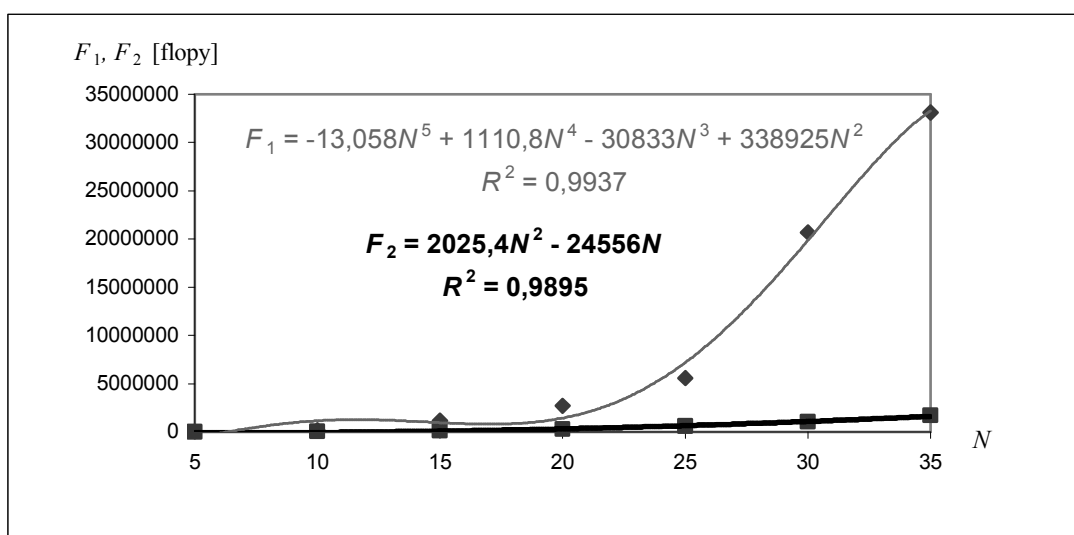
Navrhovaný přístup je pro případ více omezení definován (dle kap. 5.4.2) jakožto aplikace eliminací (kap. 5.4.1) pro MPC s využitím Hildrethovy metody (příloha A.3). Navíc je přístup zkombinován s prospěšnou Wangovou modifikací (příloha A.4). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 12. Data měření *SISO_0.05_2om_HWE.csv* jsou uložena na CD, viz. příloha B.

Tab. 12 Analýza navrhovaného přístupu k MPC definovaného v měření 3

Hildrethova metoda + Wangova modifikace + návrh eliminací							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_1	J_2
5	10	6	0,016	0,016	3466	0,1321	0,8596
10	20	11	0,016	0,016	29544	0,1267	0,8621
15	30	16	0,016	0,016	110629	0,1268	0,8624
20	40	21	0,016	0,016	276614	0,1268	0,8624
25	50	26	0,016	0,016	585482	0,1268	0,8624
30	60	31	0,032	0,032	1038620	0,1268	0,8624
35	70	36	0,032	0,032	1692519	0,1268	0,8624

Počty operací F_{opt} s přeznačením veličinou F_1 u zavedeného přístupu a veličinou F_2 u navrhovaného přístupu jsou porovnány na obrázku 31.

Z tabulek a grafu měření 3 plyne, že se podařilo navrhovaným řešením (s $m_r < m$) snížit počet operací uvažovaného algoritmu MPC, přičemž se zhoršením kvality řízení maximálně v řádu tisíců u J_1 a J_2 . I v případě $N \geq 25$, kdy nelze algoritmus provést zavedeným přístupem v rámci $T = 0,05$ s. Vliv zařazení navrhovaných přístupů je patrný z funkcí O příslušných regresních křivek aproximující průběhy funkcí F_1 a F_2 ($O_1 > O_2$). Kde $O_1 = -13,058N^5$ a $O_2 = 2025,4N^2$. Kdy funkce F_2 je řádově pomalejší oproti funkci F_1 .



Obr. 31 Analýza výpočetní náročnosti přístupů k MPC v měření 3

Měření 4: MPC modelu SISO: $G_2(z^{-1})$ (6.6), $T = 0,03$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,2$, $u_{\max} = 0,45$; působení šumu $e_s(k)$, hodnoty horizontů a rozměr problému: $N_1 = 1$; $N_u = N_2 = N = 5, 10, \dots, 35$. Měření veličin T_{opt} , T_{mpc} , F_{opt} , J_1 , J_2 dle definic (kap. 6.2, 6.4.1). Pozn.: Šedé řádky v tabulkách vyjadřují nežádoucí překročení periody vzorkování T časem T_{mpc} .

Zavedený přístup je definován jako MPC s využitím Hildrethovy optimalizační metody (příloha A.3) a Wangovou modifikací (příloha A.4) (kap. 4.3.5). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 13. Data měření *SISO_0.03_2om_HW.csv* jsou uložena na CD, viz. příloha B.

Tab. 13 Analýza zavedeného přístupu k MPC za podmínek měření 4

Hildrethova metoda + Wangova modifikace							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_1	J_2
5	10	10	0,032	0,032	52218	0,1925	1,4705
10	20	20	0,047	0,047	679729	0,1752	1,3451
15	30	30	0,078	0,078	2431381	0,1761	1,3527
20	40	40	0,11	0,11	5941257	0,1769	1,3540
25	50	50	0,14	0,14	11822034	0,1772	1,3543
30	60	60	0,172	0,172	20686385	0,1773	1,3544
35	70	70	0,203	0,203	33146987	0,1773	1,3544

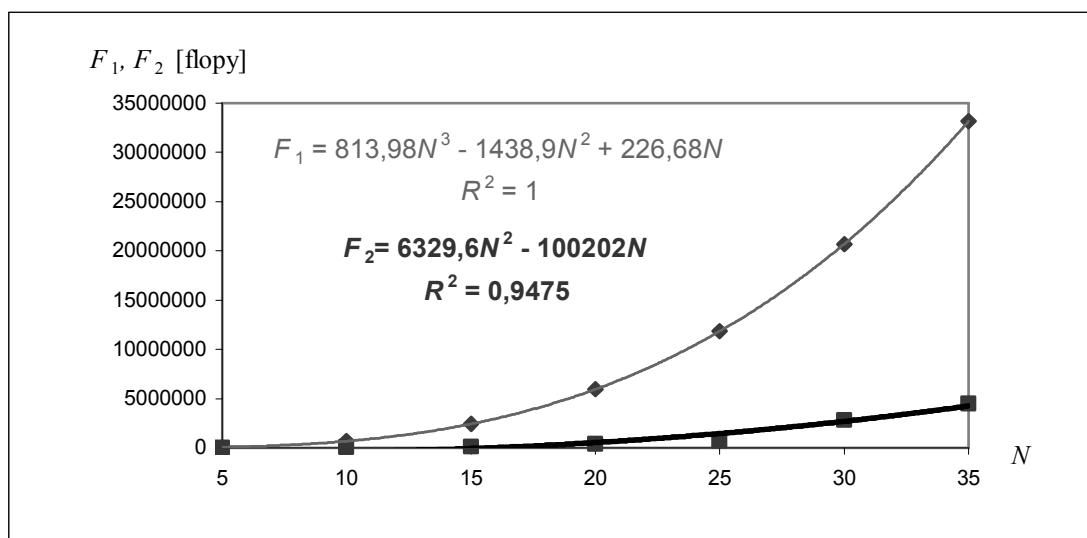
Navrhovaný přístup je pro případ více omezení definován (dle kap. 5.4.2) jakožto aplikace eliminací (kap. 5.4.1) pro MPC s využitím Hildrethovy metody (příloha A.3). Navíc je přístup zkombinován s prospěšnou Wangovou modifikací (příloha A.4). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 14. Data měření *SISO_0.03_2om_HWE.csv* jsou uložena na CD, viz. příloha B.

Tab. 14 Analýza navrhovaného přístupu k MPC definovaného v měření 4

Hildrethova metoda + Wangova modifikace + návrh eliminací							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_1	J_2
5	10	6	0,016	0,016	4886	0,1893	1,4643
10	20	11	0,016	0,016	31488	0,1732	1,3467
15	30	16	0,016	0,016	126889	0,1724	1,3474
20	40	21	0,031	0,031	353934	0,1725	1,3475
25	50	26	0,047	0,047	721940	0,1725	1,3475
30	60	31	0,078	0,078	2826445	0,1725	1,3475
35	70	36	0,094	0,094	4497907	0,1725	1,3475

Počty operací F_{opt} s přeznačením veličinou F_1 u zavedeného přístupu a veličinou F_2 u navrhovaného přístupu jsou porovnány na obrázku 32.

Z tabulek a grafu měření 4 plyne, že se podařilo navrhovaným řešením (s $m_r < m$) snížit počet operací uvažovaného algoritmu MPC, přičemž se zhoršením kvality řízení maximálně v řádu tisíců u J_1 a J_2 . I v případě $N \leq 15$, kdy nelze algoritmus provést zavedeným přístupem v rámci $T = 0,03$ s. Vliv zařazení navrhovaných přístupů je patrný z funkcí O příslušných regresních křivek aproximující průběhy funkcí F_1 a F_2 ($O_1 > O_2$). Kde $O_1 = 813,98N^3$ a $O_2 = 6329,6N^2$. Kdy funkce F_2 je řádově pomalejší oproti funkci F_1 .



Obr. 32 Analýza výpočetní náročnosti přístupů k MPC v měření 4

6.4.3 Dvourozměrové prediktivní řízení

Měření 5: MPC modelu TITO: $G_1(z^{-1})$ (6.7), $T = 0,05$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,02$; působení šumů $e_s(k)$, hodnoty horizontů a rozměr problému: $N_1 = 1; N_u = N_2 = N = 5, 10, \dots, 35$. Měření veličin T_{opt} , T_{mpc} , F_{opt} , J_3 , J_4 dle definic (kap. 6.2, 6.4.1). Pozn.: Šedé řádky v tabulkách vyjadřují nežádoucí překročení periody vzorkování T časem T_{mpc} .

Zavedený přístup je definován jako MPC s využitím Hildrethovy optimalizační metody (příloha A.3) a Wangovou modifikací (příloha A.4) (kap. 4.3.5). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 15. Data měření *TITO_0.05_1om_HW.csv* jsou uložena na CD, viz. příloha B.

Tab. 15 Analýza zavedeného přístupu k MPC za podmínek měření 5

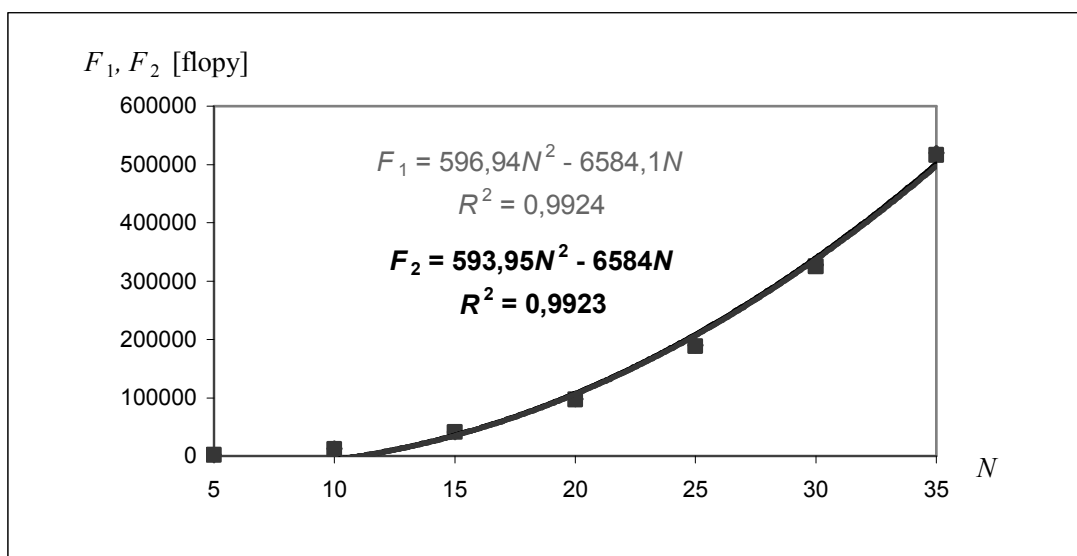
Hildrethova metoda + Wangova modifikace							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_3	J_4
5	10	10	0,016	0,016	1592	0,0364	0,0637
10	20	20	0,016	0,016	12382	0,0415	0,0536
15	30	30	0,016	0,016	41372	0,0414	0,0531
20	40	40	0,016	0,016	97562	0,0413	0,0531
25	50	50	0,016	0,016	189952	0,0413	0,0531
30	60	60	0,016	0,016	327542	0,0413	0,0532
35	70	70	0,016	0,016	519332	0,0413	0,0532

Navrhovaný přístup je pro případ TITO definován (dle kap. 5.4.2) jakožto aplikace eliminací (kap. 5.4.1) pro MPC s využitím Hildrethovy metody (příloha A.3). Navíc je přístup zkombinován s prospěšnou Wangovou modifikací (příloha A.4). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 16. Data měření *TITO_0.05_1om_HWE.csv* jsou uložena na CD, viz. příloha B.

Tab. 16 Analýza navrhovaného přístupu k MPC definovaného v měření 5

Hildrethova metoda + Wangova modifikace + návrh eliminací							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_3	J_4
5	10	3	0,016	0,016	1522	0,0364	0,0637
10	20	5	0,016	0,016	12082	0,0415	0,0536
15	30	8	0,016	0,016	40712	0,0414	0,0531
20	40	10	0,016	0,016	96362	0,0413	0,0531
25	50	13	0,016	0,016	188102	0,0413	0,0531
30	60	15	0,016	0,016	324842	0,0413	0,0532
35	70	18	0,016	0,016	515692	0,0413	0,0532

Počty operací F_{opt} s přeznačením veličinou F_1 u zavedeného přístupu a veličinou F_2 u navrhovaného přístupu jsou porovnány na obrázku 33. Z tabulek a grafu měření 5 plyne, že se podařilo oběma přístupům provést řízení v rámci $T = 0,05$ s, se zachováním kvality řízení. Mírný vliv zařazení navrhovaných přístupů je patrný z funkcí O příslušných regresních křivek aproximující průběhy funkcí F_1 a F_2 ($O_1 \cong O_2$). Kde $O_1 = 596,94N^2$ a $O_2 = 593,95N^2$.



Obr. 33 Analýza výpočetní náročnosti přístupů k MPC v měření 5

Měření 6: MPC modelu TITO: $G_2(z^{-1})(6.10)$, $T = 0,03$ s, meze žádané veličiny $\{0,5,1\}$, omezení $u_{\min} = 0,02$; působení šumů $e_s(k)$, hodnoty horizontů a rozměr problému: $N_1 = 1$; $N_u = N_2 = N = 5, 10, \dots, 35$. Měření veličin T_{opt} ,

T_{mpc} , F_{opt} , J_3 , J_4 dle definic (kap. 6.2, 6.4.1). Pozn.: Šedé řádky v tabulkách vyjadřují nežádoucí překročení periody vzorkování T časem T_{mpc} .

Zavedený přístup je definován jako MPC s využitím Hildrethovy optimalizační metody (příloha A.3) a Wangovou modifikací (příloha A.4) (kap. 4.3.5). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 17. Data měření *TITO_0.03_1om_HW.csv* jsou uložena na CD, viz. příloha B.

Tab. 17 Analýza zavedeného přístupu k MPC za podmínek měření 6

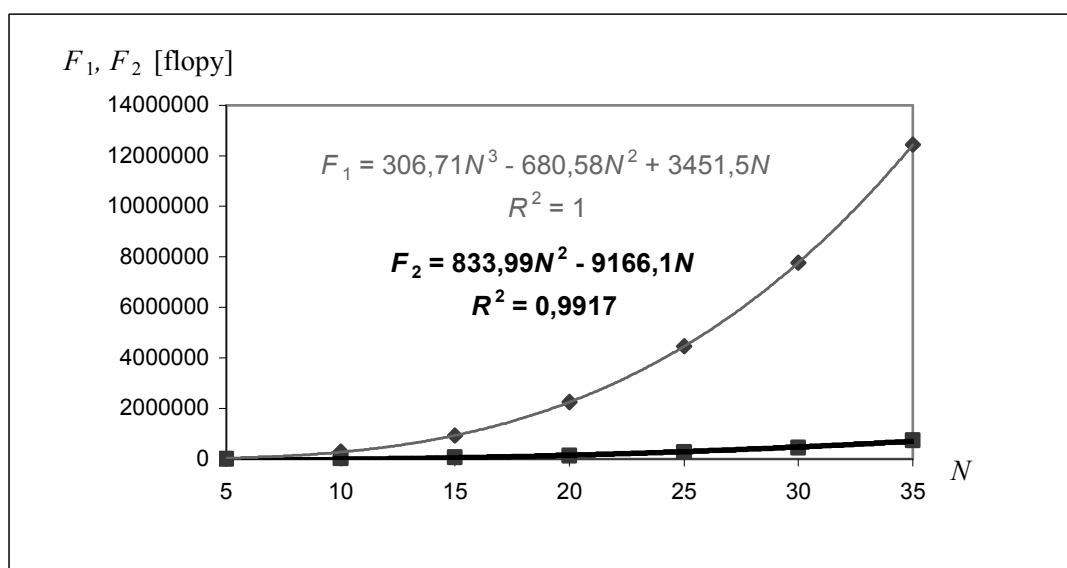
Hildrethova metoda + Wangova modifikace							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_3	J_4
5	10	10	0,016	0,016	29463	0,0448	0,1284
10	20	20	0,016	0,016	288697	0,0618	0,0861
15	30	30	0,031	0,031	925739	0,0657	0,0780
20	40	40	0,031	0,031	2247477	0,0660	0,0766
25	50	50	0,047	0,047	4454815	0,0658	0,0766
30	60	60	0,062	0,062	7775153	0,0656	0,0767
35	70	70	0,078	0,078	12435891	0,0655	0,0767

Navrhovaný přístup je pro případ TITO definován (dle kap. 5.4.2) jakožto aplikace eliminací (kap. 5.4.1) pro MPC s využitím Hildrethovy metody (příloha A.3). Navíc je přístup zkombinován s prospěšnou Wangovou modifikací (příloha A.4). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 18. Data měření *TITO_0.03_1om_HWE.csv* jsou uložena na CD, viz. příloha B.

Tab. 18 Analýza navrhovaného přístupu k MPC definovaného v měření 6

Hildrethova metoda + Wangova modifikace + návrh eliminací							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_3	J_4
5	10	3	0,016	0,016	2632	0,0425	0,1346
10	20	5	0,016	0,016	17470	0,0604	0,0880
15	30	8	0,016	0,016	60133	0,0647	0,0791
20	40	10	0,016	0,016	135381	0,0651	0,0776
25	50	13	0,016	0,016	269584	0,0649	0,0775
30	60	15	0,016	0,016	452442	0,0648	0,0776
35	70	18	0,016	0,016	729685	0,0647	0,0777

Počty operací F_{opt} s přeznačením veličinou F_1 u zavedeného přístupu a veličinou F_2 u navrhovaného přístupu jsou porovnány na obrázku 34. Z tabulek a grafu měření 6 plyne, že se podařilo navrhovaným řešením (s $m_r < m$) snížit počet operací uvažovaného algoritmu MPC, přičemž se zhoršením kvality řízení maximálně v řádu tisíců u J_3 a J_4 . I v případě $N \geq 15$, kdy nelze algoritmus provést zavedeným přístupem v rámci $T = 0,03$ s. Vliv zařazení navrhovaných přístupů je patrný z funkcí O příslušných regresních křivek aproximující průběhy funkcí F_1 a F_2 ($O_1 > O_2$). Kde $O_1 = 306,71N^3$ a $O_2 = 833,99N^2$. Kdy funkce F_2 je řádově pomalejší oproti funkci F_1 .



Obr. 34 Analýza výpočetní náročnosti přístupů k MPC v měření 6

Měření 7: MPC modelu TITO: $G_1(z^{-1})$ (6.7), $T = 0,05$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,02$; $\Delta u_{\max} = 0,005$; působení šumů $e_s(k)$, hodnoty horizontů a rozměr problému: $N_1 = 1$; $N_u = N_2 = N = 5, 10, \dots, 35$. Měření veličin T_{opt} , T_{mpc} , F_{opt} , J_3 , J_4 dle definic (kap. 6.2, 6.4.1). Pozn.: Šedé řádky v tabulkách vyjadřují nežádoucí překročení periody vzorkování T časem T_{mpc} .

Zavedený přístup je definován jako MPC s využitím Hildrethovy optimalizační metody (příloha A.3) a Wangovou modifikací (příloha A.4) (kap. 4.3.5). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 19. Data měření *TITO_0.05_2om_HW.csv* jsou uložena na příloze C.

Tab. 19 Analýza zavedeného přístupu k MPC za podmínek měření 7

Hildrethova metoda + Wangova modifikace							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_3	J_4
5	20	20	0,016	0,016	129957	0,0229	0,0967
10	40	40	0,032	0,032	1065601	0,0336	0,0623
15	60	60	0,032	0,032	11786829	0,0384	0,0556
20	80	80	0,047	0,047	13173990	0,0402	0,0539
25	100	100	0,078	0,078	20383717	0,0409	0,0534
30	120	120	0,109	0,109	47129913	0,0411	0,0532
35	140	140	0,125	0,125	51158085	0,0412	0,0532

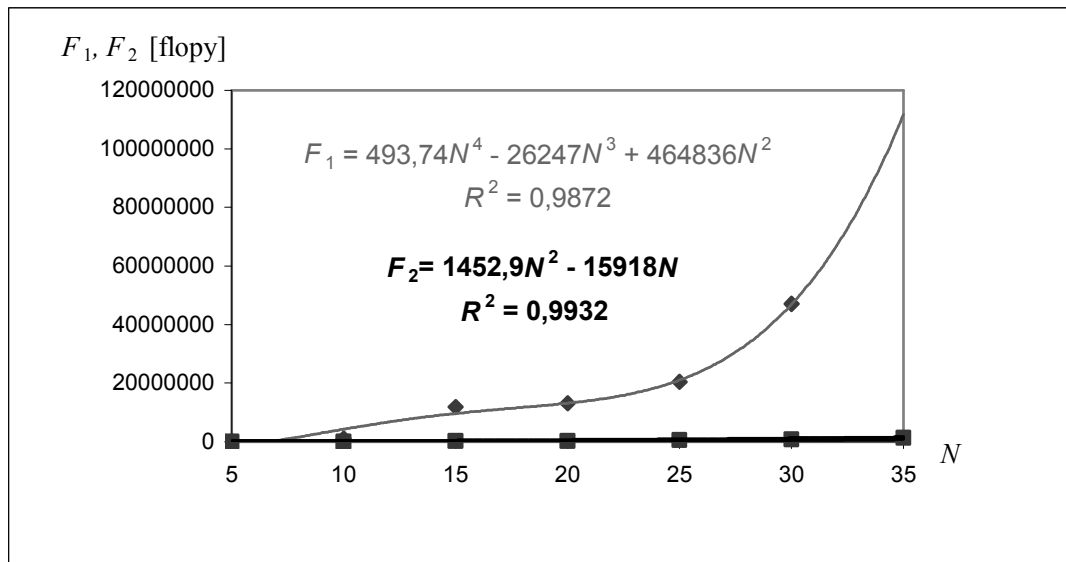
Navrhovaný přístup je pro případ TITO definován (dle kap. 5.4.2) jakožto aplikace eliminací (kap. 5.4.1) pro MPC s využitím Hildrethovy metody (příloha A.3). Navíc je přístup zkombinován s prospěšnou Wangovou modifikací (příloha A.4). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 20. Data měření *TITO_0.05_2om_HWE.csv* jsou uložena na CD, viz. příloha B.

Tab. 20 Analýza navrhovaného přístupu k MPC definovaného v měření 7

Hildrethova metoda + Wangova modifikace + návrh eliminací							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_3	J_4
5	20	5	0,016	0,016	3912	0,0348	0,0656
10	40	10	0,016	0,016	29965	0,0404	0,0545
15	60	15	0,016	0,016	102712	0,0410	0,0534
20	80	20	0,016	0,016	236121	0,0412	0,0532
25	100	25	0,016	0,016	460024	0,0412	0,0532
30	120	30	0,016	0,016	817411	0,0412	0,0532
35	140	35	0,016	0,016	1258980	0,0413	0,0532

Počty operací F_{opt} s přeznačením veličinou F_1 u zavedeného přístupu a veličinou F_2 u navrhovaného přístupu jsou porovnány na obrázku 35. Z tabulek a grafu měření 7 plyne, že se podařilo navrhovaným řešením (s $m_r < m$) snížit počet operací uvažovaného algoritmu MPC, přičemž se zhoršením kvality řízení maximálně v řádu setin u J_3 a J_4 . I v případě $N \geq 25$, kdy nelze algoritmus provést zavedeným přístupem v rámci $T = 0,05$ s.

Vliv zařazení navrhovaných přístupů je patrný z funkcí O příslušných regresních křivek aproximující průběhy funkcí F_1 a F_2 ($O_1 > O_2$). Kde $O_1 = 493,74N^4$ a $O_2 = 1452,9N^2$. Kdy funkce F_2 je řádově pomalejší oproti funkci F_1 .



Obr. 35 Analýza výpočetní náročnosti přístupů k MPC v měření 7

Měření 8: MPC modelu TITO: $G_2(z^{-1})(6.10)$, $T = 0,03$ s, meze žádané veličiny $\{0,5;1\}$, omezení $u_{\min} = 0,02$; $\Delta u_{\max} = 0,005$; působení šumů $e_s(k)$, hodnoty horizontů a rozměr problému: $N_1 = 1$; $N_u = N_2 = N = 5, 10, \dots, 35$. Měření veličin T_{opt} , T_{mpc} , F_{opt} , J_3 , J_4 dle definic (kap. 6.2, 6.4.1). Pozn.: Šedé řádky v tabulkách vyjadřují nežádoucí překročení periody vzorkování T časem T_{mpc} .

Zavedený přístup je definován jako MPC s využitím Hildrethovy optimalizační metody (příloha A.3) a Wangovou modifikací (příloha A.4) (kap. 4.3.5). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 21. Data měření *TITO_0.03_2om_HW.csv* jsou uložena na CD, viz. příloha B.

Tab. 21 Analýza zavedeného přístupu k MPC za podmínek měření 8

Hildrethova metoda + Wangova modifikace							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_3	J_4
5	20	20	0,047	0,047	679729	0,0312	0,2057
10	40	40	0,11	0,11	5825209	0,0550	0,1150
15	60	60	0,11	0,11	14820789	0,0615	0,0932
20	80	80	0,172	0,172	35198073	0,0650	0,0852
25	100	100	0,188	0,188	59788981	0,0675	0,0818
30	120	120	0,25	0,25	102296805	0,0690	0,0803
35	140	140	0,359	0,359	213853869	0,0698	0,0795

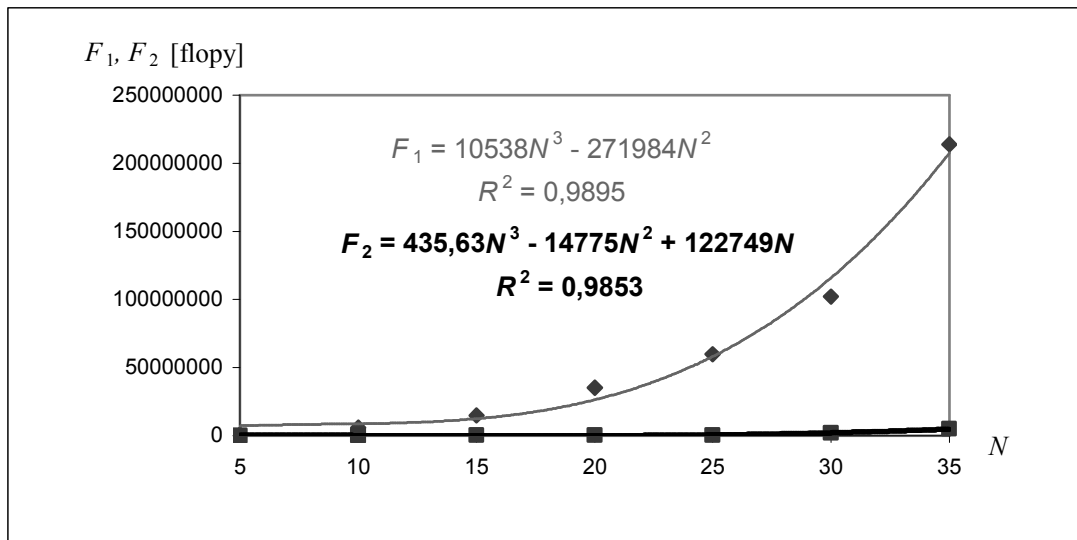
Navrhovaný přístup je pro případ TITO definován (dle kap. 5.4.2) jakožto aplikace eliminací (kap. 5.4.1) pro MPC s využitím Hildrethovy metody (příloha A.3). Navíc je přístup zkombinován s prospěšnou Wangovou modifikací (příloha A.4). Analýza výpočetní náročnosti realizace MPC se nachází v tabulce 22. Data měření *TITO_0.03_2om_HWE.csv* jsou uložena na CD, viz. příloha B.

Tab. 22 Analýza navrhovaného přístupu k MPC definovaného v měření 8

Hildrethova metoda + Wangova modifikace + návrh eliminací							
N	m	m_r	T_{opt} [s]	T_{mpc} [s]	F_{opt} [flopy]	J_3	J_4
5	20	5	0,016	0,016	4248	0,0435	0,1396
10	40	10	0,016	0,016	99927	0,0618	0,0924
15	60	15	0,016	0,016	216404	0,0662	0,0833
20	80	20	0,016	0,016	255993	0,0668	0,0816
25	100	25	0,016	0,016	500236	0,0668	0,0814
30	120	30	0,047	0,047	1909175	0,0666	0,0814
35	140	35	0,094	0,094	5018652	0,0666	0,0814

Počty operací F_{opt} s přeznačením veličinou F_1 u zavedeného přístupu a veličinou F_2 u navrhovaného přístupu jsou porovnány na obrázku 36. Z tabulek a grafu měření 8 plyne, že se podařilo navrhovaným řešením (s $m_r < m$) snížit počet operací uvažovaného algoritmu MPC, přičemž se zhoršením kvality řízení maximálně v řádu setin u J_3 a J_4 , a to pro $N > 10$. I v případě $N \leq 25$, kdy nelze algoritmus provést zavedeným přístupem v rámci $T = 0,03$ s.

Vliv zařazení navrhovaných přístupů je patrný z funkcí O příslušných regresních křivek aproximující průběhy funkcí F_1 a F_2 ($O_1 > O_2$). Kde $O_1 = 10538N^3$ a $O_2 = 435,63N^3$. Kdy funkce F_2 je řádově pomalejší oproti funkci F_1 .



Obr. 36 Analýza výpočetní náročnosti přístupů k MPC v měření 8

7. PŘÍNOS PRÁCE PRO VĚDU A PRAXI

Disertační práce využívá takových návrhů přístupů k řešení MPC procesů s rychlou dynamikou, jež jsou založeny na souvislostech teorie optimalizace. Přičemž při návrzích zlepšení zavedených metod plynule navazuje na dosavadní poznání v dané oblasti. Kromě teoretického základu vychází návrhy a jejich algoritmy též z konkrétních experimentálních případů. Tyto návrhy sledují jednotný cíl, a to snížení výpočetní náročnosti algoritmu řešení MPC se zaměřením na část optimalizace.

Interní mechanismy návrhů metod optimalizace se opírají o teorii duality a podmínky Kuhn–Tuckerovy věty, jež jsou obecně popisného charakteru a nenabízí návod, jak řešit neklasickou úlohu na vázaný extrém. Tuto skutečnost se snaží napravit algoritmizace postupu řešení úlohy kvadratického programování.

Postup návrhu optimalizační metody nejprve vycházel z porušení východisek duální úlohy, a to konkrétně podmínky nezápornosti duální proměnné (Lagrangeových multiplikátorů) ve 4. podmínce Kuhn–Tuckerovy věty. Následně byly korigovány vzniklé mezivýsledky s následným obdržením správného výsledku. Tento přístup i samotná podoba jeho návrhu může být přínosná pro oblast optimalizace. Metoda je citlivá na situace takového zadání úlohy, kdy může nastat porušení podmínek regularity. S čímž se dokáže naopak vypořádat Hildrethova metoda. Tato situace je častá v MPC se zařazením více typů omezení veličin či při mnohazměrovosti řízeného systému.

Nosným návrhem disertační práce je princip eliminací omezení MPC. Výsledky optimalizační úlohy jsou shodné, ale jsou redukovány některé operace. Na základě těchto zásahů se může mírně zhoršit kvalita řízení, avšak lze provést algoritmus řízení pro procesy s rychlou dynamikou rychleji a v rámci stanovené periody vzorkování. Výhodou navrženého principu je jeho modulárnost pro jakoukoliv interně zařazenou optimalizační metodu. Princip eliminací redukuje omezení šetrněji, a to než-li zavedená modifikace dle autora L. Wanga. Jeho aplikace výrazněji napomáhá snížit výpočetní náročnost operací spojených s algoritmem MPC pro náročnější podmínky řízení (mnohazměrovost, počet více zařazených omezení veličin nebo např. vyšší horizonty).

Dále je přínosné sestavení algoritmu pro analytické řešení úlohy kvadratického programování. Kromě názorného řešení praktického příkladu této úlohy byl aplikací tohoto postupu též dokázán výskyt jevu převodu neklasické úlohy na vázaný extrém na úlohu volného extrému. Tento jev

používají zavedené metody a je zjednodušením původní nelineární úlohy s omezeními. Nevýhodou je, že nenastává vždy. V prediktivním řízení jeho výskyt nastává především v ustálených stavech řízení.

Zavedené i navrhované přístupy navazují na zmíněný jev převodu originální úlohy na úlohu volného extrému. Navrhovaný přístup částečných úprav v zadání úlohy optimalizace sice pokračuje v myšlenkovém směru eliminací omezení, ale vždy se jedná o převod na neklasickou úlohu na vázaný extrém menšího rozsahu zadání.

Hlavním praktickým přínosem práce je snížení výpočetní náročnosti postupů, a to v oblasti optimalizace s ohledem na aplikace v prediktivním řízení procesů s rychlou dynamikou. Navíc v případech vícerozměrového řízení této kategorie procesů, a to v situaci s více omezeními či vyššími hodnotami horizontů, je cíl snižování výpočetní náročnosti nezbytný, a to pro uvažovaný online přístup řešení zákona řízení.

Praktickým výsledkem je v tomto ohledu softwarové prostředí, jež je vhodné jak pro analýzu výpočetní náročnosti daných přístupů k řešení algoritmu MPC, tak i pro simulace diskrétního MPC.

8. ZÁVĚR

V rámci disertační práce byl naplněn cíl, a to redukovat počet operací v algoritmu prediktivního řízení procesů s rychlou dynamikou. Byly navrženy takové přístupy k řešení, jež navazují na současný stav poznání dané problematiky. Ve srovnání se zavedenými přístupy se podařilo zrychlit tyto výpočty tak, aby byl algoritmus prediktivního řízení realizovatelný v rámci deklarované periody vzorkování i v takových případech, kdy zavedené přístupy toto nedokáží.

V rámci analýzy výpočetní náročnosti algoritmu MPC, se zaměřením na optimalizační subsystém, bylo zjištěno, že největší vliv na snížení počtu operací má návrh eliminací omezení v optimalizační úloze v realizaci zákona řízení. Na úkor tohoto opatření se mírně změní, v určitých případech nastavení horizontů, kvalita řízení dle definovaných kritérií.

V případě jednorozměrového řízení s jedním typem omezení veličin MPC lze aplikovat navrhovanou optimalizační metodu vyvinutou na základě teorie duality a 4. podmínky Kuhn–Tuckerovy věty. Tento přístup řeší původní úlohu speciálním převodem, a to na sekvenci úloh typu volného extrému s následnými korekcemi.

Simulace a provedená měření dále potvrdily skutečnost, že při náročnějších požadavcích na prediktivní řízení (volba modelu procesu s rychlou dynamikou, mnohorozměrovost, vyšší horizonty, více omezujících podmínek v MPC) roste výpočetní náročnost řešení.

Výhodou navrhovaného přístupu k řešení MPC je jeho modulárnost. Pokud by byla v budoucnu navržena např. rychlejší metoda optimalizace, lze princip eliminací aplikovat v nezměněné podobě.

Implementace navržených i zavedených přístupů byly zkompletovány do podoby softwarového řešení s grafickým uživatelským rozhraním pro MATLAB. Tento výstup práce je hlavním nástrojem pro měření i simulace v rámci disertační práce. Algoritmy řešení optimalizační úlohy byly verifikovány teoreticky odvozeným postupem pro úlohu kvadratického programování. Tímto ověřovacím algoritmem byl též dokázán směrodatný jev možnosti převodu neklasické úlohy na vázaný extrém (kvadratického programování) na úlohu volného extrému. S tímto související speciální tvar podmínek Kuhn–Tuckerovy věty využívá důležitá modifikace autora L. Wanga.

Na základě provedených měření výpočetní náročnosti předkládaných řešení lze v porovnání s výsledky zavedených přístupů konstatovat, že pomocí navrhovaných metod je možné provést algoritmus MPC pro procesy s rychlou dynamikou rychleji. I když měření zahrnují též časové údaje vázané ke konkrétní použité konfiguraci hardwarových prostředků, analýza počtu operací prokázala, že zavedené přístupy mají řádově vyšší výpočetní náročnost řešení MPC a redukce operací je způsobena aplikací navržených řešení.

SEZNAM POUŽITÉ LITERATURY

- [1] CAMACHO, Eduardo F., and Carlos BORDONS. *Model predictive control*. London: Springer, 2007. ISBN 978-1-85233-694-3.
- [2] KWON, Wook Hyun. *Receding horizon control: model predictive control for state models*. London: Springer, 2005. ISBN 1-84628-024-9.
- [3] RAWLINGS, James B., and David Q. MAYNE. *Model Predictive Control: Theory and Design*. Madison: Nob Hill Publishing, 2009. ISBN 978-0-9759377-0-9.
- [4] WANG, Liuping. *Model Predictive Control System Design and Implementation Using MATLAB*. London: Springer-Verlag Limited, 2009. ISBN 978-1-84882-330-3.
- [5] HUANG, Sunan. *Applied predictive control*. London: Springer, 2002. ISBN 1852333383.
- [6] CORRIOU, Jean-Pierre. *Process control: theory and applications*, London: Springer, 2004. ISBN 1-85233-776-1.
- [7] BALÁTEĚ, Jaroslav. *Automatické řízení*. Praha: BEN - technická literatura, 2004. ISBN 80-7300-148-9.
- [8] LUSE, D., KHALIL, H. Frequency domain results for systems with slow and fast dynamics, *Automatic Control*, IEEE Transactions on , vol.30, no.12, pp.1171,1179, 1985.
- [9] KUČERA, Vladimír. *Analysis and Design of Discrete Linear Control Systems*. Praha: Nakladatelství Československé akademie věd, 1991. ISBN 80-200-0252-9.
- [10] STREJC, Vladimír. *Stavová teorie lineárního diskrétního řízení*. Praha: Nakladatelství Československé akademie věd, 1978.
- [11] HOAGG J. B., and BERNSTEIN, D. S. Nonminimum-phase zeros - much to do about nothing - classical control - revisited part II, in *Control Systems*, IEEE, vol.27, no.3, pp.45,57, 2007.
- [12] BOBÁL, V., KUBALČÍK, M., DOSTÁL, P., MATĚJÍČEK, J. Adaptive predictive control of time-delay systems. *Computers & Mathematics with Applications*, Vol. 66, No. 2, 2013, 165-176.

- [13] KUBALČÍK, Marek, and Vladimír BOBÁL. Computation of predictions in multivariable predictive control. In *Proceeding ACMOS'11 Proceedings of the 13th WSEAS international conference on Automatic control, modelling & simulation*. 2011. ISBN 978-1-61804-004-6.
- [14] HAMALA, Milan. *Nelineárne programovanie*. Bratislava: Vydavateľstvo technickej a ekonomickej literatúry, 1972. 240 s. Matematické metódy v ekonomike.
- [15] LEE, G. M., TAM, N. N., and N. D. YEN. *Quadratic Programming and Affine Variational Inequalities: A Qualitative Study*. New York: Springer, 2005. ISBN 0-387-24277-5.
- [16] LUENBERGER, David G., and Yinyu YE. *Linear and nonlinear programming*. New York: Springer, 2008. ISBN 978-0-387-74502-2.
- [17] DOSTÁL, Zdeněk. *Optimal Quadratic Programming Algorithms: With Applications to Variational Inequalities*. New York: Springer, 2009. ISBN 978-0387848051.
- [18] PÁNKOVÁ, Václava. *Nelineární optimalizace pro ekonomy*. Praha: Professional Publishing, 2003. 153 s. ISBN 80-86419-50-9.
- [19] HERTOOG, D. *Interior point approach to linear, quadratic, and convex programming: algorithms and complexity*. Dordrecht: Kluwer Academic Publishers, c1994, xii, 208 s. ISBN 0-7923-2734-9.
- [20] RAO, Singiresu S. *Engineering optimization: theory and practice*. 4th ed. Hoboken, New Jersey: Wiley, c2009, 813 s. ISBN 978-0-470-18352-6.
- [21] BAKR, Mohamed. *Nonlinear optimization in electrical engineering with applications in MATLAB*. London: Institution of Engineering and Technology, 2013.
- [22] ŠUBRT, Tomáš. *Ekonomicko-matematické metody*. Plzeň: Vydavatelství a nakladatelství Aleš Čeněk, 2011. 351 s. ISBN 978-80-7380-345-2.
- [23] WANG, Yang, and BOYD, Stephen. Fast Model Predictive Control Using Online Optimization. In *Control Systems Technology*, IEEE Transactions on, vol.18, no.2, pp.267-278, 2010. ISSN 1063-6536.

- [24] ALAMIR, Mazen. On Trade-offs Governing Real-Time Implementation of Model Predictive Control. In *20th International Conference on Process Control*. Štrbské pleso, Slovak Republic. 2015. ISBN 978-1-4673-6626-7.
- [25] GUANG Gueng; GEARY, G.M.. Experimental comparisons between generalised predictive control algorithms using CARMA and CARIMA models. *Industrial Electronics, Control, and Instrumentation*, Proceedings of the 1996 IEEE IECON 22nd International Conference on, vol.1, no., pp.108,113 vol.1, 1996.
- [26] BELDA, Květoslav. On-line Solution of System Constraints in Generalized Predictive Control Design. In *20th International Conference on Process Control (PC)*. Štrbské pleso, Slovak Republic. 2015. ISBN 978-1-4673-6626-7.
- [27] JOHANSEN, Tor A., and Alexandra GRANCHAROVA. Approximate explicit constrained linear model predictive control via orthogonal search tree. In *Automatic Control*. 2003.
- [28] INGOLE, D., HOLAZA, J., TAKÁCS, B., KVASNICA, M. FPGA-Based Explicit Model Predictive Control for Closed Loop Control of Intravenous Anesthesia. In *20th International Conference on Process Control (PC)*. Štrbské pleso, Slovak Republic. 2015. ISBN 978-1-4673-6626-7.
- [29] KVASNICA, Michal, and Miroslav FIKAR. Design and implementation of model predictive control using Multi-Parametric Toolbox and YALMIP. In *Computer-Aided Control System Design (CACSD)*. 2010.
- [30] HUNGER, Raphael. Floating Point Operations in Matrix-Vector Calculus. (Version 1.3). Technical Report [online]. Technische Universität München, Associate Institute for Signal Processing, 2007, [cit. 2016-03-11]. Dostupné z: <https://mediatum.ub.tum.de/doc/625604/625604.pdf>
- [31] VANÍČEK, Jiří. *Teoretické základy informatiky*. Praha: Kernberg Publishing, 2007. ISBN 978-80-903962-4-1.
- [32] KUBALČÍK, Marek, and Vladimír BOBÁL. Control algorithms with suppression of measurable disturbances: Comparison of two methods. In *WSEAS Transactions on Systems and Control*. 2013. E-ISSN 2224-2856.
- [33] BAROT Tomáš, and Marek KUBALČÍK. Predictive Control with Particular Modifications in Quadratic Programming Problem. In *20th*

International Conference on Process Control (PC). Štrbské pleso, Slovak Republic. 2015. ISBN 978-1-4673-6626-7.

[34] KUBALČÍK, Marek a Tomáš BAROT. *Simulace a analýza prediktivního řízení z pohledu optimalizace* [software]. [cit. 2016-04-01]. Dostupné z: http://nod32.fai.utb.cz/Promotion/software_OBD/2012/barot_software.zip

SEZNAM OBRÁZKŮ

Obr. 1 Ukázka zadání diskretního matematického modelu TITO procesu	66
Obr. 2 Zadávání průběhu žádané veličiny v rámci programu	66
Obr. 3 Ukázka nastavení parametrů prediktivního regulátoru	67
Obr. 4 Přechodové charakteristiky (spojitá a diskretní) SISO modelu procesu	69
Obr. 5 Přechodové charakteristiky h_{11} (spojitá a diskretní) TITO modelu procesu	70
Obr. 6 Přechodové charakteristiky h_{12} (spojitá a diskretní) TITO modelu procesu	70
Obr. 7 Přechodové charakteristiky h_{21} (spojitá a diskretní) TITO modelu procesu	71
Obr. 8 Přechodové charakteristiky h_{22} (spojitá a diskretní) TITO modelu procesu	71
Obr. 9 Průběh vygenerované neměřitelné poruchové veličiny pro SISO řízení	73
Obr. 10 Průběh veličin prediktivního řízení pro nastavení MPC dle příkladu 1	73
Obr. 11 Průběh veličin prediktivního řízení pro nastavení MPC dle příkladu 2	74
Obr. 12 Průběh veličin prediktivního řízení pro nastavení MPC dle příkladu 3	75
Obr. 13 Průběh veličin prediktivního řízení pro nastavení MPC dle příkladu 4	75
Obr. 14 Průběh veličin y_1 a šumu pro nastavení MPC dle příkladu 5	76
Obr. 15 Průběh veličin y_2 a šumu pro nastavení MPC dle příkladu 5	76
Obr. 16 Průběh řídicích veličin pro nastavení MPC dle příkladu 5	77
Obr. 17 Průběh veličin y_1 a šumu pro nastavení MPC dle příkladu 6	77
Obr. 18 Průběh veličin y_2 a šumu pro nastavení MPC dle příkladu 6	78
Obr. 19 Průběh řídicích veličin pro nastavení MPC dle příkladu 6	78
Obr. 20 Průběh veličin y_1 a šumu pro nastavení MPC dle příkladu 7	79
Obr. 21 Průběh veličin y_2 a šumu pro nastavení MPC dle příkladu 7	79
Obr. 22 Průběh řídicí veličiny u_1 pro nastavení MPC dle příkladu 7	80

Obr. 23 Průběh řídicí veličiny u_2 pro nastavení MPC dle příkladu 7	80
Obr. 24 Průběh veličin y_1 a šumu pro nastavení MPC dle příkladu 8	81
Obr. 25 Průběh veličin y_2 a šumu pro nastavení MPC dle příkladu 8	81
Obr. 26 Průběh řídicí veličiny u_1 pro nastavení MPC dle příkladu 8	82
Obr. 27 Průběh řídicí veličiny u_2 pro nastavení MPC dle příkladu 8	82
Obr. 28 Porovnání výpočetní náročnosti Hildrethovy a navrhované opt. metody	87
Obr. 29 Analýza výpočetní náročnosti přístupů k MPC v měření 1	91
Obr. 30 Analýza výpočetní náročnosti přístupů k MPC v měření 2	93
Obr. 31 Analýza výpočetní náročnosti přístupů k MPC v měření 3	95
Obr. 32 Analýza výpočetní náročnosti přístupů k MPC v měření 4	97
Obr. 33 Analýza výpočetní náročnosti přístupů k MPC v měření 5	99
Obr. 34 Analýza výpočetní náročnosti přístupů k MPC v měření 6	101
Obr. 35 Analýza výpočetní náročnosti přístupů k MPC v měření 7	103
Obr. 36 Analýza výpočetní náročnosti přístupů k MPC v měření 8	105

SEZNAM TABULEK

Tab. 1 Definice podmínek v jednorozměrném prediktivním řízení	32
Tab. 2 Definice podmínek v dvourozměrném prediktivním řízení	36
Tab. 3 Počet řádků jednotlivých typů omezení v prediktivním řízení	58
Tab. 4 Definice modifikovaných podmínek v SISO prediktivním řízení	61
Tab. 5 Definice modifikovaných podmínek v TITO prediktivním řízení	62
Tab. 6 Počty operací frekventovaných matematických vztahů	83
Tab. 7 Analýza zavedeného přístupu k MPC za podmínek měření 1	90
Tab. 8 Analýza navrhovaného přístupu k MPC definovaného v měření 1	91
Tab. 9 Analýza zavedeného přístupu k MPC za podmínek měření 2	92
Tab. 10 Analýza navrhovaného přístupu k MPC definovaného v měření 2	93
Tab. 11 Analýza zavedeného přístupu k MPC za podmínek měření 3	94
Tab. 12 Analýza navrhovaného přístupu k MPC definovaného v měření 3	95
Tab. 13 Analýza zavedeného přístupu k MPC za podmínek měření 4	96
Tab. 14 Analýza navrhovaného přístupu k MPC definovaného v měření 4	97
Tab. 15 Analýza zavedeného přístupu k MPC za podmínek měření 5	98
Tab. 16 Analýza navrhovaného přístupu k MPC definovaného v měření 5	99
Tab. 17 Analýza zavedeného přístupu k MPC za podmínek měření 6	100
Tab. 18 Analýza navrhovaného přístupu k MPC definovaného v měření 6	100
Tab. 19 Analýza zavedeného přístupu k MPC za podmínek měření 7	102
Tab. 20 Analýza navrhovaného přístupu k MPC definovaného v měření 7	102
Tab. 21 Analýza zavedeného přístupu k MPC za podmínek měření 8	104
Tab. 22 Analýza navrhovaného přístupu k MPC definovaného v měření 8	104

SEZNAM POUŽITÝCH SYMBOLŮ

n	rozměr primární proměnné optimalizační úlohy
m	počet omezujících podmínek v optimalizační úloze, rozměr duální proměnné optimalizační úlohy
m_r	počet omezujících podmínek v optimalizační úloze zbylých po jejich redukci a rozměr duální proměnné optimalizační úlohy
i, j, l	indexy určující pořadí prvků v rámci dané množiny
k	pořadnice diskrétního prediktivního řízení nebo též krok algoritmu optimalizační metody, $k \in Z_0^+$
t	čas uvažovaný od okamžiku $t = 0$
N_1	minimální horizont MPC, uvažovaný v práci jako $N_1 = 1$
N_u	řídící horizont MPC
N_2	maximální horizont MPC
τ_i, T_j	časové konstanty modelu procesu
T	perioda vzorkování diskrétního MPC
\bar{T}	doba dosažení 95 % ustálené hodnoty přechodové charakteristiky modelu řízeného procesu
K, K_{ij}	zesílení uvažovaného modelu procesu
\mathcal{G}_i	nulový bod spojitě přenosové funkce modelu procesu
π_i	pól spojitě přenosové funkce modelu procesu
a_i	koeficienty jmenovatele diskrétní přenosové funkce modelu procesu
b_j	koeficienty čitatele diskrétní přenosové funkce modelu procesu
e_s, e_{s1}, e_{s2}	veličiny reprezentující šum; hodnoty v okamžiku k značeny s (k)
u, u_1, u_2	vstupní či akční veličiny řízení; hodnoty v okamžiku k značeny s (k)

- y, y_1, y_2 výstupní veličiny řízení; hodnoty v okamžiku k značeny s (k)
- w, w_1, w_2 žádané veličiny řízení; hodnoty v okamžiku k značeny s (k)
- $\Delta u, \Delta u_1, \Delta u_2$ přírůstky vstupních veličin řízení; hodnoty v okamžiku k značeny s (k)
- \min, \max symbolika dolních indexů značící minimální, resp. maximální hodnoty daných veličin v rámci definice omezení v MPC
- δ počet prvků množiny \mathcal{T} v rámci navrhované duální metody
- N_w, M_w, P_w rozměry pomocných matic aparátu určení výpočetní náročnosti
- ε reálná konstanta v rámci aparátu určení výpočetní náročnosti
- N rozměr problému řešeného algoritmu z pohledu výpočetní náročnosti
- R^2 koeficient spolehlivosti v rámci regresní analýzy
- $T_{mpc}(k)$ doba provedení algoritmu MPC v rámci k -té periody vzorkování
- $T_{opt}(k)$ doba provedení algoritmu optimalizace v rámci k -té periody vzorkování
- T_{mpc} maximální doba provedení algoritmu MPC v rámci celého řízení
- T_{opt} maximální doba provedení algoritmu optimalizace v rámci celého řízení
- J_1, J_2 definovaná kritéria posouzení kvality řízení u SISO MPC
- J_3, J_4 definovaná kritéria posouzení kvality řízení u TITO MPC
- F_1, F_2 počty operací posuzovaného algoritmu v jednotkách flopy
- F_{opt} počet operací algoritmu optimalizace v jednotkách flopy
- \mathcal{R}^n Euklidův prostor o rozměru n
- Z_0^+ číselný obor kladných celých čísel s uvažováním hodnoty 0

\mathcal{N}	číselný obor přirozených čísel
\mathcal{K}	množina řešitelnosti optimalizační úlohy
\mathcal{K}'	množina definující okolí bodu v \mathcal{R}^n
\mathcal{T}	množina pro uložení indexů v rámci navrhované duální metody
$\{\}$	prázdná množina
\mathbf{x}	vektor (s prvky x_j ; $j = 1, \dots, n$) – primární proměnná opt. úlohy
$\hat{\mathbf{x}}$	vektor (s prvky \hat{x}_j ; $j = 1, \dots, n$) – stacionární bod účelové funkce
\mathbf{x}^\bullet	vektor (s prvky x_j^\bullet ; $j = 1, \dots, n$) – globální extrém účelové funkce
$\boldsymbol{\gamma}$	vektor (s prvky γ_j ; $j = 1, \dots, m$) pro definici omezení opt. Úlohy
$\tilde{\boldsymbol{\gamma}}$	vektor (s prvky $\tilde{\gamma}_j$; $j = 1, \dots, m_r$) pro definici redukovaných omezujících podmínek opt. úlohy
$\boldsymbol{\lambda}$	vektor Lagrangeových multiplikátorů (s prvky λ_j ; $j = 1, \dots, m$)
$\hat{\boldsymbol{\lambda}}$	vektor Lagrangeových multiplikátorů (s prvky $\hat{\lambda}_j$; $j = 1, \dots, m$) příslušející stacionárnímu bodu
$\boldsymbol{\sigma}$	vektor koeficientů pravých stran soustavy rovnic; o $m+n$ prvcích
$\mathbf{0}$	vektor příslušných rozměrů zahrnující nulové prvky
\mathbf{b}	vektor o n prvcích v rámci úlohy kvadratického programování vázaný k lineární části zadání
\mathbf{d}	vektor (s prvky d_j ; $j = 1, \dots, m$) – duální proměnná optimalizační úlohy
\mathbf{d}^\bullet	vektor (s prvky d_j^\bullet ; $j = 1, \dots, m$) – globální extrém duální účelové funkce
$\mathbf{d}(k)$	vektor duální proměnné určený v k -tém kroku numerické metody

- \mathbf{o} vektor (s prvky o_j ; $j = 1, \dots, m$) lineární části zadání duální úlohy kvadratického programování
- \mathbf{o}° mezivýsledek vektoru lineární části zadání navrhované duální úlohy kvadratického programování, s δ prvky
- $\bar{\mathbf{y}}$ vektor m doplňkových proměnných s kvadratickými prvky \bar{y}_j^2
- $\hat{\mathbf{y}}$ vektor m doplňkových proměnných s kvadratickými prvky \hat{y}_j^2 příslušející stacionárnímu bodu
- \mathbf{y} vektor predikované výstupní veličiny, resp. veličin u TITO, se značením (k) pro aktuální okamžik diskretního řízení
- $\Delta \mathbf{u}$ vektor predikované akční veličiny, resp. veličin u TITO, se značením (k) pro aktuální okamžik diskretního řízení
- \mathbf{w} vektor žádané veličiny, resp. veličin u TITO, se značením (k) pro aktuální okamžik diskretního řízení
- \mathbf{e}_s vektor šumu u TITO, se značením (k) pro aktuální okamžik diskretního řízení
- $\boldsymbol{\chi}$ vektor koeficientů pravých stran soustavy pouze lineárních rovnic
- \mathbf{v}_λ vektor s informacemi o nulování Lagrangeových multiplikátorů
- \mathbf{v}_y vektor s informacemi o nulování doplňkových proměnných
- \mathbf{d}^\times vektor (s prvky d_j^\times ; $j = 1, \dots, m$) – mezivýsledek řešení navrhované duální metody
- \mathbf{d}° vektor (s prvky d_j° ; $j = 1, \dots, m$) – mezivýsledek řešení navrhované duální metody
- \mathbf{M} matice koeficientů levé části soustavy rovnic definující podmínky omezení (s prvky M_{ij} ; $i = 1, \dots, m$; $j = 1, \dots, n$)

- \tilde{M} matice koeficientů levé části soustavy rovnic pro definici redukovaných omezujících podmínek opt. úlohy (s prvky \tilde{M}_{ij} ; $i = 1, \dots, m_r$; $j = 1, \dots, n$)
- E čtvercová matice koeficientů levé strany soustavy rovnic; rozměru $m+n$
- H Hessián primární úlohy kvadratického programování s prvky H_{ij} , čtvercová matice o rozměru n
- N Hessián duální úlohy kvadratického programování s prvky N_{ij} , čtvercová matice rozměru m
- N° mezivýsledek pro Hessián navrhované duální úlohy kvadratického programování, čtvercová matice rozměru δ
- P, G matice soustavy predikčních rovnic, s prvky G_{ij} , resp. P_{ij}
- I jednotková matice příslušných rozměrů
- E jedničková matice příslušných rozměrů (všechny prvky rovny 1)
- A_1, \dots, A_3 pomocné matice pro zpřehlednění predikčních rovnic u TITO MPC
- B_1, B_2 pomocné matice pro zpřehlednění predikčních rovnic u TITO MPC
- G_{ij}, P_{ij} submatice matic P, G v rámci predikčních rovnic u TITO MPC
- Γ matice koeficientů levé strany soustavy pouze lineárních rovnic
- $diag(\mathbf{v})$ sestavení diagonální matice z diagonálních prvků daných vektorem \mathbf{v}
- W_1, \dots, W_3 pomocné matice aparátu určení výpočetní náročnosti maticových operací
- s komplexní proměnná v rámci Laplaceovy transformace
- z^{-1} komplexní proměnná v rámci Z-transformace
- Δ operátor zpětné difference; $\Delta = 1 - z^{-1}$

- $f(\mathbf{x})$ reálná funkce n reálných proměnných; účelová funkce
- J účelová funkce optimalizační úlohy v rámci MPC
- L Lagrangeova funkce optimalizační úlohy na vázaný extrém
- \tilde{L} Lagrangeova funkce optimalizační úlohy na vázaný extrém s doplňkovými proměnnými
- $h(t)$ spojitá přechodová charakteristika
- $h(kT)$ diskrétní přechodová charakteristika, též možné označovat $h(k)$
- $h_{ij}(t)$ dílčí spojitě přechodové charakteristiky pro TITO model procesu
- $h_{ij}(kT)$ dílčí spojitě přechodové charakteristiky pro TITO model procesu
- $O(N)$ funkce určující řád růstu funkce závislé na rozměru problému N
- O_1, O_2 funkce určující řád růstu funkce závislé na rozměru problému N u konkrétních analyzovaných algoritmů
- $O_1 > O_2$ funkce O_1 roste řádově rychleji oproti funkci O_2
- $O_1 \cong O_2$ funkce O_1 roste řádově přibližně shodně ve srovnání s funkcí O_2
- $\{i\}_{i=1}^n$ aritmetická posloupnost n prvků s hodnotami $1, \dots, n$
- $G(s)$ spojitá přenosová funkce
- $\mathbf{G}(s)$ matice spojitých přenosových funkcí; spojitá přenosová matice
- $G(z^{-1})$ diskrétní přenosová funkce uvažovaná pro danou periodu vzorkování
- $\mathbf{G}(z^{-1})$ matice diskrétních přenosových funkcí; diskrétní přenosová matice uvažovaná pro danou periodu
- $\mathbf{A}(z^{-1})$ polynomiální matice, maticový zlomek diskrétní přenosové matice

$\mathbf{B}(z^{-1})$ polynomiální matice, maticový zlomek diskrétní přenosové matice

$\mathbf{C}(z^{-1})$ polynomiální matice v rámci CARIMA modelu pro TITO MPC

$\Delta(z^{-1})$ polynomiální matice operátorů zpětné diference v rámci CARIMA modelu pro TITO MPC

$M_j(\mathbf{x})$ levá strana rovnice j -té podmínky z m omezení opt. úlohy

$A(z^{-1})$ polynom jmenovatele diskrétní přenosové funkce

$B(z^{-1})$ polynom čitatele diskrétní přenosové funkce

$\alpha_{ij}(z^{-1})$ polynomy maticového zlomku $\mathbf{A}(z^{-1})$ s koeficienty α_{ijl}

$\beta_{ij}(z^{-1})$ polynomy maticového zlomku $\mathbf{B}(z^{-1})$ s koeficienty β_{ijl}

$C(z^{-1})$ polynom v rámci CARIMA modelu pro SISO MPC

T operátor vyjadřující transpozici matici

\in operátor náležitosti prvku dané množině

\notin operátor nenáležitosti prvku dané množině

$|$ operátor připojující nutné podmínky

\langle , \rangle závorky vyjadřující dolní a horní mez uzavřeného intervalu čísel z \mathcal{R}

$(,)$ závorky vyjadřující dolní a horní mez otevřeného intervalu čísel z \mathcal{R}

arg operace vyjádření argumentu funkce či matematické operace

min operace nalezení minima

∇ gradient funkce více reálných proměnných

∇_x gradient funkce více reálných proměnných podle proměnné \mathbf{x}

∇_λ gradient funkce více reálných proměnných podle proměnné λ

$ M $	determinant matice, resp. matice M
∂	operátor parciální derivace reálné funkce více reálných proměnných
∇^2	operátor Hessiánu reálné funkce více reálných proměnných
∇_x^2	Hessián funkce více reálných proměnných pouze podle proměnné x
$^{-1}$	operátor vyjadřující inverzi matici
\forall	kvantifikátor vyjadřující zohlednění každého prvku dané množiny
\wedge	operátor logického součinu
\neg	operátor negace
\max	operace nalezení maxima
\cong	operátor přibližné rovnosti
\equiv	operátor totožnosti
\Leftrightarrow	operátor ekvivalence
sgn	funkce určení znaménka výrazu
$:=$	operátor přiřazení hodnoty dané výrazem
\rightarrow	operátor vyjadřující sled operací
\cup	operátor sjednocení množin

SEZNAM ZKRATEK

MPC	Model Predictive Control – Prediktivní řízení založené na modelu řízeného procesu.
LDDS	Linear Discrete Dynamical Systems – Lineární diskrétní dynamické systémy.
SISO	Single Input Single Output – Systém s jedním vstupem a výstupem.
MIMO	Multi Input Multi Output – Systém s více vstupy a více výstupy.
TITO	Two Input Two Output – Systém s dvěma vstupy a dvěma výstupy.
CARIMA	Controlled Autoregressive Integrated Moving Average.
GUI	Graphical User Interface – Grafické uživatelské rozhraní programu.
SQP	Sequential Quadratic Programming – Sekvenční kvadratické programování.
SLP	Sequential Linear Programming – Sekvenční lineární programování.
FLOP	Floating-point operation – Elementární operace s čísly oboru \mathcal{R} .
CSV	Comma Separated Values – Typ datového souboru s hodnotami na řádku oddělenými středníkem a poslední hodnotou oddělenou speciálním znakem ukončení řádku.

PUBLIKAČNÍ AKTIVITY AUTORA

D – Článek ve sborníku konference evidované v databázi Web of Science a SCOPUS

BAROT Tomáš [70] and Marek KUBALČÍK [30]. Predictive Control with Particular Modifications in Quadratic Programming Problem. In *20th International Conference on Process Control (PC)*. Štrbské pleso, Slovak Republic. 2015. ISBN 978-1-4673-6626-7.

BAROT Tomáš [70] and Marek KUBALČÍK [30]. Predictive Control of Systems with Fast Dynamics Using Computational Reduction Based on Feedback Control Information. In *Intelligent Systems in Cybernetics and Automation Theory. Proceedings of the 4th Computer Science On-line Conference 2015 (CSOC2015), Vol 2*. Advances in Intelligent Systems and Computing 348, Springer. Switzerland, 2015. ISBN 978-3-319-18502-6.

BAROT Tomáš [80] and Marek KUBALČÍK [20]. Predictive Control of Non-Minimum Phase Systems. In *15th International Carpathian Control Conference (ICCC)*. Velke Karlovice, Czech Republic, 2014. ISBN 978-1-4799-3527-7.

D – Článek ve sborníku konference evidované v databázi SCOPUS

BAROT Tomáš [70] and Marek KUBALČÍK [30]. Predictive Control of Processes with Fast Dynamics. In *16th International Carpathian Control Conference (ICCC)*. Szilvásvárad, Hungary, 2015. ISBN 978-1-4799-7369-9.

BAROT Tomáš [80] and Marek KUBALČÍK [20]. Primary Method of Quadratic Programming in Multivariable Predictive Control with Constraints. In *Nostradamus 2014: Prediction, Modeling and Analysis of Complex Systems*, Advances in Intelligent Systems and Computing 289, Springer. Switzerland, 2014. ISBN 978-3-319-07400-9.

J_{neimp} – Článek v recenzovaném časopise uvedeném v seznamu českých recenzovaných periodik (po aktualizaci r. 2014)

BAROT, Tomáš [80] and Marek KUBALČÍK [20]. Equality Constraints in Predictive Control of Laboratory Model. *Trilobit* [online]. 2014, roč. 6, č. 1 [cit. 2016-04-01]. ISSN 1804-1795.

J_{neimp} – Článek v recenzovaném časopise uvedeném v seznamu českých recenzovaných periodik (po aktualizaci r. 2010)

BAROT, Tomáš. Specifické využití Microsoft Visual Basic for Application s principy modelování systémů. *Journal of Technology and Information Education*. 2012, Olomouc - EU, Univerzita Palackého, Ročník 4, Číslo 1, s. 84 - 88. ISSN 1803-537X (print). ISSN 1803-6805 (on-line).

J_{neimp} – Článek ve vědeckém nebo odborném časopise - ostatní jazyky

BAROT, Tomáš [80] a Marek KUBALČÍK [20]. Porovnání GPC řízení procesů s dopravním zpožděním a jeho korigované varianty za použití Smithova prediktoru. *POSTERUS.sk, portál pre odborné publikovanie* [online]. 2012, roč. 5, č. 8 [cit. 2016-04-01]. ISSN 1338-0087.

BAROT, Tomáš. Vyšetření stability mnohorozměrových diskretních systémů v souvislosti s GPC prediktivním řízením. *POSTERUS.sk, portál pre odborné publikovanie* [online]. 2012, roč. 5, č. 8 [cit. 2016-04-01]. ISSN 1338-0087.

BAROT, Tomáš [80] a Marek KUBALČÍK [20]. Studium závislosti výpočetního času algoritmu GPC prediktivního řízení na volbě typu popisu matematického modelu v regulátoru. *POSTERUS.sk, portál pre odborné publikovanie* [online]. 2012, roč. 5, č. 6 [cit. 2016-04-01]. ISSN 1338-0087.

BAROT, Tomáš. Využití alternativních výpočetních prostředků ke zkvalitnění studijní průpravy studentů. *Trilobit* [online]. 2011, roč. 3, č. 1 [cit. 2016-04-01]. ISSN 1804-1795.

Software

KUBALČÍK, Marek [20] a Tomáš BAROT [80]. *Simulace a analýza prediktivního řízení z pohledu optimalizace* [software]. [cit. 2016-04-01]. URL: http://nod32.fai.utb.cz/Promotion/software_OBD/2012/barot_software.zip

Dolinay, Viliam [40] a Tomáš BAROT [60]. *Časový souhrn audiometrických vyšetření* [software]. [cit. 2016-04-01]. URL: http://web.fai.utb.cz/?id=0_5_2_8_1&lang=cs&type=0

ODBORNÝ ŽIVOTOPIS

Jméno a příjmení: Ing. Tomáš Barot

Rok, místo narození: 1985, Zlín, ČR

Email: barot@fai.utb.cz

Aktuální stav: Nezaměstnaný. Student doktorského studia na FAI UTB ve Zlíně, kombinovaná forma studia Ph.D.

Průběh vzdělání: Od 1.9.2014 – kombinovaná forma studia Ph.D., FAI UTB ve Zlíně, obor Automatické řízení a informatika.

2010 - 2014 – prezenční forma studia Ph.D., FAI UTB ve Zlíně, obor Automatické řízení a informatika (volba jiné oborové profilace na Ph.D. studiu kvůli snaze studovat více do hloubky statě matematiky v rámci teorie systémů)

2007 - 2010 – denní magisterské studium FAI UTB ve Zlíně, obor Informační technologie, dosažený titul Ing.

2004 - 2007 – denní bakalářské studium FAI UTB ve Zlíně, obor Informační technologie, dosažený titul Bc. (ocenění pana děkana za Bc. práci; červený diplom).

2000 - 2004 – Gymnázium Zlín, Lesní čtvrť.

Jazykové dovednosti: Angličtina (úroveň B2 ústní projev, C1 písemný), němčina (úroveň B2 ústní i písemný projev).

Kurz angličtiny – trimestr na jazykové škole Only4.

Odborné zájmy: Vědecká činnost (prediktivní řízení, teorie systémů, matematika, statistika, nelineární optimalizace, informatika). Edukační oblast (2010-2014) z pozice doktoranda.

Práce s počítačem a programování (C++, C, MATLAB, Mathematica, MS Office s programovací nadstavbou VBfA, PHP, MySQL, JavaScript).

Doktorské studium: Studijní program: P3902 Inženýrská informatika.
Studijní obor: 3902V037 Automatické řízení a informatika.

Školitel: pan doc. Ing. Marek Kubalčík, Ph.D.
Téma práce: *Prediktivní řízení procesů s rychlou dynamikou.*

Oponent Bc. práce pana Petra Víška, r. 2013.
Vedoucí Bc. práce pana Jakuba Hromady, r. 2013.

Tři oponentní posudky vypracované pro odborný recenzovaný časopis JTIE (Journal of Technology and Information Education), ISSN 1803-537X.

Interní grantové projekty UTB ve Zlíně, IGA v r. 2012, 2014, 2015.

Pedagogické zkušenosti: 337 hodin výuky prezenčních studentů
220 hodin výuky kombinované formy studia

Objektové programování AUPKS/AXOPR.
Dálkové studium, 80 studentů, 2.roč. oboru IŘT.
Výuka objektového programování v C++.
(2010/11, 2011/12, 2012/2013, 2013/2014)

Programování AUPKS/AQPRO.
Dálkové studium, 30 studentů, 1.roč. oboru BTSM.
Výuka programování v jazyku C. (2013/2014)

Programování AUPKS/A2PRG.
Denní studium, 22 studentů, 1.roč. oboru BTSM.
Výuka programování v jazyku C.
(2012/2013, 2013/2014)

Inženýrské výpočty TUVI/T2IV.
Denní studium, 13 studentů, 1.roč. FT UTB ve Zlíně
Programová podpora výpočtů algebry a vícerozměrné matematické analýzy. (2013/2014)

Základy počítačové techniky AURP/AQZPT.
Dálkové studium, 93 studentů, 1.roč. oboru BTSM.
Výuka MS Word, Excel, PowerPoint, HTML.
(2010/11, 2011/12, 2012/2013, 2013/2014)

Základy počítačové techniky AURP/A1ZPT.
Denní studium, 57 studentů, 1.roč. oboru BTSM.
Výuka MS Word, Excel, PowerPoint, HTML.
(2010/11, 2011/12, 2012/2013)

Rodinný stav: Svobodný.

Pracovní a odborné cíle: **Práce v edukační oblasti. Publikace výzkumu v odborných časopisech spjatých s kvalifikací Ph.D., a to s reflektováním hodnocení tvůrčích činností dle metodiky Rady vlády pro vědu, výzkum a inovace. Rozvíjet se dále v edukační oblasti - zahájit doplňující studium pedagogiky.**

Ve Zlíně, dne 15. června 2016

Ing. Tomáš Barot

PŘÍLOHA A

A.1 VERIFIKAČNÍ ALGORITMUS PRO OPT. METODY

Úryvek programového kódu (řádek 36 – 69) algoritmu analytického postupu řešení úlohy kvadratického programování v programu MATLAB (soubor *Verifikacni_analyticka_metoda.m*, přílohy B):

```
% reseni soustavy rovnic pro nalezeni stacionarnich bodu s cilem najit
% vazane optimum x
for i=2^m-1:-1:0
    % vytvoreni nulovych a nenulovych prvku ve vektoru vy
    prevod=i; j=m; vy=zeros(m,1);
    while(prevod~=0)
        zbytek=mod(prevod,2); vy(j)=zbytek;
        prevod=floor(prevod/2); j=j-1;
    end;

    % reseni soustavy rovnic pro urceni stacionarniho bodu
    L=[H zeros(n,m) (M')*diag(~vy);...
        M diag(vy) zeros(m,m);...
        zeros(m,n) diag(~vy) diag(vy)...
    ];
    P=[-b'; gamma; zeros(m,1)];
    reseni=(L\P);
    x=reseni(1:n,1);
    lambda=reseni(n+m+1:n+2*m,1);

    % kontrola splneni Kuhn-Tuckerovych podminek pro stacionarni bod
    if (round(x'*H+b+lambda*M)==zeros(1,n))
        if (M*x<=gamma)
            if(lambda>=zeros(m,1))
                if(round(lambda*(M*x-gamma))==0)
                    % Vazany extrem minimum x
                    optimum=x;
                    lagrMultiplikatory=lambda;
                    return;
                end;
            end;
        end;
    end;
end;
end;
end;
end;
```

A.2 OVĚŘENÍ SPRÁVNOSTI VERIFIKAČNÍHO ALGORITMU

Úryvek programového kódu (řádek 19 – 43) algoritmu postupu ověření analytického řešení úlohy kvadratického programování v programu MATLAB (soubor *Overeni_analyticke_metody.m*, přílohy B):

```
verifikace=1;
presnost=0.001;

for n=3:1:20
    H=ones(n,n);
    b=ones(n,1);
    gama=ones(n,1);
    M=eye(n,n);
    pom=eye(n,n);
    for i=1:n % sestaveni matic/vektoru pro verifikaci
        pom(i,i)=i*2;
        b(i)=i;
        gama(i)=0.5;
    end;
    H=pom+H;

    % Provedeni optimalizace - ulohy kvadratickeho programovani
    x_QuadProg=quadprog(H,b',M,gama);
    [L,x_AnalytMet]=Verifikacni_analyticka_metoda(H,b',M,gama);

    % Detekce chybného výpočtu k dané přesnosti výsledku
    if (abs(x_QuadProg-x_AnalytMet)>presnost) % provede se nad všemi prvky
        verifikace=0;
    end;
end;
```

A.3 IMPLEMENTACE HILDRETHOVY METODY

Úryvek programového kódu (řádek 24 – 49) algoritmu zavedené Hildrethovy metody pro řešení úlohy kvadratického programování v programu MATLAB (soubor *Hildrethova_metoda.m*, přílohy B) se znázorněním jádra cyklu řešící jednotlivé iterace této numerické metody:

```
for k=1:max_pocet_iteraci
    % Pamet vektoru predchozi iterace d(k-1)=d(k)
    dk1=dk; flopy=flopy+m;

    % Urceni noveho vektoru d(k) pro k-tou iteraci,
    % vypocet celkem m prvku vektoru d(k)
    for i=1:m
        if (i==1)
            dk(i)=max(-(1/N(i,i))*(N(i,2:m)*dk1(2:m,1)+o(i,1)),0); flopy=flopy+2*m+5;
        end;
        if (i>1)&&(i<m)
            dk(i)=max(-(1/N(i,i))*(N(i,1:(i-1))*dk1(1:(i-1),1)+...
                N(i,(i+1):(m-1))*dk1((i+1):(m-1),1)+o(i,1)),0); flopy=flopy+2*i*m-6*i+5;
        end;
        if (i==m)
            dk(i)=max(-(1/N(i,i))*(N(i,1:(m-1))*dk1(1:(m-1),1)+o(i,1)),0); flopy=flopy+2*m+3;
        end;
    end;

    % Pri shode aktualniho a predchoziho vysledku je vypocet
    % uspesne ukoncen
    if (dk==dk1)
        flopy=flopy+m;
        break;
    end;
end;
```

A.4 KÓD WANGOVOY MODIFIKACE

Programový kód algoritmu implementující zavedenou Wangovu modifikaci v programu MATLAB (soubor *Wangova_modifikace .m*, přílohy B):

```
function [flopy, optimum, podminky_splneny]=Wangova_modifikace(H,b,M,gamma)

% Mereni vypocetni slozitosti ve flopech
flopy=0;

% m podminek omezeni, n rozmerny definicni obor ucelove funkce
[m,n]=size(M); flopy=flopy+m*n;

% Inicializace vysledku ulohy na optimum rovne volnemu extremu
optimum=-inv(H)*b'; flopy=flopy+0.3*n*n*n+0.6*n*2*n*n-n;

% Test, uspesnosti Wangovy modifikace prevodu puvodni ulohy na ulohu volneho extremu
podminky_splneny=1; flopy=flopy+1;
for i=1:m
    if M(i,:)*optimum>gamma(i) % Vyjadreni nesplneni puvodnich podminek omezeni
        flopy=flopy+n;
        podminky_splneny=0; flopy=flopy+1;
    end;
end;
if podminky_splneny
    flopy=flopy+1;
    return; % Je vracen vypocitany volny extrem
end;
```

A.5 KÓD NAVRHOVANÉ OPTIMALIZAČNÍ METODY

Úryvek programového kódu (řádek 56 – 81) algoritmu navrhované optimalizační metody řešící úlohu kvadratického programování v programu MATLAB (soubor *Navrhovana_dualni_metoda.m*, přílohy B):

```
% Dokud jsou obsazeny zaporne prvky ve vektoru d - opakovane reseni
% upraveno ulohy na volny extrem (sekvence modif. uloh na volny extrem)
while ((size(indexy_splnuji,2)+pocet_nulovych)<m)
    % Vypocet dane redukovane ulohy na volny extrem
    % s upravenou redukovanou matici N a vektorem o
    % (zustaly prvky, jejichz indexy radku nebo sloupce odpovidaji
    % indexu ve vektoru d, kde jsou nezaporne prvky)

    % Vysledek redukovane ulohy bude mit na danyh indexech hodnotu 0
    d=zeros(m,1); flopy=flopy+m;
    d(indexy_splnuji,:)=inv(N(indexy_splnuji,indexy_splnuji))*o(indexy_splnuji,1);
    [fl,f1 1]=size(indexy_splnuji);
    flopy=flopy+2*fl*fl-fl+0.3*fl*fl*fl+0.6*fl*fl;

    % Kontrola kladnych nebo nulovych prvku vektoru d
    indexy_splnuji=[];
    for i=1:m
        if d(i,1)>0
            % Zapamatovani poradnice prvku v ramci vektoru d s nezapornym prvkem
            indexy_splnuji=[indexy_splnuji i]; flopy=flopy+1;
        end;
        if d(i,1)==0
            pocet_nulovych=pocet_nulovych+1; flopy=flopy+1;
        end;
    end;
end;
```

A.6 KÓD VERIFIKACE OPTIMALIZAČNÍCH METOD

Úryvek programového kódu (řádek 12 – 39) algoritmu implementovaných optimalizačních metod v programu MATLAB (soubor *Overeni_opt_metod.m*, přílohy B):

```
for n=3:1:18
    H=ones(n,n);
    b=ones(n,1);
    gama=ones(n,1);
    M=eye(n,n);
    pom=eye(n,n);
    for i=1:n % sestaveni matic/vektoru pro verifikaci
        pom(i,i)=i*3;
        b(i)=i;
        gama(i)=0.5;
    end;
    H=pom+H;

    % Provedeni optimalizace - ulohy kvadratickeho programovani
    [L,x_AnalytMet]=Verifikacni_analyticka_metoda(H,b',M,gama);
    [fl,x_HildirMet]=Hildrethova_metoda(H,b',M,gama);
    [fl,podm_reg,lagr_m,x_NavrhMet]=Navrhovana_dualni_metoda(H,b',M,gama);

    % Detekce chybného výpočtu k dané přesnosti výsledku
    if (abs(x_HildirMet-x_AnalytMet)>presnost)
        verifikace=0;
        break;
    end;
    if (abs(x_NavrhMet-x_AnalytMet)>presnost)
        verifikace=0;
        break;
    end;
end;
```


A.7 KÓD POSOUZENÍ EFEKTIVITY OPT. METOD

Programový kód porovnání výpočetní náročnosti algoritmů implementujících řešení úlohy kvadratického programování v programu MATLAB (soubor *Porovnaní_opt_metod.m*, přílohy B):

```
% zalozeni souboru pro export dat mereni flopu jednotlivych metod
soubor=fopen('Porovnaní_opt_met.csv','w');
fprintf(soubor,'n; Flopy Hildr.; Flopy Navrh.met.\n');

for n=3:3:30
    H=ones(n,n);
    b=ones(n,1);
    gama=ones(n,1);
    M=eye(n,n);
    pom=eye(n,n);
    for i=1:n % sestaveni matic/vektoru
        pom(i,i)=i;
        b(i)=i;
        gama(i)=1;
    end;
    H=pom+H;

    % Provedeni optimalizace - ulohy kvadratickeho programovani
    [flopyHildr,x_HildrMet]=Hildrethova_metoda(H,b',M,gama);

    [flopyNavrhMet,podm_reg,lagr_m,x_NavrhMet]=Navrhovana_dualni_metoda(H,b',M,gama);

    % Nebot vzorce určení flopů dle zdroje [30] zahrnují zlomky, provede se
    % zaokrouhlení výsledku na celá čísla.
    flopyHildr=ceil(flopyHildr);
    flopyNavrhMet=ceil(flopyNavrhMet);

    % export dat do CSV souboru
    fprintf(soubor,num2str(n));
    fprintf(soubor,','');
    fprintf(soubor,num2str(flopyHildr));
    fprintf(soubor,','');
    fprintf(soubor,num2str(flopyNavrhMet));
    fprintf(soubor,'\n');
end;
% ukončení zápisu dat do souboru
fclose(soubor);
```

PŘÍLOHA B – STRUKTURA PŘILOŽENÉHO CD

- Disertacni_prace.pdf* Disertační práce.
- /Software** Software disertační práce s obsahem:
- /grafika* Grafické soubory pro grafické uživatelské rozhraní.
 - /procesy* Uložitelná konfigurace jednotlivých modelů procesů:
 $SISO_T_0.05.Gz1R... G_1(z^{-1}), T = 0,05 s$,
 $SISO_T_0.03.Gz1R... G_2(z^{-1}), T = 0,03 s$,
 $TITO_T_0.05.Gz2R... G_1(z^{-1}), T = 0,05 s$,
 $TITO_T_0.03.Gz2R... G_2(z^{-1}), T = 0,03 s$
 - /pracovni_prostory* Konfigurace analýzy MPC s různým počtem omezení:
 $mereni_SISO_0_05s_1om.gpc... G_1(z^{-1}), T = 0,05 s$,
 $mereni_SISO_0_03s_1om.gpc... G_2(z^{-1}), T = 0,03 s$,
 $mereni_TITO_0_05s_1om.gpc... G_1(z^{-1}), T = 0,05 s$,
 $mereni_TITO_0_03s_1om.gpc... G_2(z^{-1}), T = 0,03 s$,
 $mereni_SISO_0_05s_2om.gpc... G_1(z^{-1}), T = 0,05 s$,
 $mereni_SISO_0_03s_2om.gpc... G_2(z^{-1}), T = 0,03 s$,
 $mereni_TITO_0_05s_2om.gpc... G_1(z^{-1}), T = 0,05 s$,
 $mereni_TITO_0_03s_2om.gpc... G_2(z^{-1}), T = 0,03 s$
 - /grafika* Grafické soubory pro grafické uživatelské rozhraní.
- programPrediktivniRizeni.m* Hlavní soubor skriptu softwaru s GUI.
- programPrediktivniRizeni_info.m* Kód pro zobrazení informací o softwaru.
- Generovani_sumu.m* Skript pro jednotné generování šumu pro MPC.
- Hildrethova_metoda.m* Implementace Hildrethovy optimalizační metody.
- Navrhovana_dualni_metoda.m* Implementace navrhované opt. metody.

Wangova_modifikace.m Implementace zavedené Wangovy modifikace.

Verifikacni_analyticka_metoda.m Implementace analytického řešení úlohy kvadratického programování.

Overeni_analyticke_metody.m Ověření správnosti analytické metody pro řešení kvadratického programování.

Overeni_opt_metod.m Ověření správnosti implementace opt. metod.

Porovnani_opt_metod.m Porovnání opt. metod z pohledu výpočetní náročnosti.

programPrediktivniRizeni.fig Konfigurace GUI rozhraní.

programPrediktivniRizeni_info.fig Konfigurace GUI rozhraní pro zobrazení informací o softwaru.

Casy_kriteria.csv CSV soubor s výsledky analýzy algoritmu MPC.

Export.csv CSV soubor s exportem průběhů veličin MPC.

Sum.csv CSV soubor s hodnotami generovaného šumu.

/Zpracovani_mereni Analýza výpočetní náročnosti implementace MPC:

Zpracovani_SISO.xls Analýza MPC pro SISO modely procesů.

Zpracovani_TITO.xls Analýza MPC pro TITO modely procesů.

Porovnani_opt_metod.xls Porovnání výpočetní náročnosti opt. metod.

/Analyza_postupu Měření výpočetní náročnosti daných postupů řešení:

16 souborů typu CSV pojmenovaných dle pravidla:

[SISO]_[0.05]_[1]om_[HW].csv
⋮ ⋮ ⋮ ⋮
[TITO] [0.03] [2] [HWE]
 ⋮
 [NWE]

Kde: *HW* - Hildr. met. + Wangova modifikace,
HWE - *HW* + navrhovaný princip eliminací
NWE - navrhovaná opt. metoda + Wangova

zavedená modifikace + návrhov. princip
eliminací

Např. *TITO_0.03_2om_HWE.csv*

/Simulace_MPC

Průběhy veličin MPC v rámci simulací jednotlivých
jeho nastavení:

8 souborů typu CSV pojmenovaných dle pravidla:

sim_[SISO]_[0.05]_[1]om_[N_u].csv

⋮ ⋮ ⋮

[*TITO*] [0.03] [2]

Kde: N_u - uvažovaná hodnota řídicího horizontu

Např. *sim_SISO_0.05_2om_N5.csv*