

Návrh prediktívnych algoritmov riadenia

(Model: *M6 Hydraulický systém – dve nádoby v interakcii*)

Cieľom návrhu prediktívnych algoritmov riadenia bude:

1. zabezpečiť stabilizáciu regulovanej veličiny na ustálenej hodnote v okolí pracovného bodu (výška hladiny v druhej nádobe $h_3(t)$)
2. zabezpečiť stabilizáciu regulovanej veličiny pri zmene ustálených hodnôt v okolí pracovného bodu (výška hladiny v druhej nádobe $h_3(t)$)

Úlohy:

1. Zostavte matematicko – fyzikálny model nelineárneho dynamického systému *M6* s využitím analytickej identifikácie
 2. Získajte lineárny model *Hydraulického systému M6* v stavovom opise vo vhodne zvolenom pracovnom bode
 3. Navrhňte algoritmus zovšeobecneného prediktívneho riadenia (GPC)
 4. Navrhňte algoritmus stavového prediktívneho riadenia (MPC)
 5. Simulačne overte algoritmy prediktívneho riadenia v prislúchajúcich spätnoväzobných riadiacích štruktúrach:
 - 5.a simulačné overenia prediktívnych algoritmov riadenia do ustáleného stavu na nelineárnom modeli
 - 5.b simulačné overenie prediktívnych algoritmov riadenia do ustáleného stavu s pôsobením poruchy na nelineárnom modeli
 - 5.c simulačné overenie prediktívnych algoritmov riadenia do rôznych ustálených stavov na nelineárnom modeli
-

Úloha č.1 Zostavte matematicko – fyzikálny model nelineárneho dynamického systému *M6* s využitím analytickej identifikácie

Odvodenie matematického popisu modelu *M6* s využitím postupu analytickej identifikácie sa nachádza na stránke predmetu *Optimálne a nelineárne systémy* v časti *Simulačné modely*.

http://matlab.fei.tuke.sk/ons/pdfModely/M6_dveNadobyInter.pdf

Úloha č.2 Získajte lineárny model Hydraulického systému *M6* v stavovom opise vo vhodne zvolenom pracovnom bode.

Postup pre získanie lineárneho modelu *M6* v stavovom opise ako aj vhodný pracovný bod PB (*pracovný bod A*) sa nachádzajú na stránke predmetu *Optimálne a nelineárne systémy* v časti *Simulačné modely*. http://matlab.fei.tuke.sk/ons/pdfModely/M6_dveNadobyInter.pdf

Úloha č.3 Navrhnete algoritmus zovšeobecneného prediktívneho riadenia (GPC)

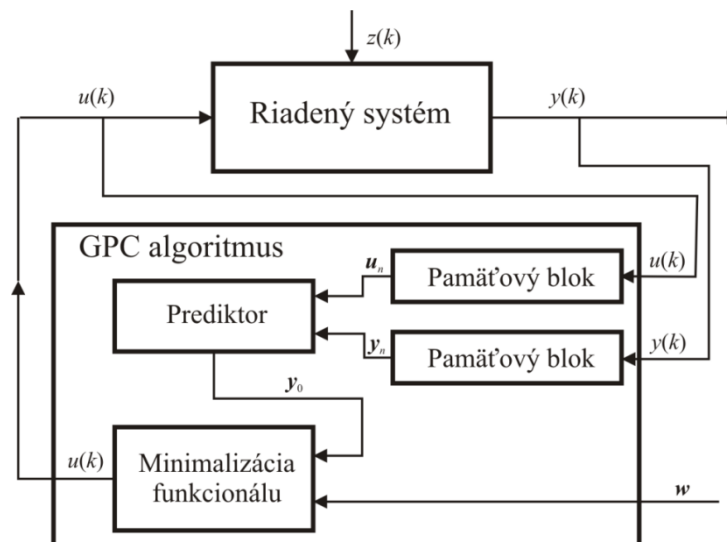
Pri návrhu algoritmu GPC riadenia vychádzame z vstupno-výstupného opisu lineárneho modelu SISO dynamického systému M6 vo forme CARIMA modelu:

$$A_z(z^{-1})y(k) = B_z(z^{-1})u(k-1) + \frac{C_z(z^{-1})}{\Delta}\zeta(k) \quad (1)$$

kde $A_z(z^{-1})$ je polynómom menovateľa diskkrétnej prenosovej funkcie (1) $B_z(z^{-1})$ je polynómom čitateľa diskkrétnej prenosovej funkcie (1), $C_z(z^{-1})$ je polynóm, $\zeta(k)$ je chyba merania výstupu systému a $\Delta = 1 - z^{-1}$ predstavuje integrátor.

Riadiaca štruktúra s GPC algoritmom pre SISO systém je znázornená na

Obr. 1:



Obr. 1 Riadiaca štruktúra zovšeobecneného prediktívneho riadenia

Postup návrhu algoritmu zovšeobecneného prediktívneho riadenia je daný v nasledujúcich krokoch:

1. GPC algoritmus založený na CARIMA modeli, minimalizuje funkcionál v tvare:

$$J_{GPC} = \sum_{i=N_1}^{N_p} [\hat{y}(k+i) - w(k+i)]^2 + \lambda \sum_{i=1}^{N_u} [\Delta u(k+i-1)]^2 \quad (2)$$

kde $\hat{y}(k)$ je predikovaná hodnota regulovanej veličiny, $\Delta u(k)$ je zmena akčného zásahu λ je pomerný váhový koeficient vyjadrujúci pomer váh medzi regulačnou odchýlkou a akčným zásahom.

2. Majme odvodený prediktor $\hat{\mathbf{y}}$ v tvare:

$$\hat{\mathbf{y}} = \mathbf{G}_{\text{PCH}}\Delta\mathbf{u} + \mathbf{y}_0 \quad , \quad (3)$$

kde je potrebné v každom kroku k vypočítať voľnú odozvu systému \mathbf{y}_0 na horizonte predikcie N_p a taktiež vypočítať maticu \mathbf{G} , ktorá je tvorená koeficientami prechodovej charakteristiky,

$$\mathbf{G}_{\text{PCH}} = \begin{bmatrix} g_{\text{PCH}0} & 0 & \dots & \dots & \dots & 0 \\ g_{\text{PCH}1} & g_{\text{PCH}0} & 0 & \dots & \dots & 0 \\ \vdots & & \dots & & & \\ \vdots & & & \dots & & \\ \vdots & & & & g_{\text{PCH}0} & 0 \\ g_{\text{PCH}N_p-1} & \dots & \dots & \dots & \dots & g_{\text{PCH}0} \end{bmatrix} . \quad (4)$$

3. Dosadením prediktora (3) do funkcionálu (2) môžeme získať kvadratickú formu

$$\begin{aligned} J_{\text{GPC}} &= (\hat{\mathbf{y}} - \mathbf{w})^T (\hat{\mathbf{y}} - \mathbf{w}) + \lambda \Delta\mathbf{u}^T \Delta\mathbf{u} = \\ &= (\mathbf{G}_{\text{PCH}}\Delta\mathbf{u} + \mathbf{y}_0 - \mathbf{w})^T (\mathbf{G}_{\text{PCH}}\Delta\mathbf{u} + \mathbf{y}_0 - \mathbf{w}) + \lambda \Delta\mathbf{u}^T \Delta\mathbf{u} = \\ &= c + 2\mathbf{g}^T \Delta\mathbf{u} + \Delta\mathbf{u}^T \mathbf{H} \Delta\mathbf{u}, \end{aligned} \quad (5)$$

kde \mathbf{g} je gradient, \mathbf{H} je Hessian a c je konštanta.

Platí:

$$\mathbf{g} = \mathbf{G}_{\text{PCH}}^T (\mathbf{y}_0 - \mathbf{w}), \quad \mathbf{H} = \mathbf{G}_{\text{PCH}}^T \mathbf{G}_{\text{PCH}} + \lambda \mathbf{I} . \quad (\mathbf{I} \text{ je jednotková matica})$$

4. Pre výpočet optimálneho akčného zásahu na dĺžke riadiaceho horizontu N_u analytickým spôsobom platí:

$$\Delta\mathbf{u} = -\mathbf{H}^{-1} \mathbf{g}, \quad (6)$$

alebo v prípade požiadavky zakomponovania ohraničení fyzikálnych veličín do výpočtu optimálneho akčného zásahu je na tento účel vhodná funkcia *quadprog*, ktorá je súčasťou *Optimization Toolboxu* v prostredí Matlab a ktorej tvar vyzerá takto:

$$\mathbf{du} = \text{quadprog}(\mathbf{H}, \mathbf{g}, \mathbf{A_cela}, \mathbf{C_cela}, [], [], \mathbf{dumin}, \mathbf{dumax}, [], \mathbf{opts}) \quad (7)$$

kde A_{cela} a C_{cela} sú matice, ktoré obsahujú fyzikálne obmedzenia systému, $dumin$ a $dumax$ reprezentujú ohraničenie zmeny akčného zásahu a $opts$ zahrňuje inicializačné nastavenia.

Vývojový diagram algoritmu zovšeobecného prediktívneho riadenia v kroku k znázorňuje Obr. 2.



Obr. 2 Vývojový diagram algoritmu zovšeobecného prediktívneho riadenia

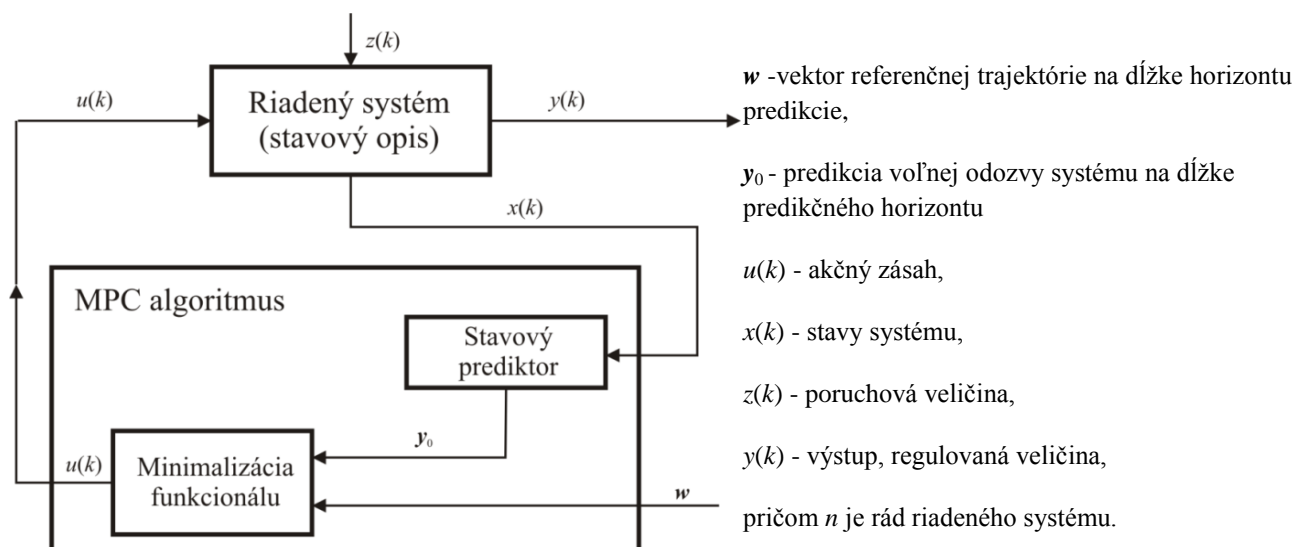
Úloha č.4 Navrhnete algoritmus stavového prediktívneho riadenia (MPC)

Pri návrhu algoritmu stavového prediktívneho riadenia (MPC) vychádzame zo stavového popisu diskretného systému (8)

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{F}\mathbf{x}(k) + \mathbf{G}u(k) \\ y(k) &= \mathbf{C}\mathbf{x}(k) \end{aligned} \quad (8)$$

kde matice \mathbf{F} , \mathbf{G} sú získané diskretizáciou systému s periódou vzorkovania T_s .

Riadiaca štruktúra s MPC algoritmom pre SISO systém je znázornená na Obr. 3.



Obr. 3 Riadiaca štruktúra stavového prediktívneho riadenia

Postup návrhu algoritmu stavového prediktívneho riadenia je uvedený v nasledujúcich krokoch:

1. Majme funkcionál

$$J_{MPC} = \sum_{i=1}^{N_p} \mathbf{Q}[\hat{y}(k+i) - w(k+i)]^2 + \sum_{i=1}^{N_u} \mathbf{R}[u(k+i-1)]^2, \quad (9)$$

ktorý je minimalizovaný riadiacou postupnosťou $u(k)$ na horizonte predikcie N_p , pričom \mathbf{Q} a \mathbf{R} sú váhové koeficienty.

Zvyčajne sa v kritériu (9) neváži priamo akčná veličina $u(k)$, ale jej zmena $\Delta u(k) = u(k) - u(k-1)$. Kritérium prediktívneho riadenia má potom tvar

$$J_{MPC} = \sum_{i=1}^{N_p} \mathbf{Q}[\hat{y}(k+i) - w(k+i)]^2 + \sum_{i=1}^{N_u} \mathbf{R}[\Delta u(k+i-1)]^2. \quad (10)$$

kde

$$\Delta \mathbf{u}_k = \tilde{\mathbf{I}} \mathbf{u}_k - \tilde{\mathbf{u}}_k,$$

$$\tilde{\mathbf{I}} = \begin{bmatrix} 1 & & & & & & \\ -1 & 1 & & & & & \\ & \cdot & \cdot & & & & \\ & & \cdot & \cdot & & & \\ & & & \cdot & \cdot & & \\ & & & & \cdot & \cdot & \\ & & & & & -1 & 1 \end{bmatrix}, \quad \tilde{\mathbf{u}}_k = \begin{bmatrix} u(k-1) \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix}.$$

2. Postupným iterovaním rovníc diskrétného stavového opisu (8) pre ďalšie kroky je možné jednoducho odvodiť prediktor v tvare

$$\hat{\mathbf{y}} = \mathbf{V} \mathbf{x}(k) + \mathbf{S} \mathbf{u}_k, \quad (11)$$

kde je potrebné vypočítať odozvu systému \mathbf{V} na počiatočnú podmienku $\mathbf{x}(k)$ a odozvu systému \mathbf{S} na postupnosť riadenia \mathbf{u}_k na horizonte predikcie N_p a platí

$$\hat{\mathbf{y}} = \begin{bmatrix} y(k) \\ y(k+1) \\ y(k+2) \\ \cdot \\ \cdot \\ \cdot \\ y(k+N_p-1) \end{bmatrix}, \quad \mathbf{u}_k = \begin{bmatrix} u(k) \\ u(k+1) \\ u(k+2) \\ \cdot \\ \cdot \\ \cdot \\ u(k+N_p-1) \end{bmatrix}, \quad (12)$$

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CF}_B \\ \mathbf{CF}^2 \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{CF}^{N_p-1} \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 0 & & & & & & \\ \mathbf{CG} & 0 & & & & & \\ \mathbf{CFG} & \mathbf{CG} & 0 & & & & \\ \cdot & \cdot & \cdot & & & & \\ \cdot & \cdot & \cdot & \cdot & & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & & \\ \mathbf{CF}^{N_p-2} \mathbf{G} & & & & \mathbf{CG} & 0 & \end{bmatrix}. \quad (13)$$

Pri numerickom výpočte môžeme zahrnúť obmedzujúce podmienky pre jednotlivé veličiny. Najčastejšie používané je obmedzenie na amplitúdu akčného zásahu alebo zmeny akčnej veličiny

$$\mathbf{u}_{\min} \leq \mathbf{u}(k) \leq \mathbf{u}_{\max} \quad \text{resp.} \quad \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}(k) \leq \Delta \mathbf{u}_{\max}.$$

3. Vypočet Hessianu \mathbf{H} a gradientu \mathbf{g}

a. pre stavový MPC algoritmus vážiaci akčný zásah

- Pre úlohu kvadratického programovania minimalizujúcu kritérium (9) dostávame Hessian \mathbf{H} a gradient \mathbf{g} v tvare:

$$\begin{aligned}\mathbf{H} &= \mathbf{S}^T \mathbf{Q} \mathbf{S} + \mathbf{R}, \\ \mathbf{g} &= (\mathbf{V} \mathbf{x}(k) - \mathbf{w}_k)^T \mathbf{Q} \mathbf{S}\end{aligned}$$

b. pre stavový MPC algoritmus vážiaci zmenu akčného zásahu

- Pre úlohu kvadratického programovania minimalizujúcu kritérium (10) dostávame Hessian \mathbf{H} a gradient \mathbf{g} v tvare:

$$\begin{aligned}\mathbf{H} &= \mathbf{S}^T \mathbf{Q} \mathbf{S} + \tilde{\mathbf{I}}^T \mathbf{R} \tilde{\mathbf{I}}, \\ \mathbf{g} &= (\mathbf{V} \mathbf{x}(k) - \mathbf{w}_k)^T \mathbf{Q} \mathbf{S} + \tilde{\mathbf{u}}_k^T \mathbf{R} \tilde{\mathbf{I}}\end{aligned}$$

c. pre stavový MPC algoritmus s diskretným integrátorom vážiaci akčný zásah

- Do algoritmu stavového MPC riadenia zavedieme sumátor regulačnej odchýlky popísaný rovnicou

$$\begin{aligned}\mathbf{x}_{se}(k+1) &= \mathbf{x}_{se}(k) + \mathbf{w}(k) - \mathbf{y}(k), \\ y_{se}(k) &= \mathbf{x}_{se}(k)\end{aligned}\tag{14}$$

a rozšírime kritérium (9) na tvar

$$J_{MPC} = \sum_{i=1}^{N_p} \mathbf{Q} [\hat{y}(k+i) - w(k+i)]^2 + \sum_{i=1}^{N_p} \mathbf{Q}_{se} [y_{se}(i)]^2 + \sum_{i=1}^{N_u} \mathbf{R} [u(k+i-1)]^2.\tag{15}$$

- Pre minimalizáciu kritéria (15) dostávame vzťahy pre Hessian \mathbf{H} a gradient \mathbf{g} v tvare

$$\begin{aligned}\mathbf{H} &= \mathbf{S}^T \mathbf{Q} \mathbf{S} + \mathbf{R} + \mathbf{S}^T \mathbf{I}_v^T \mathbf{Q}_{se} \mathbf{I}_v \mathbf{S}, \\ \mathbf{g} &= (\mathbf{V} \mathbf{x}(k) - \mathbf{w}_k)^T \mathbf{Q} \mathbf{S} + (\hat{\mathbf{I}} \mathbf{x}_{se} + \mathbf{I}_v (\mathbf{w}_k - \mathbf{V} \mathbf{x}(k)))^T \mathbf{Q}_{se} \mathbf{I}_v \mathbf{S}\end{aligned}$$

kde

$$\mathbf{I}_v = \begin{bmatrix} 0 & & & & & \\ 1 & 0 & & & & \\ \cdot & 1 & \cdot & & & \\ \cdot & \cdot & \cdot & \cdot & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & \\ 1 & 1 & \cdot & \cdot & 1 & 0 \end{bmatrix}, \quad \hat{\mathbf{I}} = \begin{bmatrix} 1 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix}.$$

d. pre stavový MPC algoritmus s diskrétnym integrátorom vážiaci zmenu akčného zásahu

- Rovnaký postup realizujeme aj pre stavový MPC algoritmus riadenia vážiaci zmenu akčného zásahu. Rozšírime kritérium (10) o sumátor regulačnej odchýlky (14). Po roznásobení a úpravách dostávame výsledné vzťahy pre Hessian \mathbf{H} a gradient \mathbf{g} v tvare

$$\mathbf{H} = \mathbf{S}^T \mathbf{Q} \mathbf{S} + \tilde{\mathbf{I}}^T \mathbf{R} \tilde{\mathbf{I}} + \mathbf{S}^T \mathbf{I}_v^T \mathbf{Q}_{se} \mathbf{I}_v \mathbf{S},$$

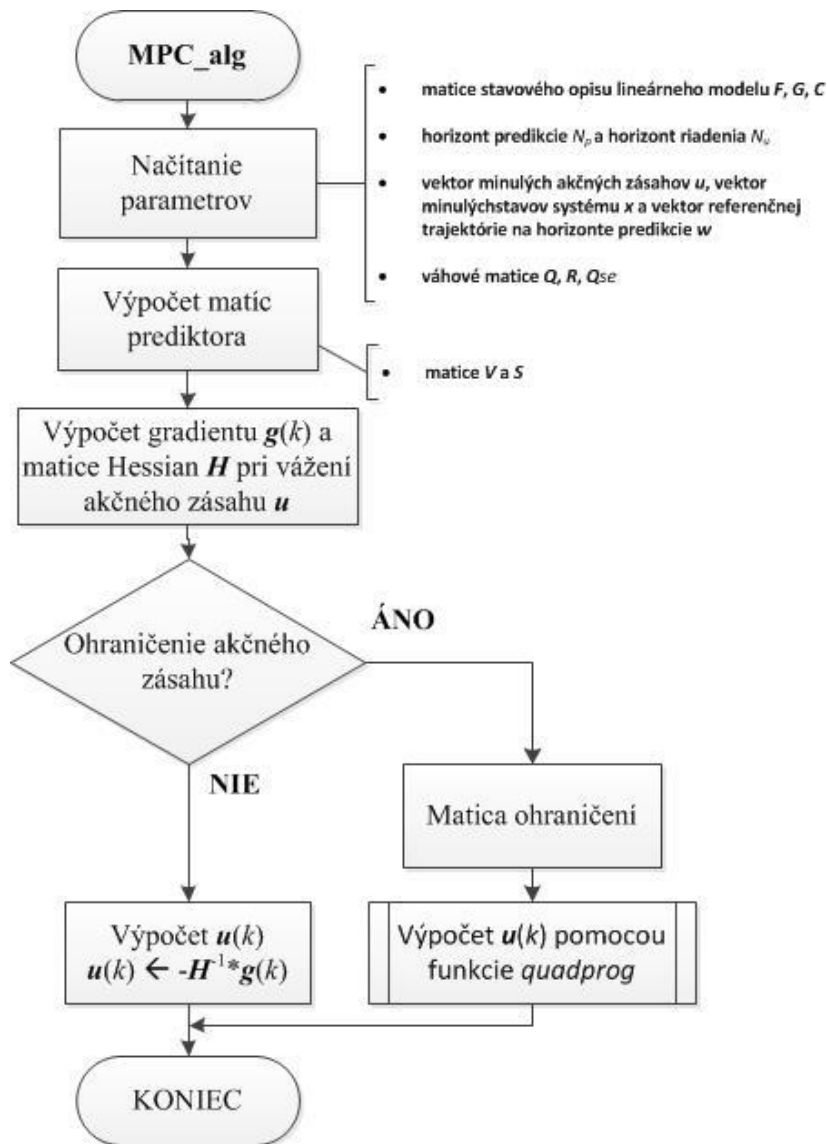
$$\mathbf{g} = (\mathbf{V} \mathbf{x}(k) - \mathbf{w}_k)^T \mathbf{Q} \mathbf{S} + (\hat{\mathbf{I}}_{se} + \mathbf{I}_v (\mathbf{w}_k - \mathbf{V} \mathbf{x}(k)))^T \mathbf{Q}_{se} \mathbf{I}_v \mathbf{S} + \mathbf{u}_k^T \mathbf{R} \tilde{\mathbf{I}}$$

4. Rovnako ako pri algoritme GPC riadenia aj v tomto prípade pre výpočet optimálneho akčného zásahu na dĺžke riadiaceho horizontu N_u analytickým spôsobom platí vzťah (6) alebo v prípade požiadavky zakomponovania ohraničení fyzikálnych veličín do výpočtu optimálneho akčného zásahu je na tento účel vhodná funkcia *quadprog*, ktorej tvar vyzerá takto:

$$\mathbf{u} = \text{quadprog}(\mathbf{H}, \mathbf{f}^T, [], [], [], [], \mathbf{Umin}, \mathbf{Umax}, [], \text{opts}), \quad (16)$$

kde \mathbf{Umin} a \mathbf{Umax} reprezentujú ohraničenie akčného zásahu a *opts* zahrňuje inicializačné nastavenia.

Vývojový diagram výpočtu akčného zásahu v kroku k pomocou stavového MPC algoritmu riadenia vážiaceho akčný zásah je znázornený na Obr. 4. Pre ostatné tri modifikácie váženia akčného zásahu bude vývojový diagram (Obr. 4) obmenený o výpočet Hessianu \mathbf{H} a gradientu \mathbf{g} .

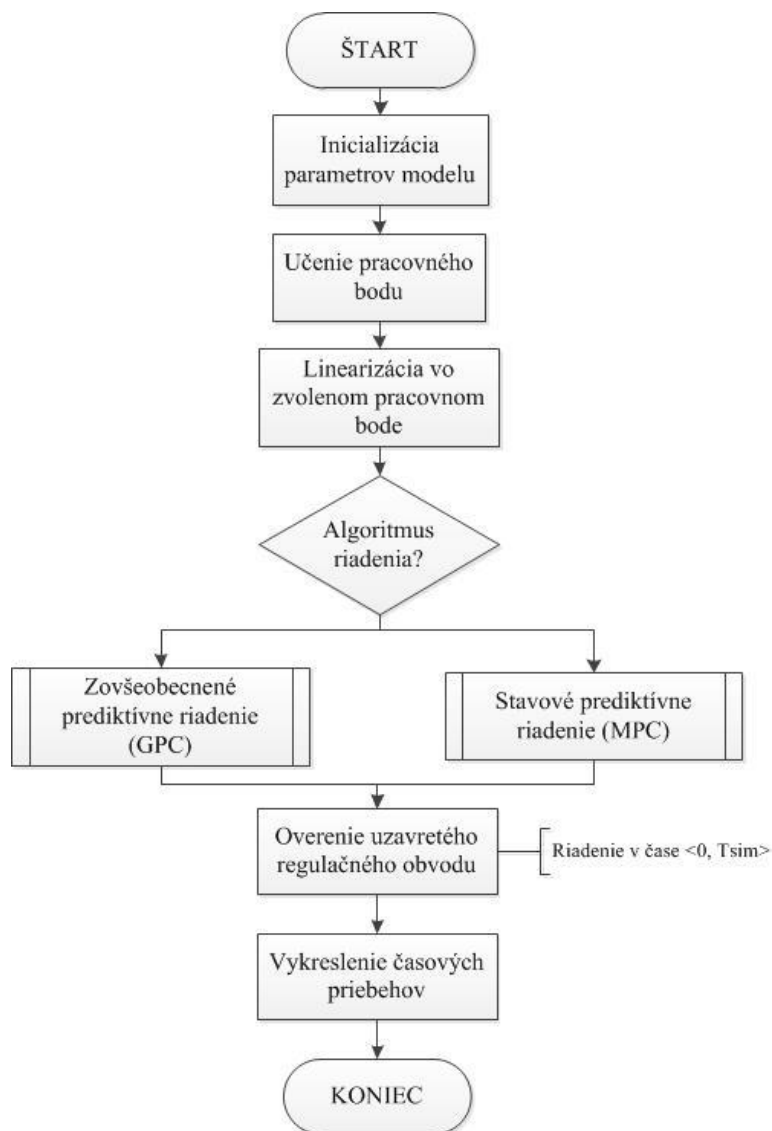


Obr. 4 Vývojový diagram stavového MPC algoritmu riadenia pre výpočet akčného zásahu v kroku k pri vážení akčného zásahu

Úloha č.5 Simulačne overte algoritmy prediktívneho riadenia v prislúchajúcich spätnoväzobných riadiacich štruktúrach

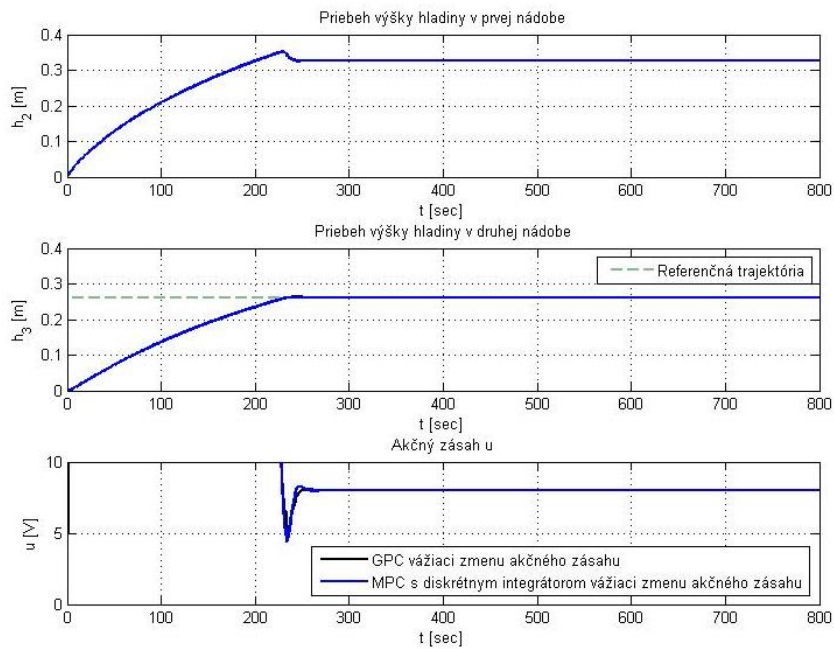
Pri overení algoritmu zovšeobecneného prediktívneho riadenia (GPC) vychádzame z riadiacej štruktúry znázornenej na Obr. 1 a pri overení stavového prediktívneho riadenia (MPC) z riadiacej štruktúry na Obr. 3.

Vývojový diagram zobrazený nižšie zobrazuje algoritmus pre overenie algoritmov prediktívneho riadenia.



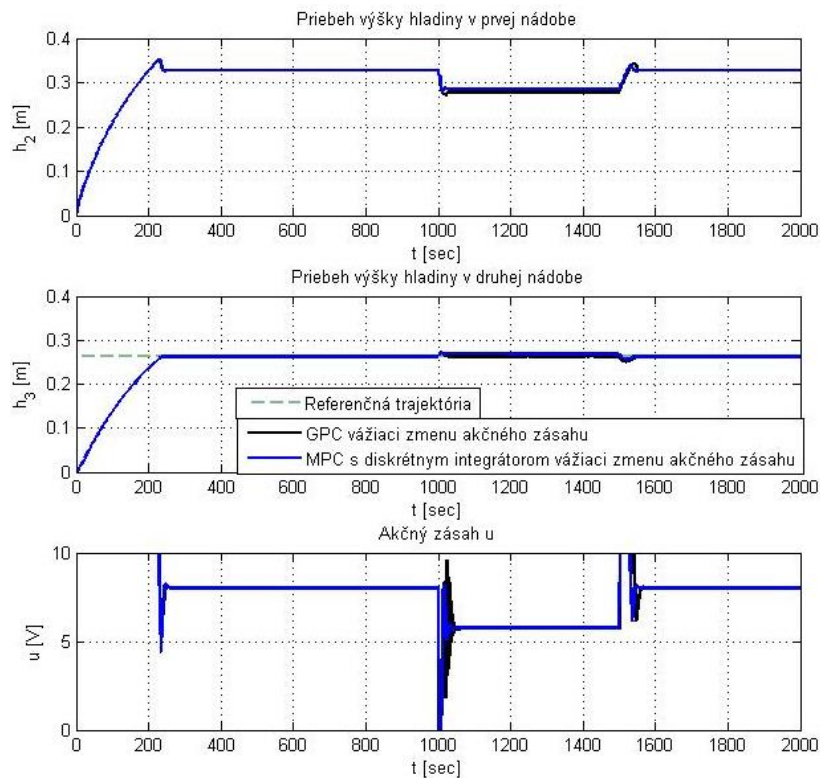
Obr. 5 Algoritmus verifikácie prediktívneho riadenia

5.a simulačné overenia prediktívnych algoritmov riadenia do ustáleného stavu na nelineárnom modeli

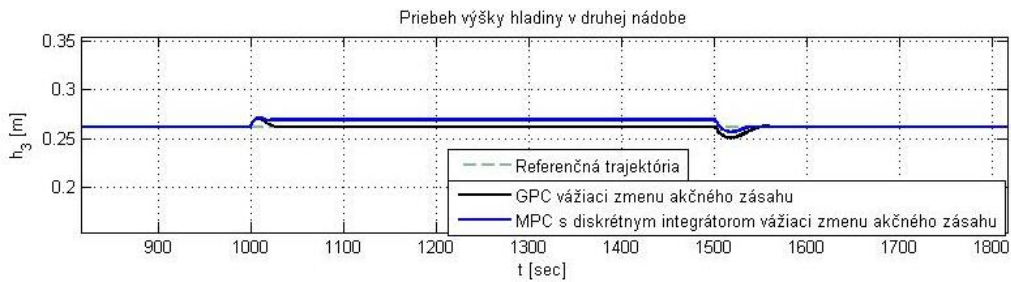


Obr. 6 Porovnanie časových priebehov – stabilizácia regulovanej veličiny na ustálenú hodnotu – nelineárny model

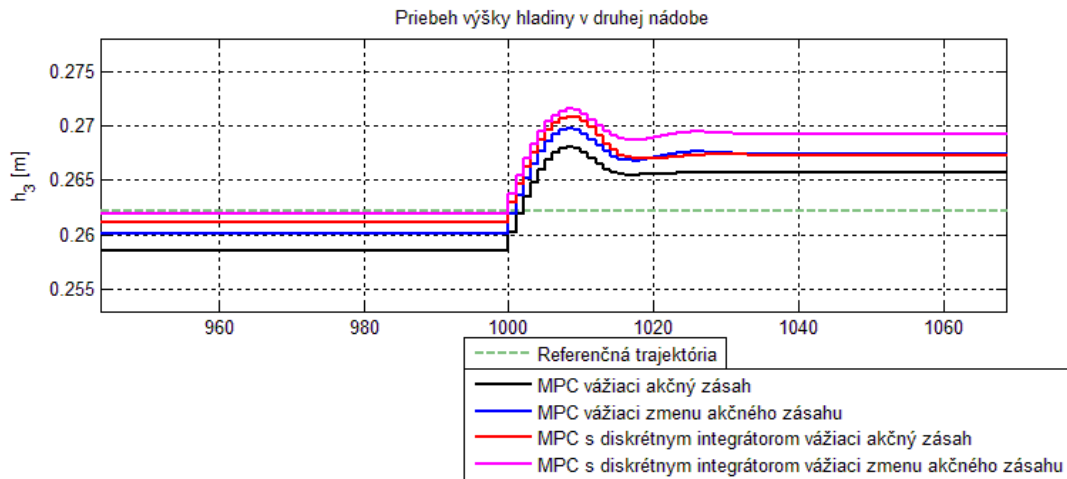
5.b simulačné overenie prediktívnych algoritmov riadenia do ustáleného stavu s pôsobením poruchy na nelineárnom modeli



Obr. 7 Porovnanie časových priebehov – stabilizácia regulovanej veličiny na ustálenú hodnotu s poruchou – nelineárny model

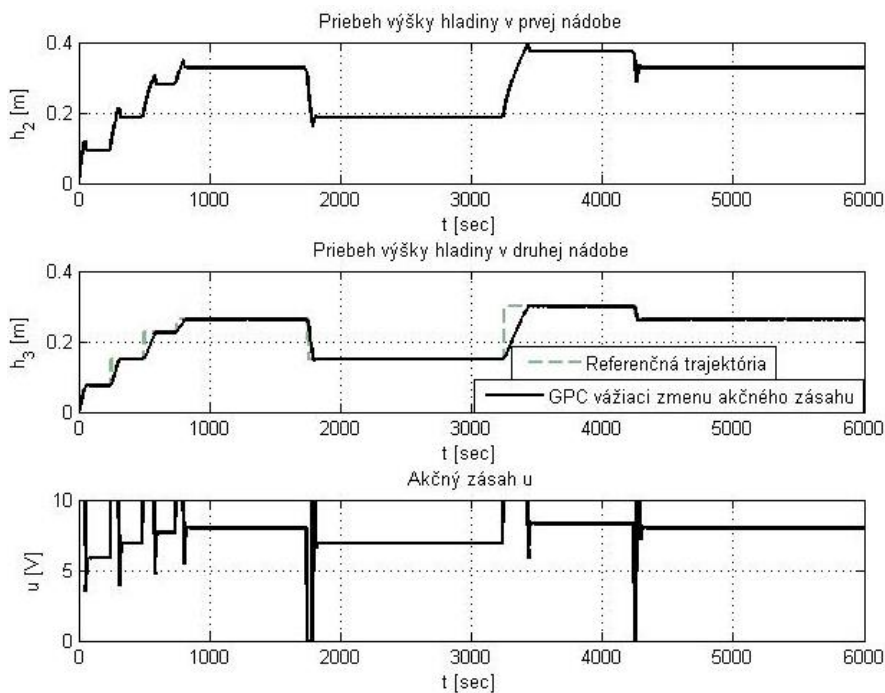


Obr. 8 Porovnanie časových priebehov – stabilizácia regulovanej veličiny na ustálenú hodnotu s poruchou (PRIBLIŽENIE)– nelineárny model – nelineárny model

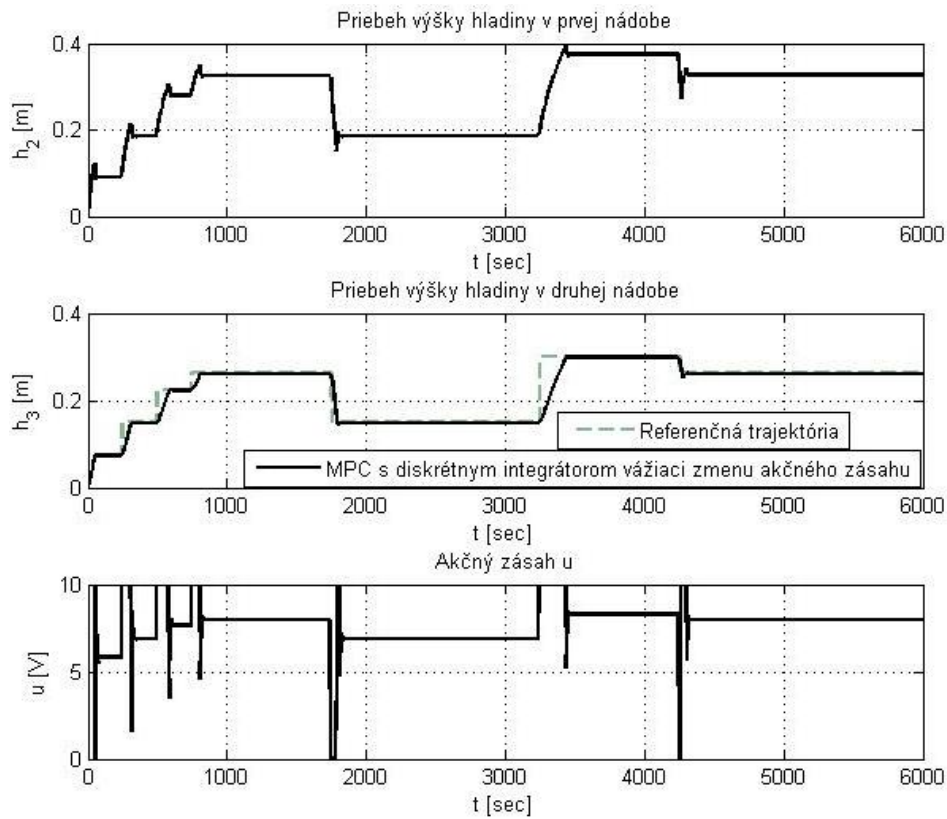


Obr. 9 Porovnanie časových priebehov výšok hladín v druhej nádobe pre Hydraulický systém pri použití stavového MPC algoritmu v riadiacej štruktúre – nelineárny model

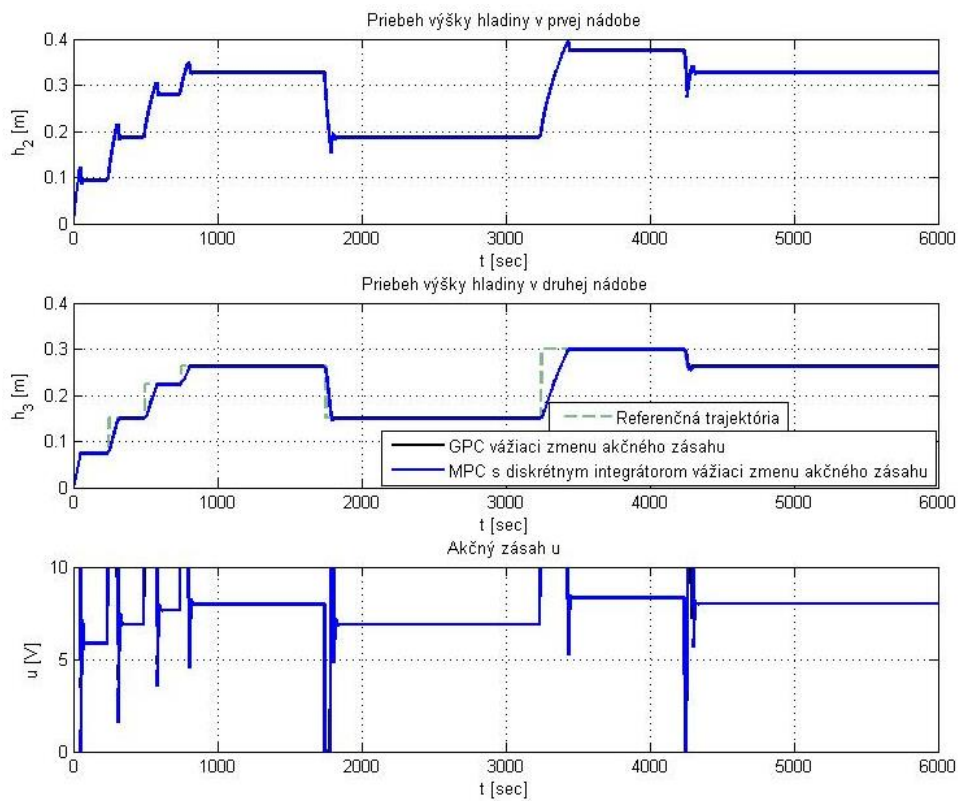
5.c simulačné overenie prediktívnych algoritmov riadenia do rôznych ustálených stavov na nelineárnom modeli



Obr. 10 Časové priebehy výšok hladín a akčný zásah pre Hydraulický systém pri skokových zmenách referenčnej trajektória (GPC algoritmus riadenia) – nelineárny model



Obr. 11 Časové priebehy výšok hladín a akčný zásah pre Hydraulický systém pri skokových zmenách referenčnej trajektórie (stavový MPC algoritmus riadenia) – nelineárny model



Obr. 12 Porovnanie časových priebehov – stabilizácia regulovanej veličiny pri zmene ustálených hodnôt – nelineárny model

Záver

Simuláciou dvoch typov algoritmov prediktívneho riadenia na nelineárnom modeli Hydraulického systému M6, pri vhodne zvolenom pomernom váhovom koeficiente λ pre algoritmus GPC riadenia, zvolených váhových maticiach pre algoritmus MPC riadenia, sme získali výsledky vo forme časových priebehov uvedených vyššie.

Na Obr. 6 a Obr. 12 možno vidieť, že časové priebehy výšky hladiny v druhej nádobe $h_3(t)$ sú prípadoch použitia algoritmu GPC a MPC totožné. Rozdiel je vo veľkosti akčného zásahu. Na tento rozdiel vplýva fakt, že algoritmus GPC je založený na báze diskkrétnej vstupno-výstupnej prenosovej funkcie a algoritmus MPC vychádza z diskrétného stavového opisu lineárneho SISO modelu.

Detailnejší pohľad na porovnanie časových priebehov stabilizácie regulovanej veličiny na ustálenú hodnotu s poruchou znázorňuje Obr. 8. Algoritmus GPC efektívne odstránil vplyv poruchy na rozdiel od algoritmu MPC, ktorý nedokázal poruchu dostatočne eliminovať, dôsledkom čoho došlo k trvalej regulačnej odchýlke.

Na Obr. 9 je znázornené porovnanie časových priebehov výšky hladiny $h_3(t)$ v druhej nádobe pri použití štyroch modifikácií stavového algoritmu MPC. Z hľadiska odstránenia trvalej regulačnej odchýlky bolo výhodne použiť stavový algoritmus MPC s diskrétnym integrátorom vážiaci zmenu akčného zásahu. Na druhej strane pri použití tohto algoritmu, v čase keď nastala chyba, došlo k najväčšej regulačnej odchýlke, pretože algoritmus vniesol do systému integračný charakter. Pri návrhu tejto modifikácie algoritmu bolo potrebné zvoliť váhovú maticu sumátora regulačnej odchýlky Q_{se} . Použitie ostatných modifikácií nemá žiadne opodstatnenie, pretože pri stabilizácii na referenčnú hodnotu a taktiež aj v čase keď pôsobila porucha dochádzalo k trvalej regulačnej odchýlke.

Veľkosť preregulovania a dobu nábehu pre algoritmus GPC riadenia je možné upraviť voľbou vhodnejšieho pomerného váhového koeficientu λ , podobne aj pre algoritmus MPC riadenia voľbou iných váhových matic Q , Q_{se} a R .