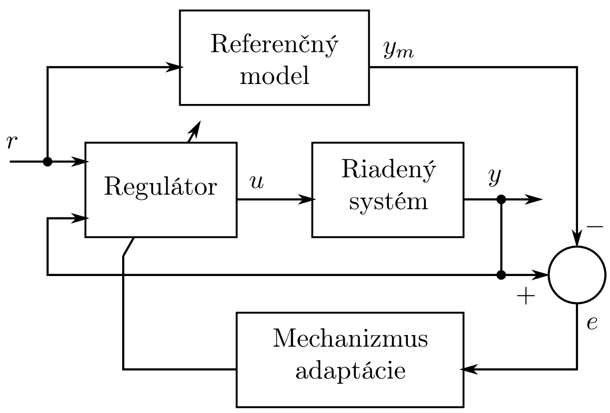
Riadenie modelu lode

Obsah

1 Úvod

Referát sa zaoberá adaptívnym riadením zjednodušeného modelu lode. Zvoleným adaptačným algoritmom na riadenie modelu lode bol MRAC resp. adaptívne riadenie s referenčným modelom.

Pri adaptívnom riadení s referenčným modelom obsahuje riadiaci systém dve spätnoväzbové slučky. Prvá spätnovazbová slučka obsahuje riadený systém a regulátor.



Obr. 1: Adaptívne riadenie s referenčným modelom – principiálna schéma

Druhá spätnoväzbová slučka obsahuje adaptačný mechanizmus, ktorý upravuje parametre regulátora. Vstupom do mechanizmu adaptácie je rozdiel medzi výstupom riadeného systému a referenčného modelu. Mechanizmus, ktorý adaptuje parametre regulátora môže byť v adaptívnom riadení s referenčným modelom získaný použitím gradientnej metódy. Žiadané vlastnosti uzavretého regulačného obvodu sú opísané referenčným modelom.

2 Simulačný model

V tomto referáte sa použil zjednodušený model lode tzv. Nomotov model, ktorý má tvar prenosovej funkcie:

$$\varphi(s) = \frac{\frac{K}{\tau_1}}{s^2 + \frac{1}{\tau_1}s}\delta(s) \tag{1}$$

kde $\varphi(s)$ je uhol natočenia lode v radiánoch, $\delta(s)$ je uhol vychýlenia kormidla v radiánoch. Definujme nákladnú loď nasledovnými parametrami.

Tabuľka 1: Parametre lode

Parameter	Hodnota
L	161 m
K_0	$-3,86$
τ_{10}	$5,66$
v	5 m s^{-1}

kde L je dĺžka lode v metroch, K_0, τ_{10} sú konštanty, v je rýchlosť lode v smere daného uhla $\varphi(s)$ v metroch za sekundu. Parametre, ktoré boli uvedené vyššie v prenosovej funkcii (1), sú definované nasledovne:

$$K = K_0\frac{v}{L} \tag{2}$$

$$\tau_1 = \tau_{10}\frac{L}{v} \tag{3}$$

Model sa vytvoril v programe python tak, že sa definovala matica A a vektori b a c.

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -a1 \end{bmatrix} \tag{4}$$

$$b = \begin{bmatrix} 0 & b0 \end{bmatrix}^T \tag{5}$$

$$c = \begin{bmatrix} 1 & 0 \end{bmatrix}^T \tag{6}$$

pričom a1 je parameter z menovateľa prenosovej funkcie (1) a to isté platí pre b0, ale to je čitateľ. Na realizáciu simulácie sa ďalej použila ode funkcia v pythone.

3 Referenčný model

Referenčný model vyjadruje to, ako chceme aby sa správal riadený systém. Vstupom je referenčná r veličina (kurz) a výstup y_m definuje požadované správanie systému. Spomenutý referenčný model je vyjadrený v tvare prenosovej funkcie:

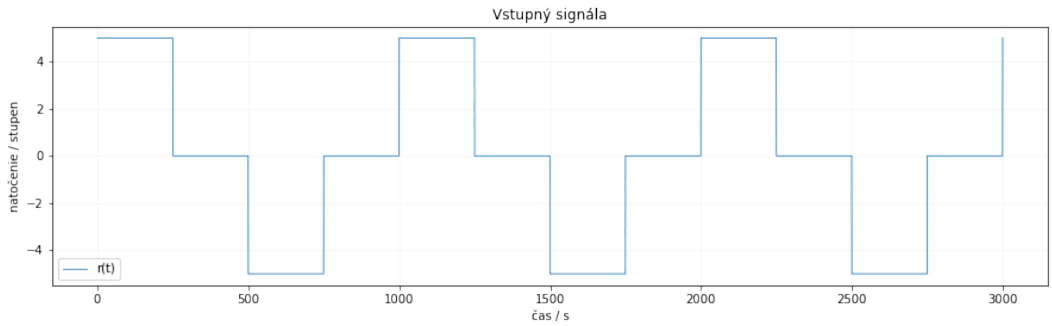
$$\frac{y_m(s)}{r(s)} = \frac{0,0025}{s^2 + 0,1s + 0,0025} \tag{7}$$

4 Adaptívne riadenie: MRAC gradientný

Úlohou adaptívneho riadenia je schopnosť reagovať na rozumné zmeny systému a následne prispôsobiť k nim aj riadenie celého systému. Inak povedané, takto sa môže vytvoriť robustnejší systém.

4.1 Vstupný signál

Uvažujme, že použijeme vstupný signál skokovej zmeny. Jedna perióda trvajúca 1000 s. Rozsah hodnôt vsutpného signálu je -5 až 5. Dĺžka celej simulácie je 3000 s. Daný vstupný signál bol tak zvolený, aby sa dala pozorovať adaptácia riadiaceho systému. Z toho to dôvodu ide o periodicky sa opakujúci signál.



Obr. 2: Vstupný signál do modelu

4.2 Riadenie

Uvažujme riadený systém daný prenosovou funkciou v tvare

$$\frac{y(s)}{u(s)} = \frac{b_0}{s^2 + a_1 s} \tag{8}$$

kde $y(s)$ je obraz výstupného signálu, $u(s)$ je obraz vstupného signálu a a_1, b_0 sú reálne konštanty – neznáme parametre sústavy.

Jedná sa o zjednodušenie nášho modelu lode v rovnici (1). Parameter $b_0 = K/\tau_1$ a $a_1 = 1/\tau_1$. Následne sa len dosadia hodnoty z tabulky (1). Ďalej si treba vyjadrit zákon riadenia a vytvoriť URO. Ďalším krokom je adaptácia.

4.2.1 Zákon riadenia

Na riadenie systému sa používa zákon riadenia, ktorý je v tvare

$$u(s) = \Theta_1 (r(s) - y(s)) - \Theta_2 s y(s) \tag{9}$$

kde r je žiadaná hodnota. Dôvodom použitia PD regulátora je fakt, že prenosová funkcia je druhého rádu s astatizmom. Zákon riadenia (9) vznikne úpravou štandardného PD regulátora v tvare

$$u(s) = (\Theta_1 + \Theta_2 s)\, e_r(s) \tag{10}$$

kde $e_r = r - y$ je regulačná odchýlka, pričom sa predpokladá, že $r(t) =$ konšt. a teda $\dot{r}(t) = 0$. V časovej oblasti možno napísať štandardný PD regulátor (10) v tvare

$$u(t) = \Theta_1 (r(t) - y(t)) + \Theta_2 (\dot{r}(t) - \dot{y}(t)) \tag{11}$$

a upravený PD zákon riadenia (9) má v časovej oblasti tvar

$$u(t) = \Theta_1 (r(t) - y(t)) - \Theta_2 \dot{y}(t) \tag{12}$$

4.2.2 Analitické URO

Na získanie URO zlúčime dve funkcie a to prenosovú funkciu (8) a získaný zákon riadenia u (12). Výsledkom je URO v tvare

$$\frac{y(s)}{r(s)} = \frac{b_0\Theta_1}{s^2 + (a_1 + b_0\Theta_2)\, s + b_0\Theta_1} \tag{13}$$

4.2.3 Podmienka zhody

Na určenie existencie a riešiteľnosti podmienky zhody je treba si definovať referenčný model. Konkrétny referenčný model bol už uvedený v kapitole (3). Avšak tu sa uvedie vo všeobecnosti takto

$$\frac{y_m(s)}{r(s)} = \frac{b_{0m}}{s^2 + a_{1m}s + a_{0m}} \tag{14}$$

kde $b_{0m} = a_{0m}$ a a_{1m} sú konštanty, ktorých hodnoty sú uvedené v kapitole (3). Ak URO (13) s RM (14) dostaneme ideálne parametre regulátora

$$\Theta_1^\star = \frac{a_{0m}}{b_0} \tag{15}$$

$$\Theta_2^\star = \frac{a_{1m} - a_1}{b_0} \tag{16}$$

4.3 Adaptácia

Cieľom adaptácie je minimalizácia účelovej funkcie, resp. dosiahnuť nulovú adaptačnú odchýlku. V prípade ideálnych parametroch $\Theta^\star$ je adaptívna odchýlka e nulová.

$$J(\Theta) = \frac{1}{2}e^2(\Theta, t) \tag{17}$$

Zákon adaptácie definujme nasledovne

$$\dot{\Theta} = -\alpha\frac{\partial J}{\partial \Theta} \tag{18}$$

kde α je konštanta určujúca krok, resp. rýchlosť pohybu proti smeru gradientu. Záporný gradient sa uvažuje preto aby sme minimalizovali účelovú funkciu, resp. našli jej minimum. V prípade nájdenia minima účelovej funkcie dosiahneme aj nulovú adaptačnú odchylku. Vďaka tejto skutočnosti docielime aby y kopírovalo y_m. Na získanie konečných rovníc využijeme rovnicu pre adaptačnú odchylku. Následne za y dosadíme (13). Potom vykonáme parciálnu deriváciu podľa Θ_1 a Θ_2. Po ďalších úpravách dostaneme rovnice

$$\frac{\partial e}{\partial \Theta_1} = \frac{1}{(s^2 + a_{1m}s + a_{0m})}\,(r - y) \tag{19}$$

$$\frac{\partial e}{\partial \Theta_2} = \frac{-s}{(s^2 + a_{1m}s + a_{0m})}\,y \tag{20}$$

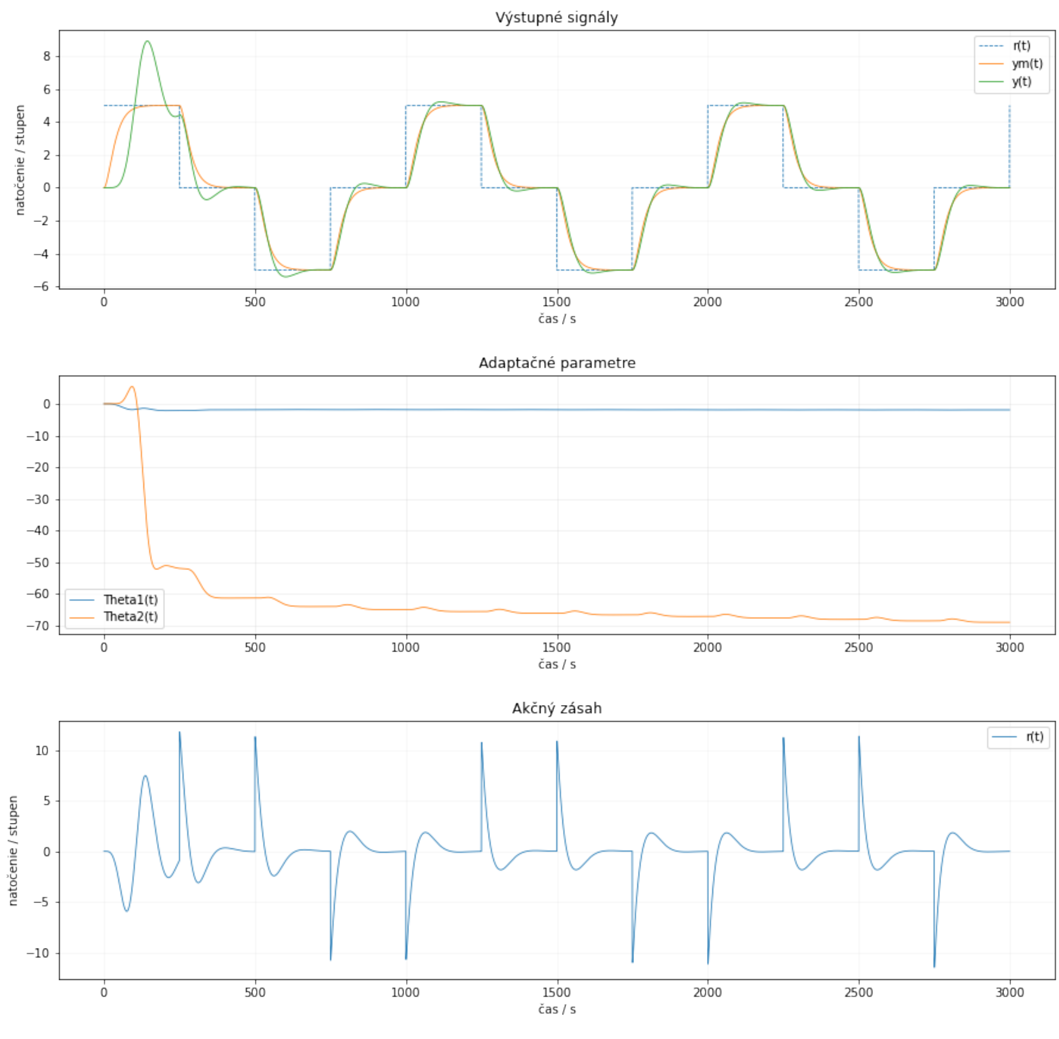
Ďalej môžeme ešte upraviť a dostaneme konečnú verziu

$$\Theta_1 s = -\alpha_1\left(\frac{1}{(s^2 + a_{1m}s + a_{0m})}\,(r - y)\right)\,e \tag{21}$$

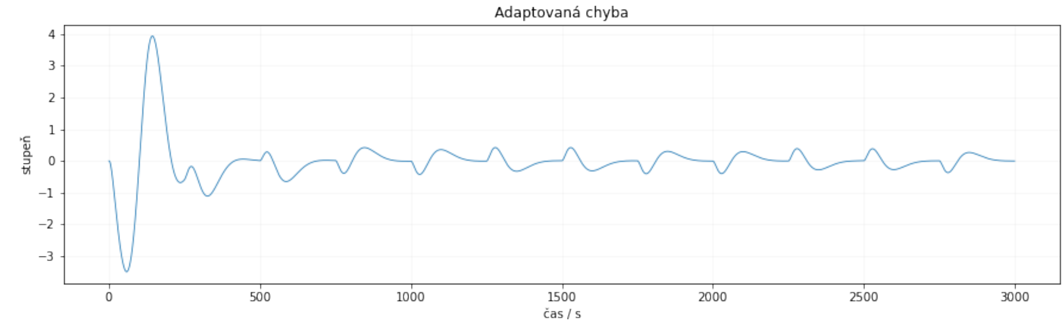
$$\Theta_2 s = -\alpha_2\left(\frac{-s}{(s^2 + a_{1m}s + a_{0m})}\,y\right)\,e \tag{22}$$

4.4 Výsledok simulácie riadenia

Po spracovaní všetkých rovníc a parametrov daného algoritmu sa spustila simulácia pre adaptívne riadenie modelu lode. Použité α_1 bolo 0.025 a α_2 25. Dané hodnoty sa zvolili preto aby bolo možné odkontrolovať správnosť referátu. Samozrejme tieto hodnoty sa za iných okolností získavajú experimentálne a nedá sa vopred stanoviť, aká má byť daná hodnota. Algoritmus MRAC je schopný už po prvej perióde pomerne presne sledovať referenčný model, resp. minimalizuje adaptačnú odchylku. Prva simulacia mala parameter rýchlosti nastavený na hodnotu $v = 5m/s$. Je vidieť pomerne velký prekmit na začiatku adaptácie, ale následne je systém schopný sa relatívne adaptovať na požadovanú hodnotu y_m. Taktiež je viditelné, že najväčšie zmeny sa odohrávajú počas adaptácie pre hodnotu parametra Θ. Tretí graf vyjadruje priebeh akčného zásahu počas adaptácie resp. riadenia daného systému. Hodnoty akčného zásahu sú spojité a konzistentné.

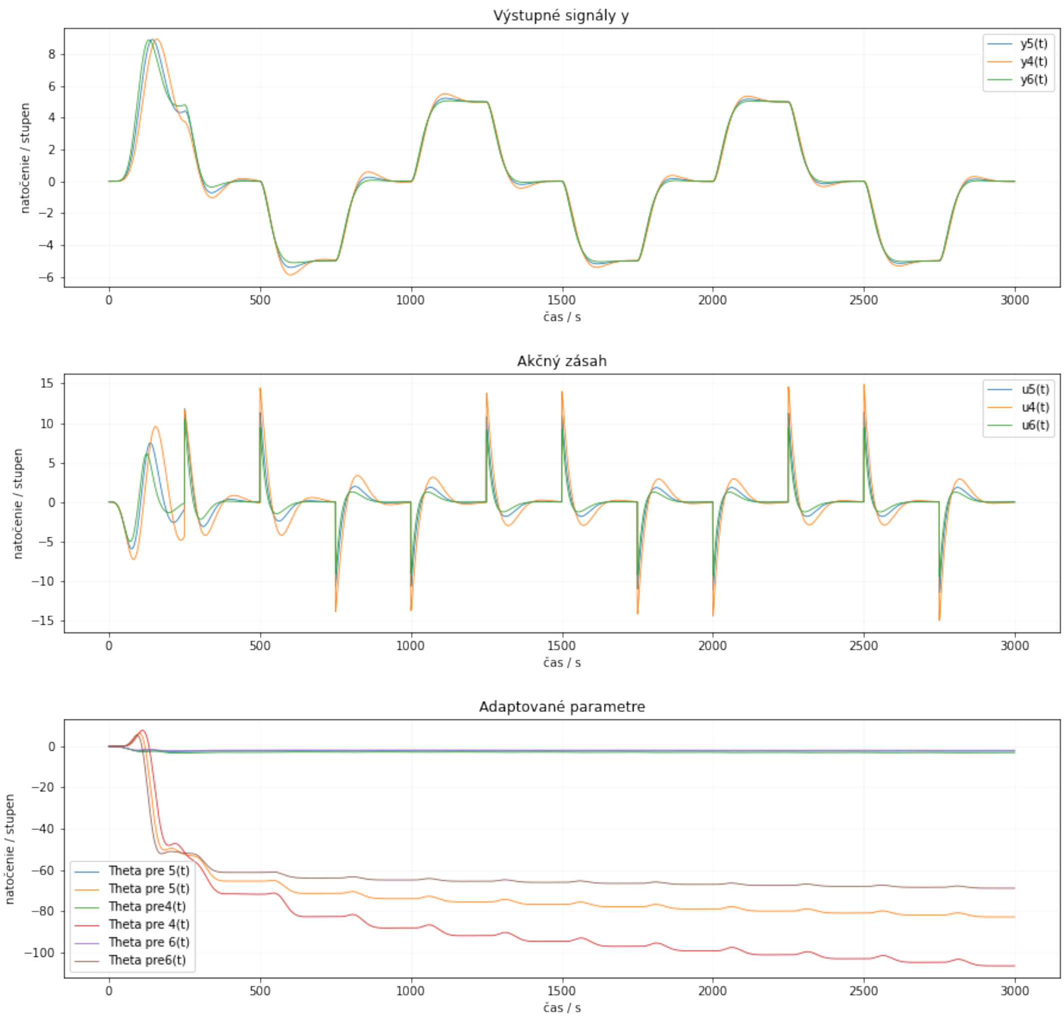


Obr. 3: Simulácia pri $v = 5$ [m/s]

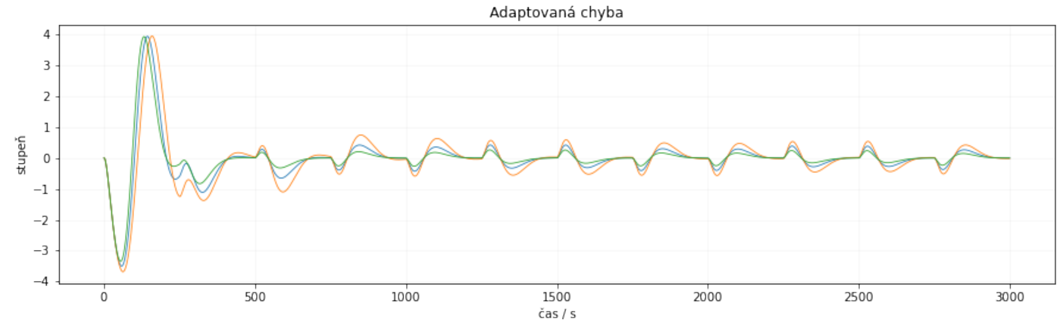


5 Testovanie rýchlosti

Testovali sa adaptívne riadenia pre dve rôzne rýchlosti, resp. tri. Použité rýchlosti boli $v = 5$ [m/s], $v = 4$ [m/s], $v = 6$ [m/s]. Z obr. 4 je vidieť, že rýchlosť má vplyv na reguláciu systému, avšak minimálnu. Najlepšie bol systém schopný kopírovať referenčné hodnoty v prípade najvyššej rýchlosti $v = 6m/s$. Naopak najväčšia adaptačná odchýlka bola pri rýchlosti $v = 4m/s$. Taktiež sa skúšala extrémna hodnota rýchosti, aby sa overilo, čo všetko vie daný systém riadiť. V prípade rýchlosti $v = 0.5m/s$ systém absolútne nebol schopný sledovať referenčnú hodnotu y_m. Vďaka tomuto je zrejmé, že systém je schopný adaptovať sa len v nejakom rozhraní hodnôt.



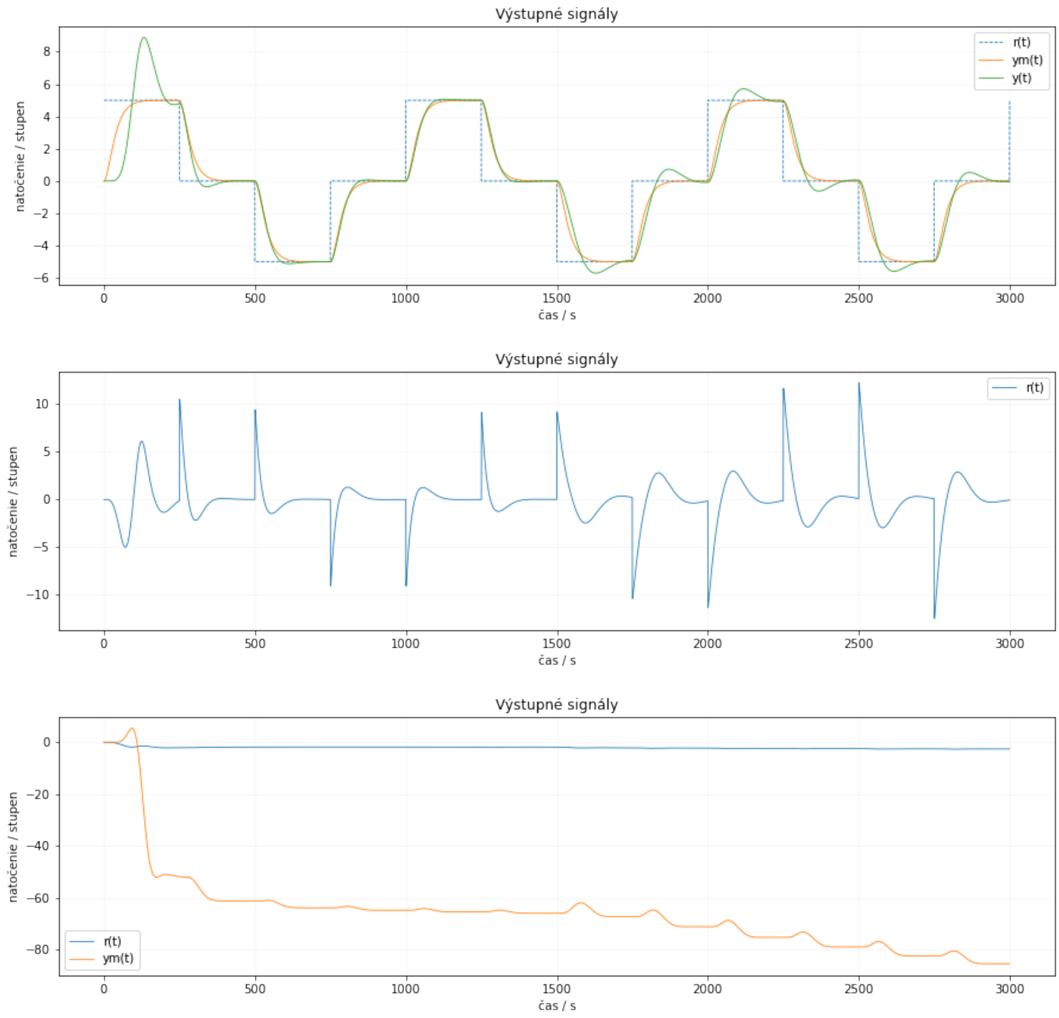
Obr. 4: Simulácia pri $v = 4$ [m/s], $v = 5$ [m/s], $v = 6$ [m/s]



Obr. 5: Simulácia pri $v = 4$ [m/s], $v = 5$ [m/s], $v = 6$ [m/s]

6 Meniaca sa rýchlosť

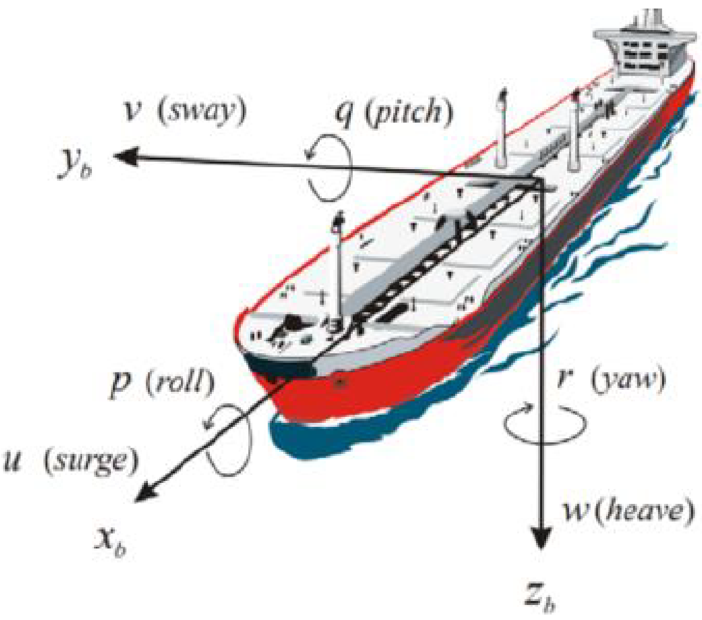
Rýchlosť sa mení v polovici simulácie z hodnoty $v = 6$ na $v = 4$. Je vidiet, že pri vyššej rýchlosti je viac schopný adaptovať sa a kopírovať referenčný model. Avšak pri znížení rýchlosti sa vyskitnú mierne prekmity pri kopírovaní referenčného modelu.



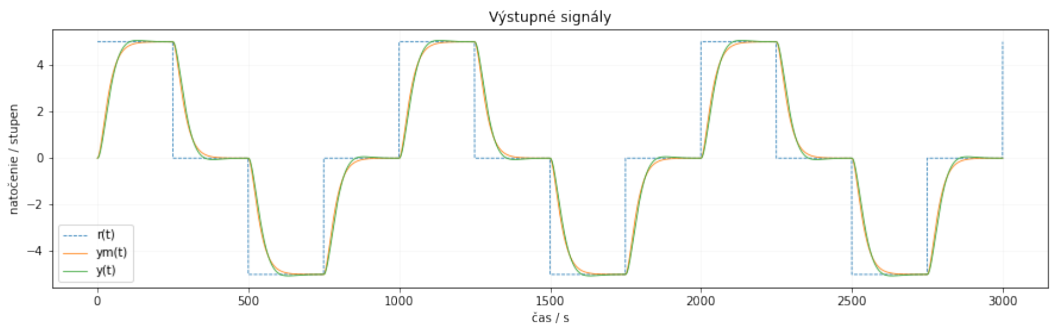
Obr. 6: Simulácia keď sa mení rýchlosť z $v = 6$ [m/s] na $v = 4$ [m/s]

7 Neadaptované

V prípade neadaptovaného riadenia sa použijú ideálne parametre Θ^*. V prípade malej zmeny parametrov modelu sa može stať, že systém nebude stabilný resp. nebude schopný vhodne riadiť požadovaný systém. Inymi slovami Θ^* je nemenna hodnota, ktorá sa neprispôsobuje okolnostiam.



Obr. 7: Loď



Obr. 8: Neadaptívne $v = 4$ [m/s]

8 Záver

V priebehu tohto referátu sa navrhol a otestoval riadiacy systém MRAC gradientný. Ako riadiacy systém sa uvažoval Nomotovov model lode. Simulácia spolu s riadením bola vytvorená v Pythone. Referát pozostával z vytvorenia simulačného modela, referenčnej prenosovej funkcie, zákona riadenia a adaptačného algoritmu. Taktiež bola dokázaná schopnosť adaptácie algoritmu pri zmene vnútorných parametrov ako je rýchlosť. Okrem dokázania adaptácie bolo dokázané, že systém je schopný sa adaptovať len pre určité rozharnie hodnôt.

Literatúra

[1] Marián Tárnik (2022), MRAC gradientný referát.

[2] Marián Tárnik (2022), MRAC gradientný AR04.

[3] Nur Assani (2020), ANALYSIS OF THE NOMOTO SHIP MODEL RESPONSE TO COURSE CHANGES USING PID CONTROLLER IN MATLAB/SIMULINK