

Učiaci algoritmy pre časovo suboptimálne riadenie servomechanizmu

M. Alexik

katedra Technickej kybernetiky,

Žilinská univerzita v Žiline, Univerzitná 8215/1, 010 26 Žilina, SR

mikulas.alexik@fri.uniza.sk

ABSTRAKT

Tento článok popisuje tri stratégie pre realizáciu časovo suboptimálneho učiaceho algoritmu aplikovaného na riadenie polohového servomechanizmu. Servomechanizmus bol realizovaný v laboratóriu a riadenie bolo realizované v reálnom čase. Nutnosť použitia učiaceho algoritmu vyplývala z požiadavky aby riadenie polohy bolo časovo optimálne aj keď sa jeho záťaž mení v širokom rozsahu. Okamžitá hodnota momentu zotrvačnosti nie je známa, preto nie je možné použiť deterministické časovo optimálne riadenie s prepínacou krivkou. Autor odvodil tri rôzne učiaci algoritmy pre "nájdanie" časovo sub optimálnej trajektórie. Efektivita (doba učenia algoritmu) je rôzna pre každú stratégiu. Stratégie pre nájdanie prepínacej krivky sú založené na sklzovom režime kombinovanom s: postupné hľadanie vhodného sklonu prepínacej priamky, 2) priebežná identifikácia parametrov serva a výpočet prepínacej krivky, 3) off line výpočet inverzného neurónového modelu serva s prepínacou krivkou a následný výpočet časovo suboptimálneho riadenia včítane klasifikácie..

KEY WORDS

Sliding mode, hardware in loop simulation, neural nets.

1. Úvod

Systémy riadenia s učením majú výhodu oproti systémom s klasickými algoritmi riadenia v prípadoch, ak sa vnútorné alebo vonkajšie podmienky riadenia menia. Klasická stratégia môže použiť naučené informácie z predchádzajúceho procesu riadenia alebo situácie. Ktoré sú potom uložené v pamäti a po úspešnom rozpoznaní situácie môže dosiahnuť optimálne výsledky v kratšom čase. Lepšie výhod učenia možno úspešne aplikovať tiež pri riadení v sklzovom režime s pomocou umelých neurónových sietí. Učiaci systém môže pracovať s pamäťou, ktorá ukladá predchádzajúce výsledky adaptácie. V procese učenia, nasledujúcom po adaptačných výsledkoch, systém vyberie tie najlepšie z nich. Teda cieľom je minimalizovať stratu:

$$Q(x, \Omega, \omega_r) = \min_{\omega} Q(x, \Omega, \omega) \quad (1)$$

kde x je stav systému, Ω je informácia učiteľa a ω je zákon riadenia. Obecná strata optimalizačného kritéria, v učiacom sa systéme po procese učenia, je vždy menšia

ako v adaptívnom systéme. Najlepšia stratégia, z pohľadu autora, je kombinácia priebežnej identifikácie parametrov servomechanizmu s výpočtom prepínacej krivky v reálnom čase z modelu neurónovej siete.

Čas potrebný na učenie sa systému je špecifikovaný rýchlosťou riešenia rovnice (1) a značne závisí od množstva apriórnych informácií o riadenom systéme. Výhoda učiacich sa systémov oproti optimálnym algoritmom je, že ich návrh nevyžaduje všetky apriórne informácie o prostredí alebo riadenom systéme.

Článok je organizovaný nasledovne. Časť 2 popisuje problém optimálneho riadenia, časť 3 popisuje riadenie v sklzovom režime, časť 4 popisuje učiaci algoritmus a časť 5 popisuje simulačné experimenty v reálnom čase a praktické výsledky. Článok končí záverom a výhľadom ďalšieho riešenia v časti 6.

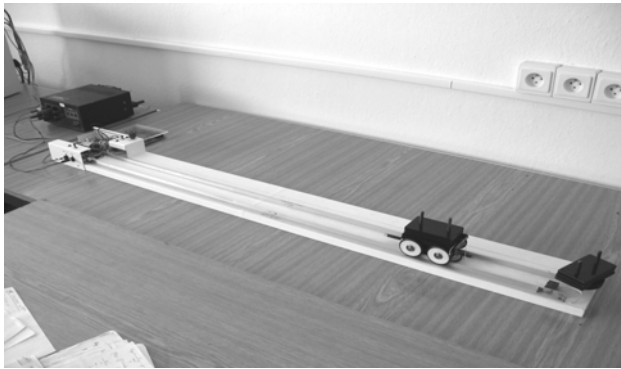
2. Problém časovo optimálneho riadenia.

Úloha t- optimálneho riadenia patrila medzi prvé problémy, ktoré boli riešené v teórii automatického riadenia a optimalizácii systémov. Iba formulácia princípu minima umožnila obecný pohľad na otázku t- optimálneho riadenia lineárnych systémov s obmedzením akčnej veličiny. Vlastnosti optimálnej trajektórie sú často používané v nelineárnych systémoch, v časovo sub optimálnych servomechanizmoch robotov a v adaptívnych a učiacich sa algoritmoch.

Učiaci sa regulátor je navrhnutý pre laboratórny model vozíka, poháňaný jednosmerným - motorom. Cieľom je nájsť t - optimálne riadenie jeho polohy. Fotografia modelu je na Obrázku 1. Prenosová funkcia tohto systému môže byť redukovaná do tvaru

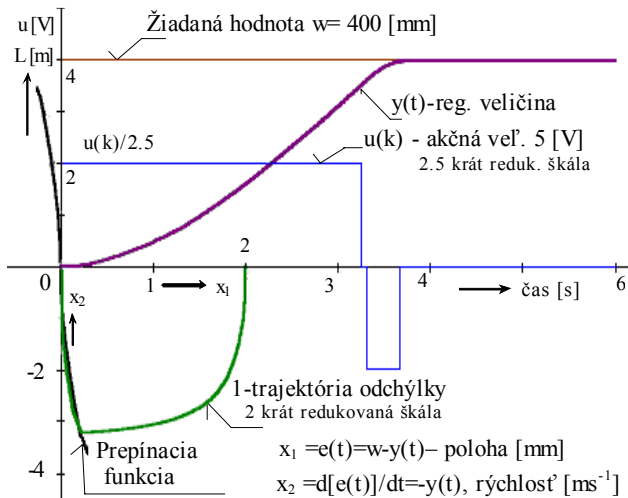
$$S(p) = \frac{K}{s(Ts + 1)} \quad (2)$$

kde K je zosilnenie a T je časová konštanta modelu vozíka. Laboratórny model vozíka môže byť zaťažovaný s 0 až 6 rôznymi závažiami (1 závažie = 0.6 kg). Potom prenosová funkcia (2) má 7 rôznych zosilnení K a časová konštanta T , ktorá závisí tiež od trenia, rovnako. Reálne trenie je nelineárne, model (2) predpokladá lineárne coulombovské trenie.



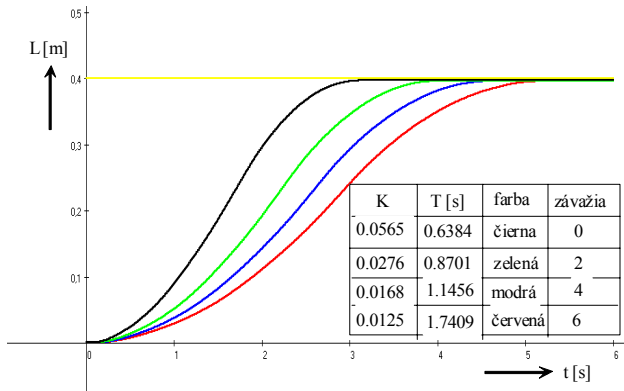
Obr. 1 Laboratórny model vozíka

Model vozíka je systém s the dvoma kapacitnými oneskoreniami pre ktoré je možné odvodiť časovo optimálne riadenie podľa riadiacej slučky v Obr. 4 a odozvy na Obr.2. Trajektória pod osou času reprezentuje proces riadenia vo fázovom priestore $[x_1, x_2] = [e(t), e'(t)]$. Na fázovej trajektórii je jeden t bod prepnutia.



Obr. 2. Odozva a stavová trajektória časovo optimálneho riadenia riadeného systému druhého rádu.

Ak regulátor pozná “K” a “T” presne hoci závažia na vozíku sa menia, potom môže počítať prepínicu funkciu (4), a tak sa proces riadenia stane t- optimálnym.



Obr. 3 Odozvy a parametre pre niektoré závažia

3. Riadenie v sklzovom režime.

Riadenie v sklzovom režime (SMC) je veľmi populárne a široko používané. Výhodou je jeho skutočne jednoduchý návrh, invariantnosť a robustnosť. Reléové riadenie (bang-bang) patrí medzi prvé aplikácie [2], [4], keď bol akčný signál obmedzený. Preto t- optimálne riadenie vyžaduje iba minimálnu alebo maximálnu akčnú veličinu. SMC regulátor je veľmi jednoduchý. Akčná hodnota je predpísaná podľa polohy v stavovom priestore. Fázový priestor je rozdelený podľa prepínacej plochy $s(x)$:

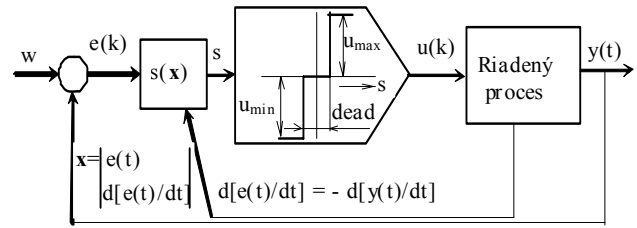


Fig. 4 Bloková schéma riadiaceho obvodu optimálneho riadenia.

Najviac používaným tvarom prepínacieho povrchu je lineárna funkcia [3],[5]:

$$s(\mathbf{x}) = C\mathbf{x} = 0 \quad (3)$$

Obyčajne táto prepínicia funkcia nevyhovuje požiadavkám t- optimálneho riadenia a riadenie sa môže dosiahnuť sklzovým režimom. Vhodná t- optimálna prepínicia funkcia je nelineárna. Prvým cieľom učiaceho algoritmu je nájsť túto prepínicu funkciu. Prepínicia funkcia je odvodená z prechodovej odozvy (4) riadeného procesu (2) a jr popísaná s (5). Potom riadená veličina z Obr. 4 je popísaná podľa (6).

$$y(t) = K[t - T + T \exp(-t/T)]U_i \quad (4)$$

$$s(\mathbf{x}) = U_i K T \ln\left(\frac{1 - x_2(t)}{K U_i}\right) + T x_2(t) + x_1(t)$$

$$\text{if } x_2 < 0 \text{ then } U_i = u_{\max} \quad (5)$$

$$\text{if } x_2 \geq 0 \text{ then } U_i = u_{\min}$$

$$x_1(t) = e(t) = w(t) - y(t); \quad x_2(t) = d[e(t)/dt]$$

$$\text{if } (s \geq \text{dead}) \text{ then } u(k) = u_{\max} \quad (6)$$

$$\text{if } (\text{abs}(s) < \text{dead}) \text{ then } u(k) = 0$$

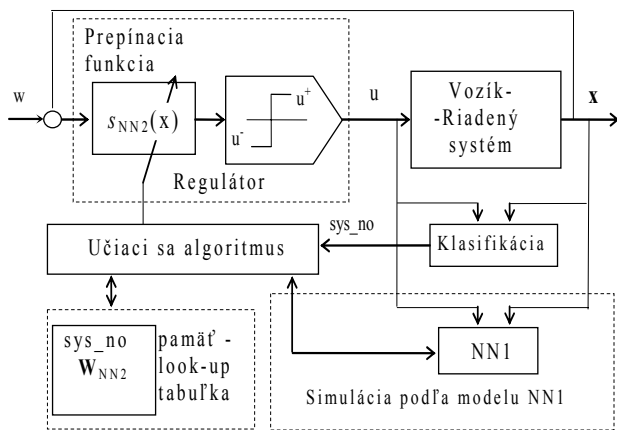
$$\text{if } (s < -\text{dead}) \text{ then } u(k) = u_{\min}$$

Pretože závažia na vozíku sa môžu meniť nevieme okamžitú hodnotu časovej konštanty T a zosilnenie K riadeného procesu, preto nemôžeme vypočítať prepínicu funkciu a realizovať t- optimálne riadenie. Učiaci sa regulátor tromi spôsobmi môže riešiť tento problém. Meranie polohy vozíka v reálnom čase v každom intervale vzorkovania (5 [ms] do 20 [ms]) a jej filtrácia je veľmi dôležitý krok v každej stratégii riadenia ale nie je opísaná detailne. Nasledujúca časť opisuje tri spôsoby realizácie učiaceho sa regulátora na báze t – optimálneho riadenia.

4. Učiaci sa regulátor.

Prečo potrebujeme učiaci sa regulátor? Keď záťaž - vozíka sa mení znamená to, že parameter riadeného procesu sa menia tiež. Je šesť rôznych závaží a preto ide o systém so siedmimi rôznymi dvojicami parametrov. Ak regulátor je nastavený na jeden systém a prepínacia funkcia sa nájde, je uložená do pamäte učiaceho sa regulátora pre prípad keď sa znovu bude riadiť taká záťaž. Učiaci sa regulátor môže riadiť systém t - optimálne aj keď parameter systému sda budú meniť. Základy pre formuláciu učiaceho sa algoritmu boli publikované v [7]. Niektoré výstupy možno vidieť na Obr. 6 až Obr. 9 a v sekcii so simulačnými experimentmi.

Stratégie hľadania prepínacej funkcie pre časovo suboptimálne riadenie sú založené na riadení v sklzovom režime. Sú nasledovné: 1.) postupné hľadanie vhodného sklonu prepínacej priamky, 2) priebežná identifikácia parametrov servomechanizmu v reálnom čase a výpočet prepínacej krivky, 3) off line výpočet inverzného modelu servomechanizmu s výpočtom prepínacej krivky, potom klasifikácia v reálnom čase s časovo suboptimálnym riadením. Idea učiaceho sa regulátora pre tieto stratégie je spoločná, zobrazená na Obr. 5.



Obr. 5 Bloková schéma učiaceho sa regulátora.

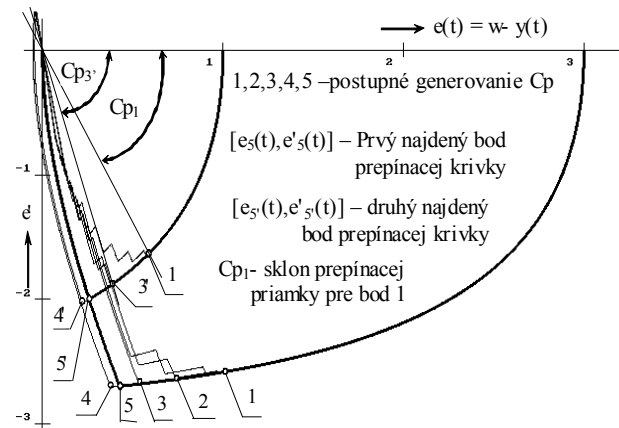
Bloky na obrázku sú nasledovné:

Blok klasifikácie je zodpovedný za výber systému. generuje číslo systému, v závislosti od parametrov systému. Pretože je možnosť použiť uložené výsledky pre správny systém, je nutné aby regulátor klasifikoval okamžitý systém. Klasifikácia, ktorá je použitá v tomto článku je na báze, parametrickej identifikácie alebo ART siete [1].

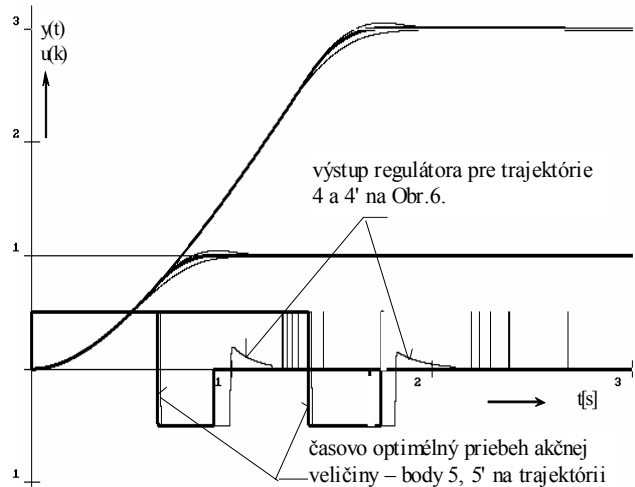
Blok regulátora je zodpovedný za akčnú veličinu. v tomto bloku prvýkrát opisujeme stratégiu postupného hľadania vhodného sklonu prepínacej priamky.

V priebehu postupného generovania sklonu prepínacej priamky C_p , adaptívnym sklzovým algoritmom, body z prepínacej krivky pre viacero žiadaných hodnôt sú nahrané do pamäte. Takým postupom, viac bodov pre prepínanie krivku môže byť nájdených a parametre prepínacej funkcie môžu byť vypočítané alebo

interpolované. Optimálna skoková odozva pre vybrané žiadané hodnoty a body z prepínacej krivky je vybraná zo všetkých generovaných odoziev podľa toho, ktorá odozva má minimálnu dobu regulácie bez preregovania. Na Obr. 6 a Obr. 7 je ilustrovaný proces pre hľadanie bodov prepínacej krivky. Ako možno vidieť je potrebné 5 až 10 odoziev pre nájdenie bodov prepínacej krivky pre jeden pár parametrov $[K, T]$ modelu (2). Táto stratégia učenia nemôže byť realizovaná v reálnom čase, ale je to prvý krok pri riešení problému.



Obr. 6 Stavové trajektórie v priebehu procesu učenia pri hľadaní bodov prepínacej krivky



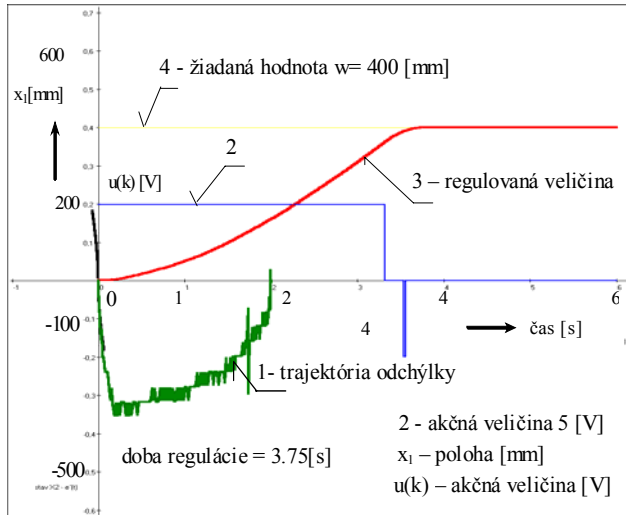
Obr. 7 Skokové odozvy v priebehu procesu učenia sa pri hľadaní bodov z prepínacej krivky.

Pre stratégiu riadenia s identifikáciou parametrov procesu a následným výpočtom prepínacej krivky (4) v reálnom čase je potrebný nový spôsob pre výpočet parametrov zo skokovej odozvy (6) riadeného procesu (2). On line priebežná identifikácia nemôže byť použitá. Parametre K , a T musia byť vypočítané pred časom keď akčná veličina prepne. Z prechodovej funkcie (4) možno odvodiť bodové odhady parametrov prenosovej funkcie (2) v tvare (7).

$$K = [x_2(t/2)]^2 / \{U_{\max} [2x_2(t/2) - x_2(t)]\} \quad (7)$$

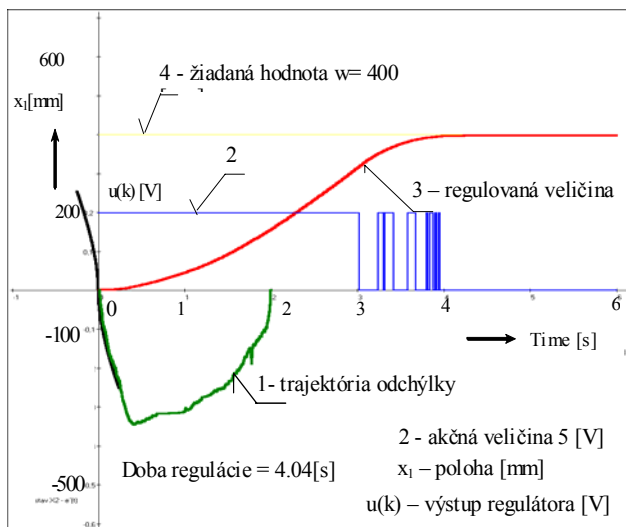
$$T = -t / \{\ln[1 - x_2(t)/K]\}$$

Nedostatkom výpočtu je, že potrebujeme poznať okamžitú hodnotu derivácie riadenej veličiny. Ako je vidieť na Obr.8 hodnoty derivácie sa menia iba v 9-15 úrovniach (použitý bol senzor s prírastkom 0.28 mm). V odozva uzavretej slučky na Obr. 8 sa predpokladalo, parametre riadeného procesu a tiež prepínacej krivky sú známe, preto identifikácia nie je potrebná.



Obr. 8 Odozva reg. obvodu, stavová trajektória a akčná veličina časovo suboptimálneho riadenia pre systém vozík.

Hoci na Obr. 9 bol použitý filtrovaný signál (100 [ms] časová konštanta filtrácie), hodnoty derivácie boli stále príliš narušené šumom, a preto hodnoty derivácie regulovanej veličiny nemohli byť vypočítané presne, čím doba regulácie nebola časovo optimálna a akčná veličina prepínala iba v jednej polarite.



Obr. 9 Odozva reg. obvodu, stavová trajektória, a akčná veličina pri časovo optimálnom riadení.

V tejto stratégii je pre klasifikáciu a t – suboptimálne riadenie použitá identifikácia v reálnom čase a nie je potrebný špeciálny blok pre simuláciu a učiaci sa regulátor.

Tretou použitou stratégiou na aproximáciu prepínacej funkcie je dopredná neurónová sieť NN2 [5]. Na začiatku NN2 aproximuje lineárnu prepínaciu funkciu. Sieť NN2 je trénovaná podľa bodov z fázy simulácie. Neskôr je NN2 nastavená na aproximáciu nelineárnej t- optimálnej prepínacej funkcie. Prepínacia funkcia podľa NN2je:

$$s_{NN2}(\mathbf{x}) = x_n(k) - f_{NN2}(x_1(k), x_2(k), \dots, x_{n-1}(k)) = 0 \quad (8)$$

kde n je rád systému. Zložitosť siete NN2 závisí od rádu riadeného systému. Pre systém druhého rádu (2) má NN2 jeden vstup a jeden výstup. Mala by mať najmenej 10 neurónov nelineárnou aktivačnou funkciou.

Keď sa parameter systému menia, učiaci sa algoritmus nastavuje NN2 (váhy jej matice W_{NN2}) podľa klasifikovaného systému (sys_no).

Blok simulácie obsahuje diskretný lineárny neurónový model (dopredná NN1) systému v tvare:

$$y(k) = f_{NN1}(y(k+1), y(k+2), \dots, y(k+n), u(k), u(k+1), \dots, u(k+n-1)) \quad (9)$$

Počet vstupov NN1 závisí od rádu systému. Pre prípad druhého rádu (2), bude mať NN1 najmenej 6 vstupov a jeden výstup. preto je systém lineárny a NN1 môže mať dvojice lineárnych neurónov. Takýto model je použitý pre simuláciu. Simulácia generuje body t - optimálnej fázovej trajektórie. Podľa jej bodov je trénovaná neurónová sieť NN2. Potom, NN2 aproximuje t - optimálnu prepínaciu funkciu.

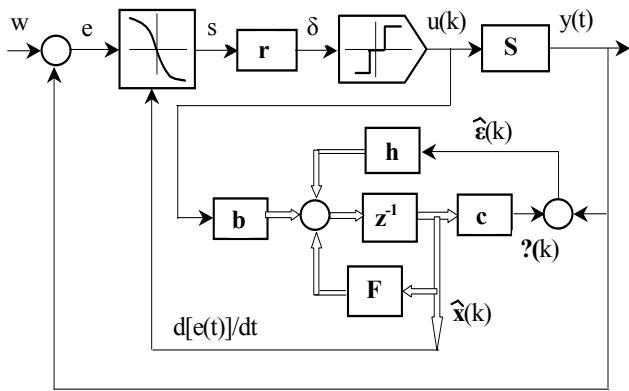
Blok učiaceho sa algoritmu zodpovedá za úlohu kooperácie, organizácie pamäte, trénovanie neurónovej siete a simuláciu.

Obidva siete (NN1, NN2) boli trénované metódou Levenberg-Marquardt [5]. Táto metóda je rýchlejšia ako všeobecná back-propagation.

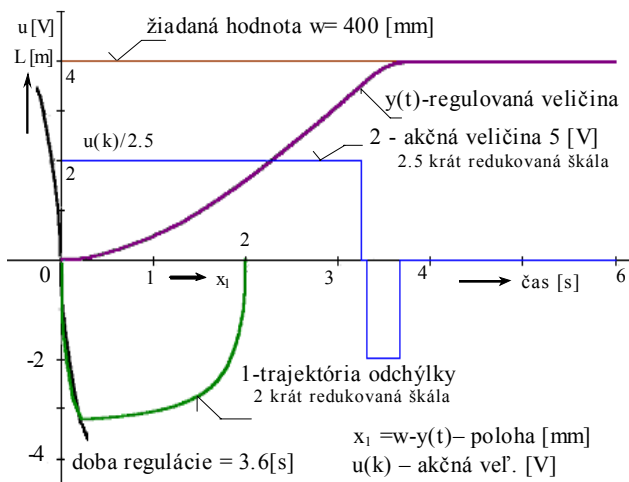
Učiaci sa regulátor, popísaný vyššie, je veľmi efektívny, pretože je schopný nájsť t - optimálne riadenie v dvoch krokoch učenia pre každú jednotlivú zmenu parametrov riadeného procesu. V prvom kroku učenia, sa proces riadenia správa podľa prepínacej funkcie a priori definovanej v NN1. V druhom kroku učenia, je riadený proces t - optimálny.

5. Simulačné experimenty.

Najlepšie výsledky boli dosiahnuté pri kombinácii merania regulovanej veličiny v reálnom čase a výpočtu exaktnej hodnoty jej derivácie z odhadu stavov. V stavovom modeli odhadu stavov riadeného procesu bol predpokladaný systém s 2 závažiami pre všetky situácie zmien závaží. Bloková schéma t – optimálneho riadenia so stavovým estimátorom je na Obr. 10. Všetky odozvy uzavretej slučky so stavovým estimátorom možno vidieť na Obr. 11.



Obr.10 Bloková schéma t - optimálneho riadiaceho obvodu s estimátorom stavov.



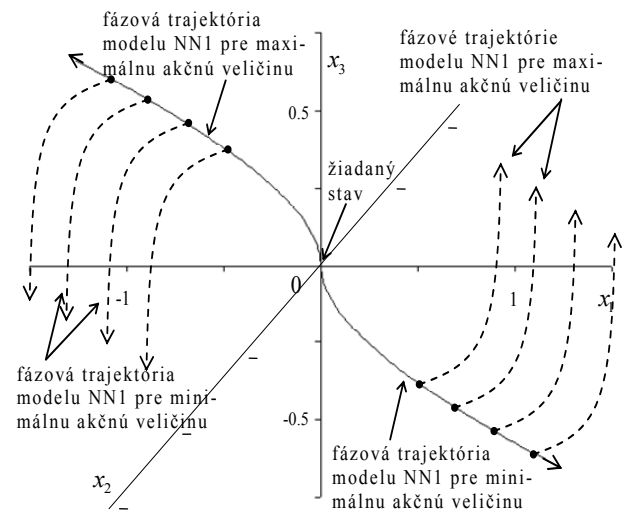
Obr.11 Odozva reg. obvodu, stavová trajektória a výstup regulátora pre regulačný slučku s estimátorom stavov.

Nasledujúce simulačné experimenty sú pre regulačný obvod s neurónovou sieťou.

Simulácie v stavovom priestore môžu byť použité na nájdenie prepínacej funkcie pre t- optimálne riadenie, z toho dôvodu lebo časť t- optimálnej fázovej trajektórie je totožná s prepínacou funkciou. Simulácia beží s modelom systému (NN1). Prepínacia funkcia prejde cez body fázovej trajektórie, ktoré sú výsledkami zo simulácie. Aproximácia prepínacej funkcie s NN2 môže byť zlepšená, ak počet simulovaných bodov fázovej trajektórie je vyšší. Potom t- optimálne riadenie má špeciálnu vlastnosť z pohľadu akčnej veličiny. Ak riadenie musí byť optimálne, akčná veličina musí mať iba extrémne hodnoty.

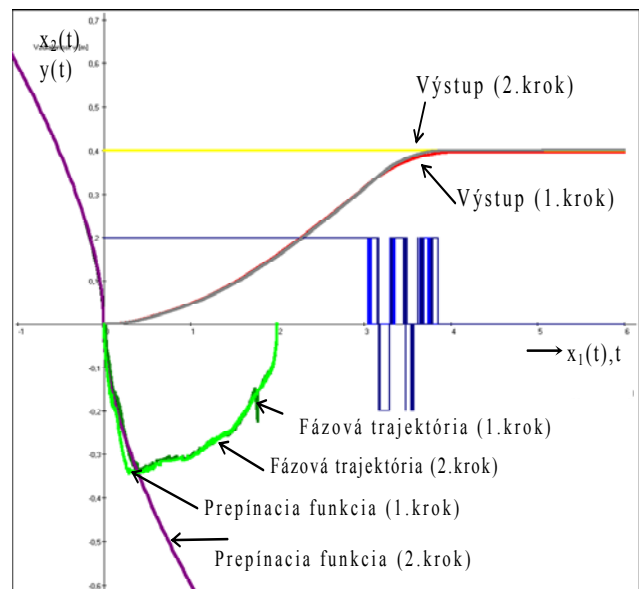
Hlavná úloha simulácie je simulovať inverzný t- optimálny proces riadenia. Tento proces začína v žiadanom stave (výstup systému a žiadaná hodnota sú identické) a potom maximálny alebo minimálny akčný signál začne prepínať. Body fázovej trajektórie tohto procesu simulácie sú uložené pre tréning NN2. Proces simulácie pre systém druhého rádu vo fázovom priestore je ukázaný v Obr. 13. Po úspešnej simulácii, sú tam dve krivky fázových bodov, ktoré sú použité na aproximáciu

t- optimálnej prepínacej funkcie pomocou neurónovej siete NN2.



Obr. 12 Simulácia vo fázovom priestore pre systém II. rádu.

Očividne doba regulácie v druhom kroku je kratšia ako čas v prvom kroku učenia. Žiadaná riadiaca veličina nebola identická. Ako vidieť na Obr. 14, regulátor bude pracovať úspešne aj pri zašumenom meranom signáli o polohu vozíka.



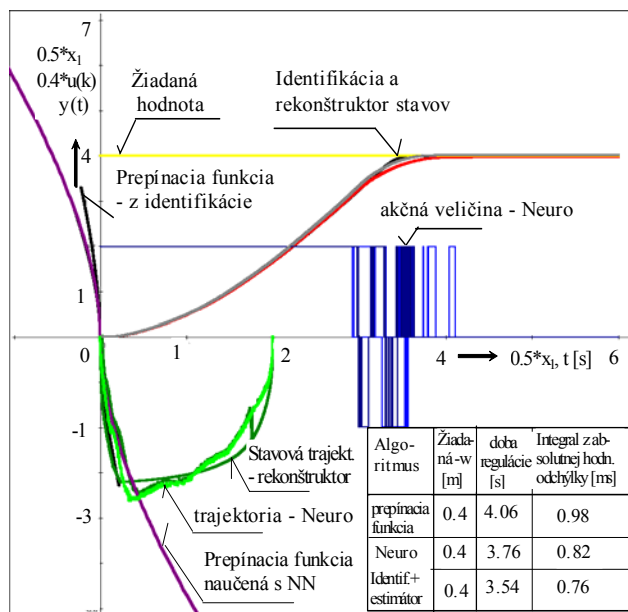
Obr. 13 Simulačný experiment v reálnom čase pre časovo suboptimálny riadiaci obvod s modelom neurónovej siete.

V prípade systému vyššieho rádu, bude simulácia viac komplikovaná. Napríklad, systém 3 rádu musí byť simulovaný v 3D fázovom priestore a výsledná t- optimálna prepínacia funkcia bude reprezentovaná ako 3D plocha.

Simulácia pozostáva z dvoch kombinácií akčných veličín: maximálna => minimálna akčná veličina
minimálna => maximálna akčná veličina

Všetky body simulovanej fázovej trajektórie sú na povrchu 3D prepínacej plochy. Simulácia systémov vyššieho rádu trvá dlhšie, lebo množstvo simulovaných bodov rapidne stúpa z dimenziou fázového priestoru. Výber vhodnej presnosti aproximácie alebo zanedbanie nevýznamných rádov systému pomôže pri zjednodušení problému. Pre stimulačný experiment (číslcová simulácia) so systémom tretieho rádu, ukázaný v prezentácii, bolo realizovaných 5000 bodov fázovej trajektórie. Sieť NS2 bola tvorená z 3 vrstiev a 13 neurónov (6 vstupná vrstva, 6 skrytá vrstva 1 výstupná vrstva).

Posledný simulačný experiment je porovnanie dvoch stratégií učenia, ktoré sú zobrazené v Obr. 14. Sú tam regulované veličiny, fázové trajektórie a prepínacie funkcie pre obidve stratégie: Prvá je výpočet prepínacej funkcie v reálnom čase z identifikácie a druhá je výpočet prepínacej funkcie po off line učení neuronovou sieťou.



Obr. 14 Porovnanie simulačných experimentov v reálnom čase

6. Záver

V mnohých praktických aplikáciách špeciálne v servomechanizmoch, problém t- optimálneho riadenia je obyčajne riešený pre potrebný systém a potom aplikovaný so špecifickými riadiacimi pravidlami. Pre triedu systémov druhého rádu s jedným vstupom a jedným výstupom sa používa riadenie v sklzovom režime. Ak systém nie je stacionárny alebo je tam možnosť že parameter systému sa menia, klasický sklzový režim nemôže byť použitý a učiaci sa regulátor na báze riadenia v sklzovom režime potom môže zabezpečiť požiadavku optimálneho riadenia. Článok popisuje tri učiace sa algoritmy. Prvý je založený na riadení v klasickom sklzovom režime, druhý na sklzovom režime kombinovanom s neuronovou sieťou a tretí je založený na priebežnom výpočte parametrov riadeného procesu

a následnom výpočte prepínacej krivky v reálnom čase v každom intervale vzorkovania, kombinovaným so stavovým estimátorom. Prvý algoritmus je prehľadný, ale učí sa veľmi pomaly, pretože potrebujeme odmerať 5 až 9 odoziev reg. obvodu pre jeden výpočet prepínacej funkcie. Druhý algoritmus, ktorý používa neuronové siete, sa učí rýchlejšie a aby bol plne pochopený je nutné vedieť, ako pacuje prvý. Obidva učiace sa algoritmy opísané v článku nájdu t- optimálnu prepínaciu funkciu pre riadený systém druhého rádu. Kombinovaný algoritmus s NN to dokáže dokonca pre riadený systém tretieho rádu. Tretí algoritmus je najlepší z pohľadu efektívnosti ale nemôže byť použitý pri problémoch kde prepínacia krivka nie je známa ako funkcia.

Apriórnymi podmienkami, je existencia počiatočného stabilného riadenia, napríklad riadenie v sklzovom režime založenom na prepínacej krivke a prepínacej priamke. Popisované algoritmy boli overované na laboratórnom zariadení simulačnými experimentmi v reálnom čase.

References

- [1] Carpenter, G., Grossberg, S.: ART: Stable Self-organization of Neural Recognition Codes in Response to Arbitrary Lists of Input Patterns, *8th Annual Conference of the Cognitive Science Society*, 45-62, 1986.
- [2] Cypkin, J. Z.: *Teorija relejnych sistem avtomatičeskogo regulirovanija*, Gostechizdat, Moskva, 1955.
- [3] Ferrara, A., Giacomini, L.: First and Second Order Sliding Mode Control for a Class of Single-input Nonlinear Systems with Nonmatched Uncertainties, *IMA Journal of Mathematic Control and Information*, 18, 253-268, 2001.
- [4] Liu, T. S., Lee S. W.: A Repetitive Learning Method Based on Sliding Mode for Robot Control, *Journal of Dynamic Systems, Measurement and Control*, ASME, 2000.
- [5] Efe, M. Ö., Kaynak, O., Wilamowski, B. M., Yu, X.: A Robust On-line Learning Algorithm for Intelligent Control Systems, *International Journal of Adaptive Control and Signal Processing*, 17, 489-500, 2003.
- [6] Alexík M., Vittek J.: Adaptive Sliding Mode Control of Position Servosystem. *Preprints of 1st IFAC Workshop on: New Trends in Design of Control Systems*. 7.-9. 9. 1994, Smolenice. pp. 278 - 283.
- [7] Alexík M.: Learning Algorithm for II. Order Systems and Analytic PID Algorithm for III. Order Systems. (in Slovak). In: *Proceedings of the 4th International Scientific-Technical Conference Process Control 2000*, June 11-14, 2000, Kouty nad Desnou, Czech Republic, pp. RIP213. ISBN 80-7194-271-5.