

Optimalizace regulátorů

Garant předmětu:

Prof. Ing. Petr Pivoňka, CSc.

Autor textu:

Prof. Ing. Petr Pivoňka, CSc.

Obsah

1	ÚVOD	7
2	OPTIMALIZACE V ŘÍZENÍ	13
2.1	ŘEŠENÍ OBEČNÉHO OPTIMALIZAČNÍHO PROBLÉMU:	13
2.2	STATICKE A DYNAMICKÉ OPTIMALIZACE	13
2.3	ADAPTIVNÍ ŘÍDICÍ SYSTÉMY	15
2.3.1	<i>Adaptivní regulátory založené na heuristickém přístupu</i>	19
2.3.2	<i>Adaptivní systémy s referenčním modelem</i>	20
2.3.3	<i>Samočinně se nastavující regulátory</i>	21
2.4	OPTIMÁLNÍ ADAPTIVNÍ ŘÍDICÍ SYSTÉMY	26
3	MODEL Y A IDENTIFIKACE PROCESŮ	28
3.1	ÚVOD	28
3.2	NEPARAMETRICKÉ METODY IDENTIFIKACE	29
3.3	PARAMETRICKÉ METODY IDENTIFIKACE MODEL Y MA, AR, ARMA	29
3.4	PARAMETRICKÉ METODY IDENTIFIKACE MODEL ARMAX, ARX	33
3.5	IDENTIFIKACE PROCESU PRO ADAPTIVNÍ ŘÍZENÍ	35
3.6	PRINCIP METODY NEJMENŠÍCH ČTVERCŮ	37
3.6.1	<i>Jednorázová metoda nejmenších čtverců</i>	37
3.6.2	<i>Průběžná metoda nejmenších čtverců</i>	38
3.6.3	<i>Exponenciální zapominání</i>	40
3.7	VOLBA ŘÁDU MODEL U A POČÁTEČNÍ NASTAVENÍ ALGORITMU	41
4	ADAPTIVNÍ REGULÁTORY	43
4.1	VÝPOČET KRITICKÝCH PARAMETRŮ PRO MODEL N-TÉHO ŘÁDU	43
4.1.1	<i>Výpočet kritického zesílení pro model prvního řádu</i>	46
4.1.2	<i>Výpočet kritického zesílení pro model druhého řádu</i>	46
4.1.3	<i>Výpočet kritického zesílení pro model třetího řádu</i>	51
5	OPTIMÁLNÍ REGULÁTORY	54
5.1	DYNAMICKÉ PROGRAMOVÁNÍ	54
5.1.1	<i>Úloha dynamického programování</i>	54
5.1.2	<i>Princip dynamického programování</i>	55
5.1.3	<i>Optimální zvyšování rychlosti a výšky</i>	57
5.2	LINEÁRNÍ OPTIMÁLNÍ REGULÁTORY	58
5.2.1	<i>Úvod</i>	58
5.2.2	<i>Co rozumíme pod pojmem optimální řízení</i>	59
5.2.3	<i>LQ, LQG regulátory</i>	60
5.2.4	<i>Stavový regulátor</i>	61
5.2.5	<i>Definice diskrétního optimálního řízení</i>	66
5.3	ODVOZENÍ LQ ŘÍZENÍ	66
5.3.1	<i>Hledání optimálního řešení v předposledním kroku</i>	66
5.3.2	<i>Dynamické programování</i>	68
5.3.3	<i>Stacionární regulátor</i>	68
5.3.4	<i>Diskrétní LQ regulátor s nekonečným horizontem</i>	69
6	FUZZY REGULÁTORY	71
6.1	FUZZY MNOŽINY A LINGVISTICKÉ PROMĚNNÉ	71

6.2	FUZZY REGULÁTOR, PRINCIPY INFERENCE, FUZZIFIKACE A DEFUZZIFIKACE.....	72
6.3	FUZZY PI, PD, PID REGULÁTORY	76
6.4	FUZZY PI REGULÁTOR	79
6.5	FUZZY PD REGULÁTOR.....	80
6.6	FUZZY PID REGULÁTOR	81
6.7	FUZZY PI/PD/PID REGULÁTORY S NORMALIZOVANÝM TVAREM UNIVERSA	83
6.7.1	<i>Vliv periody vzorkování</i>	83
6.7.2	<i>Metoda návrhu fuzzy PI regulátoru s normalizovaným universem</i>	84
6.7.3	<i>Metoda návrhu fuzzy PD regulátoru s normalizovaným universem.....</i>	90
6.7.4	<i>Metoda návrhu fuzzy PD+PI regulátoru.....</i>	91
6.7.5	<i>Metoda návrhu fuzzy PD+I regulátoru</i>	93
6.7.6	<i>Metoda návrhu fuzzy PI+D regulátoru</i>	95
6.7.7	<i>Metoda návrhu fuzzy P+I+D regulátoru.....</i>	96
6.8	NĚKTERÉ PROBLÉMY VZNIKAJÍCÍ PŘI POUŽITÍ FUZZY REGULÁTORŮ	97
6.9	FUZZY SUPERVIZOR	98
6.10	FUZZY PŘEPÍNAČ.....	98
6.11	FUZZY REGULÁTOR S VÍCE VSTUPY.....	100
7	NEURONOVÉ SÍTĚ V ŘÍDICÍ TECHNICE	102
7.1	ÚVOD.....	102
7.2	OFF-LINE A ON-LINE UČENÍ	104
7.3	VARIANTY ZAPOJENÍ MODELU	105
7.4	NEURONOVÁ SÍŤ JAKO JEDNODUCHÝ NEURONOVÝ REGULÁTOR TYPU PID.....	106
7.5	NEURONOVÉ REGULÁTORY S MODELEM	108
7.6	ADAPTIVNÍ REGULÁTOR S NEURONOVÝM MODELEM.....	109

Seznam obrázků

OBR. 1.1: POROVNÁNÍ PŘECHODOVÝCH CHAR. REGULÁTORŮ PID, PSD A PS (PI) TYPU.....	10
OBR. 1.2: POROVNÁNÍ ODEZEV PŘI POTLAČENÍ PŘEKMITU.....	11
OBR. 2.1: VNITŘNÍ STRUKTURA ADAPTIVNÍHO SYSTÉMU.....	16
OBR. 2.2: OBECNÉ BLOKOVÉ SCHÉMA ADAPTIVNÍHO ŘÍDICÍHO SYSTÉMU.....	17
OBR. 2.3: KLASIFIKACE ADAPTIVNÍCH ŘÍDICÍCH SYSTÉMŮ.....	18
OBR. 2.4: BLOKOVÉ SCHÉMA HEURISTICKÉHO PŘÍSTUPU K ADAPTIVNÍMU ŘÍZENÍ.....	19
OBR. 2.5: ZÁKLADNÍ BLOKOVÉ SCHÉMA ADAPTIVNÍHO SYSTÉMU S REFERENČNÍM MODELEM ..	21
OBR. 2.6: ZÁKLADNÍ BLOKOVÉ SCHÉMA ČÍSLICOVÉHO ADAPTIVNÍHO REGULAČNÍHO OBVODU.	21
OBR. 2.7: VNITŘNÍ ALGORITM. STRUKTURA SAMOČINNĚ SE NASTAVUJÍCÍHO REGULÁTORU	23
OBR. 2.8: BLOKOVÉ SCHÉMA EXPLICITNÍHO SAMOČINNĚ SE NASTAVUJÍCÍHO REGULÁTORU	24
OBR. 2.9: BLOKOVÉ SCHÉMA IMPLICITNÍHO SAMOČINNĚ SE NASTAVUJÍCÍHO REGULÁTORU	25
OBR. 2.10: BLOKOVÉ SCHÉMA AUT. SE NASTAVUJÍCÍHO REGULÁTORU (AUTO-TUNING).	25
OBR. 3.1: MODEL MA (MOVING AVERAGE).....	30
OBR. 3.2: MODEL ARMA.....	32
OBR. 3.3: BLOKOVÉ SCHÉMA REGRESNÍHO MODELU ARX.....	34
OBR. 3.4: SOUVISLOST MEZI PROCESEM A MODELEM PROCESU.....	37
OBR. 3.5: VLIV PERIODY VZORKOVÁNÍ A KVANTIZAČNÍHO EFEKTU A/D PŘEVODNÍKU.....	42
OBR. 4.1: UMÍSTĚNÍ KRITICKÝCH PÓLŮ NA JEDNOTKOVÉ KRUŽNICI.....	44
OBR. 4.2: VÝPOČET PARAMETRŮ REGULÁTORU PRO MODEL DRUHÉHO ŘÁDU	47
OBR. 4.3: UMÍSTĚNÍ KRITICKÝCH PÓLŮ PRO MODEL DRUHÉHO ŘÁDU	48
OBR. 4.4: VÝPOČET PARAMETRŮ REGULÁTORU PRO MODEL TŘETÍHO ŘÁDU	52
OBR. 5.1: VÝBĚR ŘÍZENÍ PRO OPTIMÁLNÍ HODNOTU KRITERIÁLNÍ FUNKCE.....	56
OBR. 5.2: SPOTŘEBA PALIVA PŘI ZMĚNĚ RYCHLOSTI A VÝŠKY	57
OBR. 5.3: KLASICKÉ ŘÍZENÍ VLKA VE SROVNÁNÍ S JINÝMI PŘÍSTUPY.	59
OBR. 5.4: LQG REGULÁTOR V UZAVŘENÉ SMYČCE.	60
OBR. 5.5: VNITŘNÍ STRUKTURA LQG REGULÁTORU V UZAVŘENÉ SMYČCE.....	60
OBR. 5.6: STRUKTURA REGULAČNÍHO OBVODU SE STAVOVÝM REGULÁTOREM S MATICÍ C	62
OBR. 5.7: STRUKTURA REGULAČNÍHO OBVODU SE STAVOVÝM REGULÁTOREM.....	62
OBR. 5.8: STAVOVÝ MODEL PODLE ROVNICE (5.12) A (5.13).....	63
OBR. 5.9: STAVOVÝ REGULÁTOR PRO MODEL PODLE ROVNIC (5.12) A (5.13).....	63
OBR. 5.10: ŘEŠENÍ PŘÍKLADU 5.1.	64
OBR. 5.11: REGULAČNÍ OBVOD PODLE PŘÍKLADU 5.1.....	65
OBR. 5.12: STAVOVÝ MODEL PODLE ROVNICE (5.27) A (5.28).....	65
OBR. 5.13: STAVOVÝ REGULÁTOR PRO MODEL PODLE ROVNIC (5.27) A (5.28).....	65
OBR. 6.1: FUZZY REGULÁTOR VE ZPĚTNOVAZEBNÍM ZAPOJENÍ.....	73
OBR. 6.2: A) SYMETRICKÉ B) NESYMETRICKÉ ROZLOŽENÍ FUNKCÍ PŘÍSLUŠNOSTÍ	74
OBR. 6.3: NELINEÁRNÍ ROZLOŽENÍ FUNKCÍ PŘÍSLUŠNOSTÍ PRO AKČNÍ ZÁSAH.....	75
OBR. 6.4: FUZZY INFERENCE METODOU MIN-MAX	77
OBR. 6.5: FUZZY INFERENCE METODOU PROD-MAX	78
OBR. 6.6: STRUKTURA FUZZY PI REGULÁTORU. RB JE DVOUDIM. BÁZE PRAVIDEL.....	79
OBR. 6.7: FUZZY PI REGULÁTOR S ÚPRAVOU PRO SNADNĚJŠÍ NASTAVOVÁNÍ	79
OBR. 6.8: STRUKTURA FUZZY PD REGULÁTORU. RB JE DVOUDIM. BÁZE PRAVIDEL.....	80
OBR. 6.9: FUZZY PD REGULÁTOR S ÚPRAVOU PRO SNADNĚJŠÍ NASTAVOVÁNÍ.....	80
OBR. 6.10: FUZZY PID REGULÁTOR DANÝ (6.20). RB JE TŘÍDIM. BÁZE PRAVIDEL.....	81
OBR. 6.11: DALŠÍ Z MOŽNÝCH STRUKTUR FUZZY PID REGULÁTORU.....	81
OBR. 6.12: STRUKTURA FUZZY PI+PD REGULÁTORU	82
OBR. 6.13: STRUKTURA FUZZY PI+PD REGULÁTORU PRO SNADNĚJŠÍ NASTAV.	83
OBR. 6.14: NORMALIZOVANÉ SYMETRICKÉ ROZLOŽENÍ FUNKCÍ PŘÍSLUŠNOSTÍ.....	84

OBR. 6.15: VÝPOČET DIFERENCE REGULAČNÍ ODCHYLKY S FILTRACÍ VÝST. VELIČINY	84
OBR. 6.16: TRAJEKTORIE REGULAČNÍHO OBVODU S PI REGULÁTOREM A ZESÍLENÍM K_1	85
OBR. 6.17: PRŮBĚHY VELIČIN PŘI STEJNÉM ZADÁNÍ A $K_2 > K_1$	86
OBR. 6.18: STAVOVÁ TRAJEKTORIE REGULAČNÍHO OBVODU S PI REGULÁTOREM.....	87
OBR. 6.19: MAPOVÁNÍ BÁZE PRAVIDEL FUZZY PI REG. DO DISKR. STAV. ROVINY	87
OBR. 6.20: STRUKTURA FUZZY PI REGULÁTORU S NORMALIZ. ROZSAHEM UNIVERSA	88
OBR. 6.21: PRŮBĚHY VELIČIN V REGULAČNÍM OBVODU S PI REGULÁTOREM	88
OBR. 6.22: PRŮBĚHY VELIČIN V REGULAČNÍM OBVODU S FUZZY PI REGULÁTOREM.....	89
OBR. 6.23: FUZZY PI REGULÁTOR A) $M=10, w=8$; B) $M=3, w=8$	89
OBR. 6.24: ZMĚNA ČASOVÉHO MĚŘÍTKA PŘI SIMULACI	90
OBR. 6.25: REALIZACE FUZZY PD REGULÁTORU S NORM. UNIVERSEM PODLE (6.42)	91
OBR. 6.26: STRUKTURA FUZZY PD+PI REGULÁTORU S NORM. ROZSAHEM UNIVERSA	92
OBR. 6.27: FUZZY PI+PD REGULÁTOR A) OPTIM. NA Ž.H. B) OPTIM. NA PORUCHU	92
OBR. 6.28: NELINEÁRNÍ ROZLOŽENÍ FUNKCÍ PŘÍSLUŠNOSTÍ U FUZZY PI REGULÁTORU	93
OBR. 6.29: FUZZY PD+PI REG. S INFERENCÍ MIN-MAX A NELIN. ROZL. FUNKCÍ PŘÍSL.....	93
OBRÁZEK 6.30: STRUKTURA FUZZY PD+I REGULÁTORU S NORM. ROZSAHEM UNIVERSA	94
OBRÁZEK 6.31: PRŮBĚHY VELIČIN V REGULAČNÍM OBVODU S FUZZY PD+I REGULÁTOREM....	95
OBRÁZEK 6.32: STRUKTURA FUZZY PI+D REGULÁTORU S NORM. TVAREM UNIVERSA.....	95
OBRÁZEK 6.33: STRUKTURA FUZZY P+I+D REGULÁTORU S NORM. TVAREM UNIVERSA	96
OBRÁZEK 6.34: PRŮBĚHY VELIČIN A) S FUZZY PI+D, B) FUZZY P+I+D REGULÁTOREM.....	96
OBRÁZEK 6.35: POSUV FUNKCE PŘÍSLUŠNOSTI ZO V UNIVERSU PRO AKČNÍ ZÁSAH.....	97
OBRÁZEK 6.36: VLIV POSUNU FUNKCE PŘÍSLUŠNOSTI ZO NA OSCILACE	97
OBR. 6.37: REGULAČNÍ OBVOD S FUZZY SUPERVIZOREM	98
OBR. 6.38: FUZZY ADAPTIVNÍ REGULÁTOR	99
OBR. 6.39: JEDNODUCHÁ VARIANTA REALIZACE FUZZY PŘEPÍNAČE	99
OBR. 6.40: INFERENCE MIN-MAX U REGULÁTORU S DVĚMA VSTUPY A JEDNÍM VÝST.....	101
OBR. 6.41: INFERENCE PROD-MAX U REGULÁTORU S DVĚMA VSTUPY A JEDNÍM VÝST.....	101
OBR. 7.1: SYMBOLICKÉ ZNÁZORNĚNÍ UMĚLÉHO NEURONU	102
OBR. 7.2: AKTIVAČNÍ FUNKCE NEURONU	103
OBR. 7.3: TŘÍVRSTVÁ DOPŘEDNÁ NEURONOVÁ SÍŤ (FEED-FORWARD)	104
OBR. 7.4: NEURONOVÝ MODEL SE ZPOŽDĚNÝMI VSTUPY	105
OBR. 7.5: MODEL S NEURONOVOU SÍŤÍ S REKONSTRUKTORY STAVU	105
OBR. 7.6: ŘÍZENÍ PROCESU POMOCÍ NEURONOVÉHO REGULÁTORU.....	107
OBR. 7.7: STRUKTURA JEDNODUCHÉHO NEURONOVÉHO REGULÁTORU PID TYPU.....	107
OBR. 7.8: NEURONOVÝ REGULÁTOR S MODELEM	108
OBR. 7.9: ADAPTIVNÍ REGULÁTOR S NEURONOVÝM MODELEM	109

Seznam tabulek

TABULKA 6.1: A) DVOUDIMENZIONÁLNÍ BÁZE PRAVIDEL B) JEDNA Z ŘADY MODIFIKACÍ.....	74
TABULKA 6.2: DVOUDIMENZIONÁLNÍ BÁZE PRAVIDEL PRO JEMNĚJŠÍ ROZLIŠENÍ.....	75
TABULKA 6.3: JEDNODIMENZIONÁLNÍ BÁZE PRAVIDEL	82

1 Úvod

Skripta jsou určena pro předmět Optimalizace regulátorů a budou využita i jako doplňková skripta pro postgraduální studium předmětu Inteligentní regulátory. Jejich hlavním úkolem je uceleným způsobem seznámit čtenáře se základními přístupy a problémy při návrhu, realizaci, nastavování a optimalizaci parametrů moderních řídicích algoritmů a jejich spojení se základním přístrojovým vybavením. Při jejich tvorbě byl kladen důraz na pochopení fyzikální podstaty problematiky ve spojení potřebné teorie a praktické realizace. Cílem kurzu je seznámit studenty s praktickým návrhem, realizací a nastavováním parametrů pokročilých řídicích algoritmů a algoritmů s principy umělé inteligence při řízení reálných technologických procesů. Podrobně budou probrány všechny aspekty nasazení regulátoru v řídicí smyčce. V semestrálním projektu student navrhne, odladí a ověří jednoduchý adaptivní a optimální regulátor a ověří si vlastnosti fuzzy regulátorů a algoritmů realizovaných na bázi neuronových sítí. Absolvent kurzu by měl být schopen navrhovat, realizovat a seřizovat řídicí systémy se standardně vyráběnými regulátory. Dále by měl zvládnout návrh, nastavení a seřízení složitých řídicích algoritmů, případně být schopen řídicí systémy doplnit novými náročnými řídicími algoritmy a zařadit je do řídicího systému. Předmět navazuje na teoretické základy položené především v předmětech Signály a systémy a Řízení a regulace 1 a Číslicová řídicí technika.

Metody moderní teorie řízení se do praxe dostávají stále velmi omezeně. Rovněž řídicí algoritmy založené na fuzzy logice a neuronových sítích jsou v praxi málo používané. Fuzzy logika přináší nové pohledy na problémy obecně se týkající nejen automatizace. Její výhoda pak je zřejmá u systémů s více vstupy a výstupy.

Ve vývoji metod používajících umělou inteligenci hrají významnou roli rovněž umělé neuronové sítě, jejichž struktury se snaží napodobit biologické neuronové sítě. Použití neuronové sítě v zapojení jako neuronový regulátor nebo model na bázi neuronových sítí představuje velmi zajímavou variantu, která by mohla mít ve vývoji adaptivních systémů velký význam. Proto ve skriptech jsou zpracovány příslušné poznatky.

Podstatný vliv na činnost regulačního obvodu má působení rušivých signálů po celé trase přenosu. Nesprávné provedení i jen malé části celého regulačního řetězce může vést k nevyhovující činnosti celého regulačního obvodu. Proto je velmi důležité správné připojení a zemnění všech prvků regulačního obvodu. Relativně velká úroveň rušivých signálů může rozkmitávat veličiny regulačního obvodu a ve snaze zmenšit tyto kmity pak musíme omezit vliv zejména derivačních složek řídicího algoritmu. Dalším řešením je filtrace rušivých signálů analogovými a číslicovými filtry. Ovšem je třeba mít na zřeteli, že časové konstanty filtrů se přidávají k celkové dynamice řídicího systému. Ve svém důsledku může být v obou případech následkem zpomalení přechodového děje.

Rovněž nesmíme zapomínat, že diskrétní regulátory vnášejí do regulačního obvodu dopravní zpoždění, které zhoršuje fázové poměry. Situace může být dále zhoršena při použití průmyslových sítí, které mohou zavádět další dopravní zpoždění při přenosech údajů. Proto je důležité s tímto vlivem počítat a správně nastavit periodu vzorkování.

Kvantizační šum A/D převodníků může podstatným způsobem ovlivnit zejména průběh identifikace. Proto tyto nelinearity při testování řídicích algoritmů je vhodné při ověřování použít.

Jako velmi důležitý předpoklad pro návrh a ověření jiných typů řídicích algoritmů považují schopnost posluchače realizovat a nastavit diskrétní regulátor typu PID. Realizace regulační smyčky a nastavení PID regulátoru představuje základní dovednost pro regulační techniky v praxi. Dalším důvodem je, že heterogenní regulátory bývají často srovnávány s klasickými PID regulátory. Tato srovnání v naprosté většině případů nejsou korektní, struktura PID regulátoru není správně realizována a jeho parametry bývají často evidentně špatně určeny. Navíc jsem přesvědčen, že bez této základní znalosti není možné dobře seřizovat složitější řídicí algoritmy. Dobře naprogramovaný a nastavený PID regulátor stále zůstává v praxi daleko nejčastěji používaným regulátorem při relativně jednoduchém nastavování. Každá význačnější regulační firma má vypracovány své vlastní varianty řídicích algoritmů a metodiku jejich nastavování. Tyto postupy se vzájemně liší a někdy bývá obtížné najít společnou podstatu, protože algoritmus PID regulátoru je možné realizovat mnoha způsoby. Nejen v české, ale i v cizojazyčné literatuře bývá spojité PID regulátor často odbyt „knižní“ verzí PID regulátoru a z diskrétních tvarů algoritmů bývá uveden paralelní a přírůstkový tvar PSD regulátoru. Ale spojité PID regulátor a jeho diskrétní tvar mají řadu dalších variant, které mají mnohem vhodnější vlastnosti pro implementaci v průmyslových procesech, než jaké nabízí základní definice PID regulátoru. Navíc verze používaná v průmyslových aplikacích musí mít možnost beznázarového přepínání mezi ručním řízením a dalšími řídicími algoritmy používanými v regulačním obvodu. Důležitá je rovněž ochrana před přebuzením integrační složky regulátoru. Pro kvalitní nastavování parametrů řídicích algoritmů by bylo výhodné zajistit jejich průběžné automatické nastavování. Používání simulačních programů, kde je spojité PID regulátor realizován diskrétními simulačními metodami způsobilo, že pokud položíme otázku, za jakých podmínek se simulovaný spojité PID regulátor bude chovat podobně jako reálný (nebo diskrétní PID regulátor chovat podobně jako spojité), pak nejčastější odpověď, pokud ji vůbec dostaneme je, že musí být splněn vzorkovací teorém. Přitom nejen u diskrétních, ale i u simulovaných spojitých PID regulátorů je to podmínka nutná, ale ne postačující. Důsledkem výše uvedené skutečnosti je, že adaptivní, fuzzy, neuronový, regulátor s identifikací pomocí delta operátoru či jiný regulátor vykazuje zdánlivě mnohem výhodnější vlastnosti než špatně realizovaný, nastavený a ověřený spojité nebo diskrétní PID regulátor, přičemž opak by mohl být často pravdou. Důsledkem je pak skepse praxe na tyto nové metody, které by však mohly být v řadě případů přínosem.

Většina vyráběných průmyslových regulátorů je v současné době diskrétního typu, i když výrobci udávají přenos regulátoru jako spojitou přenosovou funkci v Laplaceově transformaci. Na vstupu regulátoru bývá dvanáctibitový (výjimečně i šestnáctibitový) A/D převodník s filtry a zpracovaný signál je z mikroprocesoru vyslán na D/A převodník. Levnější verze mají jen výstup tyristorového či reléového typu se šířkovou modulací výstupního signálu. Typická perioda zpracování kroku algoritmu je od 0,01 do 0,3 s. Reálný proces má časové konstanty zpravidla minimálně o řád větší. Výhodou je možnost realizace řady dalších variant řídicích algoritmů jako je filtrace žádané hodnoty, použití nelineárních řídicích algoritmů a speciálních algoritmů pro procesy s dopravním zpožděním. Vzhledem k tomu, že řídicí algoritmy jsou realizovány v pohyblivé řádové čárce, je rozsah nastavitelných parametrů regulátoru mnohem větší než je tomu u spojitých PID regulátorů. U nepřítliš dražších systémů je samozřejmostí vybavení s hlídáním technologických mezí, sledováním trendů a spolupráce s jinými řídicími systémy pomocí průmyslových sítí. Ovládání usnadňuje grafický panel zobrazující průběhy regulačních pochodů. Řada systémů má i automatické počáteční nastavování parametrů regulátoru, většinou ale jen u smyček s regulací teploty. Pokročilejší regulátory dovolí uživateli používat nejen standardní implementace řídicích algoritmů, ale umožní vytvoření uživatelských specializovaných řídicích algoritmů.

Při vlastní realizaci diskrétního regulátoru musíme zejména dbát na to, aby byl nejen splněn vzorkovací teorém, ale aby byly rovněž spojitým filtrem dostatečně potlačeny všechny rušivé signály vyšší, než odpovídají frekvenci vzorkování, perioda vzorkování byla dostatečně krátká (nebo u simulovaného PID regulátoru simulační krok dostatečně malý) vzhledem k dynamice reálného nebo modelovaného systému a derivační složka PSD regulátoru byla dostatečně vyfiltrována, podobně jako u PID regulátoru. Pouze při splnění těchto předpokladů můžeme uvažovat nad otázkou, je-li porovnání mezi diskrétním PSD a spojitým simulovaným či realizovaným PID regulátorem korektní.

A/D a D/A převodníky zavádějí v diskrétních obvodech kvantování a vytvářejí schodovitou funkci veličin. Důsledkem kvantování je, že nepočítáme s přesností na řádově deset či více platných míst, ale maximálně na 4 platná místa. Délka periody vzorkování vytváří přidavné dopravní zpoždění, které se přičítá k fázi regulovaného procesu a které můžeme aproximovat časovou konstantou dopravního zpoždění $\tau \approx T/2$, kde T je perioda vzorkování. To ve svém důsledku znamená, že nastavení PSD regulátoru podle první metody Zieglera a Nicholse nemusí při nesprávné volbě T odpovídat nastavení PID regulátoru a odezva PSD regulátoru s relativně delší periodou vzorkování při změně žádané hodnoty bude pomalejší a kmitavější než při použití PID regulátoru. Rovněž porucha vstupující do procesu může mít větší překmit a horší regulační pochod.

Vlivy kvantování, působení poruchových veličin a rušení pak např. u adaptivních systémů vyžadují při průběžné identifikaci procesu delší periodu vzorkování, protože prodloužení periody vzorkování ve své fyzikální podstatě způsobuje zvýšení filtračního účinku a získání reálnějšího odhadu procesu při průběžné identifikaci. Porovnávání heterogenních regulátorů s pevně nastaveným PID regulátorem se často omezuje jen na porovnání dynamiky při změně žádané hodnoty a neověřuje se vliv působení poruchových veličin na proces, které bývají spojitým PID regulátorem mnohem lépe vykompenzovány. Proto je žádoucí zkrátit periodu vzorkování. V poslední době se uvádí delta modely, jako prostředky dovolující výrazné zkrácení periody vzorkování [11] , [12]. Pro vytvoření adaptivního modelu s použitím delta operátoru však potřebujeme z reálného procesu získat derivace vyššího řádu, což v reálné praxi představuje omezení pro použití modelů do maximálně druhého řádu bez dopravního zpoždění, tedy pro procesy, kde PID regulátor, případně nelineární PID regulátor velmi dobře vyhovuje. Spojitý PID regulátor můžeme uvažovat také jako jednokrokový prediktivní regulátor, kde optimalizací nastavení D složky regulátoru představuje predikci pro jeden krok. Proto prediktivní regulátory bychom měli porovnávat rovněž s PID regulátory a ne jenom s PI regulátory jak lze v literatuře rovněž někdy najít.

Velmi důležité je rovněž připojení čidla při měření výstupní veličiny regulačního obvodu. Často používané zkušební zapojení s PC a s kartou pro unipolární vstupy a výstupy vykazuje poměrně velkou úroveň rušivých signálů a při zesílení signálu z čidla řádově desítkrát bývá na hranici použitelnosti. Výhodnější řešení představují např. programovatelné automaty, kde volíme bipolární zapojení (plovoucí vstup), nebo lépe proudový vstup 4-20 mA a připojení kroucenými stíněnými vodiči, které správně uzemníme. Ještě lepší je zároveň použít galvanické oddělení. Všechny tyto úpravy mají za cíl zmenšit úroveň rušivých signálů. Pro větší vzdálenosti pak můžeme použít přenos průmyslovými sítěmi. Zde však nesmíme zapomenout na možná zpoždění přenosu dat T_p . Při použití průmyslových sítí je třeba zmínit i fakt, že ne všechny sítě a komunikační protokoly umožní synchronní vzorkování a ne vždy je k dispozici časová značka, tj. údaj o okamžiku kdy byl příslušný vzorek sejmuto. Celková velikost dopravního zpoždění s uvažováním zpoždění ve výpočtu v počítači T_v mezi změřením výstupní veličiny a vysláním akčního zásahu je $\tau_c \approx (T/2 + T_v + T_p)$, kde T_p je zpoždění přenosu dat

Dále si připomeneme vzájemné porovnání různých struktur PID a PSD regulátorů z literatury [4] PIVOŇKA, P.: Číslicová řídicí technika. Skriptum. VUT, Brno, 2004.. Ve všech případech byly regulátory nastavovány podle klasické metody Zieglera a Nicholse [1] na přenosovou funkci:

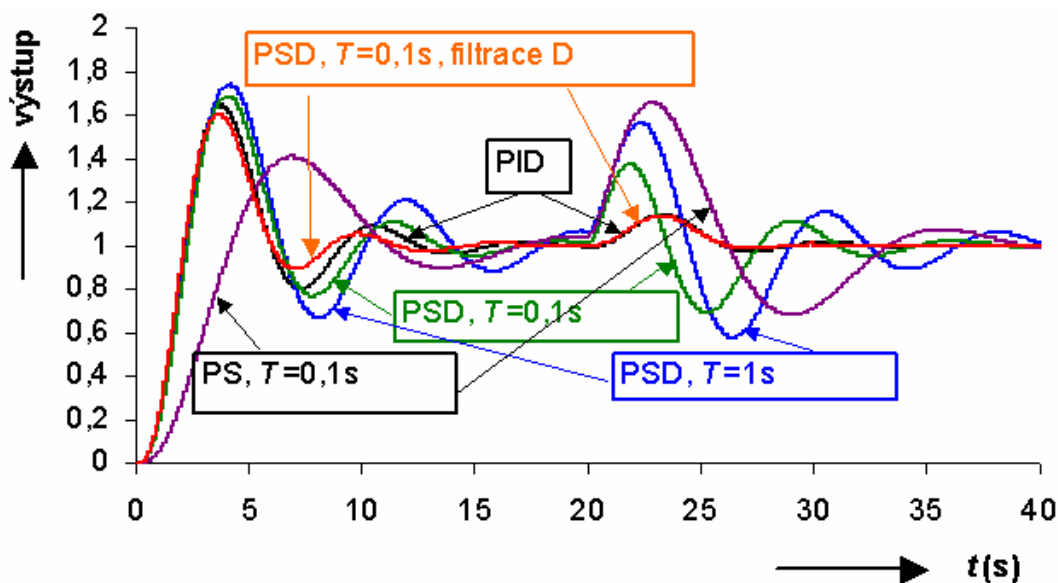
$$F_s(s) = \frac{2}{(10s+1)(s+1)^2} \quad (1.1)$$

V čase 20 s působí jednotkový skok poruchy do části soustavy s přenosovou funkcí:

$$F_{SU}(s) = \frac{1}{(s+1)^2} \quad (1.2)$$

Parametry nastavení regulátoru PID (RK 4, krok integrace 0,002) a PSD (v periodě vzorkování $T = 0,1$ s) byly určeny simulací: $K = 0,6K_{TKRIT} = 0,6 \cdot 12,1 = 7,26$; $T_I = 0,5T_{TKRIT} = 0,5 \cdot 5,7 = 2,85$ s; $T_D = 0,125T_{TKRIT} = 0,125 \cdot 5,7 = 0,712$ s; $N = 3$. PS (stejné parametry dává i PI) regulátor byl nastaven na $K = 0,2K_{TKRIT} = 0,2 \cdot 12,1 = 2,42$; $T_I = T_{TKRIT} = 5,7$ s; $T = 0,1$ s. Spojitá část přenosu je simulována vždy v kroku 0,002 s. Průběh odezvy s PI regulátorem je velmi podobný jako při použití PS regulátoru, kde integrace je nahrazena sumací. S PSD regulátorem v periodě vzorkování $T=1$ s se stejnými parametry je regulační obvod nestabilní, novým experimentem určíme parametry $K_{TKRIT} = 6,4$; $T_{TKRIT} = 8,1$ s; $K = 0,6K_{TKRIT} = 0,6 \cdot 6,4 = 3,84$; $T_I = 0,5T_{TKRIT} = 0,5 \cdot 8,1 = 4,05$ s; $T_D = 0,125T_{TKRIT} = 0,125 \cdot 8,1 = 1,01$ s. V obou případech použití PSD regulátoru bez filtrace derivační složky by bylo výhodnější omezit kmitání nastavením podle upravené metody ZN, uvedené v [4] PIVOŇKA, P.: Číslicová řídicí technika. Skriptum. VUT, Brno, 2004..

Z průběhů vidíme nejen značně horší průběhy při delší periodě vzorkování při změně žádané hodnoty, ale zejména špatné vyregulování poruchy ve srovnání s PID nebo PSD regulátorem s filtrací derivační složky. Přitom perioda vzorkování $T = 1$ s je u přenosové funkce $2/((10s+1)(s+1)^2)$ na hranici použitelnosti u adaptivního regulátoru při řízení reálného systému přes A/D a D/A převodníky. V reálné praxi se pro potlačení překmitů regulované veličiny při změně žádané hodnoty proto používá spíše Takahashiho regulátor [9].



Obr. 1.1: Porovnání přechodových char. regulátorů PID, PSD a PS (PI) typu.

PSD regulátor se v odborných publikacích často používá pro referenční testování nově vyvinutých řídicích algoritmů. Obvyklý scénář těchto zpravidla nekorektních porovnání je, že se zvolí příliš krátká, případně příliš dlouhá perioda vzorkování, nepoužije se patřičný filtr a parametry diskrétního PSD regulátoru se nastaví pomocí pravidel pro spojité PID regulátory první nebo druhou metodou Zieglera a Nicholse. Je zřejmé, že všechny parametry by bylo možné optimalizovat podle typu regulátoru na požadovaný průběh. Cílem bylo ukázat na ošidnost takových porovnávání.

Podobně jako u spojitého I-PD regulátoru můžeme realizovat i diskrétní regulátor I-PD podle rovnice (1.1).

$$U(z) = K (\beta W(z) - Y(z) + \frac{Tz^{-1}}{T_I (1 - z^{-1})} (W(z) - Y(z)) - N \frac{1 - z^{-1}}{1 - e^{-\frac{T}{T_D}} z^{-1}} Y(z)) \quad (1.3)$$

Pro $\beta = 0$ dostaneme S-PD regulátor s filtrací derivační složky, pro $\beta = 1$ diskrétní PS-D regulátor.

Takahashiho regulátor je popsán následující rovnicí:

$$u(k) = K_P [y(k-1) - y(k)] + K_I [w(k) - y(k)] + K_D [2y(k-1) - y(k-2) - y(k)] + u(k-1) \quad (1.4)$$

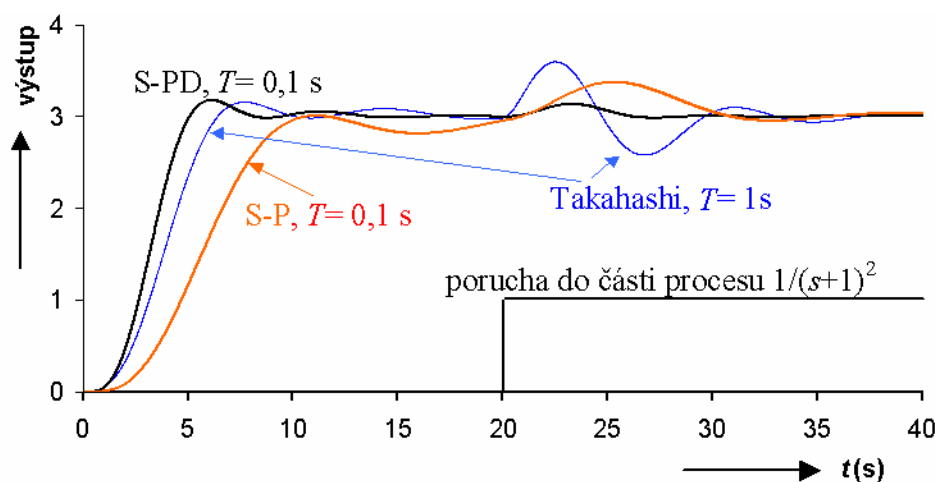
$$K_P = 0,6 K_{TKRIT} - \frac{K_I}{2}; K_I = \frac{1,2 K_{TKRIT} T}{T_{TKRIT}}; K_D = \frac{3 K_{TKRIT} T_{TKRIT}}{40T}$$

Pro stejnou přenosovou funkci a periodu vzorkování $T=1$ s je $K_{TKRIT} = 6,4$; $T_{TKRIT} = 8,1$ s;

$$K_I = \frac{1,2 K_{TKRIT} T}{T_{TKRIT}} = \frac{1,2 \cdot 6,4 \cdot 1}{8,1} = 0,948; \quad K_P = 0,6 K_{TKRIT} - \frac{K_I}{2} = 0,6 \cdot 6,4 - \frac{0,948}{2} = 3,336;$$

$$K_D = \frac{3 K_{TKRIT} T_{TKRIT}}{40T} = \frac{3 \cdot 6,4 \cdot 8,1}{40} = 3,888$$

Porovnání regulátorů je na **Obr. 1.2**. S-PD regulátor s filtrací derivační složky má nastaveny stejné parametry jako PSD s filtrací derivační složky v ($K = 7,26$; $T_I = 2,85$ s; $T_D = 0,712$ s; $N = 3$, $T=0,1$). U regulátoru S-P s původními parametry $K = 2,42$; $T_I = 5,7$ s; $T = 0,1$ s musela být integrační konstanta změněna na hodnotu $T_I = 4,8$ s – pro malou hodnotu prvního překmitu.



Obr. 1.2: Porovnání odezev při potlačení překmitu.

Z průběhů na **Obr. 1.2** a **Obr. 1.2** vyplývá, že při porovnávání jde zejména o působení poruchy v regulačním obvodu. Především relativně dlouhá perioda vzorkování způsobuje, že adaptivní regulátor s průběžnou estimací parametrů bude mít s poruchami problémy. Ve skutečnosti stav bude ještě horší, protože zde nejsou zohledněny vlivy kvantování (které má vliv na proces estimace parametrů u adaptivních regulátorů) a to, že neměřitelná porucha zkresluje identifikaci a určení parametrů je pak podstatně tímto faktem ovlivněno, což celou situaci dále zhoršuje.

Položme si otázku, proč uvažujeme působení poruchy právě do této části. Největší časová konstanta může reprezentovat tzv. globální časovou konstantu procesu, která je určena technologickým procesem. Řádově menší časové konstanty pak určují senzory, ochranná pouzdra atd. Použitím časové transformace pak získáme nekonečně mnoho řídicích smyček, které budou mít podobné vlastnosti (vynásobíme-li všechny časové konstanty u soustavy i regulátoru stejným koeficientem, pak výsledné průběhy mají podobný průběh na časové ose, která je násobkem koeficientu). Globální konstanta představuje filtr, který případné šumy velmi účinně tlumí. V podstatě v procesu existují dva typy rušení. První způsobuje samotný proces (např. nerovnoměrné spalování paliva, poruchy v síťovém napájení atd.) a druhý způsobuje samotné čidlo. Na vedení a napájení čidla se dostávají rušivé signály, které jsou způsobeny především síťovým napětím, ale mohou se tam dostávat i impulsní signály, které způsobují např. silová vedení (změny proudu řádově desítky i stovky ampér, podle typu technologie). Užitečný i rušivý signál přichází z čidla přes zesilovače do řídicího algoritmu. Rušivé signály vyšších frekvencí působí zejména na D složku algoritmu. Proto je potřebné zejména tuto část dobře filtrovat. Používáme i další spojitý filtr na analogové části signálu před jeho převodem na A/D převodníku, jeho časová konstanta by neměla být příliš velká, protože se stává součástí přenosové funkce celého otevřeného obvodu a má tedy samozřejmě vliv na rychlost přechodného děje!

2 Optimalizace v řízení

Cílem kapitoly je připomenout či případně definovat některé základní používané pojmy a struktury používané při řízení procesů. Optimalizace je vyhodnocována podle zvolených hledisek. Pro každý daný případ mohou být zvolena různá hlediska. Často bývá matematicky pro optimální hledisko určena kritériální funkce.

Pojmem optimalizace procesu rozumíme nalezení takového řízení procesu, při němž zajistíme dosažení extrému kritéria optimality.

Optimální systémy v řízení- v určitém smyslu nejlepší vlastnosti řízení – **velký počet hledisek:**

- **optimální řízený pochod** nejlepší průběh přechodových charakteristik- na řízení, na poruchu
- **dosažení nejlepších technologických a ekonomických výsledků řízení** (ztráty, zisk)
 - max. energetická účinnost, nejvyšší jakost, spolehlivost
- **optimální plánování výroby**
- **optimální řízení zásob**

Deterministická optimalizace – dány (známe) všechny vazby.

Nedeterministická optimalizace – některé vazby neznáme.

Statická optimalizace. V podstatě hledání extrému funkce více proměnných – většina ekonomických a technologických problémů.

Dynamická optimalizace – hledání extrému funkcionálu, minimalizace kritéria.

Extremální řízení, regulace úlohou je zpravidla najít nejvyšší účinnost či optimální seřízení z hlediska poměru palivo-vzduch

2.1 Řešení obecného optimalizačního problému:

1. Sestavení matematického modelu systému (analýza a formulace úlohy, formulace problému, sběr a zpracování informací, konstrukce matematického modelu (technologický, ekonomický, fyzikální či jiný model systému), testování modelu.
2. Stanovení kritériální (účelové, cílové, preferenční, optimalizační) funkce – kritérium optimality – optimalizovaná technologická proměnná – zisk, ztráty, produktivita, účinnost.
3. Volba vhodné optimalizační metody – algoritmus optimalizace.
4. Interpretace výsledků.
5. Implementace (praktická realizace)

2.2 Statické a dynamické optimalizace

Statické optimalizace jsou formulovány soustavami algebraických nebo transcendentních rovnic, dovolující optimalizaci ustálených stavů technologických procesů nebo technicko-ekonomických procesů.

Pro hodnocení míry optimality se využívá kritérium ve tvaru:

$$J = f(x) \quad \dim x = n$$

Úkolem optimalizace je nalezení extrému, obvykle je to minimum J^* kritéria optimality pro vektor nezávisle proměnných x z přípustné oblasti Ω :

$$J^* = \min_{x \in \Omega} J(x)$$

Pokud je oblast Ω totožná s množinou reálných čísel R^n , pak se jedná o volný extrém (podmínka nutná pro extrém může mít relativní minimum – maximum jen v bodech pro které platí $\text{grad} f(x) = 0$). V praxi se však vyskytuje vázaný extrém, kdy vazební podmínky můžeme vyjádřit obvykle ve tvaru následujících rovnic a nerovnic:

$$\begin{array}{ll} h(x) = 0 & \dim h = m, m < n & \text{omezení definičního oboru funkce} \\ g(x) \geq 0 & \dim g = p & \text{množina přípustných řešení} \\ x \geq 0 & & \end{array}$$

kde

$h(x) = 0$ podmínka pro vázaný extrém

Výsledkem je nastavení optimálního pracovního bodu, jako jediného cíle. Další nevýhodou je že kritérium se nepřizpůsobuje změnám v procesu, je bez dynamiky.

Možné tvary kritérií použitelných pro statickou optimalizaci:

Lineární:

$$J = c_1 x_1 + c_2 x_2 + c_3 x_3 + \dots + c_n x_n$$

Nelineární:

$$J = c_1 J_1 + \frac{c_2}{J_2}$$

Optimalizace z více hledisek.

Matematická formulace obecné úlohy statické optimalizace je velmi často vzdálená praktickým problémům, protože jediným kritériem s několika pevně nastavenými omezeními nemůžeme řešit problém s dostatečnou přesností. V praxi se proto používá vyjádření kritéria pomocí vektoru

$$f(x) = [f_1(x), f_2(x), f_3(x), \dots, f_n(x)]$$

přičemž relativní významy dílčích kritérií (cílů) $f_i(x)$ jsou základy pro hledání optimálního kompromisu. Vzhledem k tomu, že $f(x)$ je vektor, nemá takto definovaná úloha jediné řešení. Řešení úlohy optimalizace z více hledisek se potom převádí na řešení úlohy nelineárního programování.

Integrální kritéria používaná pro dynamickou optimalizaci:

$$J_a = \int_0^{\infty} |e(t) - e(\infty)| dt$$

$$J_k = \int_0^{\infty} \{ e(t) - e(\infty) \}^2 dt$$

$$J_k = \int_0^{\infty} e(t)^2 dt \text{ pro obvody s astatismem}$$

$$J_{ITAE} = \int_0^{\infty} |e(t) - e(\infty)| t dt$$

$$J(k) = \frac{1}{2} \sum_{i=0}^k c_1 (y(i) - w(i))^2 + \frac{1}{2} \sum_{i=0}^k c_2 u(i)^2 + \frac{1}{2} \sum_{i=0}^k c_3 u(i)^2 \text{ (cena za akční zásah, cena za opotřebení)}$$

2.3 Adaptivní řídicí systémy

Převážná většina procesů, se kterými se v průmyslové praxi setkáváme, má proměnné dynamiky výstupních veličin. Změna parametrů procesu je způsobena změnami v provozních režimech (zejména při změně výkonu), změnami vlastností surovin, paliva, zařízení (stárnutím). Klasické regulátory s pevně nastavenými parametry často nevyhovují pro řízení takových procesů, neboť při změnách parametrů procesu je řízení neoptimální a dochází ke ztrátám materiálu, energie, ke snižování životnosti zařízení atd., se kterými se pevně seřízené regulátory často nemohou vyrovnat.

Jednou s možností zvýšení kvality řízení takových procesů je použití adaptivních řídicích systémů.

Formulace problému adaptivního řízení

Adaptace byl původně jev vlastní živé hmotě, u které se také projevuje v nejuplněnější podobě a s největší mnohotvárností. Je to vlastnost organismů přizpůsobovat svoje chování změnám okolního prostředí, i když jsou tyto nepříznivé.

Každá adaptace představuje pro organismus jistou ztrátu, ať již jde o materiál, energii nebo informace. Živé organismy při mnohonásobném opakování adaptace na určitou změnu prostředí dokáží tyto ztráty minimalizovat. Opakování adaptace je v podstatě akumulace zkušenosti, kterou organismus vyhodnocuje tak, že postupně minimalizuje ztráty vynaložené na adaptaci. Tento jev nazýváme učením.

Mezi systémy, které jsou schopny adaptace, můžeme zařadit vedle systémů přírodních i technické. Jedná se tedy o systémy velmi různorodé, pro jejichž popis se používá nejrůznějších matematických prostředků a proto není možno při definici adaptivních systémů nalézt jednotný matematický aparát. Pro potřeby definice adaptivních systémů pro řízení se omezíme na kybernetické systémy, které splňují následující předpoklady:

- mohou měnit svůj stav nebo strukturu
- můžeme ovlivňovat stav nebo výstup systému.

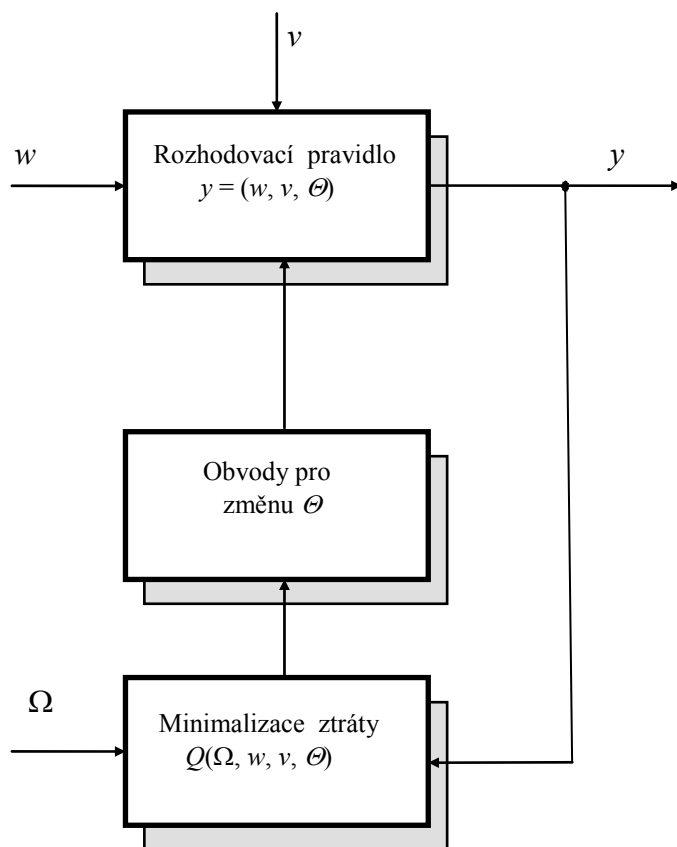
Jedna z možných obecných definicí adaptivního systému:

Adaptivní systém je systém se třemi vstupy a jedním výstupem. Na adaptivní systém působí prostředí tvořené dvěma složkami: řídicí veličinou w a poruchou v . Řídicí veličina je zadávána uživatelem, porucha v bývá zpravidla neměřitelná. Dále na systém přichází informace o jeho požadovaném chování Ω , výstupem systému je chování systému (rozhodovací pravidlo)

$$y = f(w, v, \Theta) \quad (2.1)$$

které přiřazuje každému projevu prostředí w a v jediný výstup y . Změna chování, tj. změna této funkčnosti, se uskutečňuje změnou parametrů Θ . Za parametr Θ se volí pro každou kombinaci (w, v, Θ) takový parametr Θ^* , aby minimalizoval ztrátu Q (za jednotku času nebo určitý časový úsek)

$$Q(\Omega, w, v, \Theta^*) = \min (\Omega, w, v, \Theta) \quad (2.2)$$



Obr. 2.1: Vnitřní struktura adaptivního systému

Adaptace je tedy proces, kterým se hledá Θ^* a trvá tak dlouho, pokud tento parametr není nalezen. Charakteristickým rysem adaptivního systému je skutečnost, že k procesu adaptace dochází vždy, pokud dojde ke změně projevu prostředí w nebo v nebo ke změně požadovaného chování Ω . Nastává-li změna projevu prostředí po každém časovém intervalu T_0 , dochází by mělo dojít k adaptaci vždy na začátku tohoto intervalu. Trvá-li adaptace při tom po dobu τ (po kterou se ztráta Q snižuje), pak střední ztráta bude menší s menším poměrem $\frac{\tau}{T_0}$. Převrácenou hodnotu střední ztráty nazýváme efektivností adaptace.

Učící se systém.

Učící se systém lze chápat jako systém, který po ukončení adaptace pro danou m -tou trojici (w_m, v_m, Ω_m) posloupnosti $\{(w_k, v_k, \Omega_k)\}$, pro $k = 1, 2, \dots, m, \dots, \infty$ si zapamatuje optimální hodnotu parametru Θ^* , čímž si vytváří v paměti funkci

$$\Theta^* = f(w, v, \Omega) \quad (2.3)$$

Po ukončení učení pak pro každý projev prostředí w a v se přímo volí rozhodovací pravidlo výběrem příslušné hodnoty parametru Θ^* z paměti bez adaptace.

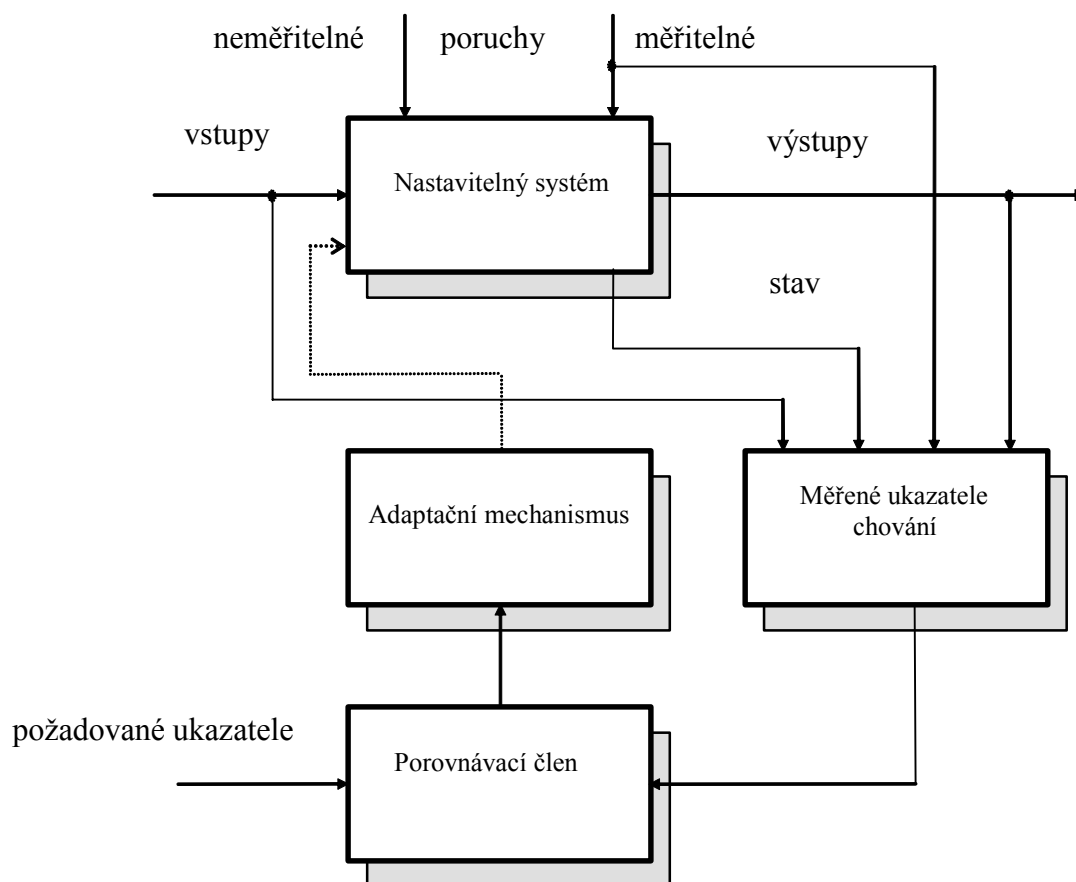
Lze tedy závěrem shrnout, že adaptivní systém adaptaci stále opakuje i při opětovném projevu prostředí a vyžaduje stále informaci o požadovaném chování, učící se systém zhodnocuje opakované adaptace tak, že si pamatuje stav nalezený při adaptaci a při novém projevu prostředí nehledá optimum řešením rovnice (2.5), ale obrátí se ke své paměti.

Pomocí adaptivních a učících se systémů lze řešit následující úlohy:

- Průběžnou (rekurzivní) identifikaci - tj. vytvoření matematického popisu řízeného objektu; používají se samočinně se přizpůsobující modely.
- Řízení systémů, o nichž máme málo apriorních informací a nedovedeme proto předem stanovit strukturu a parametry řídicích algoritmů, případně takových systémů, jejichž přenosové vlastnosti se v průběhu řízení mění.
- Rozpoznávání předmětů nebo situací (scén) a jejich třídění (klasifikace). Adaptivní a učící se systémy jsou pak součástí tzv. klasifikátorů.
- Manipulace s předměty, tj. záměna jejich polohy v prostoru. Adaptivní a učící se systémy jsou pak součástí robotů.

V dalším textu se zaměříme pouze na úlohy adaptivního řízení. Na Obr. 2.2 je znázorněno obecné blokové schéma adaptivního systému, na jehož základě můžeme pro naše další účely formulovat tuto zjednodušující definici:

Adaptivní systém měří určité ukazatele chování daného nastavitelného systému pomocí jeho vstupů, stavů nebo výstupů. Na základě porovnání těchto měřených ukazatelů a množiny požadovaných ukazatelů modifikuje parametry nebo strukturu nastavitelného obvodu nebo generuje pomocný vstup tak, aby měřené ukazatele chování se udržovaly na hodnotách co nejblíže k žadáným ukazatelům.



Obr. 2.2: Obecné blokové schéma adaptivního řídicího systému

Definice je dosti obecná a dovoluje zahrnout převážnou většinu adaptivních úloh technické kybernetiky. Ukazatele chování mohou být v těchto úlohách nejrůznějšího charakteru. Bude-li adaptivní systém využit pro řízení, může být ukazatelem chování např.:

- poloha pólů a nul přenosu uzavřeného regulačního obvodu,

- žádaný překmit přechodové charakteristiky uzavřeného regulačního obvodu na skokovou změnu řídicí nebo poruchové veličiny,
- doba regulace,
- minimální hodnota různých integrálních (v diskrétní verzi sumačních) kritérií,
- amplituda a frekvence vlastních kmitů u nelineárních obvodů,
- frekvenční spektrum uzavřeného regulačního obvodu,
- zvolená hodnota amplitudové a fázové bezpečnosti apod.

Pro účely automatické regulace lze definici adaptivního systému formulovat ještě jednodušším způsobem:

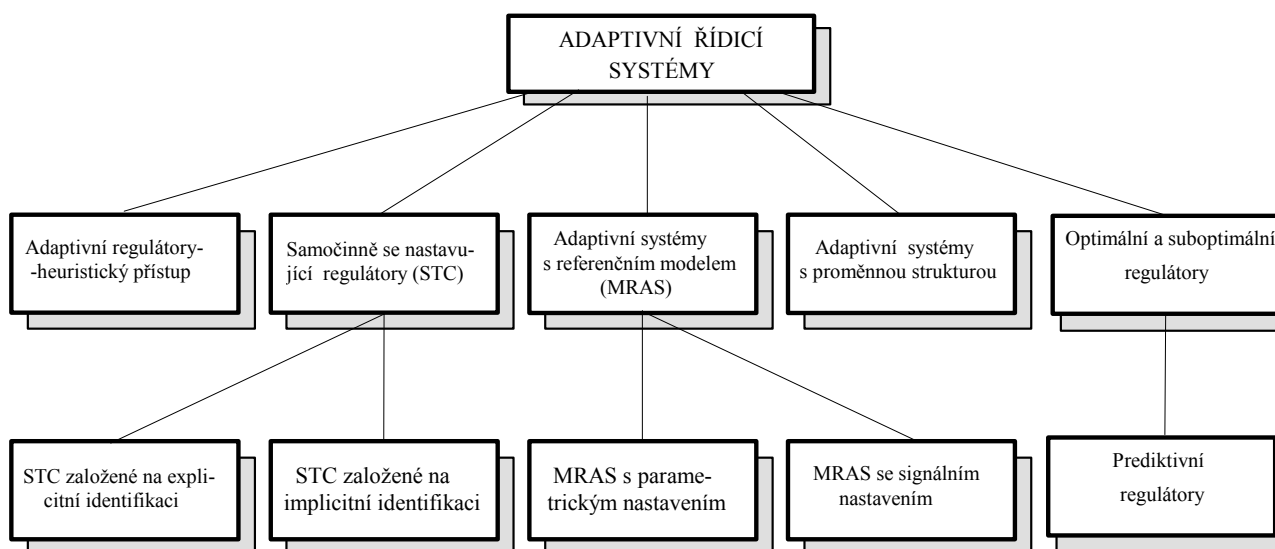
Adaptivní řídicí systémy přizpůsobují parametry nebo strukturu jedné části systému (regulátoru) změnám parametrů nebo struktury jiné části systému (regulované soustavy) tak, aby celý systém měl neustále optimální chování podle zvoleného kritéria nezávisle na změnách, které nastaly.

Adaptaci na změnu parametrů nebo struktury soustavy lze uskutečnit v podstatě třemi způsoby:

- vhodnou změnou stavitelných parametrů regulátoru,
- změnou struktury regulátoru,
- generováním vhodného přídavného vstupního signálu (adaptace vstupním signálem).

Klasifikace adaptivních řídicích systémů

Rozdíl mezi klasickým zpětnovazebním řídicím algoritmem a adaptivním řídicím algoritmem spočívá v tom, že klasický regulátor využívá principu zpětné vazby k tomu, aby kompenzoval neznámé poruchy a stavy v procesu a sledoval změny žádané hodnoty. Zpětná vazba určuje velikost regulační odchylky $e = w - y$. Regulační odchylka vstupuje do pevně nastaveného řídicího algoritmu, který vypočítá hodnota vstupního signálu (akční veličiny) u do soustavy. V každé situaci je způsob zpracování regulační odchylky stejný. Podstatou adaptivního systému je, že mění způsob zpracování regulační odchylky, tj. adaptuje řídicí zákon na neznámé podmínky a rozšiřuje oblast praktických případů, ve kterých lze dosáhnout kvalitní regulaci. Adaptivitu lze chápat jako zpětnou vazbu vyšší úrovně, která mění parametry regulátoru podle požadované kvality regulačního pochodu.



Obr. 2.3: Klasifikace adaptivních řídicích systémů.

Teorie adaptivního řízení zaznamenala v posledních letech značný rozvoj. Samozřejmě jako v každé nově se vyvíjející vědní disciplíně, ani v teorii adaptivního řízení není jednotný přístup ke klasifikaci systémů, pracujících na tomto principu. Pro naše účely vystačíme s tříděním naznačeným na obr. 2.3, učící se systémy nejsou uvažovány.

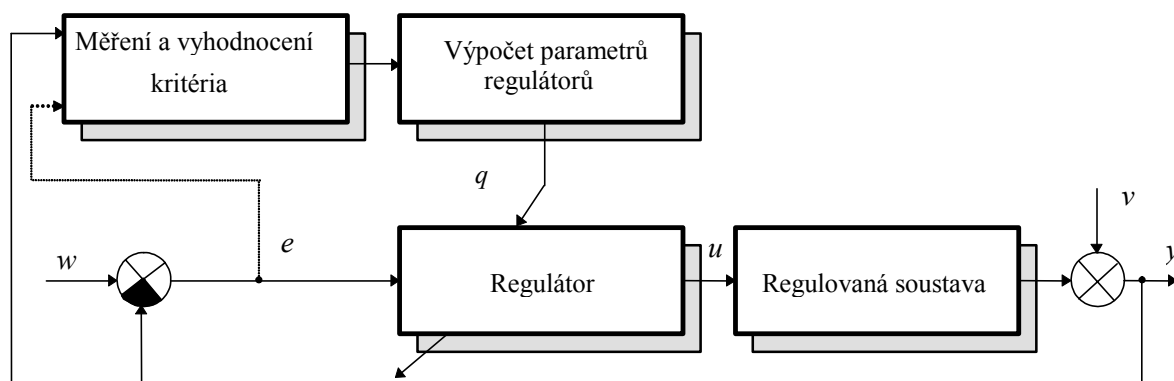
Adaptivní systémy založené na heuristickém přístupu, samočinně se nastavující regulátory (Self-Tuning Controllers - STC) a adaptivní systémy s referenčním modelem (Model Adaptive Reference Systems - MRAS) jsou tři základní přístupy k problému adaptivního řízení v současné době. U adaptivních systémů s proměnnou strukturou se záměrně podle stanovených pravidel mění jejich struktura. Protože takový systém mění svou strukturu na základě zkušeností v podmínkách své předchozí činnosti, je možno jej zařadit mezi samočinně se organizující systémy.

2.3.1 Adaptivní regulátory založené na heuristickém přístupu

Při metodách využívajících tohoto přístupu se zajišťuje adaptivita přímo vyhodnocováním průběhu regulované veličiny (případně její odchylky) nebo vybraného kritéria kvality regulačního pochodu. Často se v těchto případech využívá algoritmu číslicového PID regulátoru a jako kritérium se obvykle volí míra kmitavosti regulované veličiny nebo její odchylky. Tyto metody nevyžadují identifikaci regulované soustavy, v některých případech není třeba ani sledovat poruchové veličiny, či zavádět zvláštní zkušební signály. Blokové schéma těchto metod je uvedeno na obr. 2.4. Regulovaná veličina y , případně regulační odchylka e se vyhodnotí vzhledem k žádanému kritériu a následuje změna nastavení parametrů PID regulátoru.

Při syntéze takových regulátorů je snahou optimalizovat kritérium, které kvantifikuje kvalitu průběhu regulačního pochodu. I když tento přístup často vyhovuje požadavkům praxe a je rovněž výhodný z hlediska robustnosti regulátorů, naráží na řadu fyzikálních problémů a podařilo se jej úspěšně realizovat jen v jednodušších případech. Ukázal se však jako výhodný pro regulaci teploty zejména u procesů s delšími časovými konstantami.

Jednou z úspěšných aplikací je přístup navržený Maršíkem [36], [37]. Metoda je založena na nastavování zesílení PID regulátoru, jako přímo měřitelné kritérium je volena míra kmitavosti. Je známo, že regulační pochod, který se blíží hranici stability je kmitavější, na druhé straně příliš tlumený pochod nemívá překmit. Existuje několik modifikací tohoto typu regulátoru, některé z nich jsou nenáročné na výpočetní prostředky i paměť.



Obr. 2.4: Blokové schéma heuristického přístupu k adaptivnímu řízení

Åström v [38] zavedl vedle pojmu „self-tuning“ (samočinné nastavování) pojem „auto-tuning“ (automatické nastavování). Automatické nastavování nelze považovat za nastavení samočinné. Typem tohoto regulátoru, jenž našel poměrně široké praktické uplatnění, je automaticky se nastavující regulátor navržený Åströmem a Hägglundem [39], [40], u něhož se do zpětné vazby zařadí vedle PID regulátoru paralelně nelinearita typu relé. V seřizovací fázi je do zpětné vazby zařazeno relé, čímž se regulační obvod rozkmitá na kritické frekvenci. Poněvadž akční veličina u nabývá pouze dvou hodnot $\pm R$ a má tedy obdélníkový průběh, má regulovaná veličina y průběh přibližně sinusový, jehož zkreslení závisí na tom, jak soustava odfiltruje z akční veličiny vyšší harmonické. Kritické zesílení K_{krit} se potom vypočítá jako poměr amplitudy 1. harmonické obdélníkového průběhu vypočítané z Fourierova rozvoje a amplitudy regulační odchylky e_{max} podle vztahu:

$$K_{\text{krit}} = \frac{4R}{\pi e_{\text{max}}} \quad (2.1)$$

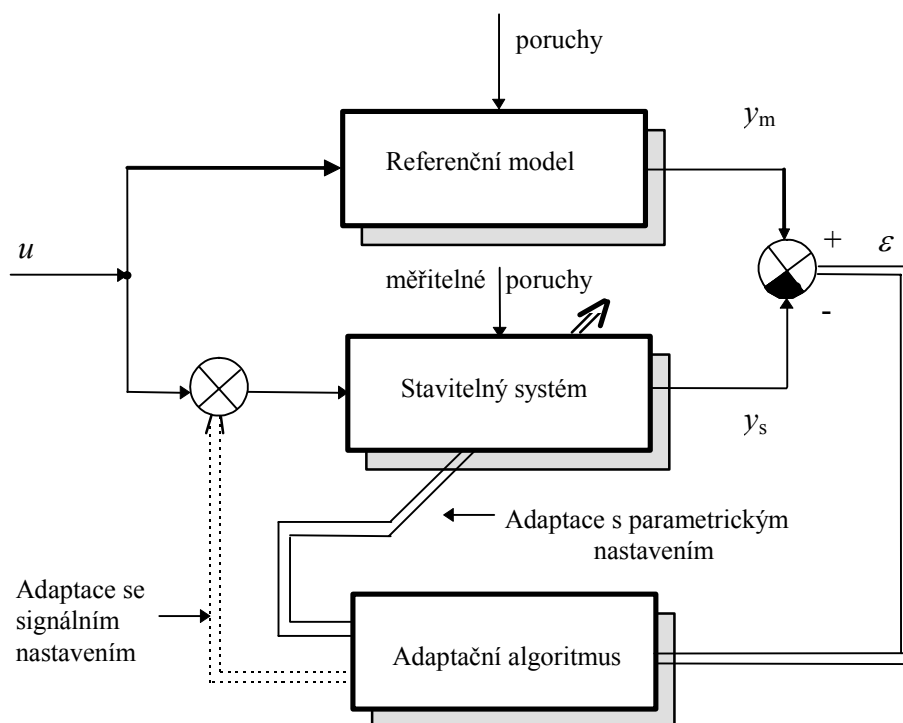
Kritická perioda kmitů T_{krit} se změří. Pro seřízení PID regulátoru se použije Ziegler - Nicholsova metoda a do zpětné vazby se připojí místo relé PID regulátor.

Regulátor je možné automaticky seřídít rovněž vyhodnocením přechodových dějů. Na tomto principu je založen regulátor EXACT (Expert Adaptive Controller Tuning) firmy Foxboro. Tento automaticky se nastavující regulátor je založen na metodě rozpoznávání obrazů (Pattern Recognition Approach), tj. na znalosti průběhu regulační odchylky při přechodovém ději. Regulátor využívá pro seřízení tři vrcholy průběhu regulační odchylky a z těchto tří hodnot se pak počítá překmit a tlumení, které využívá spolu s periodou kmitů pro nastavení parametrů PID regulátoru. Řada autorů navrhla seřízení PID regulátoru na základě měření odezvy na skokovou změnu žádané hodnoty regulované veličiny v otevřené nebo uzavřené smyčce. Optimalizace parametrů PID regulátoru se provádí pomocí výpočtu integrálních lineárních nebo kvadratických kritérií kvality regulačního pochodu.

2.3.2 Adaptivní systémy s referenčním modelem

V odborné literatuře je často diskutována je problematika návrhu adaptivních systémů s referenčním modelem [41], [42]. Základní princip adaptivního systému s referenčním modelem je na obr. 2.5. Referenční model dává žádanou odezvu y_m případně požadovaný stavový vektor x_m na vstupní signál u . Cílem adaptace je konvergence statických a dynamických vlastností stavitelného systému, tj. uzavřeného regulačního obvodu, k vlastnostem referenčního modelu. Jedná se v podstatě o adaptivní systém s vynuceným chováním, přičemž porovnáním tohoto vynuceného chování s chováním (odezvou) stavitelného systému (regulačního obvodu) y_s dostaneme adaptační odchylku ε . Úkolem vhodného adaptačního zařízení je minimalizovat odchylku ε , případně odchylku vektoru stavu x referenčního modelu a stavitelného systému při zadaném kritériu. Toto se děje buď změnou parametrů stavitelného systému nebo generováním vhodného vstupního signálu, jak je zřejmé z obr. 2.5.

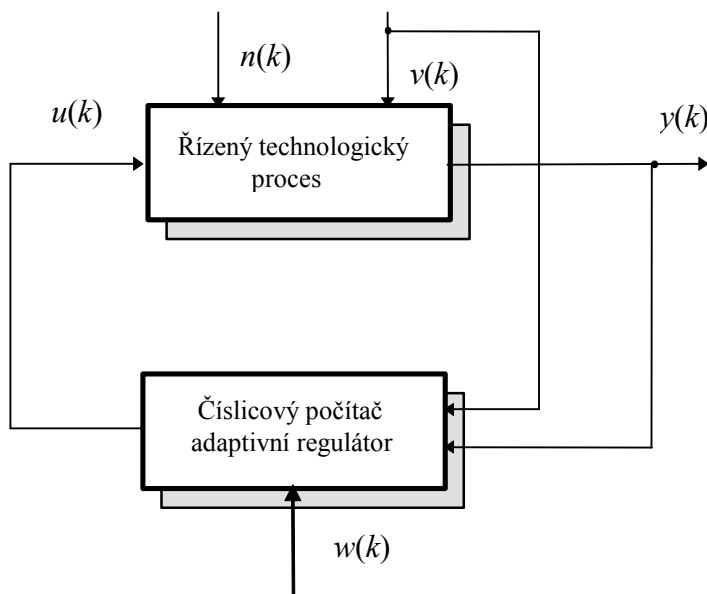
Významnou vlastností tohoto adaptivního systému je jeho duální charakter, tzn., že je možno jej použít jak pro řízení, tak i pro identifikaci parametrů modelu procesu nebo estimaci stavu systému. I když je teorii adaptivních systémů s referenčním modelem věnována v odborné literatuře značná pozornost, je zveřejněných praktických aplikací poměrně málo. Je to dáno tím, že zpočátku se vyvíjely adaptivní systémy s referenčním modelem pouze pro určitou třídu soustav (návrhy autopilotů a servomechanismů).



Obr. 2.5: Základní blokové schéma adaptivního systému s referenčním modelem

2.3.3 Samočinně se nastavující regulátory

Ve výše dvou uvedených přístupech nebyly třeba pro návrh adaptivního regulátoru detailní znalosti dynamického chování regulované soustavy. Další přístup k adaptivnímu řízení je založen na průběžném odhadování vlastností soustavy a poruch, postupném upřesňování a tím i sledování možných změn. Takový regulátor, založený na identifikaci neznámého procesu s následnou syntézou řízení (adaptivní řízení s průběžnou identifikací) je v literatuře označován jako samočinně se nastavující regulátor (Self-Tuning Controller - STC). Vývojem, výzkumem a aplikacemi tohoto přístupu se zabývá celá řada významných pracovišť jak v zahraničí, tak i v České republice.



Obr. 2.6: Základní blokové schéma číslicového adaptivního regulačního obvodu.

Předpokládejme řízený technologický proces s jednou vstupní akční veličinou $u(k)$ a s jednou výstupní regulovanou veličinou $y(k)$. Na řízený proces dále může působit měřitelná poruchová veličina $v(k)$ a neměřitelná poruchová veličina $n(k)$ - náhodný šum. K řízenému procesu je ve zpětné vazbě připojen řídicí počítač ve funkci číslicového adaptivního regulátoru, který mimo jiné zpracovává žádanou hodnotu regulované veličiny $w(k)$. Blokové schéma tohoto základního zpětnovazebního obvodu je znázorněno na obr. 2.6.

Adaptivní číslicový regulátor pracuje s pevně zadanou periodou vzorkování T . S touto periodou generuje posloupnost číselných hodnot akčního signálu $\{u(k); k = 1, 2, \dots\}$ (předpokládáme $T = 1$), kterými je přes číslicově-analogový převodník ovládán akční orgán v regulačním obvodu. Hodnota akční veličiny je po dobu intervalu vzorkování konstantní. Akční orgán včetně číslicově-analogového převodníku je zahrnut do dynamiky řízeného procesu. Výstupem procesu, který má být řízen, je fyzikální (zpravidla spojitě proměnná) veličina rovněž vzorkovaná s periodou T . Z hlediska regulátoru je tedy regulovanou veličinou posloupnost číselných hodnot $\{y(k); k = 1, 2, \dots\}$, která je jedinou informací, kterou regulátor o průběhu spojitěho výstupu získává. Někdy může být účelné spojitý výstup soustavy před vzorkováním upravit filtrací. Čidlo, analogovo-číslcový převodník a případný filtr se rovněž považuje za součást řízeného procesu.

Základní zpětnovazební obvod může být doplněn dopřednou vazbou od externí měřené poruchy $v(k)$, pokud je přístupná měření. Její průběh je rovněž vzorkován s periodou T a předáván regulátoru jako posloupnost číselných hodnot $\{v(k); k = 1, 2, \dots\}$, rovněž žádaná hodnota regulované veličiny je zadávána číslicově jako posloupnost číselných hodnot $\{w(k); k = 1, 2, \dots\}$.

Existence náhodných neměřitelných poruch $n(k)$ a případná změna žádané hodnoty regulované veličiny $w(k)$ je tedy důvodem, proč se regulace zavádí a tedy cílem regulace je co nejlepší kompenzace těchto poruch a sledování žádané hodnoty regulované veličiny. Navíc předpokládáme, že parametry řízeného procesu jsou buď konstantní ale neznámé, nebo proměnné, přičemž změna těchto parametrů je podstatně pomalejší než rychlost procesu adaptace. Použitím adaptivního řízení s průběžnou identifikací pak podle povahy řízeného procesu sledujeme splnění následujících cílů:

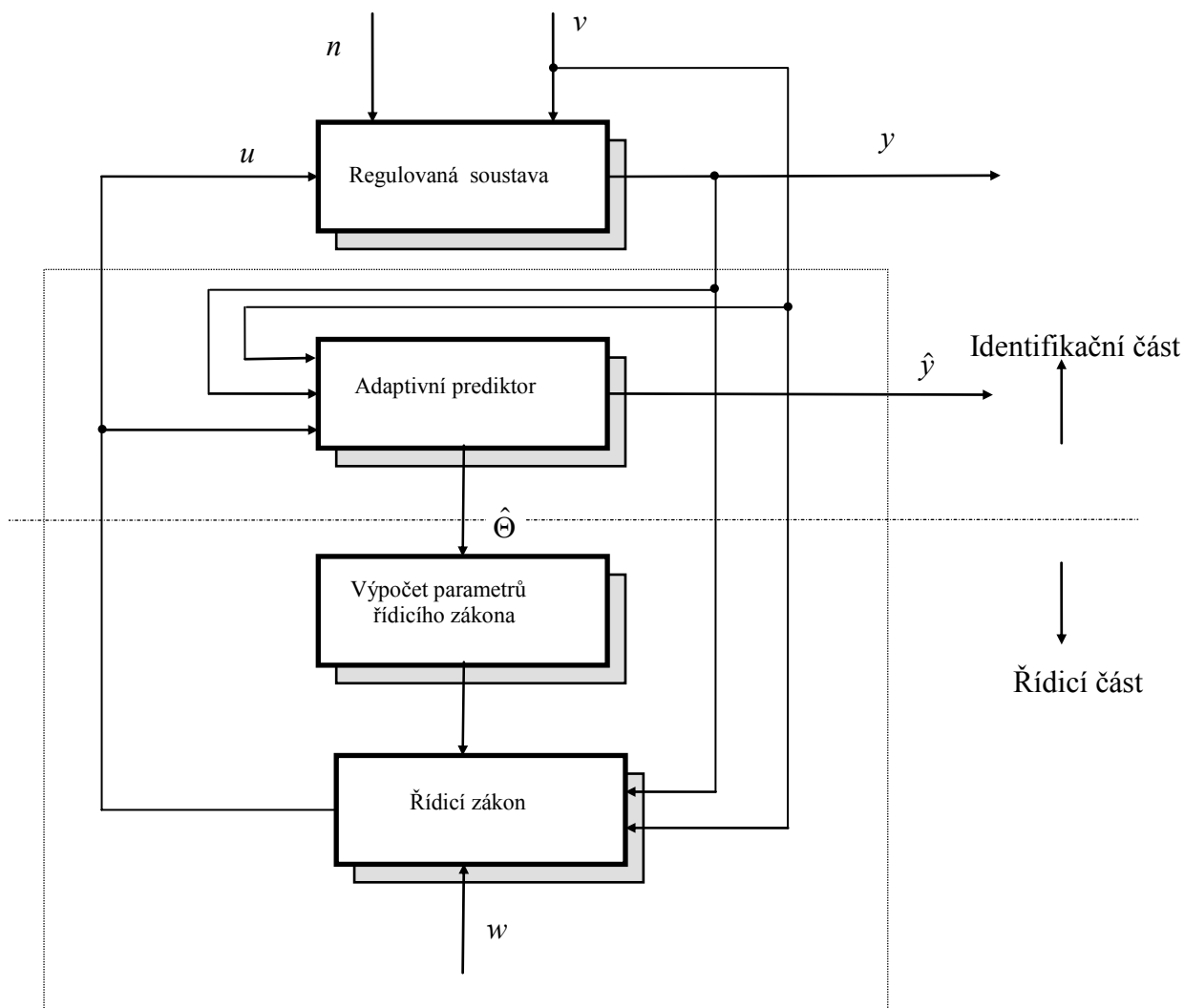
- Automatické seřízení číslicového regulátoru.
- Zlepšení regulace při přítomnosti nestacionárních poruch.
- Zachycení změn parametrů řízené soustavy, které mohou být způsobeny různými technologickými příčinami, např. provoz zařízení při různých provozních režimech.
- Následné zlepšení regulačních pochodů daného procesu vhodnou změnou parametrů číslicového regulátoru.

Algoritmická struktura samočinně se nastavujícího regulátoru

Je zřejmé, že k dosažení výše uvedených cílů, hraje důležitou roli mimo vlastní optimální strategie řízení rovněž poznání dynamických a statických vlastností daného procesu - jeho identifikace. Z teorie odhadu parametrů je však známo, že odhady parametrů jsou vždy zatíženy určitou neurčitostí - chybou. Tato neurčitost závisí nejen na počtu identifikačních kroků (tj. na počtu vzorkovaných dat) a na volbě struktury matematického modelu řízeného procesu, ale i na průběhu akčních veličin, periodě vzorkování a volbě filtrů akční a regulované veličiny. To znamená, že každá provedená změna akční veličiny kromě požadovaného řídicího účinku vyvolává rovněž řízenou soustavu a tím vytváří podmínky pro její identifikaci, jinými slovy, aby řízený proces byl co nejlépe poznán, je nutno klást na průběh akčních zásahů určité podmínky. Obecná úloha adaptivního řízení s průběžnou identifikací je tedy velmi složitá, protože je v ní nutno nalézt takovou posloupnost akčních zásahů, která zajišťuje, aby střední hodnota regulované veličiny se co nejvíce blížila k žádané hodnotě, a současně aby umožňovala co nejlepší identifikaci daného procesu. Není známo

v současné době řešení této úlohy vyhovující v praxi. Daná úloha se řeší zjednodušeně na základě experimentálních zkušeností a intuice. Toto řešení se nazývá vnucená separace identifikace a řízení (Certainty Equivalence). Princip zjednodušení spočívá v následujícím postupu:

1. Vektor parametrů Θ modelu procesu se pro daný krok řízení považuje za známý a rovný bodovému odhadu, který je v daném okamžiku k dispozici, tj. $\Theta = \hat{\Theta}(k-1)$.
2. Za tohoto předpokladu se navrhne strategie řízení pro zvolené kritérium kvality řízení a vypočítá se právě potřebný akční zásah $u(k)$.
3. Po získání nového vzorku regulované veličiny $y(k)$ (resp. externí měřené poruchy $v(k)$) a známého akčního zásahu $u(k)$ se provede další identifikační krok pomocí rekursivního identifikačního algoritmu. To znamená, že nová informace o procesu, kterou nese trojice dat $\{u(k), y(k), v(k)\}$, se použije k aktualizaci odhadu $\hat{\Theta}(k-1)$ a celý postup se opakuje pro nový odhad $\hat{\Theta}(k)$.

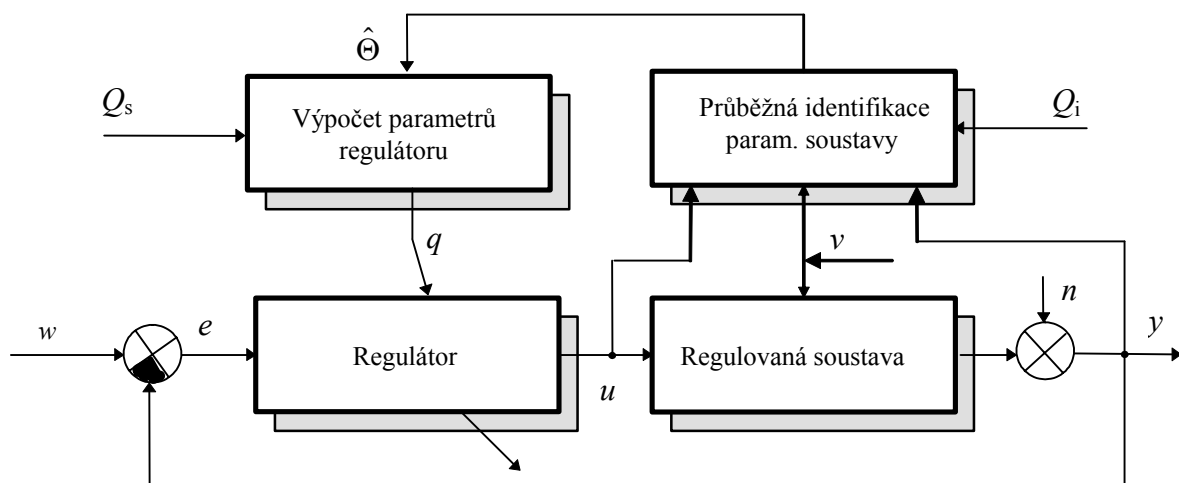


Obr. 2.7: Vnitřní algoritm. struktura samočinně se nastavujícího regulátoru

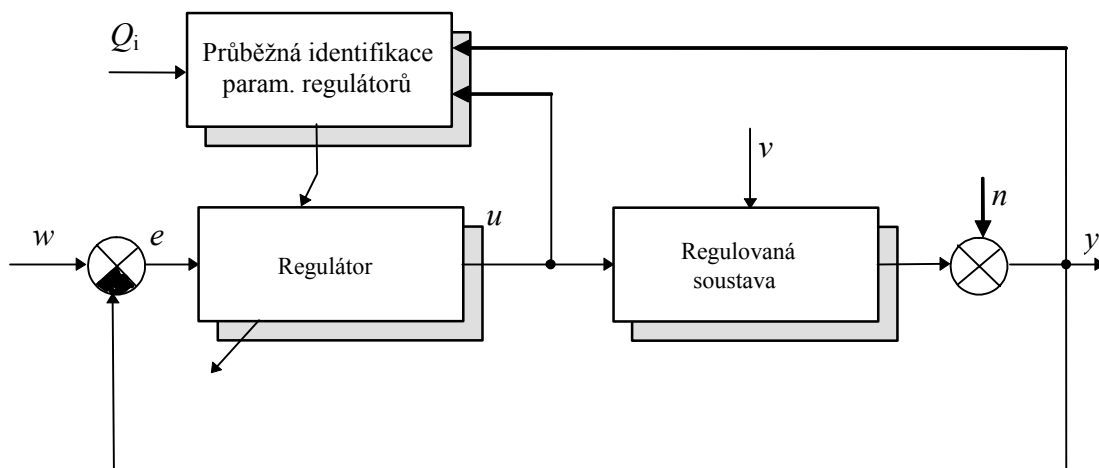
Experimentální výsledky ukázaly, že převážná část praktických úloh adaptivního řízení s průběžnou identifikací vyhovuje danému zjednodušujícímu přístupu. Vnucenou separací identifikace a řízení se rozpadá vnitřní struktura regulátoru na identifikační a řídicí část, jež jsou spojeny pouze přenosem bodových odhadů parametrů $\hat{\Theta}(k)$. V identifikační části se provádí průběžný odhad parametrů modelu procesu, pomocí nichž se počítá predikovaný odhad $\hat{y}(k)$ výstupní (regulované) veličiny $y(k)$. Řídicí část obsahuje blok pro výpočet parametrů regulátoru (řídicího zákona), které se počítají pomocí hodnot odhadů parametrů modelu procesu $\hat{\Theta}(k)$. Parametry regulátoru pak slouží pro výpočet hodnoty akční veličiny $u(k)$ v každé vzorkovací periodě.

Z uvedené struktury je patrné, že základní podmínkou dobré činnosti regulátoru je rychle konvergující a spolehlivá identifikace. Ačkoliv jsou kladeny na syntézu adaptivního řízení určité specifické podmínky, je možno říci, že při dobře fungující identifikaci lze syntézu provádět podle známých algoritmů, jako jsou algoritmy pro minimální počet kroků regulačního pochodu (dead beat), minimalizace rozptylu výstupu (MV - Minimum Variance), zobecněná metoda minimalizace výstupu (GMV - Generalized Minimum Variance), řízení podle lineárního kvadratického (Linear Quadratic), metoda přiřazení pólů (Pole Placement) a metody číslkové syntézy PID regulátorů. Algoritmy samočinně se nastavujících regulátorů, uvedených v této monografii, se budou lišit pouze v řídicí části, v identifikační části budeme používat rekurzivní metodu nejmenších čtverců (RLSM - Recursive Least Squares Method).

U některých samočinně se nastavujících regulátorů neslouží identifikační procedura k určování odhadů parametrů modelu procesu $\hat{\Theta}(k)$, ale vhodnou konfigurací regulačního obvodu lze průběžně odhadovat přímo parametry regulátoru. To znamená, že je nutné nalézt vztah mezi vstupem a výstupem procesu a definovat jej přímo pomocí parametrů regulátoru bez toho, že by se tyto parametry přepočítávaly z odhadů parametrů modelu procesu. Pro tyto regulátory se používá názvu přímé (implicitní), kdežto regulátory využívající v syntéze odhady parametrů modelu procesu se nazývají nepřímé (explicitní). Jestliže zobrazíme explicitní samočinně se nastavující regulátor blokovým schématem podle obr. 2.8, které je analogickou formou blokového schéma z obr. 2.7 (Q_i je kritériem identifikace, Q_s kritériem syntézy řízení, $q(k)$ jsou parametry regulátoru), můžeme blokové schéma implicitního samočinně se nastavujícího regulátoru znázornit obr. 2.9.

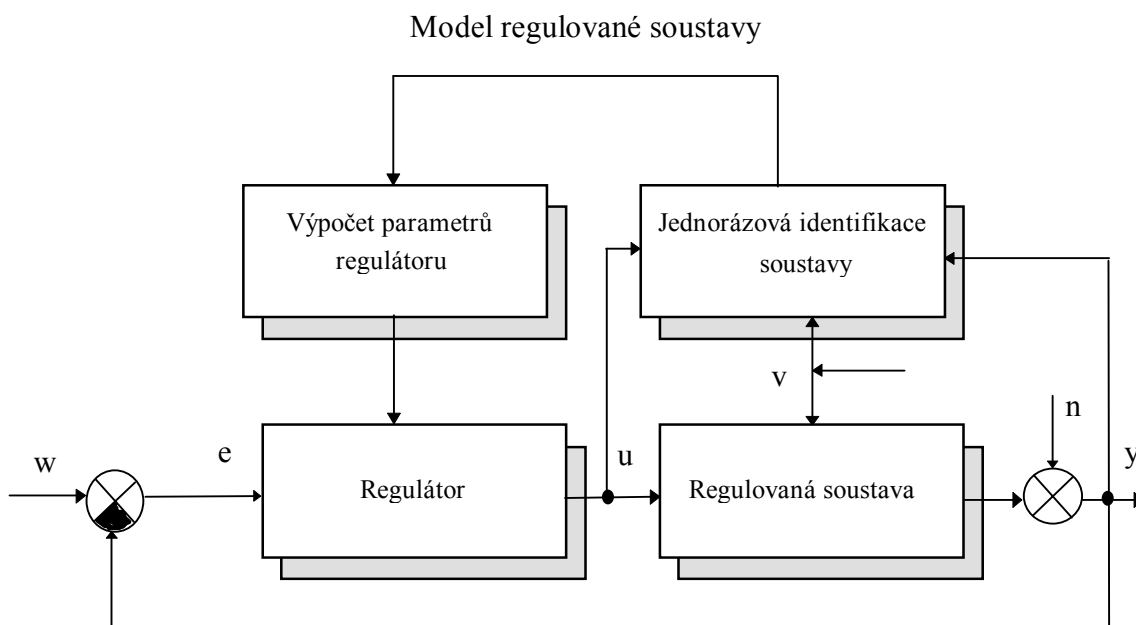


Obr. 2.8: Blokové schéma explicitního samočinně se nastavujícího regulátoru



Obr. 2.9: Blokové schéma implicitního samočinně se nastavujícího regulátoru

Principu samočinně se nastavujícího regulátoru lze použít i pro jeho jednorázové automatické nastavení (auto-tuning). Je-li pro automatické nastavování regulátoru použit algoritmus podle obr. 2.7, potom jsou bloky představující průběžnou identifikaci a výpočet parametrů regulátoru zapojeny pouze tehdy, když se provádí nastavování regulátoru - tj. v seřizovací fázi. Po seřízení regulátoru se vypojují. Soustava je pak regulována s pevně nastavenými parametry. Je zřejmé, že tento způsob regulace je vhodný pro řízení deterministických procesů, kdy identifikace se vypíná po seřízení regulátoru. Princip tohoto typu regulátoru je zřejmý z obr. 2.10.



Obr. 2.10: Blokové schéma aut. se nastavujícího regulátoru (auto-tuning).

2.4 Optimální adaptivní řídicí systémy

Řešení úlohy optimálního řízení:

Určíme cíl řízení - nejdůležitější, matematicky to formulujeme jako úlohu dosažení extrému funkcionálu J ve formě kritéria optimality:

$$J = f(x(t), y(t), u(t), v(t), t)$$

kde $x(t)$ je vektor stavových proměnných, $y(t)$ je vektor výstupních veličin, $u(t)$ je vektor akčních veličin, $v(t)$ je vektor poruchových veličin.

Jestliže optimálnímu řízení $u^*(t)$ odpovídá

$$J^* = f(x(t), y(t), u^*(t), v(t), t) \quad (2.5)$$

kde J^* nabývá hodnot maxima nebo minima, pak funkcionál J^* se nazývá kritérium optimality

Základní úloha optimálního řízení.

Základní úlohu optimálního řízení formulujeme jako úkol stanovit pro proces, jehož chování lze popsat

$\dot{x}(t) = f(x(t), u(t), t)$ s omezením na vektor stavu x a řízení u tak, aby požadavky kladené na chování byly optimálně splněny ve smyslu funkcionálu.

Dynamické optimalizace mohou nalézt optimální průběhy přechodových dějů v technologických procesech. Dynamický systém můžeme popsat vektorovou kauzální relací:

$$x(t) = \varphi [x(t_0), u(\tau)|_{\tau \in \langle t_0, t \rangle}, t, t_0] \quad t \in \langle t_0, t_k \rangle$$

Pro většinu spojitých technických systémů lze relaci vyjádřit ve tvaru

$$x'(t) = f[x(t), u(t), t] \quad t \in \langle t_0, t_k \rangle$$

s omezením (existuje derivace x')

Odpovídající kritérium optimality se obvykle uvádí ve tvaru funkcionálu, kde integrální část zhodnocuje průběh přechodného děje a člen $\Phi(x_k, t_k)$ je dán konečným stavem řízeného procesu.

$$J^* = \min_{x \in \Omega} J(x)$$

$$J = \Phi(x_k, t_k) + \int_{t_0}^{t_k} L(x, u, t) dt \quad \text{kde } L \text{ je skalární funkce} \quad (2.6)$$

$$\Phi(x_k, t_k) \geq 0$$

$$\forall t \geq t_0 \quad : \quad L(x, u, t) \geq 0$$

Doplňkové omezující podmínky obvykle popisují omezení kladená na vektor vstupních veličin $u(t)$, na interval ze kterého počítáme optimalizaci, nebo na cílovou hodnotu vektoru stavu x_k , např. ve tvaru

$$u_1 \leq u(t) \leq u_2$$

$$t_k - t_0 \leq t_{\max}$$

$$x_k = g(t)$$

Výsledkem je nalezená optimální trajektorie řízení. Nevýhodou je nutnost použít jednoduchou dynamiku s jednoduchým cílem, nebo vznikne neúměrně složité řešení úlohy.

$$J = \frac{1}{2} x_k^T(t_k) S x_k(t_k) + \frac{1}{2} \int_{t_0}^{t_k} (x^T(t) Q x(t) + u(t) R u(t)) dt \quad (2.7)$$

Řešení úlohy je (nemusí být jediné řešení)

$$u^* = u[x(t), t]$$

Řešíme pomocí:

- Lagrangeovy multiplikátory
- Hamiltonův variační počet
- Pontrjaginův princip minima (1962)
- Bellmanovo dynamické programování (1957)

Prediktivní regulátory

Návrh regulátoru je optimalizován podle kriteriální funkce, její tvar může být např.:

$$J = \sum_{j=N_1}^{N_2} (p(y) \hat{y}(t+j) - w(t+j))^2 + \sum_{j=1}^{N_u} \lambda (\Delta u(t+j-1))^2 \quad (2.8)$$

kde N_1 , N_2 a N_u jsou horizonty predikce stavu a řízení, λ je váhová konstanta (skalár), $p(y)$ je polynom pro modifikaci prediktivní trajektorie výstupní veličiny. Pokud je polynom $p(y)$ použit, tak se často v literatuře tento přístup nazývá generalizovaný prediktivní regulátor. Základem prediktivního přístupu je vytváření modelu, který je pak používán pro optimalizační výpočet. Přírůstek akčního zásahu je dán vztahem

$$\Delta u(t) = K(w - \hat{y}) \quad (2.9)$$

U prediktivního regulátoru můžeme uvažovat budoucí žádanou hodnotu (pokud je známa), můžeme uvažovat omezení stavových veličin a akční veličiny. Rovněž je velmi vhodný pro systémy obsahující dopravní zpoždění. Vzhledem k iterativnímu výpočtu mohou být časové nároky algoritmu poměrně značné.

3 Modely a identifikace procesů

3.1 Úvod

Optimalizace nastavení parametrů regulátorů se opírá o znalost regulované soustavy. Ve specifických oblastech se mohou používat různé formy modelů. Protože je však úplná znalost konkrétního procesu ve své podstatě nerealizovatelná, mluvíme o znalosti modelu soustavy či řízeného procesu. Zavedením pojmu model vyjadřujeme fyzický rozdíl mezi skutečností - regulovaným procesem a jeho abstrakcí - matematickým modelem.

Cílem modelu je přiblížit se k chování procesu. Přitom přiblížení může být interpretováno velmi různě v závislosti na tom, k jakému účelu model použijeme. Pokud model bude použit pro návrh regulátoru, pak přiblížení může být chápáno tak, že regulátor navržený pro uvažovaný model bude dostatečně dobře pracovat i se skutečným procesem. Protože můžeme pracovat s různými metodami identifikace, různými tvary modelu a různými návrhy regulátoru, bude nutné uvažovat i specifické vlastnosti u každé použité metody.

Model můžeme získat rovněž matematickou analýzou fyzikálně chemických pochodů v uvažovaném procesu nebo analýzou naměřených dat. V případě modelů pro adaptivní řízení výrazně dominuje druhý přístup, i když nelze opomíjet ani první variantu.

Při tvorbě modelu se snažíme najít funkci f , která popisuje chování výstupu soustavy $y(t)$ jako funkci vstupních veličin, typicky akční veličiny $u(t)$, případně dalších měřených veličin, které mohou ovlivňovat výstup, jako např. měřené poruchové veličiny $v(t)$. Tedy předpokládáme

$$y(t) = f(u(t), v(t), t) \quad (3.1)$$

Na reálný proces působí různé poruchy reprezentující převážně neměřitelné vlivy okolí procesu, změny pracovního bodu, působení případných dalších regulačních obvodů, změny ve složení vstupních surovin atd. Tyto vlivy zahrnujeme mezi náhodné - stochastické vlivy. Obecnější tvar modelu, potom můžeme charakterizovat vztahem

$$y(t) = f(u(t), v(t), t) + n(t) \quad (3.2)$$

kde $n(t)$ je člen respektující stochastické vlivy.

Obecně identifikace systémů, jako experimentální postup odvození matematického modelu má následující fáze:

- Plánování experimentu
- Volba struktury modelu
- Volba vhodného kritéria kvality
- Metody odhadu parametrů
- Odhad parametrů
- Test shody chování modelu a systému
- Ověřování chování modelu a systému

3.2 Neparametrické metody identifikace

- Analýza přechodové charakteristiky
- Korelační analýza
- Frekvenční analýza
- Spektrální analýza

3.3 Parametrické metody identifikace modely MA, AR, ARMA

Možnosti tvorby jednoduchých parametrických modelů budou ukázány na příkladě. Předpokládejme, že máme model ve tvaru:

$$F(s) = \frac{e^{-0,1s}}{10s + 1} \quad (3.3)$$

Model převedeme do diskrétního tvaru. Zvolíme relativně velmi dlouhou periodu vzorkování $T=10$ s.

$$F(z, m) = Z \left\{ L^{-1} \left\{ \frac{1 - e^{-Ts}}{s} \frac{e^{-0,1s}}{10s + 1} \right\} \right\}_{m = 1 - \frac{T_d}{T}} = \frac{B(z)}{A(z)} \quad (3.4)$$

$$F(z, 0) = z^{-1}(1 - z^{-1}) Z \left\{ \frac{1}{s(10s + 1)} \right\}_{m = 1 - \frac{T_d}{T}} = \frac{0,6284z^{-1} + 0,0037z^{-2}}{1 - 0,3679z^{-1}} = \frac{Y(z)}{U(z)} \quad (3.5)$$

Můžeme vypočítat odezvu modelu na jednotkový vstupní signál:

$$Y(z) = F(z)U(z) = F(z) \frac{1}{1 - z^{-1}} = 0,6284z^{-1} + 0,863z^{-2} + 0,9497z^{-3} + 0,9815z^{-4} + \dots + 0,99319z^{-5} + \dots \quad (3.6)$$

Model MA (Moving Average)

Model je určen tvarem

$$F(z) = B(z) \quad (3.7)$$

$$F(z) = \frac{0,6284z^{-1} + 0,0037z^{-2}}{1 - 0,3679z^{-1}} = \frac{Y(z)}{U(z)} \quad (3.8)$$

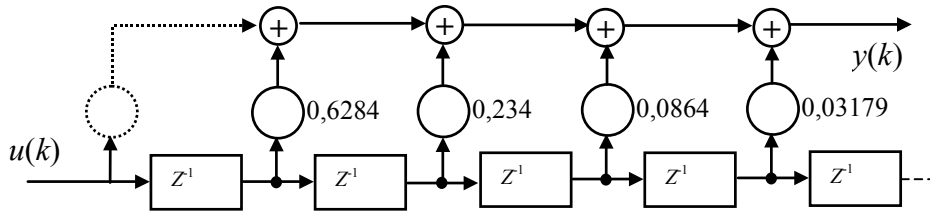
Parametry modelu určíme z následující rovnice (odezva na jediný impuls o amplitudě 1)

$$Y(z) = F(z)U(z) = F(z) 1 = 0,6284z^{-1} + 0,234z^{-2} + 0,0864z^{-3} + 0,003179z^{-4} + \dots \quad (3.9)$$

Rovnice (3.9) můžeme zapsat rovněž jako diferenční rovnici v kroku výpočtu k

$$y(k) = 0,6284u(k-1) + 0,234u(k-2) + 0,0864u(k-3) + \dots \quad (3.10)$$

Model podle rovnic (3.8), (3.9) se také nazývá filtr typu „konečná impulsová odezva“ - Finite Impulse Response FIR. Výhodou je jeho snadná realizace.



Obr. 3.1: Model MA (Moving Average)

Při vhodné volbě periody vzorkování je jeho realizace jednoduchá s řádově jednotkami až desítkami členů. Jak již bylo řečeno, odezva filtru je konečná a v tomto případě velmi rychle odchylka konverguje k nulové hodnotě. V posledním členu lze zohlednit možnou chybu zesílení.

Pokud mají všechny koeficienty filtru (včetně b_0 hodnotu)

$$b_i = \frac{1}{n+1} \quad (3.11)$$

kde n je řád filtru, jedná se pak o typ filtru „klouzavý průměr“.

Uvažme výpočet parametrů filtru pro kratší periodu vzorkování $T=1$ s.

$$F(z,0) = z^{-1}(1-z^{-1}) Z\left\{ \frac{1}{s(10s+1)} \right\}_{\bar{m}=1-0,1} = \frac{0,0861z^{-1} + 0,1056z^{-2}}{1-0,9048z^{-1}} = \frac{Y(z)}{U(z)} \quad (3.12)$$

$$Y(z) = F(z)U(z) = F(z)1 = 0,086z^{-1} + 0,1735z^{-2} + 0,25z^{-3} + 0,323z^{-4} + 0,39z^{-5} + \dots \quad (3.13)$$

Zkrácením periody vzorkování se však počet členů zvětšuje. Uvažme výpočet parametrů filtru při ještě kratší periodě vzorkování $T=0,2$ s.

$$F(z,0) = z^{-1}(1-z^{-1}) Z\left\{ \frac{1}{s(10s+1)} \right\}_{\bar{m}=1-0,5} = \frac{0,00995z^{-1} + 0,00985z^{-2}}{1-0,9802z^{-1}} = \frac{Y(z)}{U(z)} \quad (3.14)$$

$$Y(z) = 0,00995z^{-1} + 0,02955z^{-2} + 0,04877z^{-3} + 0,067z^{-4} + \dots \quad (3.15)$$

V tomto případě konverguje chyba k nulové hodnotě velmi pomalu, filtr má přes dvě stě zpoždění.

Vhodná perioda vzorkování bývá nejčastěji určena následujícími vztahy:

$$T = \frac{1}{6} \div \frac{1}{15} T_{95} \quad (3.16)$$

kde T_{95} je hodnota času, za kterou dosáhne výstup po jednotkovém vstupním skoku hodnotu 95%. Jantzen definuje vhodnou periodu vzorkování jako

$$T = 0,1 T_G \quad (3.17)$$

kde T_G je součet všech časových konstant mezi vstupem a výstupem procesu. Konkrétní určení periody vzorkování je ale velmi komplexní záležitost. V kapitole 2. bylo řečeno, že delší perioda vzorkování může způsobit horší vyregulování poruchy, a proto se volí u PSD regulátorů až o řád kratší perioda vzorkování, než je uvedeno v (3.17). V případě modelování procesů působí delší perioda vzorkování jako filtr a parametry mnohem lépe konvergují k reálným hodnotám. Proto konkrétní hodnota velikosti periody vzorkování je vždy kompromisem.

$$F(z) = \frac{1}{A(z)} = \frac{Y(z)}{U(z)} \quad (3.18)$$

Model AR (Auto-Regressive)

Model je určen přenosovou funkcí podle rovnice (3.18). Model je typu „nekonečná impulsová odezva“- Infinite Impulse Response IIR.

Model určíme roznásobením a úpravou rovnice

$$\begin{aligned} \frac{0,6284z^{-1} + 0,0037z^{-2}}{1 - 0,3679z^{-1}} &\Rightarrow A(z) = \frac{1 - 0,3679z^{-1}}{0,6284z^{-1} + 0,0037z^{-2}} = \frac{U(z)}{Y(z)} \\ A(z) &= \frac{1 - 0,3679z^{-1}}{0,6284z^{-1} + 0,0037z^{-2}} = (1 - 0,3679z^{-1}) : (0,6284z^{-1} + 0,0037z^{-2}) = \\ &= 1,591z^{-1} - 0,5945z^{-2} - 0,0035z^{-3} + 0,00002z^{-4} + .. \\ \frac{Y(z)}{U(z)} &= \frac{1}{A(z)} = \frac{1}{1,591z^{-1} - 0,5945z^{-2} - 0,0035z^{-3} + 0,00002z^{-4}} \end{aligned}$$

Další úpravou získáme

$$\frac{Y(z)}{U(z)} = \frac{0,6284z}{1 - 0,3736z^{-1} - 0,0022z^{-2}}$$

$$(z) (1 - 0,3736z^{-1} - 0,0022z^{-2}) = 0,6284zU(z)$$

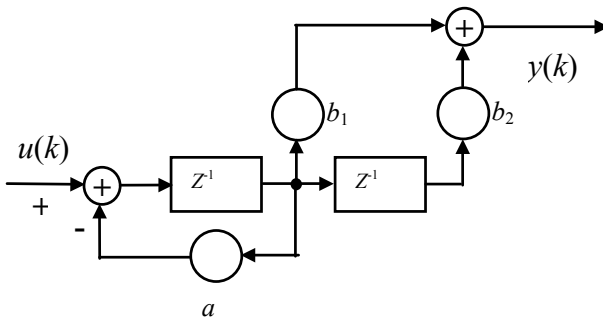
$$Y(z) = 0,6284zU(z) + (0,3736z^{-1} + 0,0022z^{-2})Y(z) \quad (3.19)$$

$$y(k) = 0,6284u(k+1) + 0,2347y(k-1) + 0,0002 y(k-2) \quad (3.20)$$

Model se vyznačuje poměrně malou numerickou stabilitou a přesností výpočtů při použití v aproximaci je uveden jen pro úplnost. Jeho význam je při zpracování stochastických procesů. Výstup zde závisí jen na okamžité hodnotě vstupního šumu a na minulých hodnotách výstupů, které jsou váženy koeficienty a_i . Proto se model nazývá autoregresní.

Model ARMA (Auto-Regressive Moving Average)

Model ARMA a jeho modifikace jsou nejpoužívanější modely při identifikaci. Model je typu „nekonečná impulsová odezva“- Infinite Impulse Response IIR. Na příkladě si ověříme určení jeho parametrů. Předpokládejme, že proces je dán přenosovou funkcí (3.8), na kterou působí jednotkový vstupní signál. Nepředpokládejme vliv poruch. Cílem je určit parametry modelu prvního řádu s dopravním zpožděním (3.21) nakresleného na **Obr. 3.2** s přesností na cca 4 platná místa.



Obr. 3.2: Model ARMA

$$F(z) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a z^{-1}} \quad (3.21)$$

$$\begin{aligned} u(0) &= 1 & y(0) &= 0 \\ u(1) &= 1 & y(1) &= 0,6284 = b_1 u(0) \\ u(2) &= 1 & y(2) &= 0,863 = b_1 u(1) + b_2 u(0) - a b_1 u(1) \\ u(3) &= 1 & y(3) &= 0,9447 = b_1 u(2) + b_2 u(1) - a b_1 u(1) - a b_2 u(0) + a a b_1 u(0) \end{aligned}$$

Parametry modelu tedy určíme z rovnic

$$b_1 = 0,6284$$

$$0,863 = b_1 + b_2 - a b_1 = 0,6284 + b_2 - a 0,6284$$

$$0,9447 = b_1 + b_2 - a b_1 - a b_2 + a a b_1 = 0,6284 + b_2 - a 0,6284 - a b_2 + a a 0,6284$$

Řešením kvadratické rovnice získáme hledané parametry.

S výhodou můžeme rovněž pro určení parametrů modelu použít výpočetní algoritmus. Rovnici (3.21) přepíšeme a vytvoříme maticový zápis:

$y(k) =$	$b_1 u(k-1)$	$+ b_2 u(k-2)$	$- a y(k-1)$
0	0	0	0
0,6284	1	0	0
0,863	1	1	-0,6284
0,9497	1	1	-0,863
0,9815	1	1	-0,9447
0,99319	1	1	-0,9815

další řádky jsou lineárně závislé

hodnota determinantu $\Delta = -0,863 + 0,6284 = -0,2346$

$$b_1 = \frac{\begin{vmatrix} 0,6284 & 0 & 0 \\ 0,863 & 1 & -0,6284 \\ 0,9497 & 1 & -0,863 \end{vmatrix}}{-0,2346} = 0,6283 \quad b_2 = \frac{\begin{vmatrix} 1 & 0,6284 & 0 \\ 1 & 0,863 & -0,6284 \\ 1 & 0,9497 & -0,863 \end{vmatrix}}{-0,2346} = 0,00209$$

$$a = \frac{\begin{vmatrix} 1 & 0 & 0,6284 \\ 1 & 1 & -0,863 \\ 1 & 1 & -0,9497 \end{vmatrix}}{-0,2346} = -0,3695$$

$$F(z) = \frac{0,6284z^{-1} + 0,0037z^{-2}}{1 - 0,3679z^{-1}} \quad F_M(z) = \frac{0,6283z^{-1} + 0,000209z^{-2}}{1 - 0,3695z^{-1}}$$

Z porovnání originálního přenosu s jeho modelem označeným $F_M(z)$ je zřejmé, že se výsledky liší. Je to způsobeno tím, že výpočet byl realizován na čtyři platná místa. Pokud bychom však uvážili, že reálný proces je běžně řízen přes dvanáctibitové D/A a A/D převodníky s maximální přesností na tři desetinná místa (třetí místo je měřeno s diskretizací 5 mV), pak by skutečný výsledek byl mnohem horší i při přesném měření bez působení šumu.

Z výše uvedeného je zřejmé, že výpočtové algoritmy pracující s daty získanými z reálných experimentů musí být poměrně robustní vlastnosti. Jako příklad vhodného postupu lze uvést metodu elementární matice rotací.

3.4 Parametrické metody identifikace model ARMAX, ARX

Všechny předchozí modely vznikly v podstatě jednorázovými experimenty i když v řadě případů by některé mohly být použity i pro průběžnou identifikaci. Cílem je odvodit metodu, která by byla vhodná pro průběžnou identifikaci v reálném procesu.

Velmi obecný diskretní popis dynamického systému se dá zapsat jako funkce předchozích hodnot měřených veličin, tedy

$$\begin{aligned} y(k) &= f[y(k-1), y(k-2), \dots, y(k-na), u(k-1), u(k-2), \dots, u(k-nb), \\ &v(k-1), v(k-2), \dots, v(k-nd), k] + n(k) \end{aligned} \quad (3.22)$$

kde $y(k)$ je hodnota výstupní veličiny v k -tém okamžiku vzorkování, tj. v čase $t = kT$ (T je perioda vzorkování, kterou v rovnici (3.22) považujeme za parametr přenosu).

Problémem je, jak blíže specifikovat stochastický člen. Poruchu $n(k)$ lze modelovat tak, že ji budeme reprezentovat signálem, který vznikne průchodem šumu známých vlastností určitým filtrem. Vlastnosti poruchy jsou pak charakterizovány tímto filtrem. Filtr, podobně jako soustavu, lze popsat závislostí zpožděných vstupních a výstupních veličin. Dostáváme tak

$$\begin{aligned} y(k) &= f[y(k-1), y(k-2), \dots, y(k-na), u(k-1), u(k-2), \dots, u(k-nb), \\ &v(k-1), v(k-2), \dots, v(k-nd), e_s(k), e_s(k-1), e_s(k-2), \dots, e_s(k-nc), k] \end{aligned} \quad (3.23)$$

kde $e_s(k)$ je náhodná, měření nepřístupná složka.

Pokud se omezíme na lineární funkci f dostáváme známý model ARMAX (3.24) (Auto-Regressive Moving Average with eXogenous variable). **Pozor: V literatuře se označení ARMA vztahuje k modelu šumu, zatímco písmeno X se vztahuje k eXogenous variable a je to vlastně náš dosavadní model procesu. Podobně pak u modelu ARX, který je vhodnější při uvažování šumu!**

$$y(k) = -\sum_{i=1}^{na} a_i y(k-i) + \sum_{i=1}^{nb} b_i u(k-i) + \sum_{i=1}^{nd} d_i v(k-i) + e_s(k) + \sum_{i=1}^{nc} c_i e_s(k-i) \quad (3.24)$$

nebo v zápise pomocí operátoru zpoždění z^{-1}

$$A(z^{-1})y = B(z^{-1})u + D(z^{-1})v + C(z^{-1})e_s \quad (3.25)$$

kde jednotlivé polynomy rovnice (3.25) mají tvar:

$$\begin{aligned}
 A(z^{-1}) &= 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{na} z^{-na} \\
 B(z^{-1}) &= b_1 z^{-1} + b_2 z^{-2} + \dots + b_{nb} z^{-nb} \\
 C(z^{-1}) &= 1 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_{nc} z^{-nc} \\
 D(z^{-1}) &= d_1 z^{-1} + d_2 z^{-2} + \dots + d_{nd} z^{-nd}
 \end{aligned} \tag{3.26}$$

Pro adaptivní řízení však model ARMAX není příliš vhodný. Pokud budeme jeho parametry (koeficienty polynomů A, B, C, D) určovat z naměřených dat, narazíme na problém určit co ve změřené hodnotě je šum a co je skutečný výstup z procesu bez působení šumu, takže neurčíme koeficienty polynomu $C(z^{-1})$, protože fiktivní šum $e_s(k)$ není měřitelný.

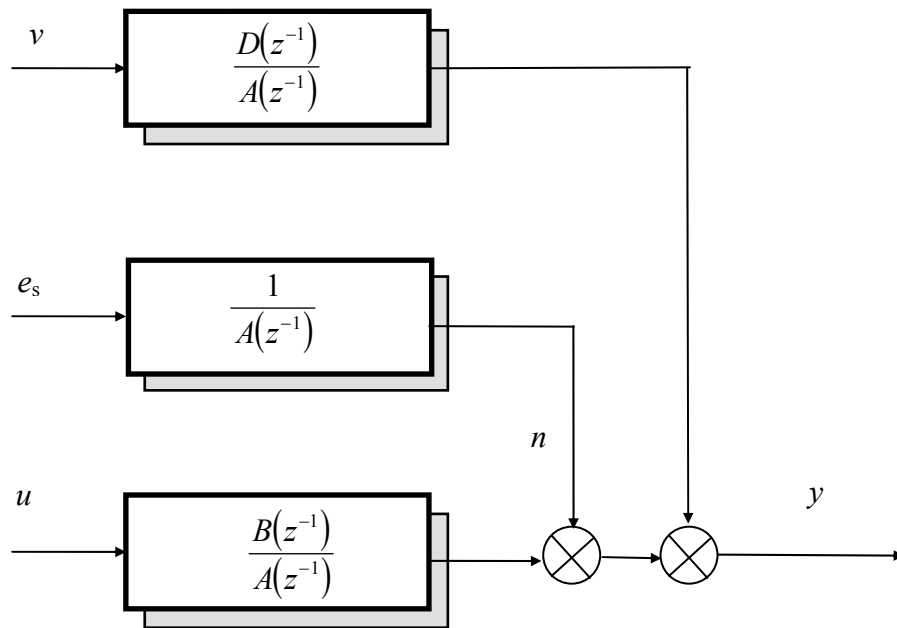
Proto se při návrzích adaptivních regulátorů vychází většinou z regresního modelu typu ARMA a bývá označován v literatuře jako (ARX), který modeluje výstup soustavy podle vztahu

$$y(k) = -\sum_{i=1}^{na} a_i y(k-i) + \sum_{i=1}^{nb} b_i u(k-i) + \sum_{i=1}^{nd} d_i v(k-i) + e_s(k) \tag{3.27}$$

neboli

$$A(z^{-1})y = B(z^{-1})u + D(z^{-1})v + e_s \tag{3.28}$$

Jeho blokové schéma je na **Obr. 3.3**.



Obr. 3.3: Blokové schéma regresního modelu ARX

Regresní model ARX se často zapisuje v kompaktní vektorové formě

$$y(k) = \theta^T(k)\varphi(k-1) + e_s(k) \tag{3.29}$$

kde

$$\theta^T(k) = [a_1, a_2, \dots, a_{na}, b_1, b_2, \dots, b_{nb}, d_1, d_2, \dots, d_{nd}] \tag{3.30}$$

je vektor parametrů vyšetřovaného modelu (neznámé parametry)

a sloupcový vektor

$$\varphi^T(k-1) = [-y(k-1), -y(k-2), \dots, -y(k-na), u(k-1), u(k-2), \dots, u(k-nb), \\ v(k-1), v(k-2), \dots, v(k-nd)] \quad (3.31)$$

je vektor dat.

O míře shody získaného modelu a reálného procesu rozhoduje v podstatné míře perioda vzorkování, úroveň a typ poruchových veličin a řád modelu. Přenosová funkce diskrétní otevřené smyčky modelu musí obsahovat alespoň jeden krok dopravního zpoždění (Předpokládáme že řízený systém je ryzí i v případě, že spojitě přenosy dopravní zpoždění neobsahují). V opačném případě by při uzavření smyčky došlo ke vzniku algebraických smyček. V literatuře se setkáváme i s případy, kdy u soustavy bez dopravního zpoždění se uvažuje polynom $B(z^{-1})$ s absolutním členem b_0 nebo bez něho. Uvažování $b_0 \neq 0$ znamená, že akční zásah $u(k)$ ovlivňuje $y(k)$. Je-li tomu tak, potom použitý regulátor nesmí používat přímo výstup $y(k)$ pro generování $u(k)$. Pokud používáme řídicí algoritmus tak že $u(k) = f[y(k)]$, musí být $b_0 = 0$. Dalším řešením je, že člen b_0 sice identifikujeme, ale v přímém řízení nepoužijeme a člen nám vlastně vyrovnává nepřesnosti způsobené identifikačním procesem. Kvalitu použitého regresního modelu posuzujeme primárně podle chyby predikce, tj. odchylky

$$\hat{e}(k) = y(k) - \hat{y}(k) \quad (3.32)$$

kde pro výpočet predikované výstupní veličiny $\hat{y}(k)$ se použije vztah (3.27) s $e_s(k) = 0$. Chyba predikce hraje klíčovou roli při identifikaci parametrů regresního modelu z naměřených dat, také hraje roli při volbě struktury (řádu) regresního modelu při vhodné periodě vzorkování. Je třeba zdůraznit, že pro kvalitní model je třeba dosáhnout nejen malé chyby predikce, ale i toho, že chyba predikce bude reprezentovat nekorelovaný šum s přibližně nulovou střední hodnotou a podle velikosti periody vzorkování budou rovněž dostatečně potlačeny vyšší kmitočty v souladu s vzorkovacím teorémem.

3.5 Identifikace procesu pro adaptivní řízení

V adaptivním řízení je úloha identifikace právě tak důležitá jako role syntézy regulátoru. Identifikace pro adaptivní řízení má ovšem svá specifika, která vedou k tomu, že se v převážné míře odhadují parametry regresního modelu (ARX) a používá se metoda nejmenších čtverců. Identifikace pro adaptivní řízení vychází z následujících podmínek:

- Vstupy do řízeného procesu jsou generovány regulátorem.
- V řídicí smyčce se vyskytují poruchy, které by měl regulátor kompenzovat a stabilizovat proces. Přítomnost poruch zhoršuje možnosti identifikace parametrů a regulátor z těchto odhadů může být špatně určen.
- Identifikační proces u adaptivního řízení trvá velmi dlouho (po dobu trvání řízení procesu). Proto lze jen stěží předpokládat, že odhadované parametry budou konstantní.
- Identifikace musí dávat výsledky za různých pracovních podmínek procesu (např. 30%, 70%, 100% požadovaného výkonu, a to i v období relativního stacionárního stavu, při poruchách nebo přechodech mezi různými stavu).
- Strukturu identifikovaného modelu obvykle neměníme.

- Identifikační algoritmus musí být numericky spolehlivý a dostatečně rychlý.
- Poměrně důležitou roli hraje počáteční nastavení parametrů regulátoru především proto, aby se zabránilo nekorektním akčním zásahům na počátku identifikace. Odhady parametrů by měly tedy buď již na počátku identifikace dostatečně reprezentovat daný proces, nebo regulátor má na počátku identifikace přednastaveny parametry tak, že regulační pochod je pro daný proces přijatelný (např. s přetlumenou dynamikou), nestačí aby byl stabilní.
- Počátečními podmínkami nejběžněji používané identifikační metody jsou počáteční odhady parametrů a jejich kovarianční matice. Zatímco role počátečních odhadů parametrů je respektována, role kovarianční matice bývá nedoceněna a její návrh je obtížný. Ukazuje se, že schůdnou a poměrně jednoduchou metodou, jak získat počáteční podmínky pro identifikaci, zahrnující v podstatě libovolnou apriorní informaci, je metoda fiktivních dat. Její podstata spočívá v tom, že pomocí velmi zjednodušeného diskrétního modelu procesu, který dostatečně reprezentuje proces, jsou vygenerovány data.
- Zpracováním těchto fiktivních dat podobně jako by byly získány z reálného procesu, lze získat počáteční odhady a kovarianční matici. Problém je, že tato data nelze zpracovat obvyklým postupem při použití metody nejmenších čtverců. Je si třeba uvědomit, že jednotlivé dílčí složky apriorní informace mohou být i částečně protichůdné, ale v každém případě je třeba tuto informaci brát jen s určitou pravděpodobností. Přitom by se mohlo stát, že použití velkého počtu dat o specifické informaci (např. zesílení) při nepřesném modelu by vedlo k tomu, že by se tato informace v odhadech tak zafixovala, že ani velké množství skutečných dat by ji již nezměnilo. Proto tato data použijeme pouze jednorázově pro návrh regulátoru. Znovu nastavíme počáteční podmínky pro identifikaci a jistý čas regulujeme pouze s takto nastaveným regulátorem a přitom stále identifikujeme proces. Po shromáždění dostatečného množství dat můžeme přejít k adaptivnímu řízení a přepsání parametrů řídicího algoritmu.
- Identifikace procesů při časově proměnných parametrech je možno řešit technikou zapomínání. Nejznámější je exponenciální zapomínání, kde vliv starších dat na odhady parametrů a jejich kovarianční matici exponenciálně klesá. Závažným nedostatkem tohoto zapomínání v adaptivním režimu je ztráta informace v případech, kdy je proces natolik ustálený, že data přináší jen málo informace o vlastnostech procesu. Tuto situaci je třeba řešit vypínáním identifikace, proměnným koeficientem zapomínání nebo jinými formami zapomínání, které v sobě obsahují schopnost měnit množství zapomínané informace podle charakteru dat. Ve své podstatě parametry procesu jsou nejlépe určeny, když na proces působí časté změny žádané hodnoty, je potřebná pestrá změna vstupních veličin. V ustáleném stavu po dosažení žádané hodnoty je lépe proces identifikace změnou koeficientu zapomínání pozastavit a tím zajistit, aby dynamika poruch se nedostávala do identifikovaných parametrů. Při další změně žádané hodnoty, která přesáhne nastavenou hranici potom opět obnovíme proces identifikace.

3.6 Princip metody nejmenších čtverců

3.6.1 Jednorázová metoda nejmenších čtverců.

Předpokládejme 1 vstup a 1 výstup, kde závislost mezi vstupem a výstupem je popsána rovnicí

$$y = b \cdot x \quad (3.33)$$

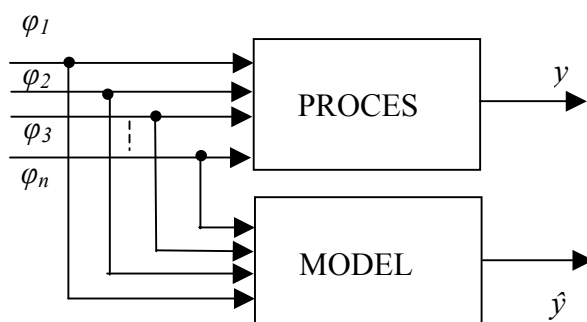
V případě j měření a při rozdílu mezi měřenými hodnotami můžeme určit metodou nejmenších čtverců optimální velikost parametru b . Určíme kritérium

měř. vyp.

$$J(b) = \frac{1}{2} \sum_{i=1}^j (y_{im} - y_{iv})^2 = \frac{1}{2} \sum_{i=1}^j (y_{im} - bx_i)^2 \rightarrow \min \quad (3.34)$$

$$\frac{\partial}{\partial b} J(b) \Big|_{b=\bar{b}} = \frac{1}{2} \sum_{i=1}^j 2(y_{im} - bx_i)(-x_i) = 0 \rightarrow b = \frac{\sum_{i=1}^j x_i y_{im}}{\sum_{i=1}^j x_i^2} \quad (3.35)$$

Metodu rozšíříme na n vstupů či vnitřních stavů



Obr. 3.4: Souvislost mezi procesem a modelem procesu

Model můžeme popsat experimentální regresní rovnicí

$$\begin{aligned} \hat{y}(i) &= \varphi_1(i) \cdot \theta_1 + \varphi_2(i) \cdot \theta_2 + \varphi_3(i) \cdot \theta_3 + \dots + \varphi_n(i) \cdot \theta_n = \varphi^T(i) \cdot \theta + \varepsilon \\ \theta &= (\theta_1 \ \theta_2 \ \theta_3 \ \dots \ \theta_n)^T \\ \varphi^T(i) &= (\varphi_1(i) \ \varphi_2(i) \ \varphi_3(i) \ \dots \ \varphi_n(i)) \end{aligned} \quad (3.36)$$

kde \hat{y} je odhad výstupní veličiny modelu, θ je hledaný vektor neznámých parametrů, φ je vektor známých měřených funkcí, i je krok výpočtu, ε je chyba v kroku výpočtu. Proměnné φ_i označujeme jako regresní proměnné a model je tedy nazýván regresním modelem

Po n měřeních můžeme učinit první odhad parametrů modelu. Za předpokladu, že parametry vektoru θ které chceme identifikovat se během trvání identifikace nemění, určíme po n -tém kroku jejich hodnotu.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} \varphi_{11} & \varphi_{12} & \varphi_{13} & \dots & \varphi_{1n} \\ \varphi_{21} & \varphi_{22} & \varphi_{23} & \dots & \varphi_{2n} \\ \varphi_{31} & \varphi_{32} & \varphi_{33} & \dots & \varphi_{3n} \\ \vdots & & & & \\ \varphi_{n1} & \varphi_{n2} & \varphi_{n3} & \dots & \varphi_{nn} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_n \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

V maticovém zápisu je

$$y = \Phi\theta + \varepsilon \quad \rightarrow \quad \varepsilon = y - \Phi\theta \quad (3.37)$$

Hledáme minimum účelové funkce

$$J(\theta) = \frac{1}{2} \varepsilon^T \varepsilon = \frac{1}{2} \|\varepsilon\|^2 = \frac{1}{2} (y - \Phi\theta)^T (y - \Phi\theta) \quad (3.38)$$

jejíž minimum získáme, když derivaci podle vektoru parametrů θ položíme rovnu 0, tj.

$$\begin{aligned} \frac{\partial J}{\partial \theta} \Big|_{\theta=\hat{\theta}} &= (y - \Phi\theta)^T (y - \Phi\theta) \quad \sim \quad \sum_{i=1}^N \varepsilon_i^2 \\ \frac{\partial}{\partial \theta} (y - \Phi\theta)^T (y - \Phi\theta) &= 0 \\ \frac{\partial}{\partial \theta} (y^T y - y^T \Phi\theta - \theta^T \Phi^T y + \theta^T \Phi^T \Phi\theta) &= 0 - \Phi^T y - \Phi^T y + 2\Phi^T \theta \\ -2\Phi^T y + 2\Phi^T \Phi\theta &= 0 \quad \rightarrow \quad \theta = (\Phi^T \Phi)^{-1} \Phi^T y \end{aligned} \quad (3.39)$$

Označíme

$$P(i) = (\Phi^T \Phi)^{-1} \quad (3.40)$$

$P(i)$ je kovarianční matice

3.6.2 Průběžná metoda nejmenších čtverců

Matice Φ by bez redukce byla v každém kroku postupně rozšiřována o další řádek. Klíčem je vytvořit rekurentní výpočet, kdy do stávajícího rozměru je začleněno další měření.

Platí

$$\theta(i) = P(i)\Phi^T(i)y \quad (3.41)$$

v kroku $i+1$ rozšíříme matici o další řádek

$$\begin{bmatrix} y_i \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} \varphi_{i,1} & \varphi_{i,2} & \varphi_{i,3} & \dots & \varphi_{i,n} \\ \varphi_{i+1,1} & \varphi_{i+1,2} & \varphi_{i+1,3} & \dots & \varphi_{i+1,n} \end{bmatrix} \begin{bmatrix} \theta_i \\ \theta_{i+1} \end{bmatrix} + \begin{bmatrix} \varepsilon_i \\ \varepsilon_{i+1} \end{bmatrix}$$

$$\theta(i+1) = P(i+1) \begin{bmatrix} \Phi^T(i), \varphi(i+1) \end{bmatrix} \begin{bmatrix} y_i \\ y_{i+1} \end{bmatrix}$$

Vypočteme $P(i+1)$ pro určení $\theta(i+1)$

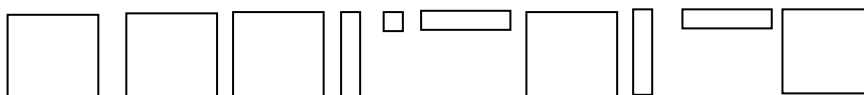
$$P(i+1) = \left[\begin{bmatrix} \Phi^T(i), \varphi(i+1) \end{bmatrix} \begin{bmatrix} \Phi(i) \\ \varphi^T(i+1) \end{bmatrix} \right]^{-1} = \left[\Phi^T(i)\Phi(i) + \varphi(i+1)\varphi^T(i+1) \right]^{-1}$$

S použitím lemma o inverzi matice

$$(A + bb^T)^{-1} = A^{-1} - A^{-1}b(I + b^T A^{-1}b)^{-1}b^T A^{-1} \tag{3.42}$$

dostaneme hledaný vztah

$$P(i+1) = P(i) - P(i) \varphi(i+1) \left[1 + \varphi^T(i+1)P(i) \varphi(i+1) \right]^{-1} \varphi^T(i+1)P(i) \tag{3.43}$$



V literatuře se můžeme často setkat s pozměněným tvarem, kdy je zavedena proměnná t ve významu diskrétního času rovnicí

$$t = i+1 \tag{3.44}$$

Potom můžeme předcházející rovnici přepsat do tvaru

$$P(t) = P(t-1) - P(t-1) \varphi(t) \left[I + \varphi^T(t)P(t-1) \varphi(t) \right]^{-1} \varphi^T(t)P(t-1) \tag{3.45}$$

V literatuře se můžeme setkat ještě s dalším tvarem (uvažme proč?)

$$P(t) = P(t-1) - \frac{P(t-1) \varphi(t)^T \varphi(t) P(t-1)}{1 + \varphi(t)^T P(t-1) \varphi(t)} \tag{3.46}$$

Pro další práci však potřebujeme odvodit i vektor odhadu parametrů

$$\theta(i+1) = P(i+1) \begin{bmatrix} \Phi^T(i), \varphi(i+1) \end{bmatrix} \begin{bmatrix} y_i \\ y_{i+1} \end{bmatrix}$$

označme

$$\rho(i+1) = \left[1 + \varphi^T(i+1)P(i) \varphi(i+1) \right]^{-1} \tag{3.47}$$

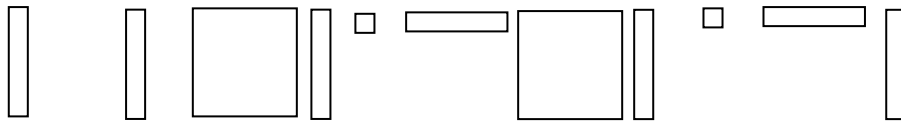
a výpočtem dostaneme

$$\theta(i+1) = P(i+1) \left[\Phi^T(i)y(i) + \varphi(i+1)y(i+1) \right] =$$

$$\begin{aligned}
&= (P(i) - P(i)\varphi(i+1)\rho(i+1)\varphi^T(i+1)P(i)) [\Phi^T(i)y(i) + \varphi(i+1)y(i+1)] \\
&= P(i) \Phi^T(i)y(i) + P(i) \varphi(i+1)y(i+1) - P(i) \varphi(i+1)\rho(i+1)\varphi^T(i+1)P(i) \Phi^T(i)y(i) - \\
&\quad - P(i) \varphi(i+1)\rho(i+1) \varphi^T(i+1)P(i) \varphi^T(i+1)y(i+1) = \\
&= \theta(i) + P(i) \varphi(i+1)y(i+1) - P(i) \varphi(i+1)\rho(i+1)\varphi^T(i+1)P(i) \Phi^T(i)y(i) - \\
&\quad - P(i) \varphi(i+1)\rho(i+1) \varphi^T(i+1)P(i) \varphi^T(i+1)y(i+1) \\
&= \theta(i) - P(i) \varphi(i+1)\rho(i+1)\varphi^T(i+1)P(i)\Phi^T(i)y(i) + P(i)\varphi(i+1)(1- \\
&\quad \rho(i+1) \varphi(i+1)P(i) \varphi^T(i+1))y(i+1) \\
&= \theta(i) - P(i) \varphi(i+1)\rho(i+1)\varphi^T(i+1)P(i)\Phi^T(i)y(i) + P(i) \varphi(i+1) \rho(i+1) (\rho(i+1))^{-1} - \\
&\quad \varphi(i+1)P(i) \varphi^T(i+1))y(i+1)
\end{aligned}$$

měřeno - predikováno

$$\theta(i+1) = \theta(i) + P(i) \varphi(i+1) [1 + \varphi^T(i+1)P(i) \varphi(i+1)]^{-1} (y(i+1) - \varphi^T(i+1)\theta(i)) \quad (3.48)$$



Předcházející rovnice přepíšeme do rekurentních vztahů

$$\begin{aligned}
\theta(i+1) &= \theta(i) + K(i+1)(y(i+1) - \varphi^T(i+1)\theta(i)) \\
K(i+1) &= P(i) \varphi(i+1) [1 + \varphi^T(i+1)P(i) \varphi(i+1)]^{-1} \\
P(i+1) &= P(i) - K(i+1) \varphi^T(i+1)P(i)
\end{aligned} \quad (3.49)$$

První rovnice v podstatě určuje ve členu měřeno – predikováno odhad parametrů pro následující krok, který se počítá z odhadu v kroku předcházejícím, který je opraven o hodnotu úměrnou rovnici

$$y(i+1) - \varphi^T(i+1)\theta(i)$$

která představuje rozdíl mezi tím co skutečně v kroku $i+1$ změříme a tím, co pro stejný krok predikuje model se stávajícím nastavením parametrů. $K(i+1)$ je v podstatě váhový součinitel, který určuje s jakým vlivem se má tento rozdíl vzít v úvahu a tím urychlit (zpozdít) nový výpočet parametrů.

3.6.3 Exponenciální zapomínání

Jestliže požadujeme, aby algoritmus byl schopen sledovat pomalé změny parametrů identifikovaného procesu, můžeme toho dosáhnout technikou exponenciálního zapomínání. Potom minimalizujeme modifikované kritérium

$$J_k(\theta) = \sum_{i=k_0}^k \lambda^{2(k-i)} \varepsilon^2(i) \quad (3.50)$$

kde $0 < \lambda^2 \leq 1$ je faktor exponenciálního zapomínání

$$\begin{aligned}
 P(i+1) &= \left[\lambda^2 \Phi^T(i) \Phi(i) + \varphi(i+1) \varphi^T(i+1) \right]^{-1} \\
 P(i+1) &= \left[\Phi^T(i) \Phi(i) \right]^{-1} \lambda^{-2} - \left[\Phi^T(i) \Phi(i) \right]^{-1} \lambda^{-2} \varphi(i+1) \left[I + \varphi^T(i+1) \left[\Phi^T(i) \Phi(i) \right]^{-1} \lambda^{-2} \varphi(i+1) \right]^{-1} \\
 &\quad \varphi(i+1) \left[\Phi^T(i) \Phi(i) \right]^{-1} \lambda^{-2} \\
 P(i+1) &= P(i) \lambda^{-2} - P(i) \lambda^{-2} \varphi(i+1) \left[I + \varphi^T(i+1) P(i) \lambda^{-2} \varphi(i+1) \right]^{-1} \varphi(i+1) P(i) \lambda^{-2} \\
 P(i+1) &= \left(P(i) - P(i) \varphi(i+1) \left[\lambda^2 I + \varphi^T(i+1) P(i) \varphi(i+1) \right]^{-1} \varphi(i+1) P(i) \right) / \lambda^2
 \end{aligned}$$

Modifikované rovnice jsou pak

$$\begin{aligned}
 \theta(i+1) &= \theta(i) + K(i+1) (y(i+1) - \varphi^T(i+1) \theta(i)) \\
 K(i+1) &= P(i) \varphi(i+1) \left[\lambda_{i+1} I + \varphi^T(i+1) P(i) \varphi(i+1) \right]^{-1} \\
 P(i+1) &= \left(P(i) - K(i+1) \varphi^T(i+1) P(i) \right) / \lambda_{i+1}
 \end{aligned} \tag{3.51}$$

3.7 Volba řádu modelu a počáteční nastavení algoritmu

V případě volby řádu a struktury modelu musíme vždy zvážit, jaké nástroje pro návrh regulátoru máme. Máme-li řídicí algoritmus pouze pro druhý či třetí řád, nemůžeme použít algoritmus pro druhý řád s dopravním zpožděním. Nejčastěji používané modely jsou druhého řádu, druhého řádu s dopravním zpožděním a třetího řádu.

Model druhého řádu můžeme popsat rovnicí

$$F_c(z) = \frac{Y(z)}{U(z)} = Z \left\{ L^{-1} \left\{ \frac{1 - e^{-Ts}}{s} \frac{K}{(T_1s + 1)(T_2s + 1)} \right\} \right\} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{3.52}$$

Model druhého řádu s dopravním zpožděním můžeme popsat rovnicí, pro $\tau < T$ platí

$$F_c(z) = \frac{Y(z)}{U(z)} = Z \left\{ L^{-1} \left\{ \frac{1 - e^{-Ts}}{s} \frac{K e^{-\tau s}}{(T_1s + 1)(T_2s + 1)} \right\} \right\} = \frac{b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{3.53}$$

Model třetího řádu můžeme popsat rovnicí

$$F_c(z) = \frac{Y(z)}{U(z)} = Z \left\{ L^{-1} \left\{ \frac{1 - e^{-Ts}}{s} \frac{K}{(T_1s + 1)(T_2s + 1)(T_3s + 1)} \right\} \right\} = \frac{b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} \tag{3.54}$$

Pro počáteční nastavení pak volíme

$$P_0 = 10^9 I \tag{3.55}$$

$$\theta_0 = (1 \ 0 \ 0 \ \dots \ 0)^T$$

$$\lambda_0 = 0,995-0,999$$

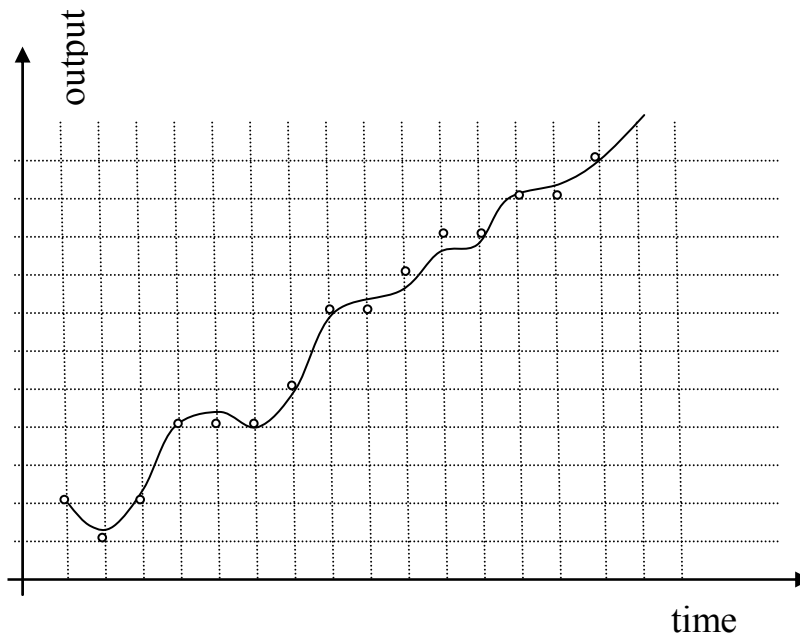
Jestliže volíme (např. pro třetí řád)

$$\varphi^T = (u(i-1), u(i-2), u(i-3), -y(i-1), -y(i-2), -y(i-3))$$

pak dostaneme odhad parametrů ve tvaru

$$\theta = (\hat{b}_1, \hat{b}_2, \hat{b}_3, \hat{a}_1, \hat{a}_2, \hat{a}_3)^T$$

V reálné praxi však dochází při identifikaci k řadě problémů. Jedním z nejdůležitějších je vliv kvantizačního efektu způsobeném A/D převodníky, které zaokrouhlí měřenou veličinu, kterou v nejlepším případě změříme na 4 platná čísla při 12-ti bitovém převodníku. Snadno na obr. 3.5 zjistíme, že z měřeného signálu dostáváme po počátečních identifikacích různé přenosové funkce podle volby periody vzorkování. Identifikační algoritmy při použití relativně krátké periody vzorkování dávají nereálný odhad parametrů identifikovaného systému.



Obr. 3.5: Vliv periody vzorkování a kvantizačního efektu A/D převodníku.

4 Adaptivní regulátory

Experimentální seřízení parametrů spojitého PID regulátoru, které navrhli Ziegler a Nichols [1] před šedesáti lety, je stále dosud používáno v průmyslové praxi. Kritické proporcionalní zesílení K_{KRIT} a kritická perioda kmitů T_{KRIT} u uzavřeného regulačního obvodu se získají postupným zvyšováním zesílení proporcionalního regulátoru K , až výstupní veličina uzavřeného regulačního obvodu kmitá s konstantní amplitudou, tzn., že regulační obvod je na mezi stability. V tomto případě jsou póly uzavřeného regulačního obvodu umístěny na imaginární ose komplexní s -roviny. Zesílení K pak odpovídá K_{KRIT} a ze záznamu průběhu výstupní veličiny se odečte tomu odpovídající kritická perioda kmitů T_{KRIT} . Konstanty PID regulátoru se určí ze vztahů podle Tabulka 2.1. Vzhledem k tomu, že klasický PSD regulátor má velký překmit, použijeme Takahashiho návrh řídicího algoritmu z [17]. Klasický PSD regulátor bez filtrace derivační složky má velký překmit při změně žádané hodnoty. Navíc kritické zesílení a kritická perioda závisí na periodě vzorkování. Proto pro Takahashiho regulátor byly uvedeny již v kapitole 3. vztahy pro výpočet jeho parametrů (kde v indexu T znamená že parametry byly určeny při této periodě vzorkování):

$$u(k) = K_P [y(k-1) - y(k)] + K_I [w(k) - y(k)] + K_D [2y(k-1) - y(k-2) - y(k)] + u(k-1)$$

$$K_P = 0,6 K_{TKRIT} - \frac{K_I}{2}; K_I = \frac{1,2 K_{TKRIT} T}{T_{TKRIT}}; K_D = \frac{3 K_{TKRIT} T_{TKRIT}}{40T} \quad (4.1)$$

Nevýhoda experimentálního určování kritických parametrů spočívá v tom, že soustava může být uvedena do nestabilního stavu a že vyhledávání meze stability může být u soustav s velkými časovými konstantami časově náročné. Rovněž samotný technologický proces nemusí i relativně malé kmitání připustit. Tyto nevýhody nemá dále navržená modifikovaná metoda pro seřizování číslicových PID regulátorů, která byla převzata z [9].

Při diskretizaci regulačního obvodu se spojitá akční veličina upraví pomocí vzorkovače a tvarovače na stupňovou funkci, kterou je možno aproximovat původní spojitou funkcí zpožděnou o polovinu vzorkovacího intervalu T . Zjednodušeně je tedy možno předpokládat, že diskrétní model soustavy se liší od spojitého tím, že obsahuje navíc dopravní zpoždění o velikosti $T/2$. Dopravní zpoždění nemění amplitudu, ale s rostoucí frekvencí lineárně zvětšuje fázový posun

$$\varphi = -\frac{T\omega}{2} \quad (4.2)$$

Na kritické frekvenci ω_K má soustava fázový posun $-\pi$ a zesílení A_K , pro které platí

$$A_K K_{TKRIT} = -1 \quad (4.3)$$

Při diskrétním řízení se vlivem fázového posunu φ , způsobeného diskretizací, změní kritická frekvence a protože na jiné frekvenci má soustava jiné zesílení, změní se i kritické zesílení. Kritické hodnoty závisí na zvolené periodě vzorkování a proto dále budou označeny jako funkce T_0 , tj. $K_{PK}(T_0) = K_{TKRIT}$ a $T_K(T_0) = T_{TKRIT}$.

4.1 Výpočet kritických parametrů pro model n -tého řádu

Předpokládejme diskrétní přenosovou funkcí regulované soustavy ve tvaru

$$G_p(z) = \frac{Y(z)}{U(z)} = \frac{z^{-d} B(z^{-1})}{A(z^{-1})} \quad (4.4)$$

s polynomy

$$A(z^{-1}) = 1 + \sum_{i=1}^n a_i z^{-i} = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n} \quad (4.5)$$

$$B(z^{-1}) = \sum_{i=1}^n b_i z^{-i} = b_1 z^{-1} + b_2 z^{-2} + \dots + b_n z^{-n} \quad (4.6)$$

kde d je počet kroků dopravního zpoždění. Dále uvažujme diskrétní přenosovou funkci proporcionálního regulátoru

$$G_R(z) = \frac{U(z)}{E(z)} = K_P \quad (4.7)$$

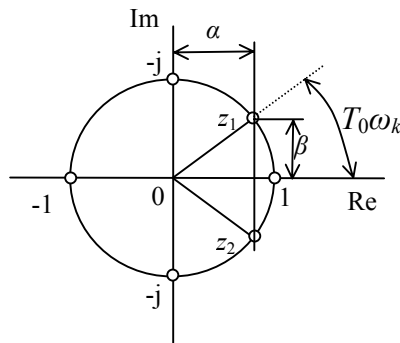
Potom přenosová funkce řízení uzavřeného regulačního obvodu má tvar

$$G_w(z) = \frac{Y(z)}{W(z)} = \frac{G_p(z)G_R(z)}{1 + G_p(z)G_R(z)} = \frac{z^{-d} K_p B(z^{-1})}{A(z^{-1}) + z^{-d} K_p B(z^{-1})} \quad (4.8)$$

Jmenovatel přenosové funkce (4.8) je charakteristický polynom

$$D(z^{-1}) = A(z^{-1}) + z^{-d} K_p B(z^{-1}) \quad (4.9)$$

jehož póly určují dynamické chování uzavřeného regulačního obvodu. Je zřejmé, že uzavřený regulační obvod bude na hranici stability, jestliže alespoň jeden pól charakteristického polynomu (4.9) bude umístěn na jednotkové kružnici a ostatní budou uvnitř jednotkové kružnice. Potom je splněna podmínka $K_P = K_{PK}(T_0)$.



Obr. 4.1: Umístění kritických pólů na jednotkové kružnici

Podle **Obr. 4.1** existují následující možnosti umístění pólů na jednotkové kružnici, které uvedou regulační obvod na mez stability:

- charakteristický polynom (4.9) obsahuje dvojici komplexně sdružených pólů $z_{1,2} = \alpha \pm j\beta; (\alpha^2 + \beta^2 = 1)$, takže příslušnou část charakteristického polynomu můžeme uvést ve tvaru součinu kořenových činitelů

$$D_1(z) = (z - z_1)(z - z_2) = z^2 - 2\alpha z + 1 \quad (4.10)$$

- charakteristický polynom (4.9) obsahuje jeden nebo více reálných pólů $z_j = \alpha_j = -1$; ($\beta_j = 0$), takže příslušnou část charakteristického polynomu můžeme uvést ve tvaru

$$D_2(z) = (z + 1)^j \quad (4.11)$$

V prvním případě musí být charakteristický polynom (4.9) dělitelný polynomem (4.10), což vede na řešení polynomiální rovnice

$$z^{n+d} \left[A(z^{-1}) + z^{-d} K_{PK}(T_0) B(z^{-1}) \right] = (z^2 - 2\alpha z + 1) z^{n+d-2} E(z^{-1}) \quad (4.12)$$

kde

$$E(z^{-1}) = 1 + \sum_{i=1}^{n+d-2} e_i z^{-i} \quad (4.13)$$

a $K_{PK}(T_0)$, α a e_i jsou neznámé parametry polynomiální rovnice (4.13).

Ve druhém případě obdržíme polynomiální rovnici

$$z^{n+d} \left[A(z^{-1}) + z^{-d} K_{PK}(T_0) B(z^{-1}) \right] = (z + 1)^j z^{n+d-j} F(z^{-1}) \quad (4.14)$$

kde

$$F(z^{-1}) = 1 + \sum_{i=1}^{n+d-j} f_i z^{-i} \quad (4.15)$$

a $K_{PK}(T_0)$, f_i jsou neznámé parametry polynomiální rovnice (4.14). V obou případech vybereme takové řešení, pro které platí $K_{PK}(T_0) > 0$ a které neobsahuje nestabilní póly.

Místo řešení rovnice (4.14) můžeme vypočítat kritické zesílení $K_{PK}(T_0)$ jednodušším způsobem. Pro pól $\alpha = -1$ vytváří kmitavou složku člen $(-1)^k$, kterému odpovídá spojitá funkce $\cos \frac{\pi}{T_0} t$. Kritická frekvence je v tomto případě dána vztahem $\omega_K = \frac{\pi}{T_0}$.

Na druhé straně uvážíme-li vlastnosti takto umístěného pólu, je zřejmé že v reálném přetlumeném systému se může vyskytovat jen tehdy, zvolíme-li příliš dlouhou periodu vzorkování, případně při velkém rušení vstupujícím do identifikačního algoritmu a které tím vytvořilo předpoklady pro nereálný odhad modelu procesu.

Výpočet kritické periody kmitů závisí na umístění pólů na jednotkové kružnici v komplexní rovině z .

Dosadíme-li za ω do základní definice pro Z – transformaci

$$z = e^{j\omega T_0} = \cos \omega T_0 + j \sin \omega T_0 \quad (4.16)$$

Z Obr. 4.1 je zřejmé, že kritickou periodu kmitů můžeme vypočítat ze vztahů

$$\cos(T_0 \omega_K) = \alpha; \quad \omega_K = \frac{1}{T_0} \arccos \alpha; \quad T_K(T_0) = \frac{2\pi}{\omega_K} \quad (4.17)$$

V případě reálných kritických pólů $z_j = -1$ platí pro kritickou periodu kmitů vztahy

$$\cos(T_0 \omega_K) = -1; \quad \omega_K = \frac{\pi}{T_0}; \quad T_K(T_0) = 2T_0 \quad (4.18)$$

Dále budou odvozeny vztahy pro výpočet kritického proporcionálního zesílení pro model prvního až třetího řádu. V případě modelu druhého řádu bude ukázáno, že kritické parametry lze odvodit i pomocí jiných přístupů. Postup je převzat z [9].

4.1.1 Výpočet kritického zesílení pro model prvního řádu

Pokud reálný proces aproximujeme modelem prvního řádu, položíme hodnoty parametrů $n = 1$, $d = 0$ v rovnicích (4.4) - (4.6). Charakteristický polynom (4.9) má potom tvar

$$D(z) = z + a_1 + K_p b_1 \quad (4.19)$$

V případě použití modelu prvního řádu může existovat pouze jeden kritický reálný pól a proto polynomiální rovnice (4.14) bude

$$z + a_1 + K_{PK}(T_0)b_1 = z + 1 \quad (4.20)$$

a pro kritické zesílení bude platit vztah

$$K_{PK}(T_0) = \frac{1 - a_1}{b_1} \quad (4.21)$$

4.1.2 Výpočet kritického zesílení pro model druhého řádu

Vztahy pro výpočet kritického zesílení je možné odvodit několika metodami.

Unifikovaný přístup

Pokud reálný proces aproximujeme modelem druhého řádu, položíme hodnoty parametrů $n = 2$, $d = 0$ v rovnicích (4.4) - (4.6). Charakteristický polynom (4.9) má potom tvar

$$D(z) = z^2 + (a_1 + b_1 K_p)z + (a_2 + b_2 K_p) \quad (4.22)$$

V prvním případě, tj. při úvaze polynomiální rovnice (4.12), bude mít tato rovnice pro model druhého řádu tvar

$$z^2 \{1 + a_1 z^{-1} + a_2 z^{-2} + K_{PK}(T_0)[b_1 z^{-1} + b_2 z^{-2}]\} = z^2 - 2\alpha z + 1 \quad (4.23)$$

Porovnáním koeficientů při stejných mocninách z obdržíme dvě rovnice

$$a_1 + K_{PK}(T_0)b_1 = -2\alpha; \quad a_2 + K_{PK}(T_0)b_2 = 1 \quad (4.24)$$

Z rovnic získáme vztahy pro výpočet kritického zesílení a reálné části komplexně sdruženého pólu

$$K_{PK}(T_0) = \frac{1 - a_2}{b_2}; \quad \alpha = \frac{a_2 b_1 - a_1 b_2 - b_1}{2b_2} \quad (4.25)$$

Ve druhém případě, tj. při úvaze polynomiální rovnice (4.14), bude mít rovnice pro model druhého řádu tvar

$$z^2 \{1 + a_1 z^{-1} + a_2 z^{-2} + K_{PK}(T_0)[b_1 z^{-1} + b_2 z^{-2}]\} = (z + 1)z(1 + f_1 z^{-1}) \quad (4.26)$$

Stejným postupem jako v předešlém případě obdržíme dvě rovnice

$$a_1 + K_{PK}(T_0)b_1 = 1 + f_1; \quad a_2 + K_{PK}(T_0)b_2 = f_1 \quad (4.27)$$

z nichž obdržíme pro kritické zesílení vztah

$$K_{PK}(T_0) = \frac{a_1 - a_2 - 1}{b_2 - b_1} \quad (4.28)$$

Kritické zesílení je vypočítáno buď z rovnice (4.25) nebo z rovnice (4.28) v závislosti na splnění podmínky

$$b^2 - 4c \leq 0 \quad (4.29)$$

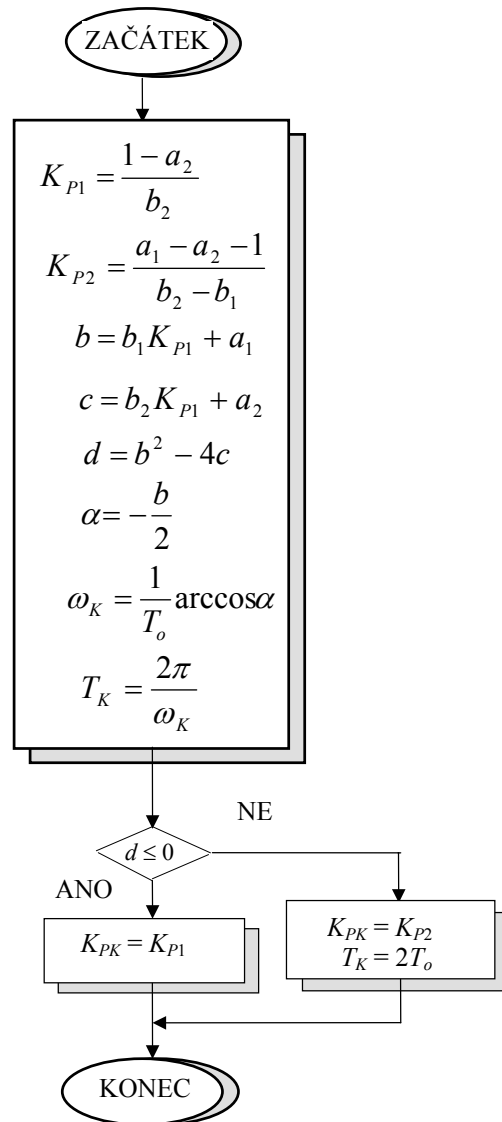
když výraz (4.29) je diskriminant charakteristické rovnice

$$z^2 + (a_1 + b_1 K_p)z + (a_2 + b_2 K_p) = 0 \quad (4.30)$$

při označení

$$b = a_1 + b_1 K_p; \quad c = a_2 + b_2 K_p \quad (4.31)$$

Vývojový diagram pro výpočet parametrů regulátoru je zobrazen na **Obr. 4.2**.



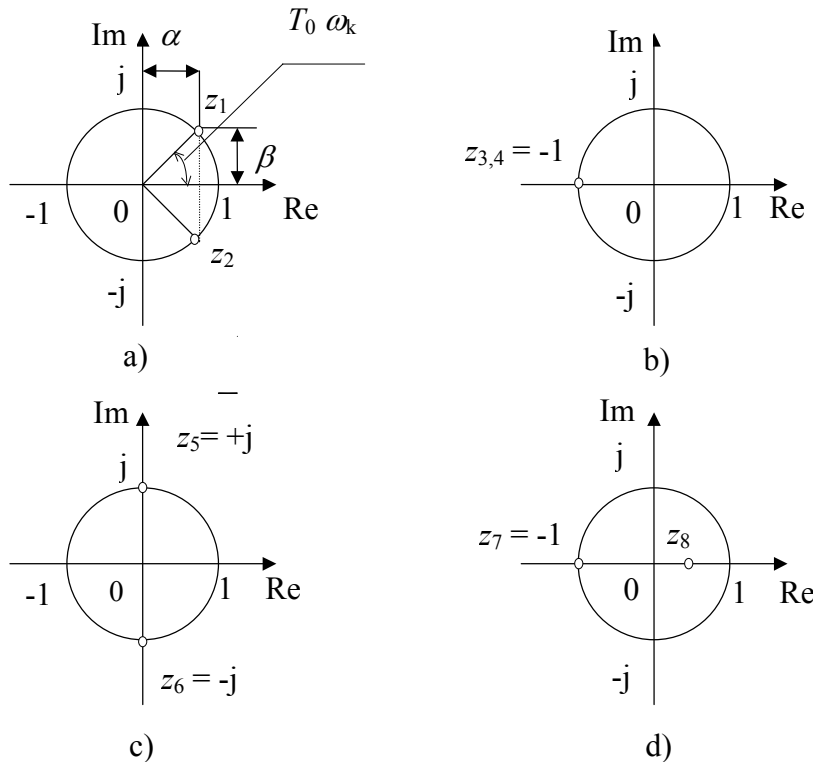
Obr. 4.2: Výpočet parametrů regulátoru pro model druhého řádu

Analýza umístění kritických pólů v z-rovině

Označíme-li charakteristický polynom

$$D(z) = z^2 + bz + c \quad (4.32)$$

s označením koeficientů podle (4.31), mohou pro uzavřený regulační obvod se soustavou druhého řádu nastat čtyři případy rozložení pólů na jednotkové kružnici, aby obvod byl na mezi stability **Obr. 4.3**:



Obr. 4.3: Umístění kritických pólů pro model druhého řádu

a) Charakteristický polynom (4.32) má dvojici komplexně sdružených pólů $z_{1,2} = \alpha \pm j\beta$ (**Obr. 4.3 a**)), pro které musí platit $\alpha^2 + \beta^2 = 1$. Charakteristický polynom (4.32) můžeme potom vyjádřit jako součin kořenových činitelů

$$D(z) = (z - z_1)(z - z_2) \quad (4.33)$$

Do rovnice (4.32) dosadíme za z_1 a z_2

$$D(z) = (z - \alpha - j\beta)(z - \alpha + j\beta) = z^2 - 2\alpha z + \alpha^2 + \beta^2 = z^2 - 2\alpha z + 1 \quad (4.34)$$

Z porovnání rovnice (4.32) a (4.34) plyne, že $b = -2\alpha$ (tedy $\alpha = -b/2$) a $c = 1$. Dosadíme-li za proměnnou c z rovnice (4.32) za předpokladu $K_P = K_{PK}(T_0)$

$$a_2 + b_2 K_{PK}(T_0) = 1 \quad (4.35)$$

obdržíme vztah pro výpočet kritického zesílení vztah a reálné složky komplexně sdruženého pólu vztahy (4.25).

b) Charakteristický polynom (4.32) má dvojnásobný reálný pól $z_{3,4} = \alpha$ (imaginární složka $\beta=0$) - **Obr. 4.3 b**). Regulační obvod je na mezi stability pouze v případě, že $\alpha =$

-1, protože kladný reálný pól $\alpha = 1$ nepřivede regulační obvod do kmitavého stavu. Charakteristický polynom (4.32), vyjádřený jako součin kořenových činitelů, má v tomto případě tvar

$$D(z) = (z + 1)^2 = z^2 + 2z + 1 \quad (4.36)$$

Poněvadž $b = 2$ a $c = 1$, lze na základě porovnání rovnic (4.31) dokázat, že ke stejné hodnotě kritického zesílení $K_{PK}(T_0)$ lze dospět v tomto případě jak použitím vztahu (4.25), tak i použitím vztahu (4.28).

c) Charakteristický polynom (4.176) má pouze ryze imaginární póly $z_{5,6} = \pm j$ (reálná složka $\alpha = 0$) - **Obr. 4.3 c)**, potom charakteristický polynom (4.32) má tvar

$$D(z) = (z + j)(z - j) = z^2 + 1 \quad (4.37)$$

Z rovnice (4.38) je zřejmé, že $b=0$, $c=1$ a proto rovněž pro výpočet $K_{PK}(T_0)$ platí první vztah (4.25).

d) Charakteristický polynom (4.32) má jeden pól $z_7 = -1$, druhý reálný $z_8 < |1|$ (stabilní ležící uvnitř jednotkové kružnice) - **Obr. 4.3 d)**. Charakteristický polynom (4.32) můžeme opět vyjádřit jako součin kořenových činitelů

$$D(z) = (z + 1)(z - z_8) \quad (4.38)$$

Z rovnic (4.32) a (4.38) plyne, že charakteristický polynom (4.32) musí být dělitelný činitelem $z + 1$ beze zbytku. Protože platí

$$\frac{z^2 + bz + c}{z + 1} = z + b - 1 + \frac{1 - b + c}{z + 1} \quad (4.39)$$

je podmínka nulového zbytku splněna v případě

$$1 - b + c = 0 \quad (4.40)$$

Dosaďme z rovnic (4.31) za proměnné b a c do rovnice (4.40).

$$1 - a_1 - K_{PK}(T_0)b_1 + a_2 + K_{PK}(T_0)b_2 = 0 \quad (4.41)$$

Po úpravě obdržíme vztah (4.41). Pól $z_8 = 1 - b$ leží uvnitř jednotkové kružnice. Kritické zesílení $K_{PK}(T_0)$ se tedy vypočítá buď z prvního vztahu rovnic (4.25) nebo z rovnice (4.28) podle toho, zda je splněna podmínka (4.29). Je zřejmé, že podmínka (4.29) bude splněna, když charakteristický polynom (4.32) má póly umístěné podle a), b) nebo c); pro výpočet kritického zesílení $K_{PK}(T_0)$ se potom použije první vztah rovnic (4.25). Jestliže podmínka (4.29) není splněna, potom charakteristický polynom (4.32) má póly umístěné podle d) a $K_{PK}(T_0)$ se vypočítá podle vztahu (4.28).

Odvození pomocí bilineární transformace

Ke stejným vztahům lze rovněž dospět (z cvičných důvodů) použitím bilineární transformace

$$z = \frac{w + 1}{w - 1}; \quad (w = \alpha + j\beta) \quad (4.42)$$

kdy se jednotková kružnice v rovině z zobrazí jako imaginární osa roviny w , přičemž bodu

$(-1;0)$ v rovině z odpovídá v rovině w počátek souřadného systému. Zavedením transformace (4.42) do charakteristické rovnice (4.30) obdržíme kvadratickou rovnici

$$w^2(1+b+c) + w(2-2c) + 1-b+c = 0 \quad (4.43)$$

Rovnici (4.43) můžeme řešit z hlediska stability způsobem známým z teorie spojitéch systémů. Spojitý regulační obvod je na mezi stability, pokud kořeny charakteristické rovnice (4.43) leží na imaginární ose komplexní roviny w , tj. pokud $\alpha = 0$. Rovnice (4.43) se potom modifikuje na tvar

$$-\beta^2(1+b+c) + j\beta(2-2c) + 1-b+c = 0 \quad (4.44)$$

a můžeme ji rozdělit na reálnou a imaginární část

$$-\beta^2(1+b+c) + 1-b+c = 0 \quad (4.45)$$

$$\beta(1-c) = 0$$

Rovnice (4.45) má dvě řešení:

$1-c=0$, odkud plyne, že $c=1$ a po dosazení do druhé rovnice (4.31) obdržíme pro výpočet kritického zesílení první vztah rovnic (4.25),

$\beta=0$, čímž po dosazení do první rovnice (4.45) obdržíme rovnicí (4.40), takže pro výpočet kritického zesílení platí vztah (4.28).

Vlastní algoritmus automaticky se nastavujícího regulátoru se skládá z identifikační části a z návrhu regulátoru. V identifikační části řídicího algoritmu je použit ARX model ve tvaru (3.49) nebo (3.51), kde regresní vektor je ve tvaru

$$\varphi^T(k-1) = (u(k-1), u(k-2), -y(k), -y(k-1)) \quad (4.46)$$

potom dostaneme odhad parametrů ve tvaru

$$\theta(k) = (\hat{b}_1, \hat{b}_2, \hat{a}_1, \hat{a}_2)^T \quad (4.47)$$

Jako regulátor použijeme Takahashiho regulátor (4.1) a jeho parametry určíme ve tvaru

$$K_R = 0,6K_{PK} - \frac{K_I}{2}; \quad K_I = \frac{1,2K_{PK}T_0}{T_K}; \quad K_D = \frac{3K_{PK}T_K}{40T_0} \quad (4.48)$$

Jestliže zavedeme následující označení pro jednotlivé prvky regresního vektoru (4.46) $d_1 = u(k-1)$, $d_2 = u(k-2)$, $d_3 = -y(k)$, $d_4 = -y(k-1)$, pomocná proměnná $d_5 = -y(k-2)$

můžeme přepsat rovnici (4.1) do následujícího tvaru

$$u(k) = K_R[d_3 - d_4] + K_I[w(k) + d_3] + K_D[d_5 - 2d_4 + d_3] + d_1 \quad (4.49)$$

Z prvního vztahu rovnic (4.25) je zřejmé, že dělení nulou nastane v případě $\hat{b}_2 = 0$, ve vztahu (4.28) k tomuto případu dojde jestliže $\hat{b}_1 = \hat{b}_2$. Dále musíme vzít v úvahu tu skutečnost, že ve druhém vztahu rovnic (4.17) musí být splněna podmínka

$$|\alpha| = \left| -\frac{b}{2} \right| \leq 1 \quad (4.50)$$

Potom algoritmus automaticky se nastavujícího regulátoru sestává v každém kroku periody vzorkování z následujících kroků:

Krok 1. Cyklická záměna dat v regresním vektoru (4.46):
pomocná proměnná $d_5 = d_4$; $d_4 = d_3$; $d_3 = -y(k)$; $d_2 = d_1$; $d_1 = u(k-1)$;

Krok 2. Odhad parametrů modelu procesu (4.47).

Krok 3. Jestliže odhad parametru \hat{b}_2 je menší než strojová nula nebo jestliže $\hat{b}_1 = \hat{b}_2$, potom se použije předchozí odhad parametru \hat{b}_2 .

Krok 4. Výpočet kritického zesílení $K_{PK}(T_0)$ a kritické periody kmitů $T_K(T_0)$ (viz vývojový diagram na **Obr. 4.2**). Jestliže $\alpha < -1$, položí se $\alpha = -1$, jestliže $\alpha > 1$, položí se $\alpha = 1$ (viz vztah (4.50)).

Krok 5. Výpočet parametrů číslicového PID regulátoru podle vztahů (4.48).

Krok 6. Výpočet akčního zásahu podle vztahu (4.49).

Krok 7. Omezíme velikost akčního zásahu $u(k)$ vzhledem k omezení akčního členu a technologickým podmínkám a zašleme jej na D/A převodník a $u(k)$ uložíme. V začátku následující periody se změní významově na $u(k-1)$

4.1.3 Výpočet kritického zesílení pro model třetího řádu

Vztahy pro výpočet kritického zesílení pro proces popsaný modelem třetího řádu ($n = 3$, $d = 0$ v rovnicích (4.4) - (4.6)) lze odvodit analogickým způsobem. Odvození provedeme pouze použitím unifikovaného přístupu, i když i v tomto případě lze použít metody bilineární transformace. Charakteristický polynom (4.9) má tvar

$$D(z) = z^3 + (a_1 + b_1 K_P)z^2 + (a_2 + b_2 K_P)z + a_3 + b_3 K_P \quad (4.51)$$

V prvním případě, tj. při úvaze polynomiální rovnice (4.12), bude mít tato rovnice pro model třetího řádu pro $K_P = K_{PK}(T_0)$ tvar

$$z^3 \{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} + K_{PK}(T_0)[b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}]\} = \\ = (z^2 - 2\alpha z + 1)z(1 + e_1 z^{-1}) \quad (4.52)$$

Porovnáním koeficientů při stejných mocninách z obdržíme tři rovnice s třemi neznámými $K_{PK}(T_0)$, α a e_1

$$\begin{aligned} a_1 + b_1 K_{PK}(T_0) &= -2\alpha + e_1 \\ a_2 + b_2 K_{PK}(T_0) &= 1 - 2\alpha e_1 \\ a_3 + b_3 K_{PK}(T_0) &= e_1 \end{aligned} \quad (4.53)$$

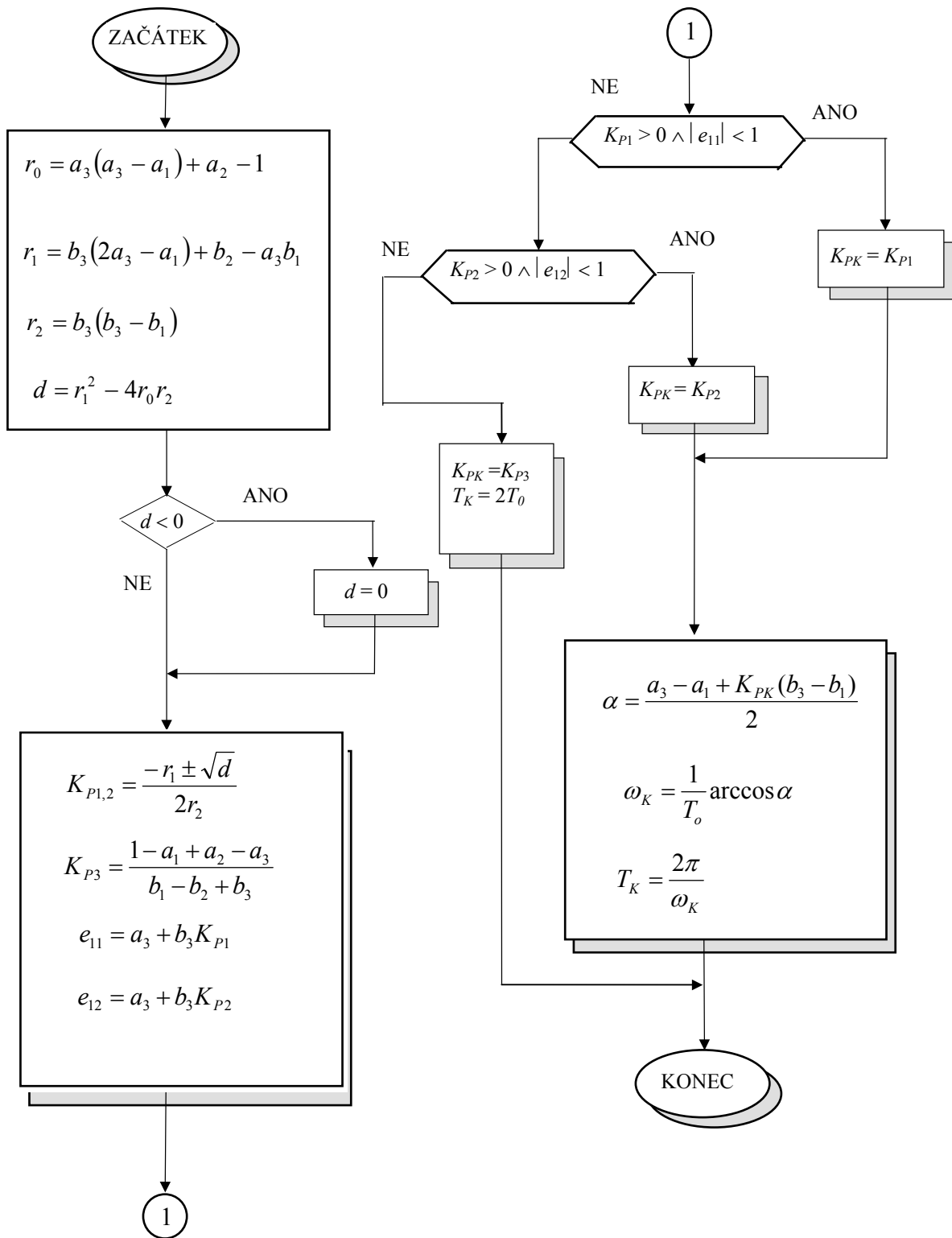
Rovnice (4.53) mají řešení

$$K_{PK1,2}(T_0) = \frac{-r_1 \pm \sqrt{d}}{2r_2}; \quad \alpha = \frac{a_3 - a_1 + K_{PK}(T_0)[b_3 - b_1]}{2} \quad (4.54)$$

kde

$$\begin{aligned} r_0 &= a_3(a_3 - a_1) + a_2 - 1; & r_1 &= b_3(2a_3 - a_1) + b_2 - a_3 b_1 \\ r_2 &= b_3(b_3 - b_1); & d &= r_1^2 - 4r_0 r_2 \end{aligned} \quad (4.55)$$

a pro výpočet e_1 se použije poslední vztah z rovnice (4.53). Správné řešení vybereme pro



Obr. 4.4: Výpočet parametrů regulátoru pro model třetího řádu

$$K_{PK}(T_0) > 0; \quad |e_i| < 1 \quad (4.56)$$

Jestliže podmínky (4.56) nejsou splněny pro $K_{PK1}(T_0)$ ani pro $K_{PK2}(T_0)$, musí se pro výpočet kritického zesílení použít rovnice (4.14), která pro model třetího řádu má tvar

$$z^3 \{1 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3} + K_{PK}(T_0)[b_1z^{-1} + b_2z^{-2} + b_3z^{-3}]\} = \quad (4.57)$$

$$= (z+1)z^2(1 + f_1z^{-1} + f_2z^{-2})$$

Porovnáním koeficientů při stejných mocninách z obdržíme tři rovnice

$$\begin{aligned} a_1 + b_1 K_{PK}(T_0) &= 1 + f_1 \\ a_2 + b_2 K_{PK}(T_0) &= f_1 + f_2 \\ a_3 + b_3 K_{PK}(T_0) &= f_2 \end{aligned} \quad (4.58)$$

jejichž řešením obdržíme pro kritické zesílení vztah

$$K_{PK}(T_0) = \frac{1 - a_1 + a_2 - a_3}{b_1 - b_2 + b_3} \quad (4.59)$$

Vývojový diagram pro výpočet parametrů regulátoru je zobrazen na **Obr. 4.4**.

Algoritmus tohoto samočinně se nastavujícího regulátoru se sestaví obdobně jako v případě regulátoru pro model druhého řádu. V identifikační části je opět použit model ARX, kde

$$\theta(k) = (\hat{b}_1, \hat{b}_2, \hat{b}_3, \hat{a}_1, \hat{a}_2, \hat{a}_3)^T \quad (4.60)$$

je vektor odhadu parametrů modelu procesu a

$$\varphi^T(k-1) = (u(k-1), u(k-2), u(k-3), -y(k), -y(k-1), -y(k-2)) \quad (4.61)$$

je regresní vektor.

5 OPTIMÁLNÍ REGULÁTORY

5.1 Dynamické programování

Dynamické programování je považováno za jednu z obecných metod navrhování **optimálního řízení** různých reálných objektů. Někdy je výraz dynamické programování nahrazován výrazem dynamické plánování. Teorie dynamického programování byla rozvíjena zejména v oblasti technicko-ekonomických úloh, např. jak nejlépe využívat zařízení, jaké jsou optimální zásoby. Nejčastěji je dynamické programování spojováno se jménem Richarda Bellmana, kterému v roce 1957 vyšla kniha *Dynamic Programming*.

5.1.1 Úloha dynamického programování

Dynamické programování umožňuje návrh optimálního řízení procesů, které mohou být v čase diskrétní nebo spojité, také jejich řízení může mít diskrétní nebo spojitý charakter. Abychom mohli úlohu optimálního řízení matematicky popsat musíme zavést **kriteriální funkci** Z , její hodnota bude charakterizovat kvalitu a úspěšnost operace.

Veličina Z může být vybrána různým způsobem. Při plánování činnosti výrobní organizace to může být objem výroby za jednotku času, u dopravní organizace počet tunokilometrů nebo střední cena za tunokilometr. Z teorie řízení jsou známá kritéria minimální plochy kvadrátu odchylky, nejkratší doba dosažení zadaného tolerančního pásma při dalším setrvání v tomto pásmu a pod.

Úlohou optimálního řízení je nalezení takové posloupnosti řídicích zásahů, aby hodnota kriteriální funkce Z byla maximální (nebo minimální) ve srovnání s jakoukoliv jinou posloupností. U spojitého řízení je posloupnost nahrazena průběhy hodnot řídicích proměnných. Úlohu vedoucí k minimalizaci hodnoty Z lehce převedeme např. změnou znaménka na úlohu maximalizace, budeme proto v dalším hovořit pouze o maximalizaci kritéria. Označíme-li řídicí proměnnou u (může to být i vektor s více složkami), řekneme, že cílem je nalezení takového řízení u , při kterém hodnota Z bude maximální.

Specifikou dynamického programování je rozkládání řízeného procesu na řadu po sobě jdoucích kroků nebo etap. Některé procesy jsou víceetapové již svým charakterem, u jiných je možno rozložení na více kroků realizovat uměle. Také proces hledání optimálního řízení je víceetapový a rozvíjí se postupně.

Přirozeně víceetapovým je např. proces výroby probíhající ve směnách, dnech, týdnech, měsících a rocích. Podobně let vícestupňové rakety, dopravní procesy rozložitelné na pohyby od jednoho dopravního uzlu ke druhému, obchodní i vojenské operace sestávající z jednotlivých akcí, šachové úlohy s posloupností tahů, Rubikova kostka atd. Spojité procesy rozkládáme uměle na takový počet etap, aby vyhovovala přesnost výsledku.

Předpokládejme, že řídicí proměnná je vektorem s n složkami a v i -tém kroku bude použito řízení

$$u_i = (x_i^1, x_i^2, x_i^3, \dots, x_i^n).$$

Úplné řízení během m -etapového procesu vyjádříme pomocí řady

$$u_1, u_2, \dots, u_i, \dots, u_m$$

Kriteriální funkce Z je tedy závislá na uvedené řadě a hledání jejího optima je hledáním posloupnosti m bodů v n -rozměrném prostoru nebo jinými slovy hledáním bodu v prostoru $m \times n$ s maximální hodnotou Z . Funkce Z však nebývá jednoduchá a hledání maxima klasickým způsobem (první derivace podle jednotlivých proměnných rovna nule atd.) vede málokdy k cíli. Maximum může být na okraji prostoru $m \times n$, v řadě případů se funkce ani derivovat nedá. Hledání maxima pomocí variačního počtu převádí obvykle danou úlohu na úlohy, které nejsou jednodušší.

Hlavní myšlenkou DP je převedení úlohy nalezení maxima funkce velkého počtu proměnných na vícenásobně opakovanou úlohu hledání maxima funkce s menším počtem proměnných, která je podstatně jednodušší.

V dalším se budeme zabývat systémy, u kterých bude **součtová (additivní) kriteriální funkce**. Nejedná se o velké omezení, u většiny systémů a problémů je additivnost při posuzování optima charakteristickou vlastností. Spotřeba zdrojů, finančních prostředků, dosažený čas a pod. jsou dány pro celý proces součtem položek za jednotlivé etapy. Také **multiplikativní kriteria** (např. pravděpodobnost úspěšného přechodu je dána součinem pravděpodobností dílčích přechodů) je možno převést na additivní pomocí logaritmů. Budeme tedy předpokládat, že

$$Z = z_1 + z_2 + z_3 + \dots + z_m$$

5.1.2 Princip dynamického programování

Dynamické programování převádí úlohu nalezení maxima Z na mnohonásobně opakovanou úlohu hledání maxima pro jednotlivé etapy. Přitom se však nezapomíná na ostatní etapy. Naopak, řízení v každém kroku je vybíráno s ohledem na všechny možné důsledky v budoucnosti. Dynamické programování je tedy možno považovat za plánování s uvažováním perspektivy, tedy plánování vycházející ze širších zájmů celého procesu.

Mezi všemi etapami existuje však jedna, u které úzké zájmy jednoho kroku se ztotožňují se zájmem celého procesu. Je to etapa poslední. Jestliže je tato optimálně navržena, můžeme se zabývat rozšířením problému o etapu předposlední, později o etapu předcházející atd.

Proto se proces dynamického programování vždy z hlediska času rozvíjí od konce k počátku. Nejprve se plánuje poslední etapa. Protože však není známo, jak skončila předposlední etapa, musíme udělat o výsledku předposlední etapy celou řadu předpokladů. Pro každý z předpokladů pak musíme určit optimální řízení v poslední etapě. Toto můžeme nazvat **podmíněným optimálním řízením**. Princip dynamického programování požaduje nalezení podmíněného optimálního řízení pro libovolný možný výsledek předchozího kroku.

Procesy a jejich stavy charakterizujeme celou řadou proměnných. Množinu hodnot těch proměnných, které ovlivňují budoucí chování procesu nazveme **stav procesu (řízení soustavy)**. Například při řízení dopravního prostředku můžeme za stavové proměnné považovat: polohu, orientaci v prostoru, rychlost, hmotnost prostředku, otáčky motorů, zařazený rychlostní stupeň a podobně. Chceme-li vysokou přesnost, pak se počet proměnných vztahujících se k požadovaným veličinám může zvýšit (teplota, vlhkost, atd.). Do charakterizované soustavy je účelné zahrnout i vlivy nejbližšího okolí, které chování ovlivňuje.

Pokud bude počet složek vektoru stavu nedostatečný, projeví se to tím, že při opakovaných pokusech se stejným výchozím stavem a při stejném řízení se nebude soustava

pokaždé chovat stejně. Toto bude signalizovat nutnost najít další proměnnou, jejíž hodnoty chování ovlivňují. Stavem procesu v určitém čase budeme tedy rozumět vektor z hodnot jednotlivých stavových proměnných. Stav m -etapového procesu po i -té etapě označíme s_i . Jednotlivé hypotézy o tomto stavu označíme $1s_i, 2s_i, \dots$.

Předpokládejme, že pro každý hypotetický stav po předposlední etapě (tedy s indexem $m-1$) jsme našli podmíněné optimální řízení v poslední etapě s označením ($*$ bude označovat optimálnost):

$$u_m^*(1s_{m-1}), u_m^*(2s_{m-1}), \dots, u_m^*(j s_{m-1})$$

Tedy poslední etapa je optimalizována pro libovolný stav po etapě předposlední. Můžeme přejít k určení optimálního řízení v předposlední etapě. Znovu vytvoříme hypotézy o stavu po ukončení etapy s indexem " $m-2$ ".

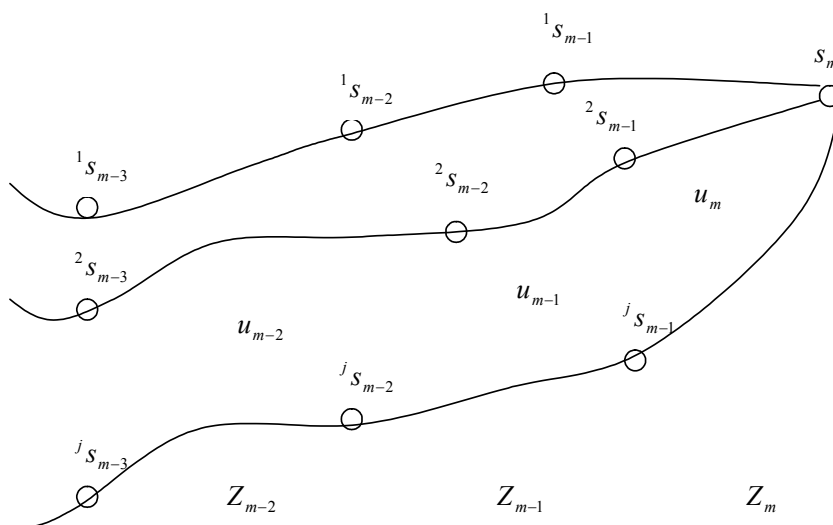
Podmíněné optimální řízení v předposlední etapě však musíme navrhnout tak, aby - současně s již vybraným řízením v poslední etapě - zajistilo optimální řízení z hlediska maxima kritériální funkce ve dvou posledních etapách. Tedy musíme pro každý hypotetický stav před předposlední etapou:

Zjistit možná řízení a na základě hodnoty kritériální funkce v předposlední etapě a optimální hodnoty kritériální funkce pro poslední etapu (pro stav, do kterého se soustava dostane na základě předposledního řízení) stanovit možnou hodnotu kritériální funkce pro obě poslední etapy. Budeme ji označovat pomocí " $+$ ":

$$Z_{m-1,m}^+(s_{m-2}, u_{m-1}) = z_{m-1}(s_{m-2}, u_{m-1}) + Z_m^*(s_{m-1})$$

Z možných řízení vybrat takové, aby hodnota kritériální funkce pro předposlední etapu byla maximální:

$$Z_{m-1,m}^*(s_{m-2}) = \max_{u_{m-1}} \{Z_{m-1,m}^+(s_{m-2}, u_{m-1})\}$$



Obr. 5.1: Výběr řízení pro optimální hodnotu kritériální funkce.

Podobně postupujeme při hledání optimálního řízení ve 3. etapě od konce atd. U první etapy je situace jednoduchá, protože známe výchozí stav a nemusíme vytvářet hypotézy. Výsledkem je u determinovaných procesů jedna nebo několik optimálních stavových trajektorií přechodu z počátečního do konečného stavu (v k -rozměrném prostoru, kde k je počet stavových proměnných) a stejný počet optimálních trajektorií řízení (v n -rozměrném prostoru). U procesů s náhodnými prvky je výsledkem celá řada pravděpodobnostních trajektorií stavů a tabulka přiřazující všem možným stavům, do kterých se proces může dostat, optimální řízení v nejbližší etapě.

Tuto část uzavřeme intuitivním principem optimálního chování, tak jak jej formuloval R. Bellman:

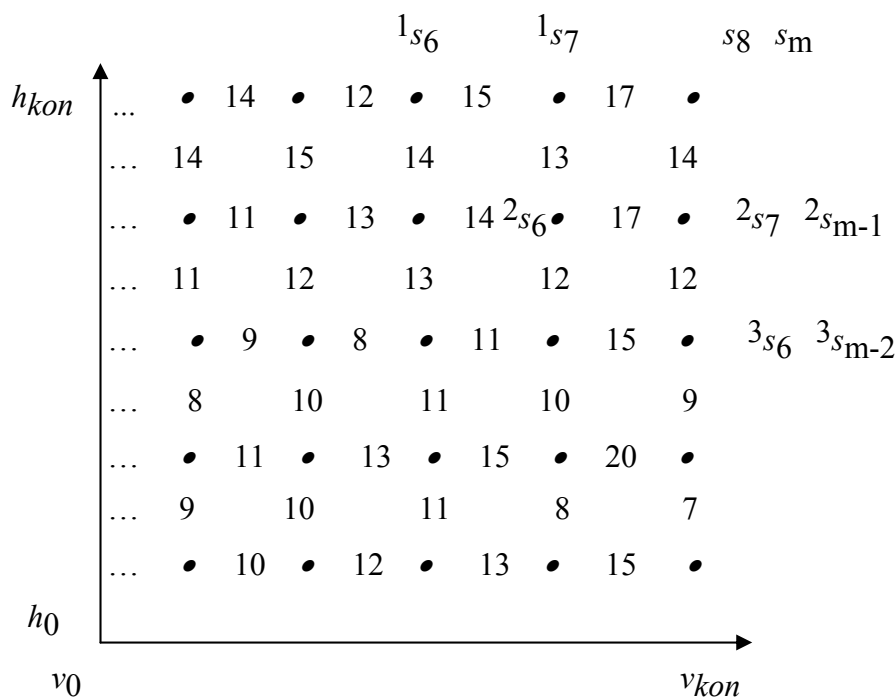
Pro jakýkoliv počáteční stav a řízení před daným okamžikem, rozhodnutí v daném okamžiku musí vycházet ze stavu soustavy v tomto okamžiku. Leží-li na optimální stavové trajektorii z počátečního do konečného stavu bod A , potom optimální trajektorie z bodu A do konečného stavu je totožná s původní optimální trajektorií mezi bodem A a konečným stavem.

5.1.3 Optimální zvyšování rychlosti a výšky

Optimální režim přechodu letounů na novou rychlost a výšku při minimální spotřebě paliva může být řešen pomocí rovněž dynamickým programováním. Uvažujme nejprve maximálně zjednodušenou úlohu.

Nechť je počáteční výška h_0 a rychlost v_0 , konečné hodnoty h_{kon} a v_{kon} . Předpokládejme 4 etapy, ve kterých se bude měnit jen výška a 4 etapy jen se změnou rychlosti, celkem tedy 8 etap, ve kterých se bude měnit výška nebo rychlost.

Je dána poslední část tabulky spotřeby paliva při jednotlivých změnách h nebo v pro 24 možných výchozích stavů.



Obr. 5.2: Spotřeba paliva při změně rychlosti a výšky

Předposlední etapa může letoun přivést do stavu 1_{s7} nebo 2_{s7} . V prvním případě je zjevné optimální řešení "zvýšit rychlost", ve druhém "zvýšit výšku". K uvedeným stavům bychom mohli připsat spotřebu: 17 a 14. Před předposlední etapou bude letoun ve stavu 1_{s6} , 2_{s6} nebo 3_{s6} . V prvním případě je jediná možnost "zvýšit rychlost", ve třetím "zvýšit výšku". Celkové spotřeby za dvě poslední etapy jsou 32 a 26. Ze stavu 2_{s6} jsou 2 možnosti: "zvýšit výšku" - pak je celková spotřeba 30 nebo "zvýšit rychlost" - pak bude spotřeba 31. Minimální je varianta "zvýšit výšku" s hodnotou 30.

Podobně bychom řešili 6. etapu a potom 5., 4. a další až k první. Označíme-li dílčí rozhodnutí "zvýšit rychlost" pomocí r a druhé rozhodnutí pomocí v , dostaneme optimální trajektorii řízení (v, v, r, r, r, r, v, v) .

Předpokládejme nyní, že obě proměnné se mohou měnit současně. Dostaneme rozšířenou tabulku, ve které budou i spotřeby pro povel rv . Postup se změní jen v tom, že v případech používání povelu rv budeme navazovat nikoliv na stavy další etapy, ale na stavy po dvou "jednoduchých" etapách. Například při řešení optima ze stavu 2_{s6} budeme vybírat ze tří variant: 30, 31 nebo spotřeba při povelu rv a přecházet do stavu $s7$ nebo $s8$.

K danému příkladu můžeme vytvořit i obrácenou úlohu, ve které bychom hledali optimální přechod ze stavu $s8$ do stavu $s0$ a tabulkové hodnoty by byly stejné. Tedy původní úlohu můžeme v tomto případě řešit i postupem "od počátku ke konci".

5.2 Lineární optimální regulátory

5.2.1 Úvod

Pojem optimální pochází již z roku 1696, kdy bratři John a James Bernoulli řešili problém o brachistochroně, což je křivka spojující dva body a v níž se dané těleso, které se pohybuje bez tření jen s pomocí zemské přitažlivosti má dostat z jednoho bodu do druhého za nejkratší čas. Motivujícím prvkem zde bylo vyřešit minimalizaci zadaného kritéria (určit minimální čas) při daných souřadnicích dvou bodů (úloha s pevnými konci). Tedy pokud se jim podařilo najít extrém řešení (optimum) pak bylo nalezeno to nejlepší možné řešení při daných podmínkách ve smyslu kritéria a jakékoliv jiné řešení již ve smyslu daného kritéria musí být stejné, maximálně jen stejně dobré nebo případně jen horší. Otázkou tedy je, proč se tedy zajímat o nějaké jiné řešení, když již bylo nalezeno řešení ve smyslu kritéria nejlepší?

V minulosti potřeba řešení této úlohy vznikla v 50-tých letech minulého století, kdy při letech do vesmíru bylo potřebné především vyřešit optimalizační problém s minimalizací paliva spotřebovaného v raketě. Daný optimalizační problém můžeme řešit různými metodami:

- Lagrangeovými multiplikátory
- Hamiltonovým principem
- Bellmanovým dynamickým programováním (1957)
- Pontrjaginovým principem minima (1962)

Pomocí uvedených metod můžeme získat exaktní řešení. Řešení optimalizačních úloh můžeme však získat i cestou opakovaných simulačních výpočtů, kdy např. bez znalosti uvedených metod můžeme získat simulační řešení (hledanou trajektorii) pomocí genetických

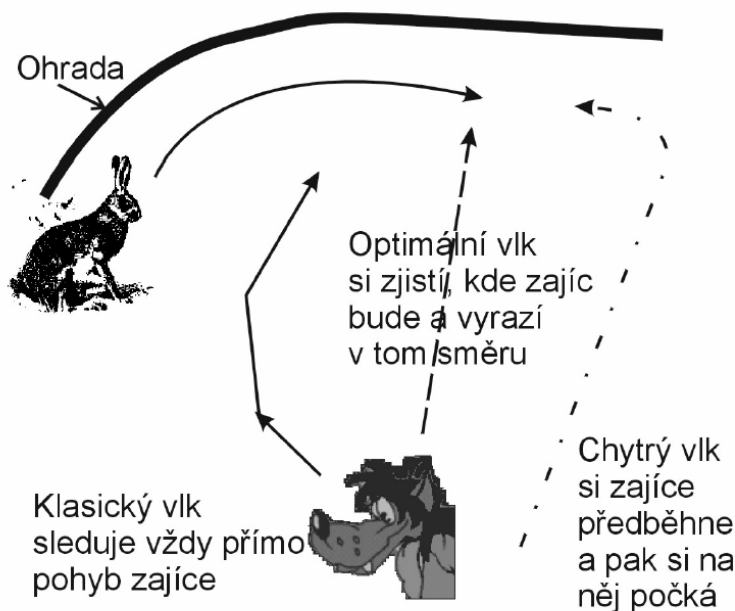
algoritmů, která se blíží optimálnímu řešení. Můžeme si tedy položit otázku, proč se těmito úlohami máme dále zabývat, když vše již bylo dávno vyřešeno. Z hlediska matematického přístupu máme sice hotový nástroj, ale z pohledu inženýra ani zdaleka ne. K tomu, aby se teorie optimálního řízení dala použít v reálném prostředí, bylo a je potřeba stále něco řešit, například:

- zajištění podmínek numerické konvergence a nalezení správného řešení (globálního extrému)
- nalezení vhodného kritéria z pohledu uživatele
- tvrdé omezující podmínky na regulaci (konečný akční zásah)
- řešení otázek časové náročnosti výpočtu oproti jiným metodám
- nalezení řešení srozumitelného pro regulační techniky v praxi atd.

Úkolem této části skript je tedy ukázat o co při optimálním řízení jde, jak je optimální řízení definováno, kde vznikají problémy a proč jsou potíže s adaptivním optimálním řízením.

5.2.2 Co rozumíme pod pojmem optimální řízení

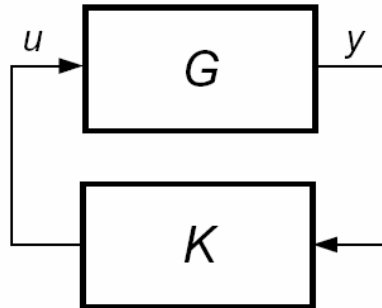
Pokud se použijeme slovo optimální, musíme mít vždy na mysli, že je pojem optimálního je dán určitými definovanými podmínkami, omezeními a daným kritériem optimálnosti (ve kterém se navíc může objevovat vliv subjektivního určení kritéria optimálnosti). Poznamenejme, že definujeme - li pevně podmínky, omezení a kritérium optimálnosti, pak nalezneme-li optimální řešení, žádné jiné řešení nemůže být ve smyslu kritéria optimálnější. Pro ilustraci podívejme se na rozdíl mezi klasickým řízením (v podobě klasického vlka honícího zajíce), optimálním řízením (optimální vlk) a prediktivním řízením (v podobě chytrého vlka) (obr.1). Otázkou však samozřejmě je, jestli zajíc sám není náhodou dostatečně předvídatelný.



Obr. 5.3: Klasické řízení vlka ve srovnání s jinými přístupy.

5.2.3 LQ, LQG regulátory

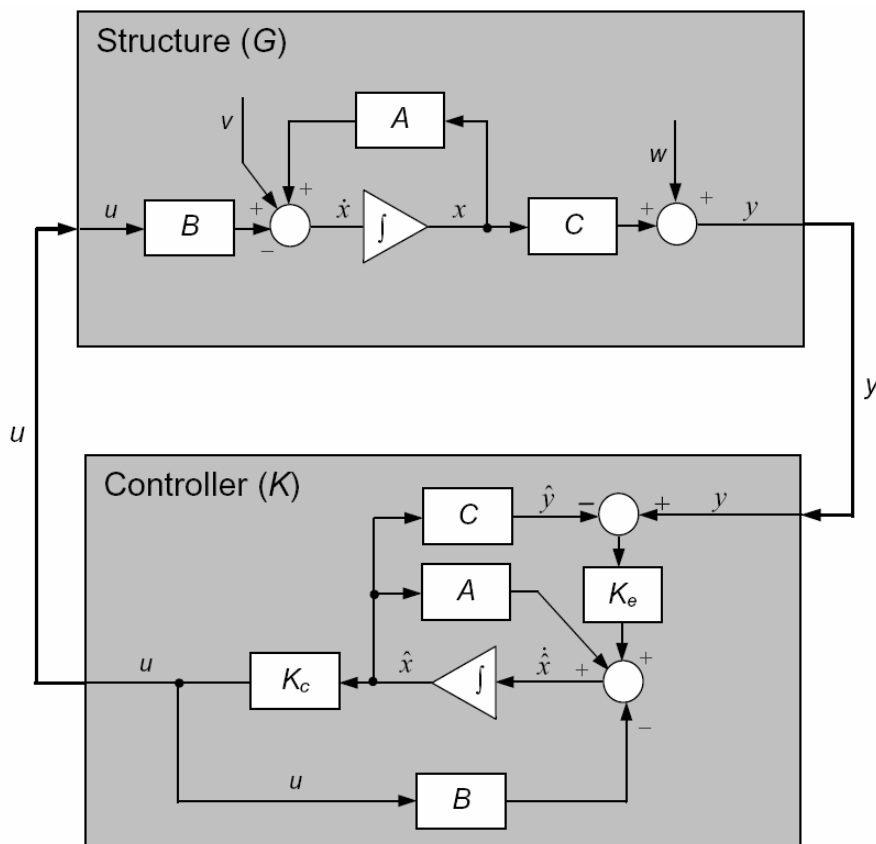
Význam zkratk je L (Linear system), Q (Quadratic cost – kvadratické kritérium) a pokud uvažujeme šum pak G (Gaussian noise – gausovský šum). Příklad realizace řídicího systému s LQG regulátorem je na obr. 2.



Obr. 5.4: LQG regulátor v uzavřené smyčce.

Význam symbolů na obr. 5.2: G je struktura procesu, K je struktura regulátoru, u – obecně vektor akčních zásahů, y – vektor výstupů z procesu.

Tak jako definice LQG regulátoru může mít mnoho variant, podobně je tomu i se strukturou regulátoru. Pro určení vnitřního stavu procesu můžeme při simulaci využít vnitřních stavů procesu, při řízení reálného procesu bývá model procesu součástí regulátoru (obr. 5.3).



Obr. 5.5: Vnitřní struktura LQG regulátoru v uzavřené smyčce.

Proces je popsán stavovými rovnicemi:

$$\begin{aligned}\dot{x} &= Ax + Bu + v \\ y &= Cx + w\end{aligned}\tag{5.1}$$

Uvnitř procesu můžeme předpokládat působení vnitřního šumu v a výstup z procesu je ovlivněn šumem w . Dále předpokládáme, že oba šумы nejsou korelované. V regulátoru je vytvořen odhad vnitřního stavu procesu pomocí rovnice estimátoru (2):

$$\hat{\dot{x}} = A\hat{x} + Bu + K_e(y - Cx)\tag{5.2}$$

Protože předpokládáme, že model procesu známe exaktně, můžeme estimovat exaktně stav s výjimkou počáteční přechodové dynamiky. Pokud uvažujeme výstup přes matici C , zavádíme estimační zisk K_e . K_c je hledané zesílení regulátoru. Obě zesílení lze určit pomocí kritéria optimality:

$$J^2 = E\left(\int_0^{\infty} (x^T Qx + u^T Ru) dt\right)\tag{5.3}$$

kde R je symetrická pozitivně definitní váhová matice vstupů a Q je symetrická pozitivně semidefinitní stavová matice

Minimum J je pak dáno použitím vztahu:

$$u^* = -K_c \hat{x}\tag{5.4}$$

Otázkou odhadu stavových proměnných se budeme zabývat později. V tomto případě bylo předpokládáno, že hodnoty stavových proměnných získáme přímo z modelu řízeného systému.

5.2.4 Stavový regulátor

V této části bude vysvětleno, jakým způsobem mohou být využity stavové veličiny při návrhu regulátoru. Je zapotřebí zdůraznit, že stavový regulátor nelze v žádném případě považovat za typ optimálního regulátoru! Stavový regulátor v tomto případě by měl mít vlastnosti klasického regulačního zpětnovazebního obvodu.

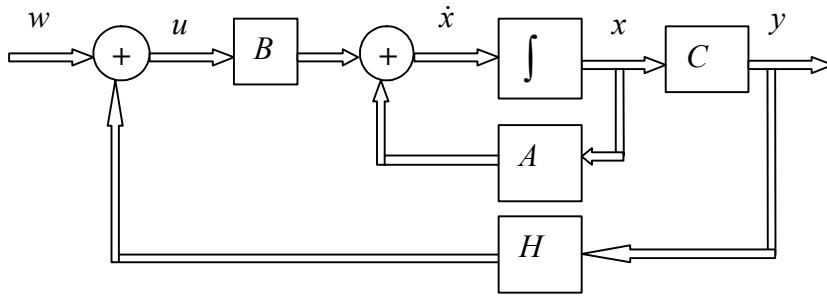
Požadavky na vlastnosti tohoto regulátoru:

- výstup regulované soustavy v ustáleném stavu je roven žádané hodnotě (nulová ustálená regulační odchylka),
- výstup regulované soustavy v ustáleném stavu nezávisí na působení konstantní poruchy (vyregulování konstantní poruchy bez chyby),
- v regulačním obvodu bude možné volit požadovanou dynamiku přechodného děje (nastavit umístění pólů regulačního obvodu).

Předpokládejme dynamický systém ve tvaru:

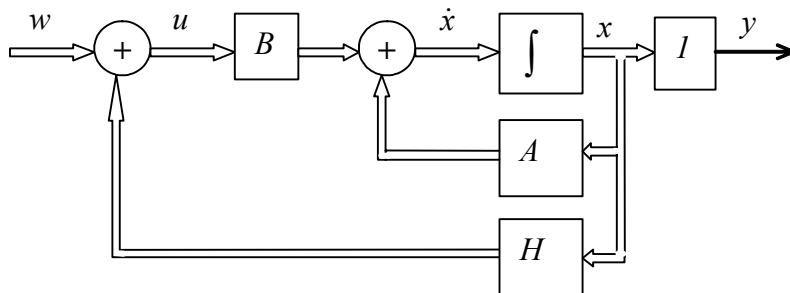
$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx\end{aligned}\tag{5.5}$$

Dalším předpokladem je, že máme přístupné všechny stavové proměnné. Vytvoříme regulační smyčku:



Obr. 5.6: Struktura regulačního obvodu se stavovým regulátorem s maticí C

Vhodnou volbou stavových proměnných a strukturou schématu upravíme matici C (jak?):



Obr. 5.7: Struktura regulačního obvodu se stavovým regulátorem

Vektor u má obecně rozměr r ($r \leq n$) – stejný rozměr by měl mít i vektor v a w . Stavový vektor je obecně n -rozměrný. Zavedená matice zpětnovazebního řízení H je tedy obecně rozměru (r, n) .

$$\dot{x} = Ax + Bu = Ax + B(w + v) = Ax + Bw + BHx \quad (5.6)$$

$$\dot{x} = (A + BH)x + Bw \quad (5.7)$$

$$\dot{x} = \bar{A}x + Bw \quad (5.8)$$

Rovnice (5.8) je rovnicí zpětnovazebního regulačního obvodu, kde matice H je zpětnovazební matice. Její prvky máme určit tak, abychom splnili zadání. Přitom využijeme skutečnosti, že v ustáleném stavu musí platit:

$$\lim_{t \rightarrow \infty} \dot{x}(t) = \dot{x}_{t \rightarrow \infty} = 0 \quad (5.9)$$

$$\bar{A}x_{t \rightarrow \infty} + Bw_{t \rightarrow \infty} = 0 \quad (5.10)$$

$$x_{t \rightarrow \infty} = -\bar{A}^{-1}Bw_{t \rightarrow \infty} \quad (5.11)$$

Příklad 5.1 Řešení klasické regulační úlohy ve stavovém prostoru.

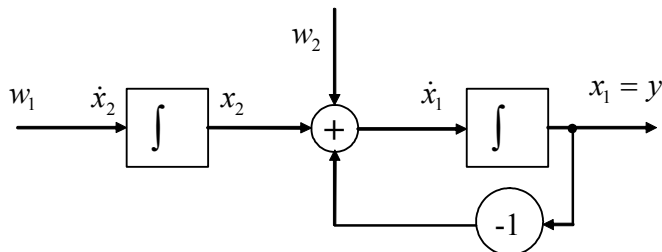
Je dána přenosová funkce regulované soustavy

$$F_S(s) = \frac{1}{s(s+1)} \quad (5.12)$$

Porucha vstupuje do části přenosu

$$F_{SU}(s) = \frac{1}{(s+1)} \quad (5.13)$$

Situace se zavedením stavových proměnných je znázorněna na následujícím obrázku.



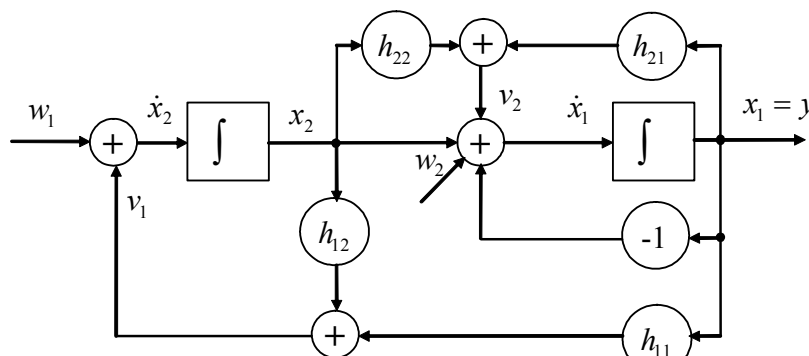
Obr. 5.8: Stavový model podle rovnice (5.12) a (5.13)

Určíme hodnoty matic:

$$A = \begin{vmatrix} -1 & 1 \\ 0 & 0 \end{vmatrix} \quad B = \begin{vmatrix} 0 & 1 \\ 1 & 0 \end{vmatrix} \quad \begin{vmatrix} v_1 \\ v_2 \end{vmatrix} = \begin{vmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{vmatrix} \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} \quad (5.14)$$

$$C = \begin{vmatrix} 1 & 0 \end{vmatrix} \quad D = 0$$

Do obrázku doplníme prvky zpětnovazební matice H . Máme dva vstupy a dvě stavové proměnné. Její rozměr je tedy $(2, 2)$.



Obr. 5.9: Stavový regulátor pro model podle rovnic (5.12) a (5.13)

Vypočítáme prvky matice H (Obr. 5.9).

$$\bar{A} = (A + BH) = \begin{vmatrix} h_{21} - 1 & 1 + h_{22} \\ h_{11} & h_{12} \end{vmatrix} \quad (5.15)$$

$$x_{t \rightarrow \infty} = -\bar{A}^{-1} B w_{t \rightarrow \infty} \quad (5.16)$$

$$\bar{A}^T = \begin{vmatrix} h_{21} - 1 & h_{11} \\ 1 + h_{22} & h_{12} \end{vmatrix}; \bar{A}_{adj} = \begin{vmatrix} h_{12} & -(1 + h_{22}) \\ -h_{11} & h_{21} - 1 \end{vmatrix}; \frac{1}{\Delta} = \frac{1}{h_{12}(h_{21} - 1) - h_{11}(1 + h_{22})} \quad (5.17)$$

$$\begin{vmatrix} x_{1t \rightarrow \infty} \\ x_{2t \rightarrow \infty} \end{vmatrix} = -\frac{1}{h_{12}(h_{21} - 1) - h_{11}(1 + h_{22})} \begin{vmatrix} h_{12} & -(1 + h_{22}) \\ -h_{11} & h_{21} - 1 \end{vmatrix} \begin{vmatrix} 0 & 1 \\ 1 & 0 \end{vmatrix} \begin{vmatrix} w_{1t \rightarrow \infty} \\ w_{2t \rightarrow \infty} \end{vmatrix} \quad (5.18)$$

$$\begin{pmatrix} x_{1t \rightarrow \infty} \\ x_{2t \rightarrow \infty} \end{pmatrix} = -\frac{1}{h_{12}(h_{21}-1) - h_{11}(1+h_{22})} \begin{pmatrix} -(1+h_{22}) & h_{12} \\ h_{21}-1 & -h_{11} \end{pmatrix} \begin{pmatrix} w_{1t \rightarrow \infty} \\ w_{2t \rightarrow \infty} \end{pmatrix} \quad (5.19)$$

Pro větší názornost rozepíšeme předchozí rovnici na dvě (5.21) a (5.23):

$$x_{1t \rightarrow \infty} = -\frac{1}{h_{12}(h_{21}-1) - h_{11}(1+h_{22})} (-(1+h_{22})w_{1t \rightarrow \infty} + h_{12}w_{2t \rightarrow \infty}) \quad (5.20)$$

Z rovnice (5.20) vyplývá požadavek:

$$x_{1t \rightarrow \infty} = w_{1t \rightarrow \infty} \Rightarrow h_{12} = 0 \quad (5.21)$$

a dále musí platit

$$x_{1t \rightarrow \infty} = -\frac{1}{-h_{11}(1+h_{22})} (-(1+h_{22})w_{1t \rightarrow \infty}) \Rightarrow h_{11} = -1 \quad (5.22)$$

$$x_{2t \rightarrow \infty} = -\frac{1}{h_{12}(h_{21}-1) - h_{11}(1+h_{22})} ((h_{21}-1)w_{1t \rightarrow \infty} - h_{11}w_{2t \rightarrow \infty}) \quad (5.23)$$

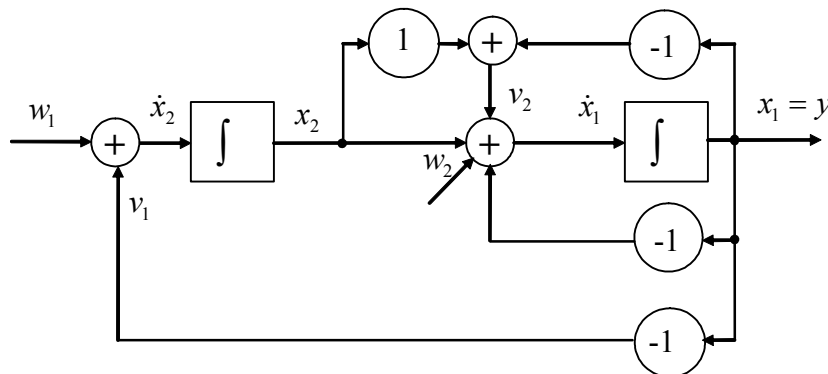
Z výše uvedené rovnice však zbývající prvky matice H nemůžeme určit. Prvky určíme z požadovaných vlastních čísel matice.

$$sI - \bar{A} = \begin{vmatrix} s - (h_{21}-1) & -(1+h_{22}) \\ 1 & s \end{vmatrix} = \quad (5.24)$$

$$s^2 - (h_{21}-1)s + 1 + h_{22} = 0 \quad (5.25)$$

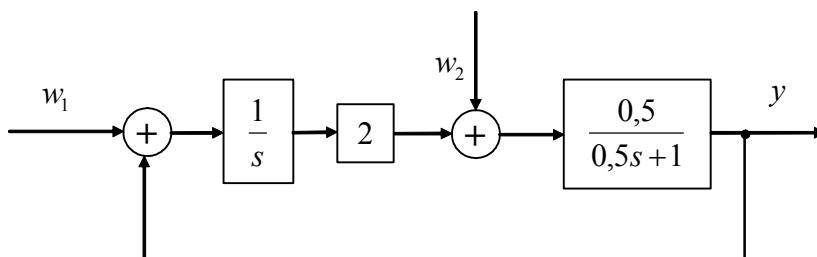
$$s_{1,2} = \frac{h_{21}-1}{2} \pm \frac{\sqrt{(h_{21}-1)^2 - 4(1+h_{22})}}{2} \quad (5.26)$$

Pokud například požadujeme umístění pólů uzavřeného obvodu $s_{1,2} = -1 \pm j$ pak z rovnice (5.26) vypočteme $h_{21} = -1$; $h_{22} = 1$.



Obr. 5.10: Řešení příkladu 5.1.

Zbývá ještě určit jaký vlastně ekvivalent přenosové funkce byl navržen.



Obr. 5.11: Regulační obvod podle příkladu 5.1.

Příklad 5.2 Řešení regulační úlohy ve stavovém prostoru se změnou zadání.

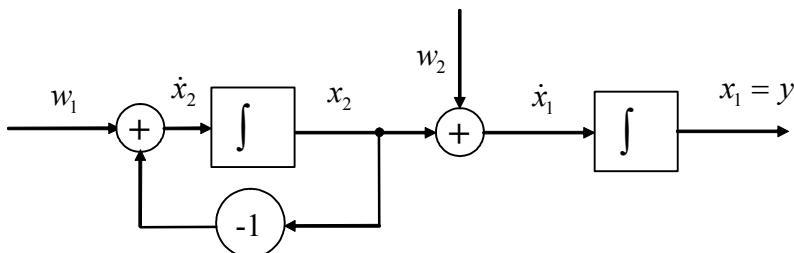
Je dána přenosová funkce regulované soustavy

$$F_s(s) = \frac{1}{s(s+1)} \tag{5.27}$$

Porucha vstupuje do části přenosu

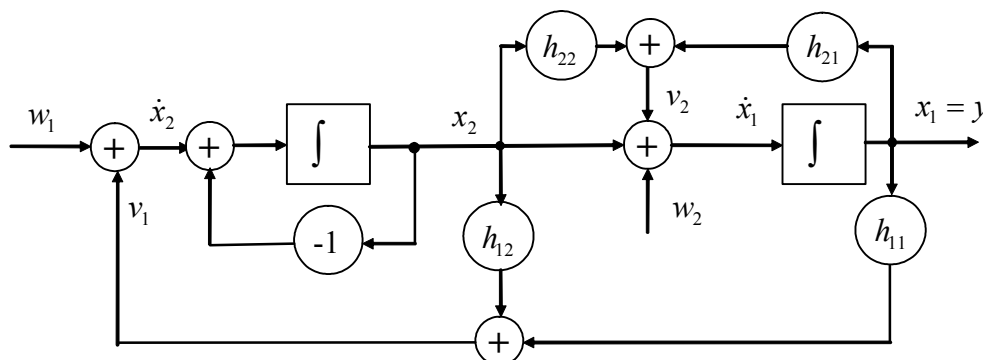
$$F_{su}(s) = \frac{1}{s} \tag{5.28}$$

Situace se zavedením stavových proměnných je znázorněna na následujícím obrázku.



Obr. 5.12: Stavový model podle rovnice (5.27) a (5.28)

Do obrázku doplníme prvky zpětnovazební matice H .



Obr. 5.13: Stavový regulátor pro model podle rovnic (5.27) a (5.28)

Z obrázku je zřejmé, že některé prvky matice stavového regulátoru musí být zvoleny jinak, pokud chceme vyhovět zadání.

Určíme hodnoty matic:

$$A = \begin{vmatrix} 0 & 1 \\ 0 & -1 \end{vmatrix} \quad B = \begin{vmatrix} 0 & 1 \\ 1 & 0 \end{vmatrix} \quad \begin{vmatrix} v_1 \\ v_2 \end{vmatrix} = \begin{vmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{vmatrix} \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} \quad (5.29)$$

$$C = \begin{vmatrix} 1 & 0 \end{vmatrix} \quad D = 0$$

$$\bar{A} = (A + BH) = \begin{vmatrix} h_{21} & 1 + h_{22} \\ h_{11} & -1 + h_{12} \end{vmatrix} \quad (5.30)$$

$$\begin{vmatrix} x_{1t \rightarrow \infty} \\ x_{2t \rightarrow \infty} \end{vmatrix} = -\frac{1}{h_{21}(h_{12} - 1) - h_{11}(1 + h_{22})} \begin{vmatrix} -(1 + h_{22}) & -1 + h_{12} \\ h_{21} & -h_{11} \end{vmatrix} \begin{vmatrix} w_{1t \rightarrow \infty} \\ w_{2t \rightarrow \infty} \end{vmatrix} \quad (5.31)$$

Z rovnice (5.31) vyplývá, že pro splnění podmínek musí platit $h_{12} = 1$ a $h_{11} = -1$.

Další parametry můžeme volit podobně jako v předešlém příkladě. Struktura regulačního obvodu je pak stejná jako na Obr. 5.11.

5.2.5 Definice diskrétního optimálního řízení

Je dán dynamický, plně říditelný systém (5.32)

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t), t) \\ y(t) &= g(x(t), u(t), t) \end{aligned} \quad (5.32)$$

s definovaným počátečním a koncovým stavem a kritériem optimálního řízení (5.33)

$$J = \frac{1}{2} (x^T(N) S x(N)) + \frac{1}{2} \left(\sum_{k=0}^{N-1} (x^T(k) Q x(k) + u^T(k) R u(k)) \right) \quad (5.33)$$

a úkolem optimálního řízení je najít takové řízení $u^*(t)$, které

1. převádí systém (5.32) z počátečního stavu do koncového stavu
2. patří do třídy přípustných řízení
3. minimalizuje dané kritérium (5.33)

5.3 Odvození LQ řízení

5.3.1 Hledání optimálního řešení v předposledním kroku

Pro odvození řídicího zákona potřebujeme spojitý systém rovnic (5.32) převést do diskrétního tvaru (5.34).

$$\begin{aligned} x_N &= Ax_{N-1} + Bu_{N-1} \\ y_N &= Cx_{N-1} \end{aligned} \quad (5.34)$$

Kritérium (5.33) přepíšeme do tvaru optimalizace posledního kroku

$$J_{N-1} = \frac{1}{2} x_N^T S_N x_N + \frac{1}{2} \sum_{k=N-1}^{N-1} (x_k^T Q x_k + u_k^T R u_k) \quad (5.35)$$

V tomto případě potřebujeme pro poslední krok určit ze znalosti stavu x_{N-1} pouze hodnotu akčního zásahu u_{N-1} a výraz $x_{N-1}^T Q x_{N-1}$ nemá na hodnotu kritéria vliv, protože jej nemůžeme akčním zásahem modifikovat.

Záměnou za $x_N = Ax_{N-1} + Bu_{N-1}$ v kritériu (5.35) získáme výraz (5.36)

$$\frac{1}{2} x_N^T S_N x_N = \frac{1}{2} (Ax_{N-1} + Bu_{N-1})^T S_N (Ax_{N-1} + Bu_{N-1}) \quad (5.36)$$

který dosadíme do (5.35)

$$\begin{aligned} J_{N-1} &= \frac{1}{2} x_N^T S_N x_N + \frac{1}{2} (x_{N-1}^T Q x_{N-1} + u_{N-1}^T R u_{N-1}) = \\ &= \frac{1}{2} (Ax_{N-1} + Bu_{N-1})^T S_N (Ax_{N-1} + Bu_{N-1}) + \frac{1}{2} (x_{N-1}^T Q x_{N-1} + u_{N-1}^T R u_{N-1}) = \\ &= \frac{1}{2} (u_{N-1}^T R u_{N-1} + x_{N-1}^T (A^T S_N A + Q) x_{N-1} + 2x_{N-1}^T A^T S_N B u_{N-1} + u_{N-1}^T B^T S_N B u_{N-1}) \end{aligned} \quad (5.37)$$

Výpočteme parciální derivaci položíme ji rovnu 0:

$$\frac{\partial J_{N-1}}{\partial u_{N-1}} = R u_{N-1} + B^T S_N A x_{N-1} + B^T S_N B u_{N-1} = 0 \quad (5.38)$$

Pozn.: pro $J = x^T P x = \sum x_i p_{ij} x_j = p_{11} x_1^2 + p_{22} x_2^2 + p_{33} x_3^2 + \dots + (p_{12} + p_{21}) x_1 x_2 + \dots$ je derivace

$$\frac{\partial x^T P x}{\partial x} = 2p_{ii} x_i + (p_{i2} + p_{2i}) x_1 + (p_{i2} + p_{2i}) x_2 + \dots = \sum_{j=1}^n (p_{ij} + p_{ji}) x_j = (P + P^T) x$$

a dále se definuje (pro zjednodušení zápisu), že platí

$$\frac{\partial J(x_1, x_2, \dots, x_n)}{\partial x} = \begin{pmatrix} \frac{\partial J}{\partial x_1} \\ \frac{\partial J}{\partial x_2} \\ \vdots \\ \frac{\partial J}{\partial x_n} \end{pmatrix}$$

Z rovnice (5.38) určíme velikost akčního zásahu

$$u_{N-1} = -(B^T S_N B + R)^{-1} B^T S_N A x_{N-1} = -K_{N-1} x_{N-1} \quad (5.39)$$

$$K_{N-1} = (B^T S_N B + R)^{-1} B^T S_N A \quad (5.40)$$

Extrém (minimální hodnotu kritéria J v kroku $N-1$) získáme dosazením (5.40) do (5.37)

$$J_{N-1}^{opt} = \frac{1}{2} x_{N-1}^T \left(K_{N-1}^T R K_{N-1} + Q + A^T S_N A - 2A^T S_N B K_{N-1} + K_{N-1}^T B^T S_N B K_{N-1} \right) x_{N-1} \quad (5.41)$$

Dosazením do rovnice (5.41) za K a úpravou získáme hledanou rovnici:

$$J_{N-1}^{opt} = \frac{1}{2} x_{N-1}^T (Q + A^T S_N A - A^T S_N B (B^T S_N B + R)^{-1} B^T S_N A) x_{N-1} \quad (5.42)$$

Pro výpočet dalšího kroku označíme jako S_{N-1} matici

$$S_{N-1} = Q + A^T S_N A - A^T S_N B (B^T S_N B + R)^{-1} B^T S_N A \quad (5.43)$$

Hledaná optimální velikost kritéria je tedy:

$$J_{N-1}^{opt} = \frac{1}{2} x_{N-1}^T S_{N-1} x_{N-1} \quad (5.44)$$

5.3.2 Dynamické programování

Připomeňme princip Bellmanova dynamického programování. Úlohu rozdělíme na N kroků a ze znalosti požadovaného konečného stavu postupně zpětnou minimalizací se dostaneme do počátečního stavu. Při daném kritériu optimalizace hledáme tedy posloupnost vektorů akčních v předchozích krocích minimalizující kritérium.

$$J_{N-2} = \frac{1}{2} \sum_{k=N-2}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2} x_N^T S_N x_N = \frac{1}{2} (x_{N-2}^T Q x_{N-2} + u_{N-2}^T R u_{N-2}) + J_{N-1} \quad (5.45)$$

Podle principu optimality použijeme dříve odvozenou rovnici $u_{N-1} = -K_{N-1} x_{N-1}$ a určíme u_{N-2} minimalizací kritéria:

$$J_{N-2}^* = \frac{1}{2} (x_{N-2}^T Q x_{N-2} + u_{N-2}^T R u_{N-2}) + J_{N-1}^{opt} = \frac{1}{2} (x_{N-2}^T Q x_{N-2} + u_{N-2}^T R u_{N-2}) + \frac{1}{2} x_{N-1}^T S_{N-1} x_{N-1} \quad (5.46)$$

$$\begin{aligned} u_{N-2} &= -(B^T S_{N-1} B + R)^{-1} B^T S_{N-1} A x_{N-2} = -K_{N-2} x_{N-2} \\ S_{N-2} &= Q + A^T S_{N-1} A - A^T S_{N-1} B (B^T S_{N-1} B + R)^{-1} B^T S_{N-1} A \end{aligned} \quad (5.47)$$

$$J_{N-2}^{opt} = \frac{1}{2} x_{N-2}^T S_{N-2} x_{N-2}$$

$$u_l = -(B^T S_{l+1} B + R)^{-1} B^T S_{l+1} A x_l = -K_l x_l \quad (5.48)$$

$$S_l = Q + A^T S_{l+1} A - A^T S_{l+1} B (B^T S_{l+1} B + R)^{-1} B^T S_{l+1} A \quad (5.49)$$

Je zřejmé, že diskrétní kroky počítáme pro $l = N-1, N-2, \dots, 3, 2, 1, 0$.
Odvozený zpětnovazební regulátor je lineární časově proměnný.

5.3.3 Stacionární regulátor

Implementace časově proměnného regulátoru je relativně náročná, problémy vznikající při jeho realizaci můžeme řešit zavedením nekonečného horizontu kritéria optimality.

$$J = \frac{1}{2} \sum_{k=0}^{\infty} (x_k^T Q x_k + u_k^T R u_k) \quad (5.50)$$

Volbou $N \rightarrow \infty$ můžeme zajistit dosažení konvergence pro start z libovolné pozitivně definitní S_N . Rovnice (5.49) přechází do diskrétní algebraické Riccatiho rovnice s neznámými elementy matice S . Systém má $n \times n = n^2$ nelineárních rovnic s n^2 neznámými. Řešení není sice jediné, ale lze ukázat, že při splnění požadovaných podmínek má jediné pozitivně definitní řešení.

$$S = Q + A^T S A - A^T S B (B^T S B + R)^{-1} B^T S A \quad (5.51)$$

Zavedme substituci pro konvergující $S \rightarrow P \rightarrow P(k) = P(k+1)$. Rovnice (5.51) pak je dána

$$P = Q + A^T P A - A^T P B (R + B^T P B)^{-1} B^T P A \quad (5.52)$$

Jedná se o nelineární maticovou rovnici s proměnnou P . Zpětnovazební matice K je pak na celém horizontu optimalizace konstantní.

Řešení odpovídající rovnici (5.52) můžeme najít jednodušším způsobem.

5.3.4 Diskrétní LQ regulátor s nekonečným horizontem

Je dán autonomní systém

$$x_{k+1} = A x_k, \quad x_0 \quad (5.53)$$

Kvadratické kritérium pro autonomní systém

$$\begin{aligned} J &= \sum_{k=0}^{\infty} x_k^T Q x_k = x_0^T Q x_0 + x_1^T Q x_1 + x_2^T Q x_2 + \dots + x_k^T Q x_k + \dots = \\ &= x_0^T Q x_0 + x_0^T A^T Q A x_0 + \dots x_0^T (A^T)^k Q A^k x_0 + \dots = \\ &= x_0^T (Q + A^T Q A + \dots + (A^T)^k Q A^k + \dots) x_0 \end{aligned} \quad (5.54)$$

$$\begin{aligned} P_k &= A^T P_{k-1} A + Q \\ P_{k+1} &= P_k + (A^T)^{k+1} Q A^{k+1} \end{aligned} \quad (5.55)$$

Limitním přechodem pro $k \rightarrow \infty$

$$\begin{aligned} P &= A^T P A + Q \\ J &= x_0^T P x_0 \end{aligned} \quad (5.56)$$

Obecné kvadratické kritérium:

$$\begin{aligned} x_{k+1} &= A x_k + B u_k, \dots x_0, \dots u_k = -K x_k, \dots A - BK \dots \text{je stabilní matice} \\ x_{k+1} &= (A - BK) x_k, \dots x_0 \end{aligned} \quad (5.57)$$

$$\begin{aligned}
 J &= \sum_{k=0}^{\infty} (x_k^T Q x_k + u_k^T R u_k) = \sum_{k=0}^{\infty} (x_k^T Q x_k + x_k^T K^T R K u_k) = \\
 &= \sum_{k=0}^{\infty} x_k^T (Q + K^T R K) x_k, \quad Q \geq 0, \quad R > 0
 \end{aligned}
 \tag{5.58}$$

Podle (5.56) pak můžeme psát

$J = x_0^T P x_0$ kde P je pozitivně definitní řešení maticové rovnice

$$P = (A - BK)^T P (A - BK) + Q + K^T R K \tag{5.59}$$

Stanovení matice P vede na řešení pomocí iteračního Kleinmannova algoritmu:

1. zvolíme matici $K^{(0)}$ tak, aby byla matice $A - BK^{(0)}$ stabilní
2. pro $i=0, 1, \dots$ určíme matici $P^{(i)}$, která je řešením rovnice

$$(A - BK^{(i)})^T P^{(i)} (A - BK^{(i)}) - P^{(i)} = -K^{(i)T} R K^{(i)} - Q$$
3. určíme zesílení

$$K^{(i+1)} = (R + B^T P^{(i)} B)^{-1} B^T P^{(i)} A$$
4. pokračujeme bodem 2

Maticovou rovnici v 2. kroku Kleinmannova algoritmu upravíme vynásobením $(A - BK^{(i)})^{-1}$ zprava. Obdržíme:

$$(A - BK^{(i)})^T P^{(i)} - P^{(i)} (A - BK^{(i)})^{-1} = (-K^{(i)T} R K^{(i)} - Q) (A - BK^{(i)})^{-1}$$

Provedeme-li substituci:

$$A = (A - BK^{(i)})^T, \quad \Pi = -(A - BK^{(i)})^{-1}, \quad \Gamma = (-K^{(i)T} R K^{(i)} - Q) (A - BK^{(i)})^{-1}$$

získáme maticovou rovnici „ $AX + XB = C$ “:

$$AP + P\Pi = \Gamma$$

jejíž řešení vede na soustavu lineárních rovnic:

$$\begin{aligned}
 \lambda_{kk} p_{kl} + p_{kl} \pi_{ll} &= \gamma_{kl} - \sum_{j=1}^{k-1} \lambda_{kj} p_{jl} - \sum_{i=1}^{l-1} p_{ki} \pi_{il} \\
 (k &= 1, 2, \dots, p; l = 1, 2, \dots, q) \\
 \lambda_{kl}, p_{kl} \text{ a } \pi_{kl} &\text{ jsou prvky matic } A, P \text{ a } \Pi.
 \end{aligned}$$

Pro systém 2. řádu bude mít soustava čtyři rovnice:

$$\begin{bmatrix}
 \lambda_{11} + \pi_{11} & \lambda_{12} & \pi_{21} & 0 \\
 \lambda_{21} & \lambda_{22} + \pi_{11} & 0 & \pi_{21} \\
 \pi_{12} & 0 & \lambda_{11} + \pi_{22} & \lambda_{12} \\
 0 & \pi_{12} & \lambda_{21} & \lambda_{22} + \pi_{22}
 \end{bmatrix}
 \begin{bmatrix}
 p_{11} \\
 p_{21} \\
 p_{12} \\
 p_{22}
 \end{bmatrix}
 =
 \begin{bmatrix}
 \gamma_{11} \\
 \gamma_{21} \\
 \gamma_{12} \\
 \gamma_{22}
 \end{bmatrix}$$

6 Fuzzy regulátory

V posledních letech se setkáváme v řízení regulaci s principy, které jsou založeny na poměrně nové vědní disciplíně, která bývá označována jako **Soft Computing** (SC). Zabývá se širokým spektrem různých výpočetních postupů, jejichž společným jmenovatelem je odklon od klasického modelování založeném na analytických modelech, booleovské logice, ostré klasifikaci a deterministickém prohledávání.

V názvu uvedené „soft“ vyjadřující lexikálně „měkkost, mírnost“ zde znamená měkké požadavky na přesnost popisovaných jevů. Mezi hlavní směry SC patří fuzzy logika, umělé neuronové sítě a genetické algoritmy.

Fuzzy logika spočívá v rozšíření logických operátorů na fuzzy množiny. Teorie fuzzy množin spočívá v zavedení tzv. stupně příslušnosti prvku k množině, který může nabývat hodnot z intervalu $\langle 0, 1 \rangle$, na rozdíl od klasické teorie množin, kdy každý prvek do množiny buď patří, nebo nepatří. Fuzzy logika nám poskytuje jazyk s vlastní syntaxí a sémantikou, který nám umožňuje bezprostřední použití kvalitativně formulovaných znalostí a zkušeností o řešeném problému.

Umělé neuronové sítě jsou velmi zjednodušené matematické modely nervových systémů živých organismů. Jeden ze směrů jejich výzkumu v této oblasti se snaží pochopit a modelovat, jakým způsobem myslíme a jak funguje náš mozek. Jejich další z řady využití je řešení úloh z umělé inteligence.

Genetické algoritmy provádějí náhodné prohledávání s pomocí imitace živé přírody. Prohledávání se uskutečňuje s modelovou evolucí od vzniku jedinců, přes jejich selekci a křížení s následným výběrem nejlépe vyhovujících jedinců.

6.1 Fuzzy množiny a lingvistické proměnné.

V klasické teorii množin se popisuje množina několika způsoby:

- výčtem prvků množiny $A = \{x_1, x_2, x_3, x_4\}$
- pravidlem, kterému musí vyhovovat
- charakteristickou funkcí $\mu_A(x)$

Příkladem charakteristické funkce může být např. funkce množiny Záporná teplota. Prvek x_1 v klasické teorii množin do ní buď patří, nebo nepatří, protože jeho charakteristická funkce nabývá hodnot buď 1 nebo 0. Mohli bychom hovořit o ostré množině a ostrém rozlišení při rozhodování o příslušnosti.

Pokud doplníme charakteristickou funkci hodnotou z intervalu $\langle 0, 1 \rangle$, s jakou prvek do množiny patří, pak tyto množiny označujeme jako neostré, fuzzy množiny.

Pro využití a popis empirických zkušeností, vlastností atd. zavádíme pojem lingvistická proměnná. **Lingvistická proměnná** je taková proměnná, jejíž hodnoty jsou výrazy nějakého jazyka. Hodnoty lingvistických proměnných můžeme interpretovat jako fuzzy množiny. Množina lingvistických hodnot se označuje jako množina **termů**. Termy jsou definovány na **universu**, které chápeme jako universální množinu. Například teplotu vody můžeme označit jako *STUDENÁ*, *VLAŽNÁ*, *TEPLÁ*, *HORKÁ*. Takto zavedená kvantifikace teplot představuje

termy, pro které je možné definovat charakteristické funkce $\mu(x)$, které se u fuzzy množin nazývají funkce příslušnosti. Např. pro teplotu $VLAŽNÁ(V)$ můžeme definovat fuzzy množinu s charakteristickou funkcí příslušnosti $\mu_V(x)$.

6.2 Fuzzy regulátor, principy inference, fuzzifikace a defuzzifikace

Nastavování fuzzy regulátorů je mnohem komplikovanější než nastavování klasických regulátorů, protože fuzzy regulátor, jako nelineární regulátor, má (zdánlivě) mnohem více stupňů volnosti, jejichž využití je však často zavádějící. Kvalitní seřízení fuzzy regulátoru vyžaduje mnohem více času než nastavení klasického regulátoru, protože dosud neexistuje metoda pro jeho nastavení (pokud pomineme možnost nastavení fuzzy regulátoru genetickými algoritmy). Navíc se v literatuře vyskytují takové struktury fuzzy regulátorů, jejichž složitost přesahuje možnosti lidského logického nastavení.

V literatuře můžeme najít mnoho nekorektních porovnání fuzzy regulátorů a PID regulátorů (za všechny případy alespoň [18] a jen někdy objektivní zhodnocení [19]). Rovněž se vyskytuje řada zapojení, jejichž složitá struktura nepřináší žádný užitek. Z poslední doby by bylo možné poukázat na [15], ve kterém autoři srovnávají fuzzy PD+I regulátor s PID regulátorem při řízení ramene robota. Srovnání je naprosto nekorektní, jsou sice uvedeny parametry nastavení diskretního PID regulátoru, ale není uvedena jeho realizace. Srovnání vychází pro PID regulátor velmi nepříznivě, protože výstup z procesu s PID regulátorem silně kmitá s amplitudou asi 0,5 V. Fuzzy PID regulátor naproti tomu vykazuje cca 10 krát menší úroveň kmitání. Pokud by byl použit diskretní PSD regulátor podle rovnice bez filtrace derivační složky, je zřejmé, že při nastavené periodě vzorkování $T = 0,001$ s, proporčním zesílením $K = 10$, derivační časové konstantě $T_D = 0,012$ s a úrovni rušivých impulsů, která se dá odhadnout na 0,1 V bude na každý rušivý impuls jen samotná derivační složka diskretního PID regulátoru reagovat změnou akčního zásahu o 12 V. To musí celý systém pochopitelně silně rozkmitat. Přitom by stačila jednoduchá filtrace derivační složky diskretního PID regulátoru a potom by srovnání vyznělo v neprospěch fuzzy PD+I regulátoru, který je zde navrhnout zbytečně složitě a těžkopádně. Faktorem ale je, že fuzzy logika vnáší nové, zajímavé pohledy na realitu z jiného úhlu, než je exaktní přístup při řešení problému.

Základní struktura fuzzy regulátoru v uzavřené smyčce je uvedena na **Obr. 6.1**. Proces je s regulátorem spojen přes A/D a D/A převodníky. Pokud jsou použity jako vstupní proměnné v regulačním obvodu regulační odchylka e v kroku k $e(k)$

$$e(k) = w(k) - y(k) \quad (6.1)$$

a její první diference v čase

$$\Delta e(k) = (e(k) - e(k-1))/T \quad (6.2)$$

kde $e(k-1)$ je regulační odchylka v předchozím kroku k periody vzorkování T . Další použité označení v obrázku:

$w(k)$ je žádaná hodnota výstupní veličiny v kroku k

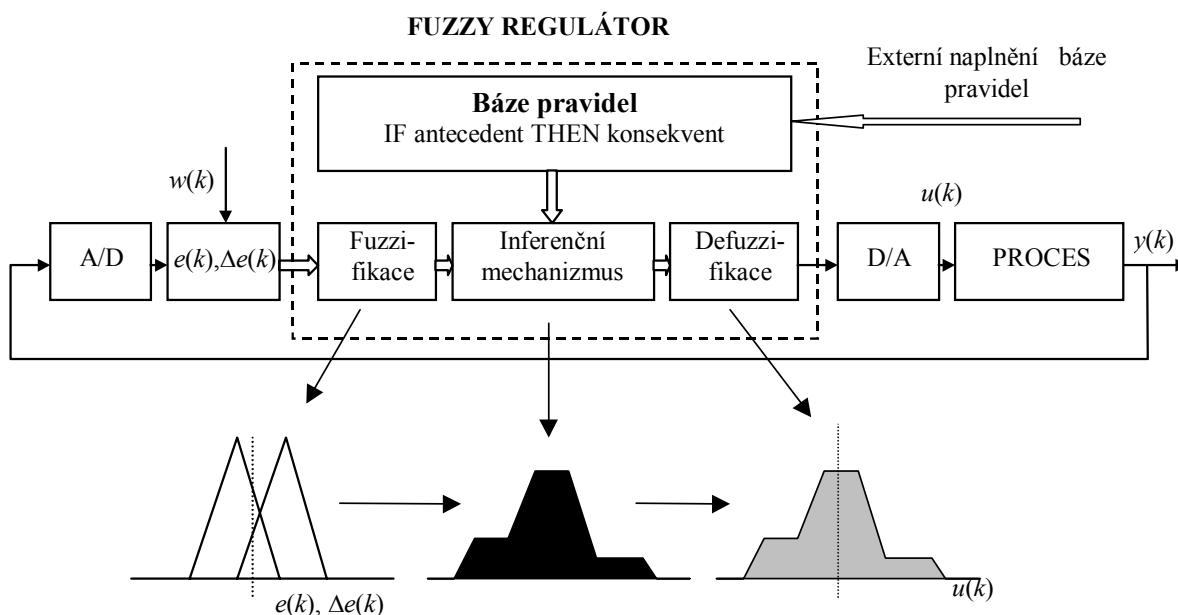
$u(k)$ je akční zásah v kroku k

$y(k)$ je hodnota výstupní veličiny z procesu v kroku k ;

pak jde o fuzzy podobu klasického PI nebo PD regulátoru. V literatuře se pro diferenci regulační odchylky používá vztah

$$\Delta_1 e(k) = e(k) - e(k-1) \quad (6.3)$$

Použití vztahu (6.2) je výhodnější, protože nastavení proměnné je nezávislé na periodě vzorkování. Při změně periody vzorkování je u (6.3) nutné upravit rozsah universa pro diferenci odchylky.



Obr. 6.1: Fuzzy regulátor ve zpětnovazebním zapojení.

Obecná forma báze pravidel je dána

$$\text{IF antecedent THEN konsekvent ELSE next_rule} \quad (6.4)$$

Antecedent je zpravidla složená logická podmínka, ve které jsou pravidla vázány logickými spojkami, ze kterých je potom vyvozována druhá logická podmínka jako konsekvent pravidla:

$$\text{IF } x_1 \text{ is } A_{1j} \text{ AND } x_2 \text{ is } A_{2j} \text{ AND } \dots \text{ AND } x_N \text{ is } A_{Nj} \text{ THEN } u \text{ is } B_j \text{ ELSE next_rule} \quad (6.5)$$

kde $j = 1, 2, \dots, M$. Zde k -té pravidlo antecedentu sestává z N částečných podmínek. Vstupní proměnné jsou x_i ; A_{ij} , B_j jsou fuzzy množiny, u je výstupní proměnná.

Konkrétní podoba pravidel pro regulátor může být rozepsána podle (6.5) nebo uložena v tabulce. Protože budeme ve fuzzy regulátorech používat fyzikální veličiny $e(k)$ a $\Delta e(k)$ může být **Tabulka 6.1** rozepsána ve tvaru (P je pozitivní, N je negativní, S je malý, M je střední, B je velký, ZO je nula)

$$\begin{aligned} &\text{IF } e(k) \text{ is } ZO \text{ AND } \Delta e(k) \text{ is } PS \text{ THEN } u(k) \text{ is } PS \\ &\text{IF } e(k) \text{ is } PS \text{ AND } \Delta e(k) \text{ is } PS \text{ THEN } u(k) \text{ is } PM \\ &\text{IF } e(k) \text{ is } PS \text{ AND } \Delta e(k) \text{ is } ZO \text{ THEN } u(k) \text{ is } PS \\ &\text{IF } e(k) \text{ is } ZO \text{ AND } \Delta e(k) \text{ is } ZO \text{ THEN } u(k) \text{ is } ZO \quad \text{atd.} \end{aligned} \quad (6.6)$$

Část pravidel (6.6) je označena kolečkem v **Tabulka 6.1**. Při 7 pravidlech pro každou vstupní veličinu dostáváme celkem 49 pravidel. Jednu z mnoha možných modifikací pravidel uvádí **Tabulka 6.1 b)**. V tomto případě dochází k pozvolnější změně akčního zásahu. Fuzzy regulátor v jednoduché smyčce podle **Obr. 6.1** má jeden nebo několik vstupů a jeden výstup.

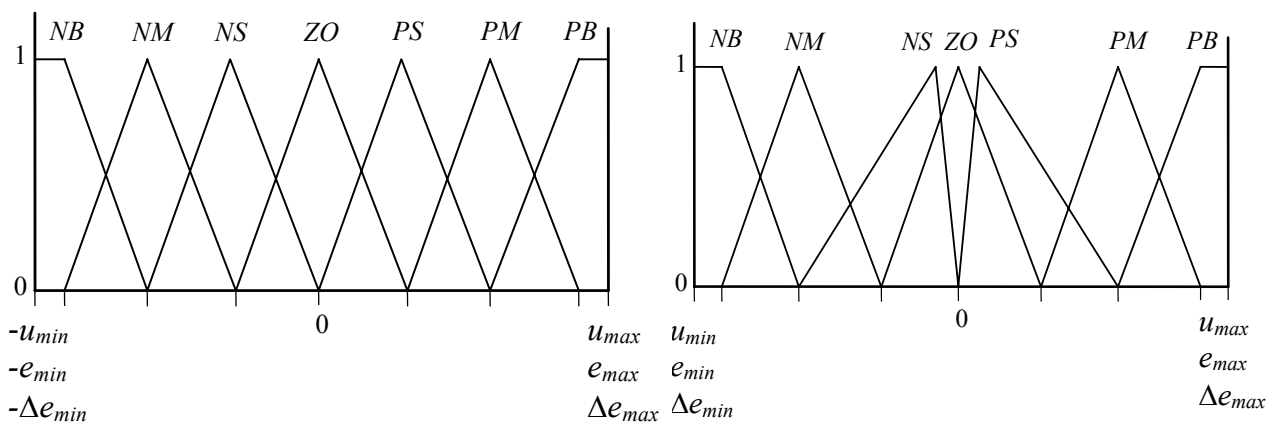
V praxi je nejčastěji používán se dvěma vstupy. Při více vstupech extrémně narůstá velikost báze pravidel. Rovněž je otázkou, jak tak rozměrnou bázi pravidel smysluplně naplnit. Přechod z dvoudimenzionální báze pravidel při 7 funkcích příslušnosti na třídimenzionální bázi pravidel má za následek vzrůst počtu pravidel z 49 na 343. V žádném případě však nelze říci, že by takový regulátor (je míněn fuzzy PID regulátor) měl lepší vlastnosti než při použití dvou fuzzy PI a PD regulátorů, každý s dvoudimenzionální bázi pravidel (celkem 98 pravidel). Navíc nastavení třídimenzionální báze pravidel bude velmi komplikované pro složitost fyzikální představy při rozložení pravidel.

Tabulka 6.1: a) Dvoudimenzionální báze pravidel b) jedna z řady modifikací

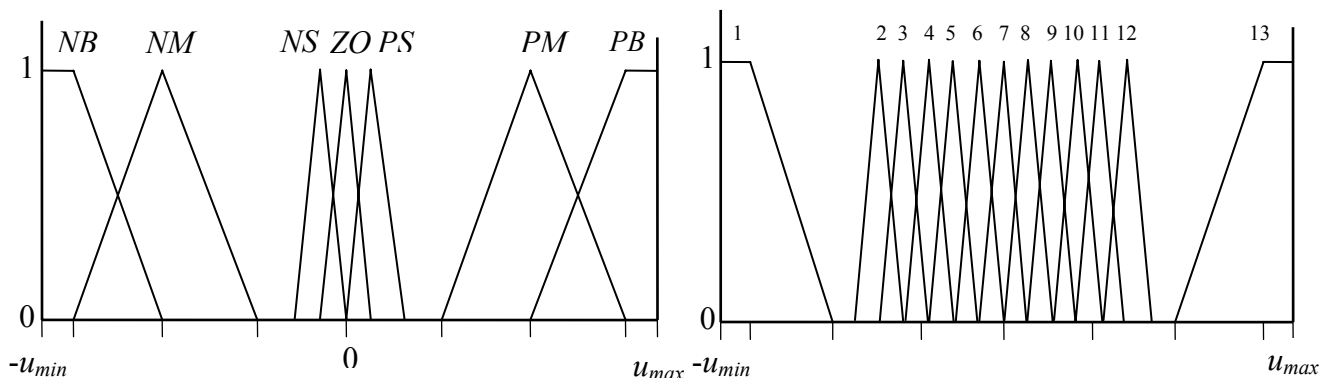
$\Delta e/e$	NB	NM	NS	ZO	PS	PM	PB
PB	ZO	PS	PM	PB	PB	PB	PB
PM	NS	ZO	PS	PM	PB	PB	PB
PS	NM	NS	ZO	PS	PM	PB	PB
ZO	NB	NM	NS	ZO	PS	PM	PB
NS	NB	NB	NM	NS	ZO	PS	PM
NM	NB	NB	NB	NM	NS	ZO	PS
NB	NB	NB	NB	NB	NM	NS	ZO

$\Delta e/e$	NB	NM	NS	ZO	PS	PM	PB
PB	ZO	PS	PS	PM	PM	PB	PB
PM	NS	ZO	PS	PS	PM	PM	PB
PS	NS	NS	ZO	PS	PS	PM	PM
ZO	NM	NS	NS	ZO	PS	PS	PM
NS	NM	NM	NS	NS	ZO	PS	PS
NM	NB	NM	NM	NS	NS	ZO	PS
NB	NB	NB	NM	NM	NS	NS	ZO

Další otázkou, kterou je možné si položit je, jak volit tvar funkcí příslušnosti. Funkce příslušnosti je jednoznačná, může být označena identifikátorem a její hodnota je vždy z intervalu $\langle 0, 1 \rangle$. Tvar a pozice funkcí příslušnosti, které návrhář definuje v pravidlech, samozřejmě rovněž ovlivňuje proces inference. Obvykle jsou používány trojúhelníkové (Δ -funkce) nebo lichoběžníkové tvary funkcí příslušnosti (Π -funkce), které nejsou tolik náročné na výpočet jako spojité funkce příslušnosti (S-funkce) a dovolují efektivněji využít paměť. Experimenty ukazují, že daleko větší vliv než tvar funkce příslušnosti má jejich rozmístění, rozsah universa na ose x a tvar báze pravidel. Obecně rozložení funkcí příslušnosti může být lineární (a symetrické) ve vstupních i výstupních proměnných **Obr. 6.2 a**), nelineární (a nesymetrické) v jednom či více vstupních a výstupních proměnných **Obr. 6.2b**). U výstupní proměnné můžeme nelineární rozložení funkcí příslušnosti využívat pro nelineární změny akčního zásahu **Obr. 6.3 a**). Možná další modifikace je na **Obr. 6.3 b**). Akční zásah je rovnoměrněji rozložen. Výsledky simulací jednoznačně ukazují, že v některých případech stupeň nesymetrie může hrát význačnou roli. V případě spolupráce více regulátorů nemusí hrát již význačnou roli.



Obr. 6.2: a) Symetrické b) nesymetrické rozložení funkcí příslušností



Obr. 6.3: Nelineární rozložení funkcí příslušností pro akční zásah

Výsledky rovněž ovlivňuje použitá metoda defuzzifikace. V dále uváděných experimentech byla používána defuzzifikační metoda využívající polohy těžiště plochy po operaci agregace (center of gravity - COG). Zajímavá úprava tabulky je uvedena v **Tabulka 6.1** (ve spojení s **Obr. 6.3**). V tomto případě dochází ke zvětšení počtu funkcí příslušnosti pro akční zásah a výsledkem je rovnoměrnější změna akčního zásahu. V literatuře můžeme najít celou řadu dalších variant.

Tabulka 6.2: Dvoudimenzionální báze pravidel pro jemnější rozlišení

$\Delta e/e$	NB	NM	NS	ZO	PS	PM	PB
PB	7	8	9	10	11	12	13
PM	6	7	8	9	10	11	12
PS	5	6	7	8	9	10	11
ZO	4	5	6	7	8	9	10
NS	3	4	5	6	7	8	9
NM	2	3	4	5	6	7	8
NB	1	2	3	4	5	6	7

Rozložení funkcí příslušností podle **Obr. 6.3** můžeme použít pouze u akčního zásahu. V případě, že bude použito pro vstupní proměnné, nedojde k průniku funkce příslušnosti se vstupní proměnnou. Akční zásah u fuzzy PD regulátoru pak bude nulový a u fuzzy PI regulátoru se přírůstek nezmění. Pro počáteční experimenty volíme rozložení podle **Obr. 6.2 a**), potom nastavíme rozsahy univers (e_{max} , e_{min} , atd.) a teprve po hrubém seřízení regulačního obvodu experimentujeme s rozložením funkcí příslušností.

Podle **Obr. 6.1** můžeme definovat pět kroků pro výpočet velikosti akčního zásahu $u(k)$ fuzzy regulátoru

1. fuzzifikaci
2. spojení v pravidlech antecedentu - *AND*
3. spojení antecedentu a konsekventu - *THEN*
4. spojení v konsekventu - *ELSE* (agregace)
5. defuzzifikaci.

Vstupní proměnné se v prvním kroku převedou do fuzzy množin. Fuzzifikovaná hodnota pak vstupuje do inferenčního mechanismu, který pracuje s bází pravidel, někdy se bází pravidel říká znalostní báze fuzzy regulátoru. Těmto krokům se říká fuzzy inference.

Inferenci se rozumí celá metoda usuzování. Výsledkem inferenčního mechanismu (v podstatě jde o vyhodnocení spojek *AND*, *THEN*, *ELSE*) je fuzzy množina, ze které je v závěrečném kroku defuzzifikací určena velikost akčního zásahu. Nejčastěji používaná metoda fuzzy inference je metoda Min-Max. Další, rovněž velmi často používanou metodou je Prod-Max. Obě tyto metody mají největší praktický význam z desítek možných inferenčních metod.

Obecně je fuzzy množina (například A) určena svou charakteristickou funkcí μ_A

$$\mu_A : X \rightarrow [0, 1] \quad (6.7)$$

kde X je univerzální množina. Charakteristická funkce mapuje hodnoty universa X do reálného spojitého intervalu $<0, 1>$. Funkce μ_A proměnné $x \in X$ (někdy se značí též $A(x)$ nebo Ax) se nazývá funkce příslušnosti (funkcí náležení prvku x do fuzzy množiny A). V podstatě každému bodu x je přiřazeno reálné číslo z jednotkového intervalu, které je chápáno jako stupeň náležení prvku x do fuzzy množiny A .

Rozšířením vlastností klasických množinových operátorů na fuzzy množiny je definována řada standardních operací. Pro odvození inferenčních metod Min-Max a Prod-Max potřebujeme operace

$$\begin{aligned} \text{konjunkce} & \quad \forall x \in X, \mu_{A \cap B}(x) = \min [\mu_A(x), \mu_B(x)] \\ \text{disjunkce} & \quad \forall x \in X, \mu_{A \cup B}(x) = \max [\mu_A(x), \mu_B(x)] \\ \text{algebraický součin} & \quad \forall x \in X, \mu_{A \cdot B}(x) = \mu_A(x) \cdot \mu_B(x) \end{aligned} \quad (6.8)$$

Inferenci Min-Max můžeme definovat vztahem

$$\max_j \{ \min_i [\mu_{A_{ij}}(x), \mu_{B_j}(x)] \} \quad (6.9)$$

a inferenci Prod-Max

$$\max_j \{ \prod_i \mu_{A_{ij}}(x) \cdot \mu_{B_j}(x) \} \quad (6.10)$$

Interpretace inference pro rozepsaná pravidla z **Tabulka 6.1** v rovnicích (6.6) jsou pro inferenci Min-Max na **Obr. 6.4** a Prod-Max na **Obr. 6.5**.

6.3 Fuzzy PI, PD, PID regulátory

Pro definování fuzzy PID regulátoru použijeme jeho podobnosti s PSD regulátorem, jehož akční zásah u v kroku k je

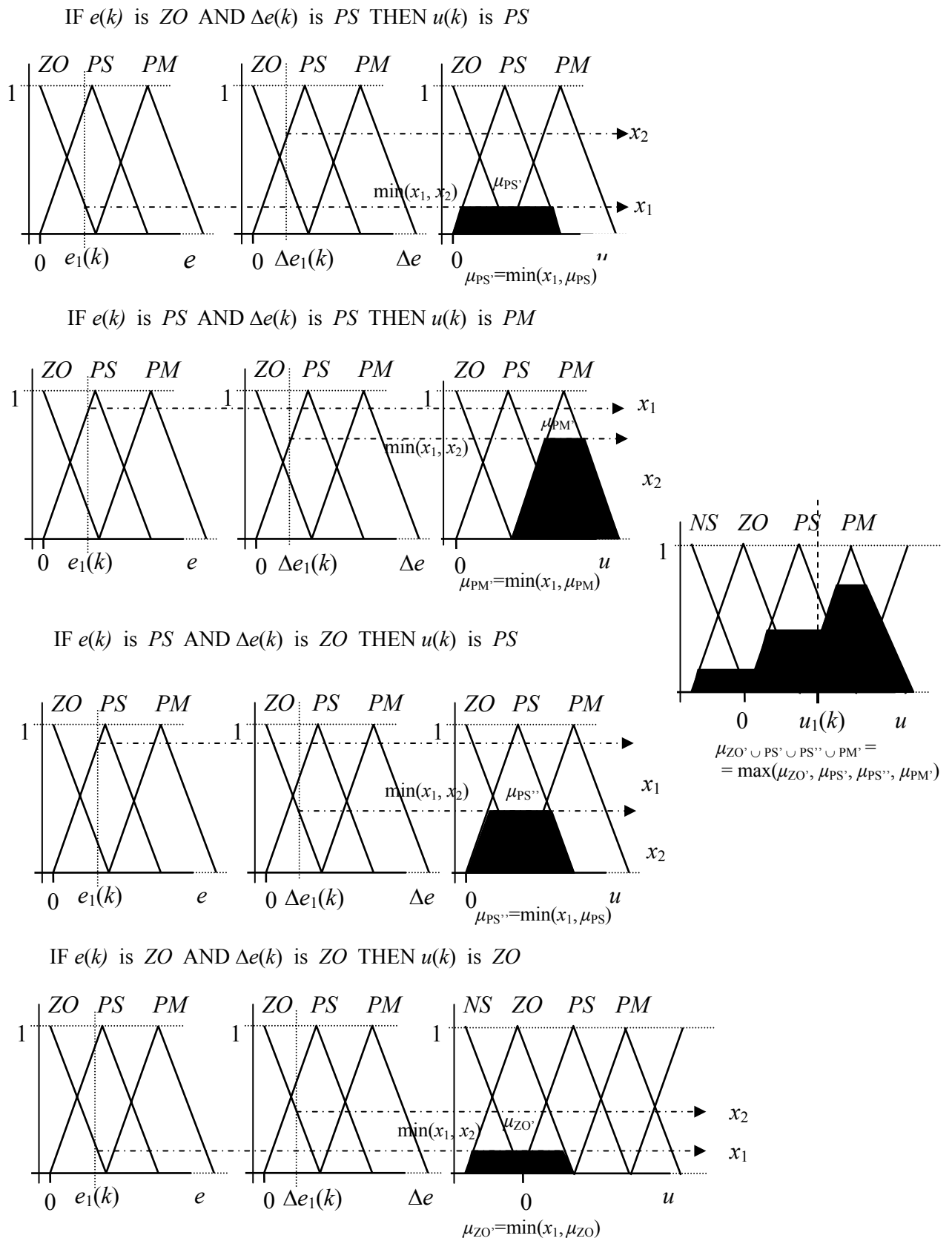
$$u(k) = K \left(e(k) + \frac{T}{T_I} \sum_{i=1}^k e(i) + \frac{T_D}{T} (e(k) - e(k-1)) \right) \quad (6.11)$$

kde K je proporcionalní zesílení, $K_I = \frac{T}{T_I}$ je integrační konstanta, $K_D = \frac{T_D}{T}$ derivační konstanta. Rovnici můžeme přepsat do přírůstkového tvaru

$$\Delta u(k) = u(k) - u(k-1) = K \left(\Delta e(k) + \frac{T}{T_I} e(k) + \frac{T_D}{T} (\Delta e(k) - \Delta e(k-1)) \right) \quad (6.12)$$

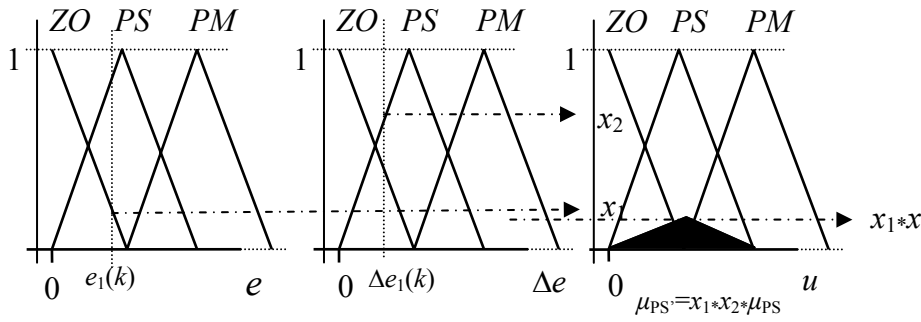
$$u(k) = u(k-1) + \Delta u(k)$$

kde

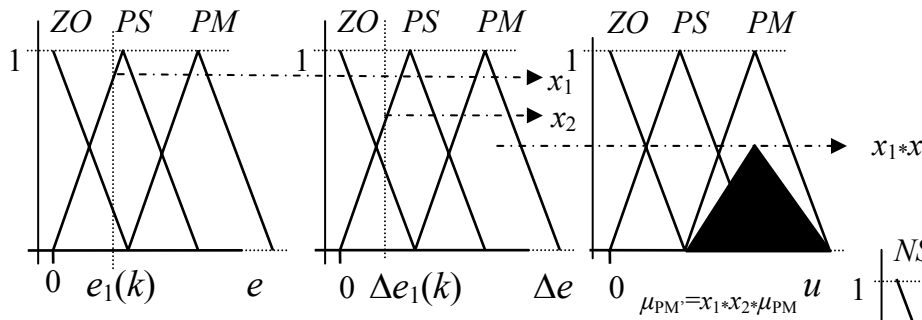


Obr. 6.4: Fuzzy inference metodou Min-Max

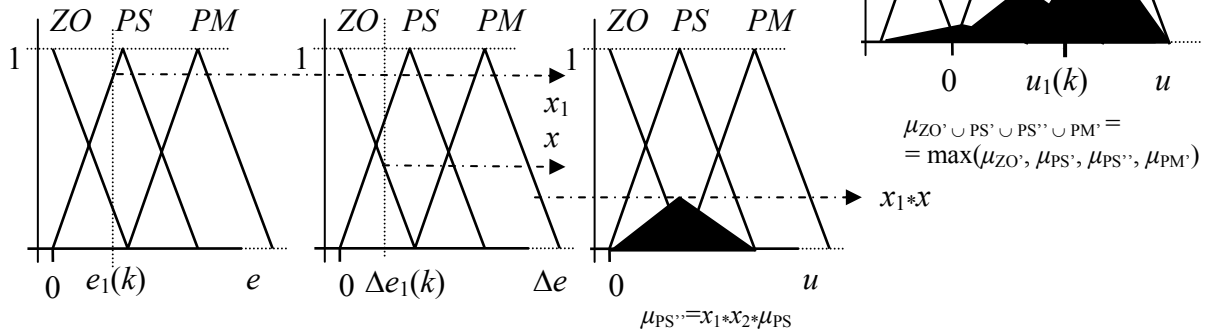
IF $e(k)$ is *ZO* AND $\Delta e(k)$ is *PS* THEN $u(k)$ is *PS*



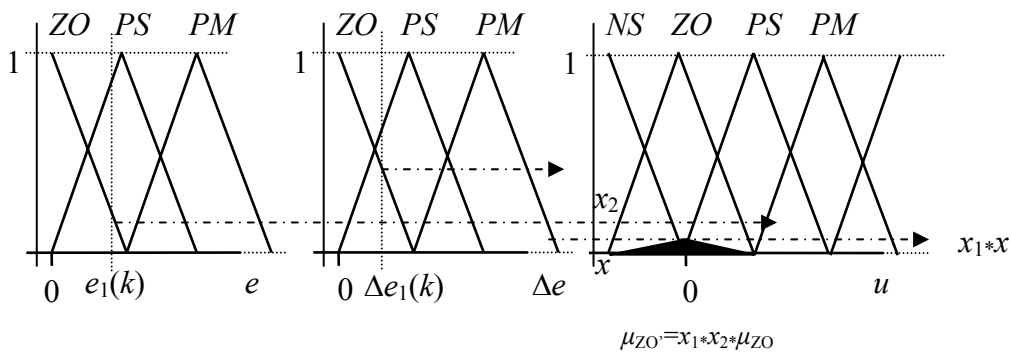
IF $e(k)$ is *PS* AND $\Delta e(k)$ is *PS* THEN $u(k)$ is *PM*



IF $e(k)$ is *PS* AND $\Delta e(k)$ is *ZO* THEN $u(k)$ is *PS*



IF $e(k)$ is *ZO* AND $\Delta e(k)$ is *ZO* THEN $u(k)$ is *ZO*



Obr. 6.5: Fuzzy inference metodou Prod-Max

$e(k)$ je odchylka

$\Delta e(k)$ je první diference odchylky

$\Delta e(k) - \Delta e(k-1)$ je druhá diference odchylky

Připomeňme, že předchozí rovnici můžeme také zapsat v již známých tvarech

$$u(k) = u(k-1) + a_0 e(k) + a_1 e(k-1) + a_2 e(k-2) \quad (6.13)$$

nebo

$$F(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - z^{-1}} = \frac{U(z)}{E(z)} \quad (6.14)$$

V rovnici (6.12) jsou použity tři vstupní proměnné veličiny, odchylka $e(k)$, její první diference $\Delta e(k)$ a druhá diference ($\Delta e(k) - \Delta e(k-1)$). Redukcí výše uvedených rovnic lze též odvodit vztahy pro jednodušší regulátory PI a PD.

6.4 Fuzzy PI regulátor

Vyjdeme-li z rovnic (6.12) a zaměříme se pouze na proporcionální P a integrační část I, dostáváme pro PI regulátor rovnice:

$$\Delta u(k) = K \Delta e(k) + \frac{KT}{T_1} e(k); \quad u(k) = u(k-1) + \Delta u(k) \quad (6.15)$$

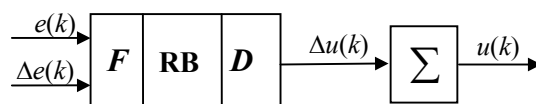
Vstupní proměnné fuzzifikujeme. Po defuzzifikaci dostaneme rovnici

$$\Delta u(k) = \mathbf{D}\{\mathbf{F}\{K \Delta e(k) + \frac{KT}{T_1} e(k)\}\} \quad (6.16)$$

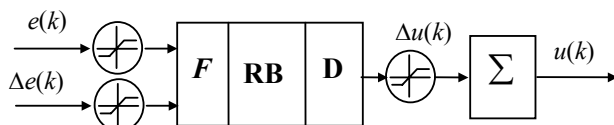
$$u(k) = u(k-1) + \Delta u(k)$$

kde \mathbf{F} je operace fuzzifikace a \mathbf{D} operace defuzzifikace. Rovnice (6.16) určují strukturu fuzzy PI regulátoru **Obr. 6.6**. Konstanty K , $\frac{KT}{T_1}$ odpovídají nastavení rozsahu universa pro

odchylku a její diferenci $e \in \langle -e_{min}, e_{max} \rangle$, $\Delta e \in \langle -\Delta e_{min}, \Delta e_{max} \rangle$, relativní velikost změny akčního zásahu v jednom kroku je určována rozsahem universa pro akční zásah $\Delta u \in \langle -u_{min}, u_{max} \rangle$. Dvoudimenzionální báze pravidel pro tento regulátor může být zapsána např. ve tvaru uvedeném v **Tabulka 6.1**.



Obr. 6.6: Struktura fuzzy PI regulátoru. RB je dvoudim. báze pravidel



Obr. 6.7: Fuzzy PI regulátor s úpravou pro snadnější nastavování

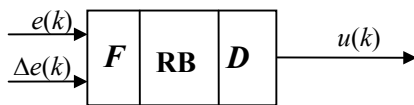
6.5 Fuzzy PD regulátor

Podobně jako u PI regulátoru můžeme strukturu PD regulátoru popsat z rovnic (6.12) vynecháním I části:

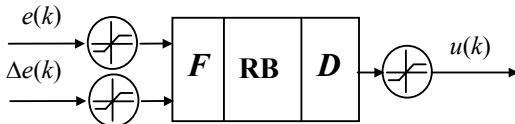
$$u(k) = Ke(k) + \frac{KT_D}{T} \Delta e(k) \quad (6.17)$$

Vstupní proměnné fuzzifikujeme. Po defuzzifikaci dostaneme rovnici

$$u(k) = \mathbf{D} \left\{ \mathbf{F} \left\{ Ke(k) + \frac{KT_D}{T} \Delta e(k) \right\} \right\} \quad (6.18)$$



Obr. 6.8: Struktura fuzzy PD regulátoru. RB je dvoudim. báze pravidel



Obr. 6.9: Fuzzy PD regulátor s úpravou pro snadnější nastavování

Fuzzy PD regulátor má dva vstupy odchylku $e(k)$ a její první diferenci $\Delta e(k)$. Konstanty K , $\frac{KT_D}{T}$ odpovídají nastavení rozsahu universa pro odchylku a její diferenci. Na rozdíl od

fuzzy PI regulátoru nemá sumaci akčního zásahu na výstupu. Zatímco u fuzzy PI regulátoru nastavení báze pravidel nezávisí na typu odezvy procesu, u fuzzy PD regulátoru závisí modifikace báze pravidel podle odezvy procesu. Jestliže má proces integrační charakter odezvy na jednotkový vstupní signál (nebo použijeme fuzzy PD regulátor v kombinaci s fuzzy PI regulátorem), není nutné posunout rozsah universa pro akční zásah $u \in \langle -u_{min}, u_{max} \rangle$. V opačném případě je nutné si uvědomit, že pro soustavy bez integračního charakteru odezvy musí regulátor v ustáleném stavu dávat akční zásah, jehož velikost závisí na zesílení procesu a žádané hodnotě. Proto musíme upravit rozsah universa pro akční zásah $u \in \langle 0, u_{max} \rangle$ s ohledem na konkrétní velikost akčního zásahu pro danou žádanou hodnotu a zesílení procesu (tím se ovšem ztratí původní jazykový význam termů v tabulce) nebo změnit bázi pravidel. Navíc velikost akčního zásahu je nepřímo úměrná zesílení systému, což komplikuje výpočet akčního zásahu pro různé žádané hodnoty. Nastavení rozsahu žádaných hodnot je tím omezeno a fuzzy PD regulátor mívá horší dynamiku a větší chybu v ustáleném stavu, než je jeho ekvivalent PD. Působení poruchových veličin rovněž výrazně ovlivňuje přesnost regulačního obvodu v ustáleném stavu.

V mnoha případech je fuzzy PD regulátor nasazen na procesy s integračním charakterem odezvy na skokový vstupní signál. Jeho výhodou může být rychlejší reakce na změny v regulačním obvodu než u fuzzy PI nebo PID regulátoru. Přítomnost poruch působících na proces však není schopen vyregulovat bez chyby. Ve snaze zmenšit chybu je často velmi zvyšováno zesílení regulátoru, což může mít za následek vznik kmitů a malou robustnost regulačního obvodu.

6.6 Fuzzy PID regulátor

Fuzzy PID regulátor může mít řadu podob. Jako regulátor se třemi vstupy je tvořen třídímní bází pravidel. Jako vstupy jsou obvykle použity proměnné odchylka $e(k)$, první diference $\Delta e(k)$, druhá diference $\Delta^2 e(k)$ viz **Obr. 6.10**, nebo sumace odchylky $\Sigma e(k)$, odchylka $e(k)$ a první diference odchylky $\Delta e(k)$ (**Obr. 6.11**). Obecně lze konstatovat, že mimo podstatného nárůstu rozměru báze pravidel, jak již bylo uvedeno, se tímto řešením ztrácí fyzikální význam pohledu. Nastavování báze pravidel se tak stává daleko obtížnější, stejně tak jako nastavení celého fuzzy regulátoru. Pro odvození struktury PID regulátoru přepíšeme rovnici (6.12) do tvaru

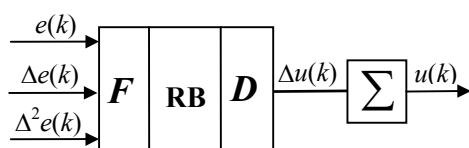
$$\Delta u(k) = K\Delta e(k) + \frac{KT}{T_I} e(k) + \frac{KT_D}{T} (\Delta e(k) - \Delta e(k-1)) \quad (6.19)$$

$$u(k) = u(k-1) + \Delta u(k)$$

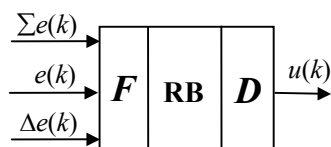
Vstupní proměnné fuzzifikujeme. Po defuzzifikaci a úpravou dostaneme rovnici

$$\Delta u(k) = \mathbf{D} \{ \mathbf{F} \{ \frac{KT}{T_I} e(k) + K\Delta e(k) + \frac{KT_D}{T} \Delta^2 e(k) \} \} \quad (6.20)$$

$$u(k) = u(k-1) + \Delta u(k)$$



Obr. 6.10: Fuzzy PID regulátor daný (6.20). RB je třídím. báze pravidel



Obr. 6.11: Další z možných struktur fuzzy PID regulátoru

Rovnice (6.20) určuje vliv parametrů na rozsah universa u příslušných vstupů fuzzy PID regulátoru. Při seřizování klasických PID regulátorů v reálném procesu se nastavují složky I a D regulátoru zvlášť (přičemž zesílení u proporcionální složky K je vytknuto) a nikoliv v jejich kombinaci, protože každá složka má svůj hluboký fyzikální význam. Proporcionální složka odpovídá přirozené řídicí akci, derivační složka je urychlující a stabilizující, integrační složka potlačuje chybu, ale zároveň zpomaluje systém a zhoršuje stabilitu. Všechna tato tvrzení jsou samozřejmě značně zjednodušená, protože parametry PID regulátoru spolu navzájem spolupracují při tvorbě akčního zásahu. Nejen velký vliv P a I složky může být destabilizující, ale v případě poruch působících v systému může rovněž vliv D složky zhoršovat stabilitu regulačního obvodu.

Obecně lze pomocí fuzzy PID regulátoru realizovat rychlejší přechodové děje než se samotným fuzzy PI regulátorem. Fuzzy regulátor se znalostí výše uvedených faktů je nejlépe

sestavit jako kombinaci dvou fuzzy regulátorů PI a PD nebo fuzzy I a PD. Modifikací rovnic (6.11), (6.12) a (6.13) dostáváme:

$$u(k) = u(k-1) + \Delta u_{PI}(k) + u_{PD}(k) \quad (6.21)$$

pro PD+PI regulátor nebo

$$u(k) = u(k-1) + \Delta u_I(k) + u_{PD}(k) \quad (6.22)$$

pro PD+I regulátor, kde

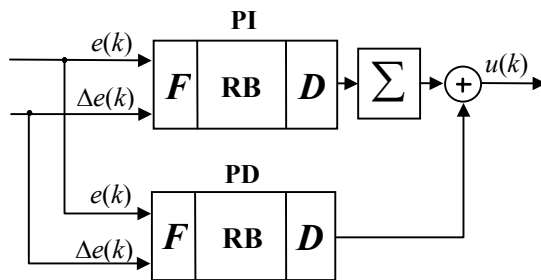
$$\Delta u_I(k) = \frac{KT}{T_I} e(k) \quad \text{I regulátor}$$

$$\Delta u_{PI}(k) = K\Delta e(k) + \frac{KT}{T_I} e(k) \quad \text{PI regulátor}$$

$$u_{PD}(k) = Ke(k) + \frac{KT_D}{T} \Delta e(k) \quad \text{PD regulátor}$$

Vstupní proměnné fuzzifikujeme. Po defuzzifikaci dostaneme rovnici (6.23) pro akční zásah fuzzy PI+PD regulátoru

$$u(k) = u_{PI}(k-1) + \mathbf{D}\{\mathbf{F}\{K\Delta e(k) + \frac{KT}{T_I} e(k)\}\} + \mathbf{D}\{\mathbf{F}\{Ke(k) + \frac{KT_D}{T} \Delta e(k)\}\} \quad (6.23)$$



Obr. 6.12: Struktura fuzzy PI+PD regulátoru

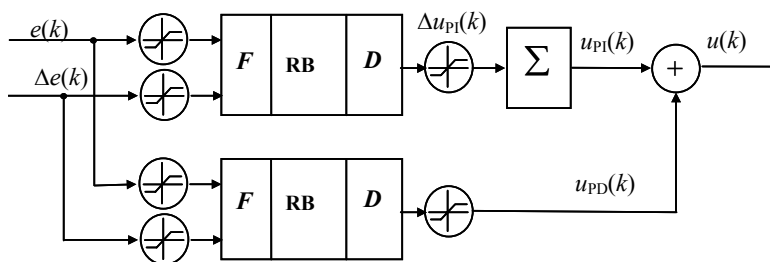
Výsledný akční zásah fuzzy PD+I regulátoru je dán rovnicí (6.24). Báze pravidel pro fuzzy I regulátor může být dána **Tabulka 6.3**.

$$u(k) = u_I(k-1) + \mathbf{D}\{\mathbf{F}\{\frac{KT}{T_I} e(k)\}\} + \mathbf{D}\{\mathbf{F}\{Ke(k) + \frac{KT_D}{T} \Delta e(k)\}\} \quad (6.24)$$

Tabulka 6.3: Jednodimenzionální báze pravidel

e	NB	NM	NS	ZO	PS	PM	PB
-----	----	----	----	----	----	----	----

Jeden z vhodných postupů pro nastavení fuzzy PI+PD regulátoru je optimálně nastavit fuzzy PD regulátor a potom postupně zvětšovat vliv PI, případně I složky. Je-li přechodný děj stále příliš kmitavý, zkusíme zvětšit rozestup mezi funkcemi příslušnosti v universu pro akční zásah u , zejména u PD regulátoru **Obr. 6.3**. Rozsah universa pro akční zásah u fuzzy PD regulátoru v tomto případě není posunut, jako bylo při použití samotného fuzzy PD regulátoru. Kombinace fuzzy regulátoru PI+PD se snadněji nastavuje a má lepší výsledky než fuzzy PD+I regulátor. Pochopitelně můžeme použít u obou fuzzy regulátorů úpravu se změnami rozsahu a zesílení univerza podle **Obr. 6.13**.



Obr. 6.13: Struktura fuzzy PI+PD regulátoru pro snadnější nastav.

Použití fuzzy PID regulátoru nejenom zlepšuje průběhy přechodových charakteristik, ale je i nezbytné v případě integračního charakteru soustavy a požadování vyregulování poruchy bez chyby. Samotný fuzzy PI regulátor zde výrazně zpomalí přechodový děj a může dojít i k nestabilitě systému, případně k stálým oscilacím s malou amplitudou. Aplikace derivačního charakteru regulátoru zde pomůže zrychlit přechodný děj, zajistit stabilitu a eliminovat oscilace. Pokud se seznámíme s vlivem jednotlivých komponent fuzzy regulátoru na průběh přechodové charakteristiky, můžeme realizovat fuzzy regulátor s poněkud výhodnějšími vlastnostmi než u klasického PID regulátoru. Společnou nevýhodou všech fuzzy regulátorů uvedených v této části je jejich nastavování. Hlavní hrubé nastavení se dělá změnou rozsahu universa pro regulační odchylku, její diferenci a pro akční zásah, další doladění pak pomocí úprav v tabulce, případně posunutím funkcí příslušností. Nastavování je převážně intuitivní, obecně použitelná metoda není známa. Výše uvedené kroky je nutné opakovat. Některé z dosud publikovaných fuzzy regulátorů s adaptací se hodí jen na velmi omezenou třídu soustav. Navíc jejich chování v průběhu seřizování dále omezuje jejich možné nasazení na reálném procesu.

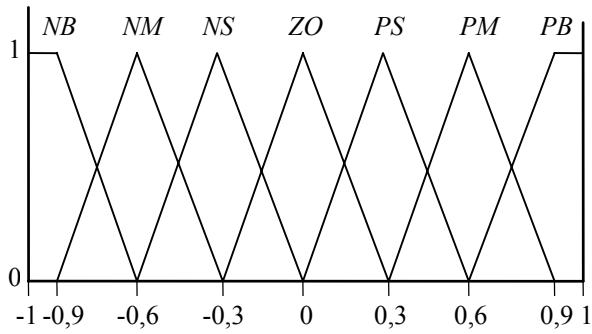
Z výše uvedeného je zřejmé, že nastavování fuzzy regulátoru je časově velmi náročný proces. Při změně periody vzorkování je pak nutné celý regulátor znovu seřídít, protože rozsah universa pro akční zásah je závislý na velikosti periody vzorkování.

6.7 Fuzzy PI/PD/PID regulátory s normalizovaným tvarem universa

Pro zjednodušení návrhu jsou rozsahy universa pro vstupní a výstupní proměnné normalizovány v intervalu $\langle -1, 1 \rangle$ (**Obr. 6.14**). Stejně jsou pak normalizovány rozsahy universa u **Obr. 6.2** a **Obr. 6.3**. Vstupující nebo vystupující proměnná veličina je pak vynásobena konstantou, která vyjadřuje skutečný rozsah universa. Vynásobíme-li hodnotu regulační odchylky e koeficientem 5 před fuzzifikací, je pak skutečný rozsah universa pro odchylku $e \in \langle -0,2 ; 0,2 \rangle$. Při vynásobení konstantou 0,1 je rozsah universa $e \in \langle -10 ; 10 \rangle$. Je zřejmé, že to vůbec není na úkor obecnosti a jak bude ukázáno dále, vede tento postup k značnému zjednodušení návrhu fuzzy regulátoru.

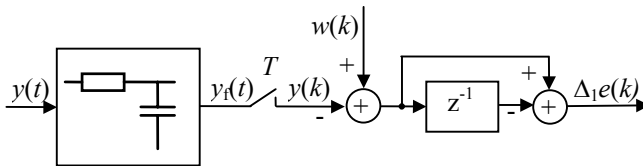
6.7.1 Vliv periody vzorkování

Podobně jako u diskrétních PID/PSD regulátorů velikost periody vzorkování může do značné míry ovlivnit přechodový děj i u fuzzy regulátorů. V literatuře můžeme najít celou řadu postupů, jak určit vhodnou velikost periody vzorkování pro klasické regulátory. U fuzzy regulátorů lze najít jen málo vhodných odkazů. Někdy je doporučena velikost periody vzorkování 0,1 až 0,2 hodnoty dominantní časové konstanty. Uvádí se zde, že při kratší periodě vzorkování je výpočet difference regulační odchylky příliš citlivý na vliv šumu.



Obr. 6.14: Normalizované symetrické rozložení funkcí příslušností

S tímto tvrzením nelze zcela souhlasit. Vhodným analogovým filtrem můžeme do značné míry šum potlačit. Výpočet difference regulační odchylky s filtrací výstupní veličiny pak znázorňuje **Obr. 6.15**. Při sledování vlivu periody vzorkování musíme mít na zřeteli nejen sledování změn žádané hodnoty regulačním obvodem, ale i vyregulování poruchových veličin působících na proces. Právě ve druhém případě lze zjistit, že kratší perioda vzorkování má příznivější vliv na regulační obvod a porucha je vyregulována rychleji a s menším překmitem. Je žádoucí, aby na frekvenci vzorkování byl dostatečně potlačen vstupující šum, na druhé straně při velké časové konstantě filtru může dojít ke zhoršení podmínek pro stabilitu systému a tím i ke zhoršení celkové dynamiky systému.



Obr. 6.15: Výpočet difference regulační odchylky s filtrací výst. veličiny

Bude-li velikost dominantní časové konstanty $T_1 = 10$ s, pak podle výše uvedeného je vhodná minimální velikost periody vzorkování $T = 1$ s. Požadujeme-li na této frekvenci vzorkování potlačení šumu o 30 dB, pak časová konstanta jednoduchého filtru prvního řádu bude $T_f = 10$ s a je srovnatelná s velikostí dominantní časové konstanty, což je nežádoucí. Proto je vhodnější používat filtry vyššího řádu s vyšší strmostí útlumu amplitudové frekvenční charakteristiky. V současné době nabízí řada výrobců aktivní dolnoprostopustné filtry s periodou vzorkování až do 10 s (maximální frekvence je 0,1 Hz). Realizace filtru je velmi jednoduchá. Uživatel může použít až 8-mý řád s aproximací podle Bessela nebo Butterwortha. Analogový filtr může být snadno doplněn číslicovým filtrem s několikanásobně kratší periodou vzorkování, než je perioda vzorkování fuzzy regulátoru. Těmito postupy je možné periodu vzorkování značně zkrátit. Na druhé straně výpočet akčního zásahu u fuzzy regulátoru je časově poměrně náročný, proto je vhodné periodu vzorkování a nastavení filtru určit až po ověření na reálném systému s konkrétním regulátorem. Rovněž musíme mít na zřeteli, že se v systému mohou vyskytovat rušivé signály o frekvenci nižší než je frekvence vzorkování. Tady nám filtrace příliš nepomůže.

6.7.2 Metoda návrhu fuzzy PI regulátoru s normalizovaným universem

Spojité PI regulátor je dán rovnicí

$$u(t) = K \left(e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau \right) \quad (6.25)$$

Derivaci získáme (zjištění lokálního extrému)

$$\dot{u}(t) = K \left(\dot{e}(t) + \frac{1}{T_I} e(t) \right) \quad (6.26)$$

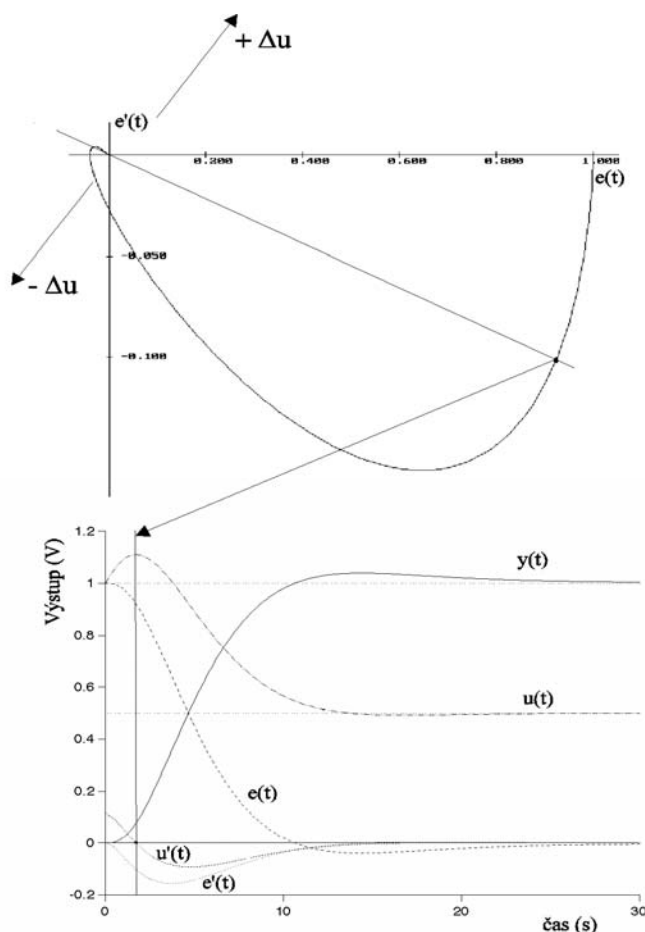
Zajímá nás kdy je derivace akčního zásahu rovna 0

$$\dot{u}(t) = K \left(\dot{e}(t) + \frac{1}{T_I} e(t) \right) = 0 \quad (6.27)$$

Řešením je rovnice

$$\dot{e}(t) = - \frac{1}{T_I} e(t) \quad (6.28)$$

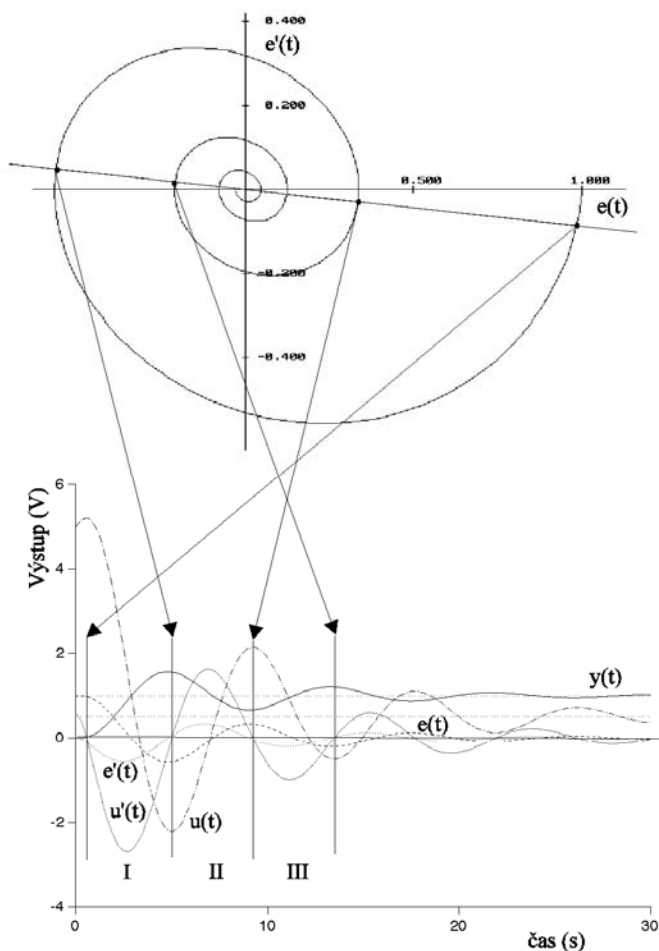
protože pro zesílení PI regulátoru musí platit $K > 0$.



Obr. 6.16: Trajektorie regulačního obvodu s PI regulátorem a zesílením K_1

Rovnice přímky (6.28) závisí jen na integrační časové konstantě PI regulátoru a její fyzikální význam spočívá v tom, že udává hranici, kde se mění znaménko derivace akčního zásahu z kladného na záporné, pokud stavová trajektorie regulačního obvodu protíná přímku při přechodu zprava doleva (či shora dolů) ve stavové rovině $\dot{e}(t)$, $e(t)$. Situace je zobrazena na **Obr. 6.16**. Při přechodu stavové trajektorie zleva doprava (či zdola nahoru) při protínání

přímky se znaménko derivace akčního zásahu mění ze záporného na kladné. Na přechodových charakteristikách je tedy místo, kde stavová trajektorie protíná přímku (pro $\dot{u}(t) = 0$), charakterizováno změnou znaménka derivace akčního zásahu $\dot{u}(t)$. Při změně zesílení $K_2 > K_1$ zůstává poloha přímky zachována **Obr. 6.17**.



Obr. 6.17: Průběhy veličin při stejném zadání a $K_2 > K_1$

Převedením rovnice (6.28) do diskrétního tvaru získáme rovnici pro přírůstek diskrétního PI regulátoru

$$\Delta u(k) = K \left(\Delta e(k) + \frac{1}{T_1} e(k) \right) \quad (6.29)$$

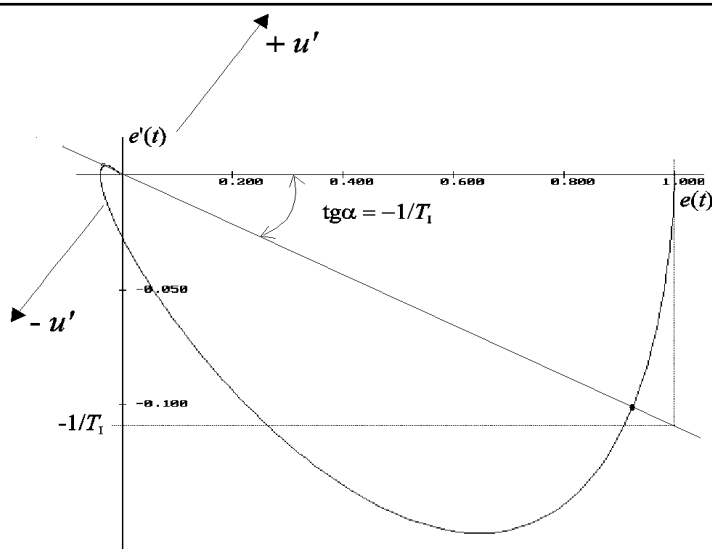
kde

$$\Delta u(k) = (u(k) - u(k-1)) / T ;$$

$$\Delta e(k) = (e(k) - e(k-1)) / T; \quad T \text{ je perioda vzorkování.}$$

Z **Obr. 6.18** je zřejmé, že časová konstanta T_1 má vztah k derivaci odchylky. Proto pro odvození fuzzy PI regulátoru upravíme

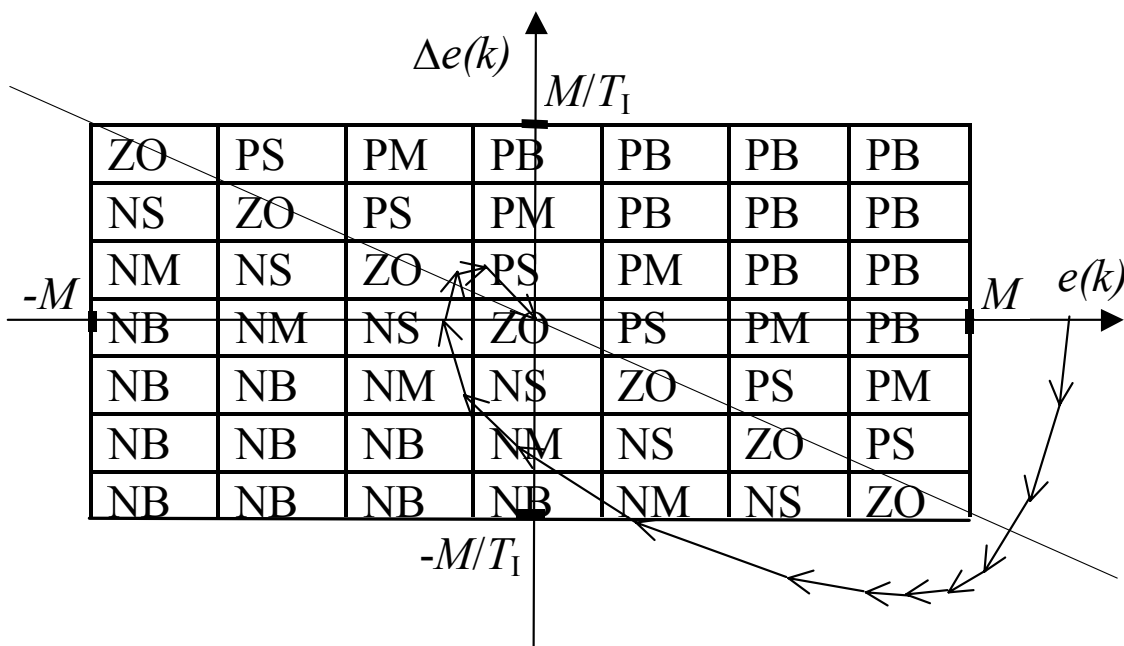
$$\Delta u(k) = K \frac{1}{T_1} (T_1 \Delta e(k) + e(k)) \quad (6.30)$$



Obr. 6.18: Stavová trajektorie regulačního obvodu s PI regulátorem

V dalším kroku je nutné namapovat bázi pravidel do diskretní stavové roviny $\Delta e(k)$, $e(k)$. Zavedeme měřítko M pro rozsah universa, $M > 0$. Toto měřítko nastaví rozsahy universa pro regulační odchylku a její první diferenci (**Obr. 6.19**). Rozšířením rovnice (4.29) dostaneme

$$\Delta u(k) = K \frac{M}{T_1} \left(\frac{T_1}{M} \Delta e(k) + \frac{1}{M} e(k) \right) \tag{6.31}$$



Obr. 6.19: Mapování báze pravidel fuzzy PI reg. do diskretní stav. roviny

Vstupní proměnné fuzzifikujeme. Po defuzzifikaci dostaneme

$$\Delta u(k) = K \frac{M}{T_1} \mathbf{D} \{ \mathbf{F} \{ \frac{T_1}{M} \Delta e(k) + \frac{1}{M} e(k) \} \} \tag{6.32}$$

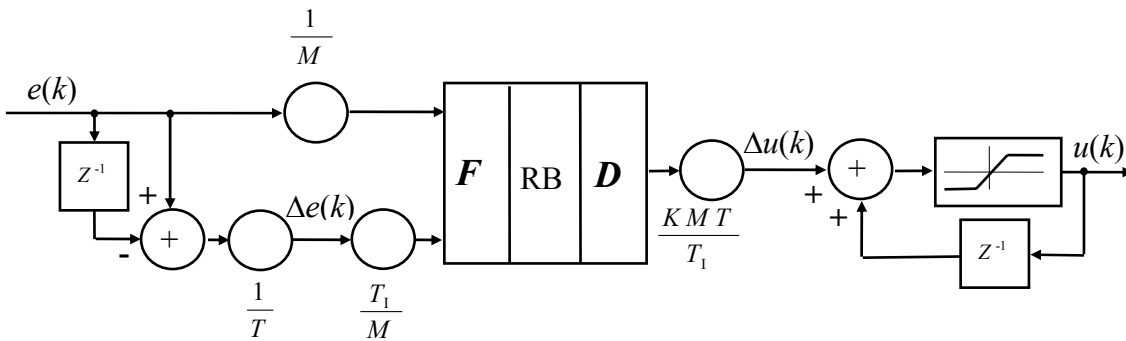
kde \mathbf{F} je operace fuzzifikace a \mathbf{D} operace defuzzifikace. Dosadíme za $\Delta u(k)$

$$\Delta u(k) = \frac{u(k) - u(k-1)}{T} = K \frac{M}{T_1} \mathbf{D} \{ \mathbf{F} \{ \frac{T_1}{M} \Delta e(k) + \frac{1}{M} e(k) \} \} \quad (6.33)$$

Výsledná hodnota akčního zásahu fuzzy PI regulátoru v kroku k je

$$u(k) = K \frac{MT}{T_1} \mathbf{D} \{ \mathbf{F} \{ \frac{T_1}{M} \Delta e(k) + \frac{1}{M} e(k) \} \} + u(k-1) \quad (6.34)$$

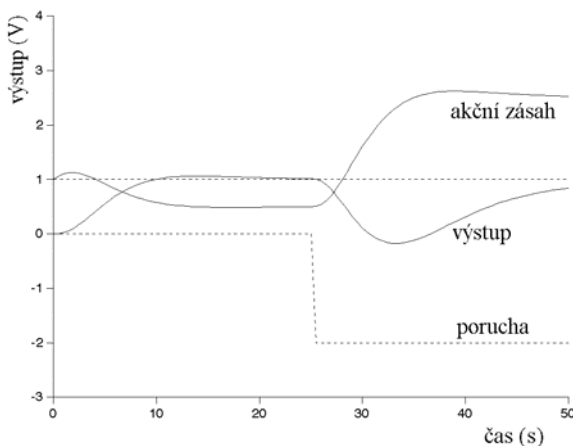
Realizace fuzzy PI regulátoru podle rovnice (6.34) je na **Obr. 6.20**.



Obr. 6.20: Struktura fuzzy PI regulátoru s normaliz. rozsahem universa

Fyzikální význam parametrů u fuzzy PI regulátoru zůstal zachován jako u PI regulátoru a to jak pro zesílení regulátoru K , tak i pro časovou konstantu integrace T_1 . Při nastavování fuzzy PI regulátoru můžeme postupovat obdobně jako při nastavování parametrů klasického PI regulátoru. Na přenosovou funkci

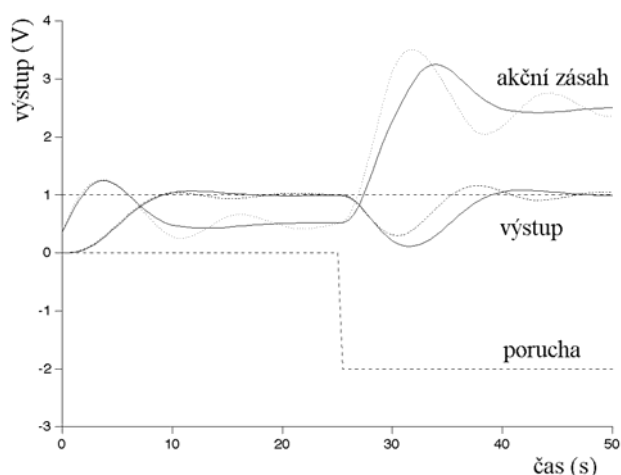
$$F(s) = \frac{2}{(10s+1)(s+1)^2} \quad (6.35)$$



Obr. 6.21: Průběhy veličin v regulačním obvodu s PI regulátorem

Vzhledem k tomu, že báze pravidel je dána pro všechny proměnné u fuzzy PI regulátoru podle **Tabulka 6.1** a) a rozložení všech funkcí příslušností podle **Obr. 6.14** nelze očekávat příliš rozdílné výsledky.

byl podle navrhnut PI regulátor s $K = 1$, $T_1 = 8,33$ s. Přechodové charakteristiky systému s tímto regulátorem jsou na **Obr. 6.21**. Na **Obr. 6.22** jsou přechodové charakteristiky systému se stejným přenosem a s nastavením fuzzy PI regulátoru $K = 2$, $T_1 = 4,7$ s, $M = 10$. V obou případech byla volena perioda vzorkování $T = 0,1$ s. Přitom u obou regulátorů lze měnit plynule periodu vzorkování bez nutnosti měnit další parametry regulátorů.

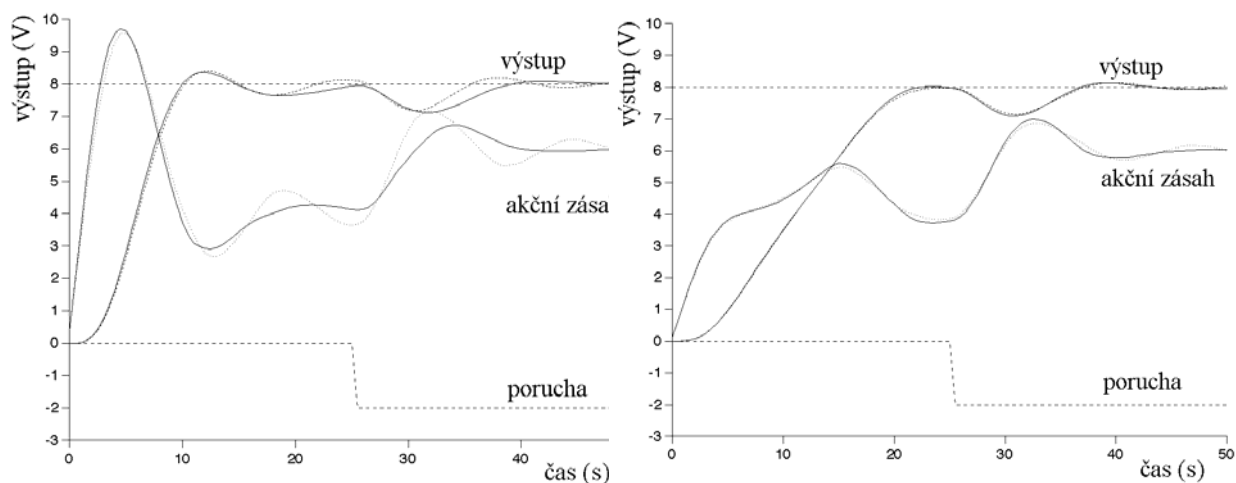


Obr. 6.22: Průběhy veličin v regulačním obvodu s fuzzy PI regulátorem

Přesto fuzzy PI regulátor v tomto případě lépe vyreguluje poruchu působící na vstup přenosové funkce (6.35).

Tečkované průběhy odpovídají fuzzy PI regulátoru s inferencí Min-Max, plně pak metodě Prod-Max. Obecně lze konstatovat, že metoda Prod-Max dává při stejném nastavení všech parametrů méně kmitavé průběhy veličin s menšími překmity, ale může se prodloužit doba pro vyregulování poruchové veličiny.

Volba parametru M ovlivňuje mapování báze pravidel do diskretní stavové roviny. Pokud stavová charakteristika systému nevybočí z namapované báze pravidel, je v tomto rozsahu změn žádaných hodnot možné očekávat přibližně stejné chování systému za předpokladu, že fuzzy regulátor je realizován s přibližně lineárním nastavením. Pokud bude M menší, než je hodnota regulační odchylky $e(k)$ viz **Obr. 6.19**, nebo dojde k vybočení z namapované báze pravidel, dojde i k omezení přírůstku akčního zásahu. Velikost přírůstku akčního zásahu je pak omezena na maximální hodnotu danou inferencí na příslušném okraji tabulky, kde k vybočení došlo. Těto vlastnosti lze využít k požadovanému omezení trendu akčního zásahu při větších odchylkách z technologických důvodů.



Obr. 6.23: Fuzzy PI regulátor a) $M=10$, $w=8$; b) $M=3$, $w=8$

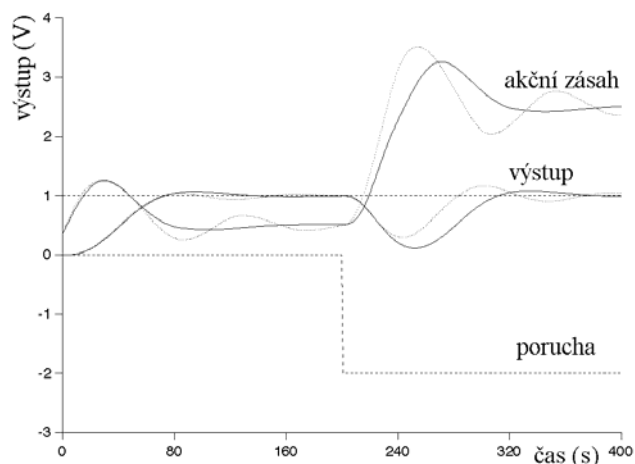
Na **Obr. 6.23** a) jsou průběhy veličin regulačního systému s přenosovou funkcí (6.35) s fuzzy PI regulátorem při nastavení $K = 2$, $T_I = 4,7$ s, $M = 10$, $w = 8$. Při stejném nastavení, ale s měřítkem $M = 3$ je omezen trend akčního zásahu **Obr. 6.23** b).

Fuzzy PI regulátor realizovaný podle (6.34) dovoluje změnu časového měřítka. Tento princip je často využíván v simulaci velmi rychlých nebo velmi pomalých dějů. Vynásobením všech časových konstant stejným násobkem (pozor při integračním charakteru odezvy!) jsou průběhy transformovány do jiného časového měřítka bez nutnosti měnit další parametry systému. Protože tato vlastnost platí jen u lineárních systémů, je zřejmé, že tento fuzzy PI regulátor je navržen s přibližně lineárním nastavením.

Na **Obr. 6.24** jsou průběhy regulačního systému s přenosovou funkcí (koeficient násobku je 8x)

$$F(s) = \frac{2}{(80s + 1)(8s + 1)^2} \quad (6.36)$$

a fuzzy PI regulátorem s parametry $K = 2$, $T_I = 4,7.8 = 37,6$ s , $M = 10$.



Obr. 6.24: Změna časového měřítka při simulaci

Dalším důkazem správnosti výše uvedeného návrhu je, že při zachování konstantního součinu zesílení v otevřené smyčce se dynamika procesu při skokové změně žádané hodnoty s fuzzy PI regulátorem se změnou zesílení soustavy nebo regulátoru nezmění.

Pokud použijeme nelineární či nesymetrické rozložení funkcí příslušností je zřejmé, že se regulační obvod stane nelineárním. Dokážeme-li funkce příslušností v konkrétním případě správně rozmístit, můžeme realizovat regulační pochod s o něco výhodnějšími vlastnostmi než při použití fuzzy PI regulátoru s přibližně lineárním rozložením funkcí příslušností. V případě změny rozložení funkcí příslušností nebo úpravě v bázi pravidel je nutné zpravidla změnit i parametry nastavení regulátoru.

6.7.3 Metoda návrhu fuzzy PD regulátoru s normalizovaným universem

Spojité PD regulátor je dán rovnicí

$$u(t) = K (e(t) + T_D \dot{e}(t)) \quad (6.37)$$

Zajímá nás kdy bude akční zásah $u(t) = 0$ při $K > 0$

$$e(t) + T_D \dot{e}(t) = 0 \quad (6.38)$$

Hledaná podmínka je

$$\dot{e}(t) = -\frac{1}{T_D} e(t) \quad (6.39)$$

Rovnice přímky závisí jen na derivační časové konstantě PD regulátoru a její fyzikální význam je podobný jako u PI regulátoru. Převedením rovnice do diskrétního tvaru získáme rovnici diskrétního PD regulátoru

$$u(k) = K (e(k) + T_D \Delta e(k)) \quad (6.40)$$

kde $\Delta e(k) = (e(k) - e(k-1)) / T$; T je perioda vzorkování

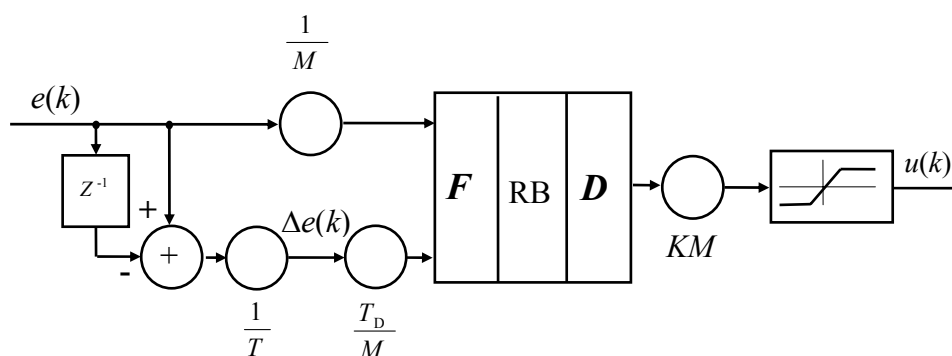
V dalším kroku mapujeme bázi pravidel do diskrétní stavové roviny $\Delta e(k)$, $e(k)$. Zavedeme měřítko M pro rozsah universa, $M > 0$. Toto měřítko nastaví rozsahy universa pro regulační odchylku a její první diferenci. Rozšířením rovnice (6.40) dostaneme

$$u(k) = KM \left(\frac{1}{M} e(k) + \frac{T_D}{M} \Delta e(k) \right) \quad (6.41)$$

Vstupní proměnné fuzzifikujeme. Po defuzzifikaci dostaneme rovnici

$$u(k) = KM \mathbf{D} \{ \mathbf{F} \{ \frac{1}{M} e(k) + \frac{T_D}{M} \Delta e(k) \} \} \quad (6.42)$$

Výsledná hodnota akčního zásahu fuzzy PD regulátoru je dána rovnicí (6.42).



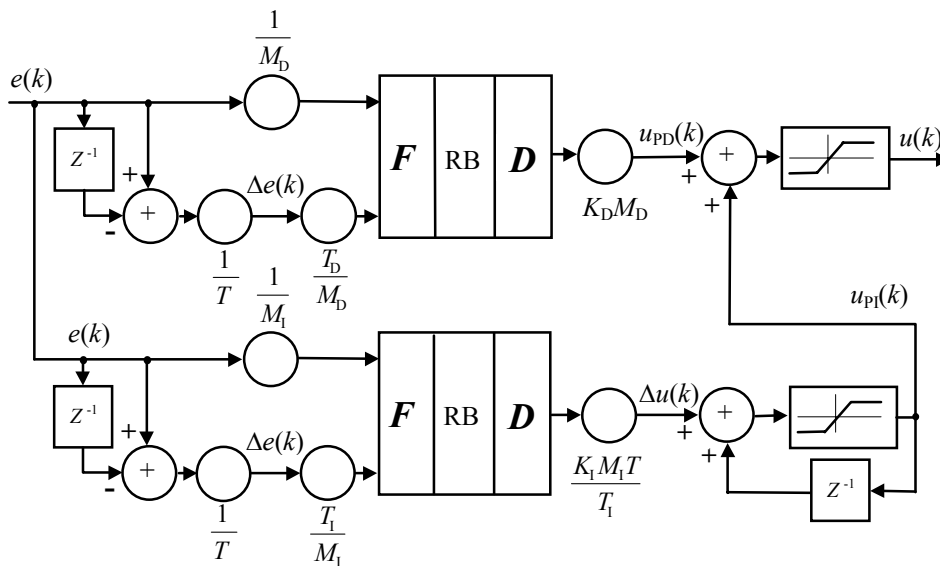
Obr. 6.25: Realizace fuzzy PD regulátoru s norm. universem podle (6.42)

6.7.4 Metoda návrhu fuzzy PD+PI regulátoru

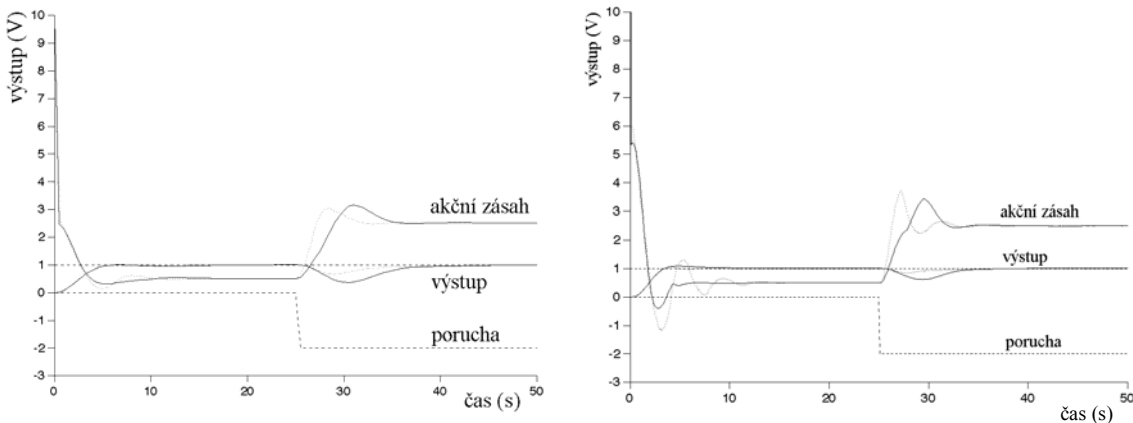
Fuzzy PID regulátor může mít řadu variant. Z praktického hlediska však nejčastěji připadají v úvahu čtyři realizace - jako paralelní kombinace regulátoru PD+PI, PD+I, P+I+D a PI+D.

Paralelním spojením fuzzy regulátorů dostáváme fuzzy PD+PI regulátor **Obr. 6.26**. Regulátor má společné tři konstanty - zesílení K , měřítko M a periodu vzorkování T . Je výhodné, jestliže si zavedeme pro každý regulátor zesílení a měřítko zvlášť, u PI regulátoru označíme zesílení jako K_I , měřítko M_I a u PD regulátoru zesílení jako K_D , měřítko M_D . Perioda vzorkování může být stejná nebo různá. Pokud optimalizujeme nastavení parametrů fuzzy regulátoru s ohledem na málo kmitavý průběh akčního zásahu při změně žádané hodnoty, můžeme parametry regulátoru nastavit na $K_I = K_D = 2$, $T_I = 3$ s, $T_D = 1,5$ s, $M_I = M_D = 10$, $T = 0,1$ s. Výsledky simulace s přenosovou funkcí (6.35) jsou na **Obr. 6.27**. Protože význam konstant regulátorů je podobný jako u klasického regulátoru, je fuzzy regulátor seřízen podstatně rychleji, než je jeho seřizování dřívějším způsobem. Vzhledem k tomu, že

měřítka M i zesílení K může být pro oba regulátory v počáteční fázi seřizování regulátoru stejné, nastavujeme pouze tři parametry $K = K_I = K_D$, T_I a T_D . Rovněž i fuzzy PD+PI regulátor dovoluje použít časovou transformaci.



Obr. 6.26: Struktura fuzzy PD+PI regulátoru s norm. rozsahem universa

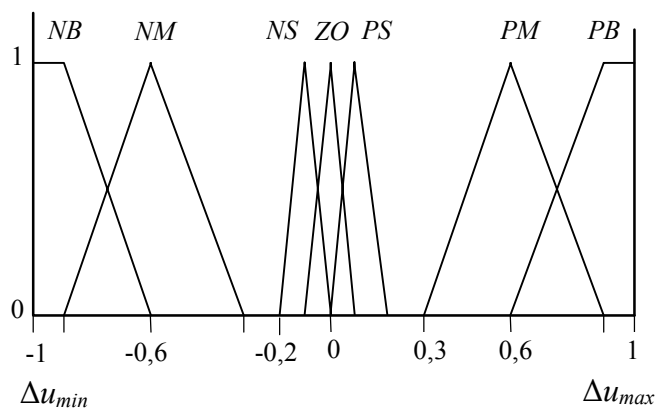


Obr. 6.27: Fuzzy PI+PD regulátor a) optim. na ž.h. b) optim. na poruchu

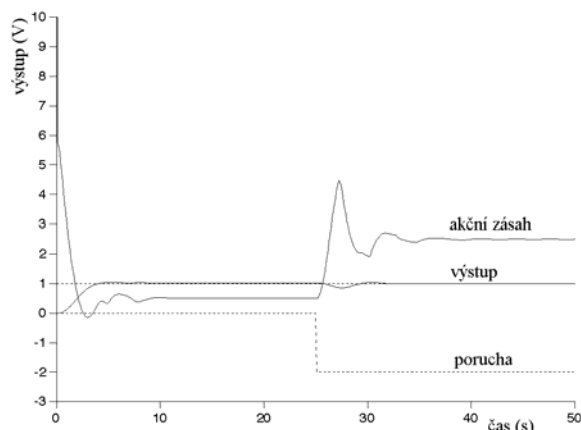
Pokud budeme nastavení fuzzy PI+PD regulátoru optimalizovat s ohledem na rychlé vyregulování poruchové veličiny působící na vstup přenosové funkce můžeme parametry regulátoru nastavit například na $K_I = K_D = 4$, $T_I = 2,2$ s, $T_D = 2$ s, $M_I = M_D = 10$, $T = 0,1$ s **Obr. 6.28 b)**. U tohoto regulátoru nemůžeme ještě použít stejného nastavení jako u klasického regulátoru, protože v jeho struktuře se vyskytuje dvakrát proporcionální složka, ale již můžeme využít svých zkušeností s klasickým nastavováním PID regulátoru.

Ve všech výše uvedených simulačních experimentech bylo použito u fuzzy regulátorů pro vstupní i výstupní proměnné pouze rozložení funkcí příslušností podle **Obr. 6.2 a)** a báze pravidel byla dána **Tabulka 6.1**. Pokud tyto výsledky porovnáme s klasickými regulátory, zjistíme, že se příliš neliší, tento závěr však bylo možné očekávat. Nelineární rozložení funkcí příslušností může přispět ke zlepšení dynamiky. Pokud má uživatel vhodný nástroj pro simulaci fuzzy regulátorů s grafickým sledováním průběhu defuzzifikace a je obeznámen se

základními principy regulace, může poměrně rychle nastavit fuzzy regulátor i při nelineárním rozložení funkcí příslušností. Pokud použijeme stejné nastavení všech parametrů fuzzy regulátoru jako u předchozí simulace mimo změn v rozložení funkcí příslušností pro přírůstek akčního zásahu u fuzzy PI regulátoru, které je realizováno podle **Obr. 6.28**, dostáváme průběhy regulačních veličin na **Obr. 6.29**. Nastavení parametrů bylo optimalizováno pro metodu inference Min-Max.



Obr. 6.28: Nelineární rozložení funkcí příslušností u fuzzy PI regulátoru



Obr. 6.29: Fuzzy PD+PI reg. s inferencí Min-Max a nelin. rozl. funkcí přísl.

6.7.5 Metoda návrhu fuzzy PD+I regulátoru

Nejprve bude odvozen fuzzy I regulátor. Rovnice popisující klasický I regulátor je dána

$$u(t) = K \frac{1}{T_I} \int_0^t e(\tau) d\tau \quad (6.43)$$

Derivací (6.43) dostaneme

$$\dot{u}(t) = K \frac{1}{T_I} e(t) \quad (6.44)$$

Převedením rovnice (6.44) do diskretního tvaru získáme rovnici diskretního I regulátoru

$$\Delta u(k) = K \frac{1}{T_1} e(k) \quad (6.45)$$

kde $\Delta u(k) = (u(k) - u(k-1)) / T$; T je perioda vzorkování

Báze pravidel fuzzy I regulátoru je dána **Tabulka 6.3**. Zavedeme měřítko M pro rozsah universa, $M > 0$. Rozšířením rovnice (6.45) dostaneme

$$\Delta u(k) = K \frac{M}{T_1} \left(\frac{1}{M} e(k) \right) \quad (6.46)$$

Vstupní proměnné fuzzifikujeme. Po defuzzifikaci dostaneme rovnici

$$\Delta u(k) = K \frac{M}{T_1} \mathbf{D} \{ \mathbf{F} \{ \frac{1}{M} e(k) \} \} \quad (6.47)$$

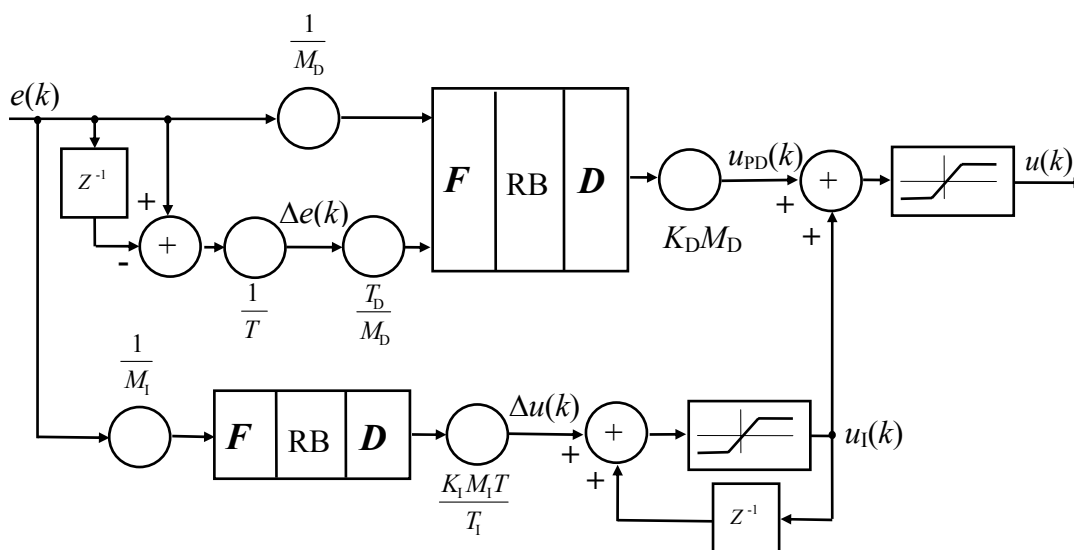
Dosadíme za $\Delta u(k)$

$$\Delta u(k) = \frac{u(k) - u(k-1)}{T} = K \frac{M}{T_1} \mathbf{D} \{ \mathbf{F} \{ \frac{1}{M} e(k) \} \} \quad (6.48)$$

Výsledná hodnota akčního zásahu I regulátoru v kroku k je

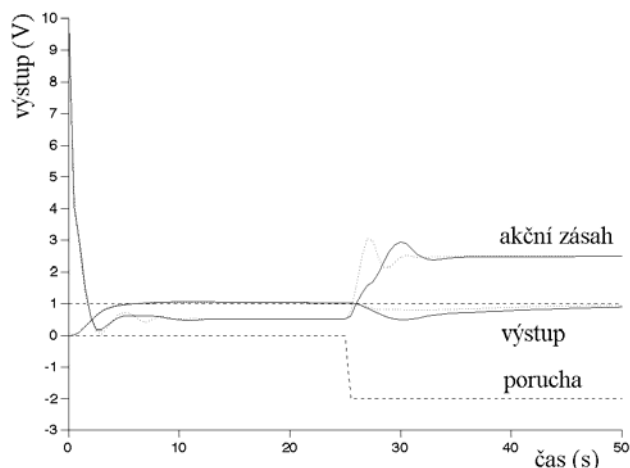
$$u(k) = K \frac{MT}{T_1} \mathbf{D} \{ \mathbf{F} \{ \frac{1}{M} e(k) \} \} + u(k-1) \quad (6.49)$$

Paralelním spojením dostaneme fuzzy PD+I regulátor



Obrázek 6.30: Struktura fuzzy PD+I regulátoru s norm. rozsahem universa

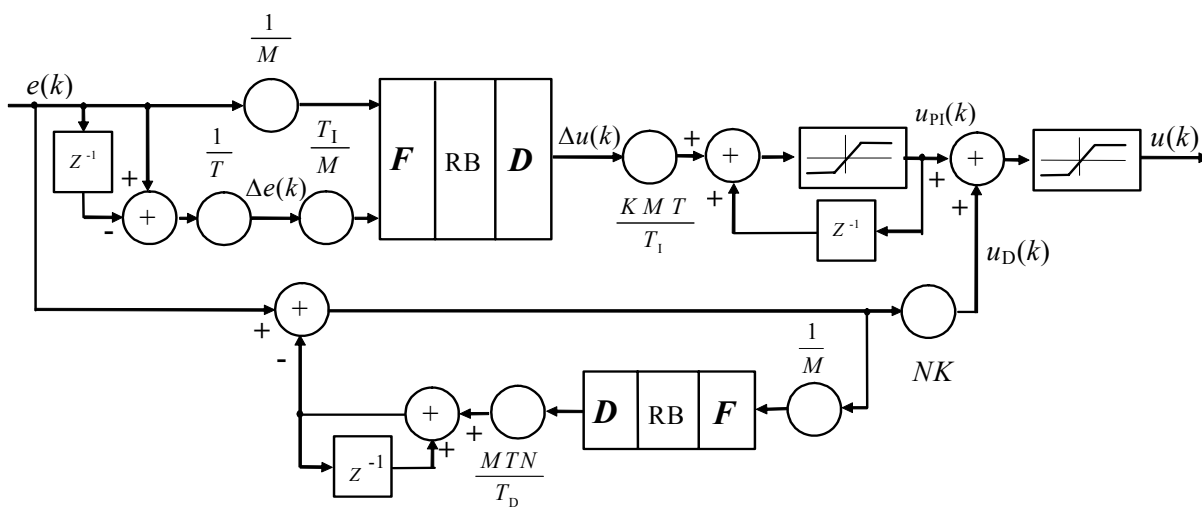
Při ověřování fuzzy PD+I regulátoru se ukázalo, že jeho nastavení je obtížnější než je nastavení fuzzy PD+PI regulátoru a to i u soustav bez integračního charakteru odezvy. Regulátor má společné tři konstanty - zesílení K , měřítko M a periodu vzorkování T . Může být výhodné, jestliže si zavedeme pro každý regulátor zesílení i měřítko zvlášť, tedy u fuzzy I regulátoru označíme zesílení jako K_I , měřítko M_I a u fuzzy PD regulátoru zesílení jako K_D , měřítko M_D . Teoreticky by bylo mělo být možné použít pro stejnou soustavu stejné parametry regulátoru. Ve skutečnosti je vliv integrační složky velmi vysoký a bylo potřebné ji upravit. Parametry fuzzy regulátoru byly nastaveny s (6.35) na $K_I = K_D = 5$, $T_I = 15$ s, $T_D = 1,5$ s, $M_I = M_D = 10$, $T = 0,1$ s. Průběhy veličin jsou na **Obrázek 6.31**.



Obrázek 6.31: Průběhy veličin v regulačním obvodu s fuzzy PD+I regulátorem

6.7.6 Metoda návrhu fuzzy PI+D regulátoru

Fuzzy PI+D regulátor je kombinace fuzzy PI a fuzzy D regulátoru. Fuzzy D regulátor využívá již odvozeného fuzzy I regulátoru ve struktuře podobné klasickému D regulátoru. Jeho nastavování se v podstatě neliší od nastavování klasického PID regulátoru. Parametry nastavení obvodu s (6.35) jsou stejné jako u PID regulátoru - $K = 7,26$, $T_I = 2,85$ s, $T_D = 0.712$ s, $M = 10$, $N = 3$, $T = 0,1$ s.



Obrázek 6.32: Struktura fuzzy PI+D regulátoru s norm. tvarem universa

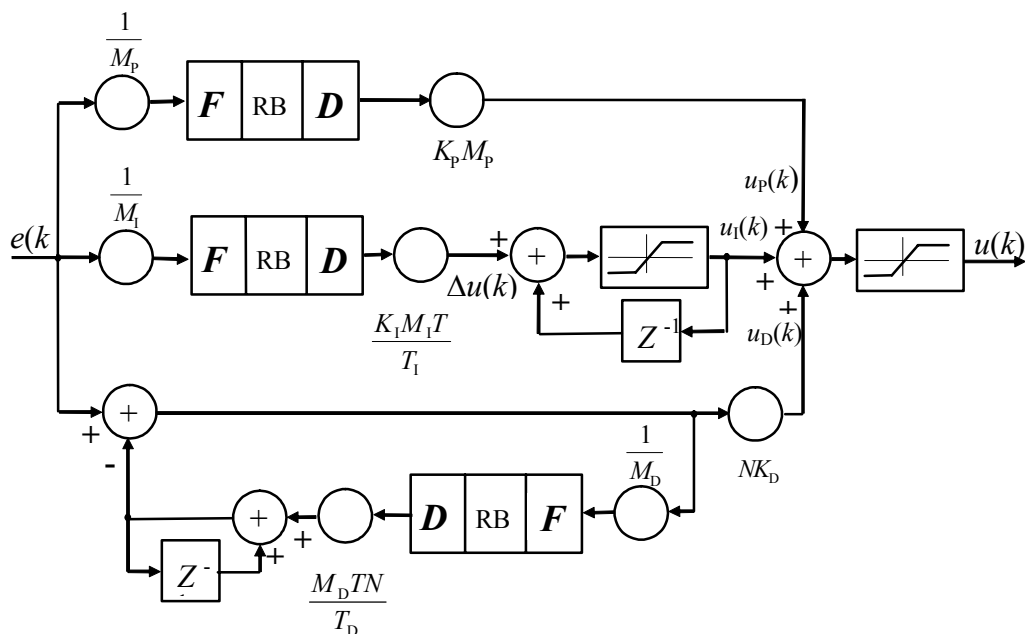
Při přibližně lineárním rozložení funkcí příslušností a stejném nastavení parametrů fuzzy regulátoru dává metoda inference Min-Max kmitavější a rychlejší přechodový děj než inference Prod-Max. Proto pro regulační obvody (např. ventily), kde není žádoucí kmitání může být použití inferenční metody Prod-Max výhodnější. Obecně je vhodnější pro fuzzy PI regulátor použít inferenci Prod-Max a pro regulátor s derivačním charakterem inferenci Min-Max.

I když v případě nelineárního nastavení funkcí příslušností fuzzy regulátorů můžeme získat v simulaci výhodnější průběhy veličin než při použití klasického PID regulátoru, mohou být u reálného systému výsledky opačné. Obvod s fuzzy regulátorem může být v důsledku rychlejší reakce akčního zásahu na odchylky od žádané hodnoty regulované veličiny méně robustní (méně odolný na změnu dynamiky a zesílení). Průběhy

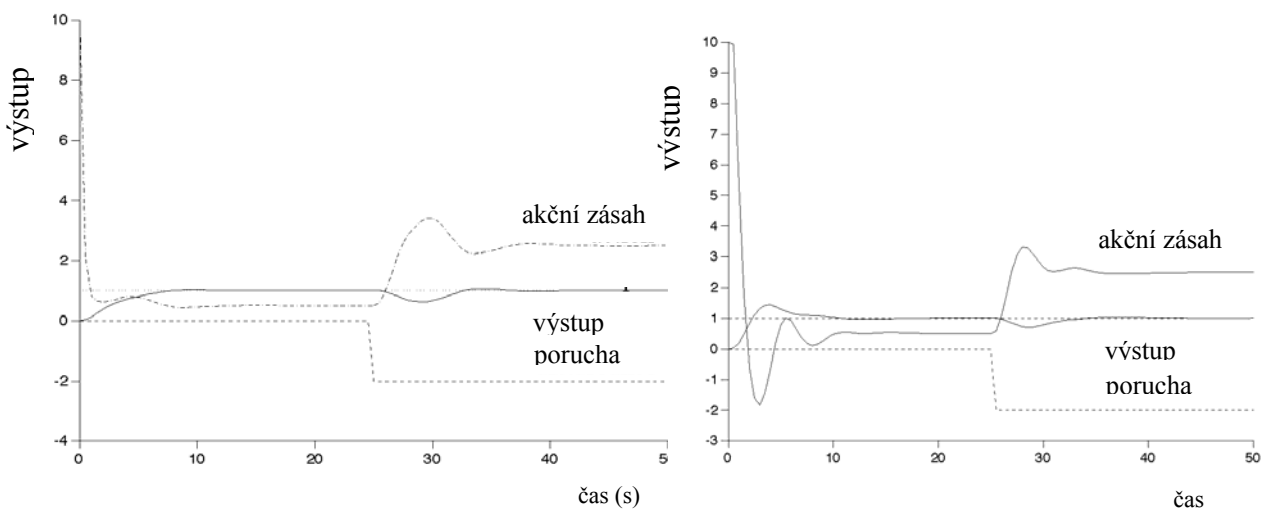
odezev v regulačním obvodu s nelineárním fuzzy regulátorem obecně závisí nejen na velikosti žádané hodnoty, ale i na amplitudě poruchových veličin vstupujících do regulačního obvodu. Proto je nutné ověřit nastavení regulátoru na reálném systému daleko pečlivěji, než při použití klasických regulátorů.

6.7.7 Metoda návrhu fuzzy P+I+D regulátoru

Spíše jako zajímavost je na **Obrázek 6.33** ukázána možnost použití fuzzy PID regulátoru s normalizovaným tvarem universa pro jednotlivé složky regulátoru, jehož parametry jsou rovněž nastavovány ve stejném fyzikálním významu jako u klasického PID regulátoru a mohou mít při použití inferenční metody Prod-Max dokonce stejné hodnoty parametrů **Chyba! Nenalezen zdroj odkazů.** Nastavení parametrů s (6.35) bylo stejné jako v předchozí části $K = K_P = K_I = K_D = 7,26$; $T_I = 2,85$ s; $T_D = 0,712$ s; $M_P = M_I = M_D = 10$, zesilovacího činitele $N = 3$ a perioda vzorkování $T = 0,1$ s.



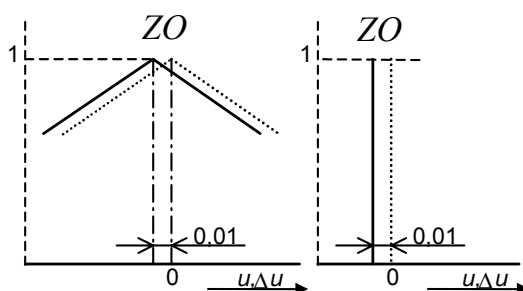
Obrázek 6.33: Struktura fuzzy P+I+D regulátoru s norm. tvarem universa



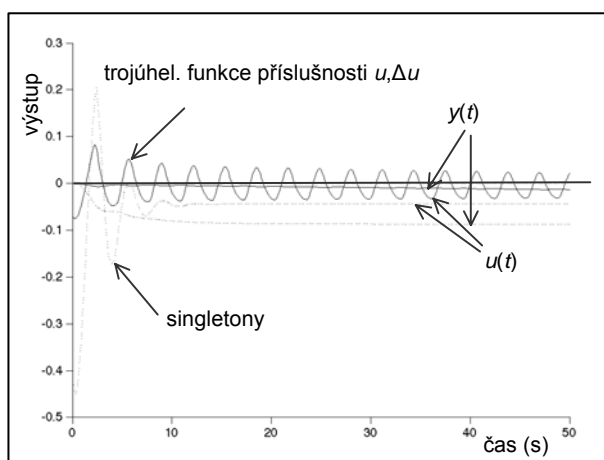
Obrázek 6.34: Průběhy veličin a) s fuzzy PI+D, b) fuzzy P+I+D regulátorem

6.8 Některé problémy vznikající při použití fuzzy regulátorů

Fuzzy regulátory jsou náchylné ke vzniku oscilací. Příčinou zpravidla nebývá jen špatné nastavení parametrů fuzzy regulátoru, ale i použitá inferenční metoda a metoda defuzzifikace ve spojení s konkrétní realizací výpočtového mechanismu a rozložením funkcí příslušnosti. Posunutí vrcholu funkce příslušnosti ZO o velikost 0,01 v normalizovaném universu pro akční zásah vyvolá oscilace akčního zásahu bez jakýkoliv vnějších zásahů. Pokud použijeme pro defuzzifikaci singletony (trojúhelníkové funkce příslušnosti v universu pro akční zásah jsou nahrazeny úsečkami spojujícími vrcholy funkcí příslušnosti s osou universa – s výhodou je pak zjednodušena defuzzifikační metoda), je kmitání sice potlačeno, ale vzniká trvalá regulační odchylka i při přítomnosti I regulátoru!



Obrázek 6.35: Posuv funkce příslušnosti ZO v universu pro akční zásah



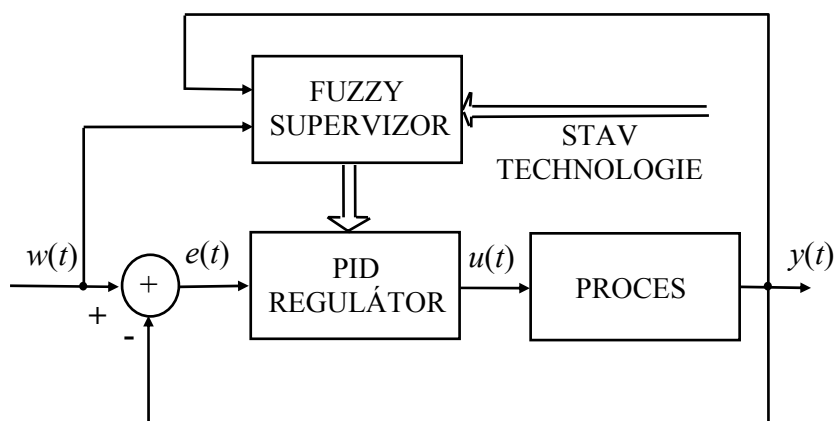
Obrázek 6.36: Vliv posunu funkce příslušnosti ZO na oscilace

Průběhy při použití metody defuzzifikace s trojúhelníkovými funkcemi příslušnosti jsou vykresleny plně, se singletony tečkovaně. Bližší analýzou vzniku oscilací zjistíme, že počáteční výchylku způsobuje v obou případech fuzzy PD regulátor, zatímco na dalším oscilačním průběhu se podílí zejména integrační část regulátoru. V obou případech je výsledkem trvalá regulační odchylka, která může vysoko převyšovat nepřesnost implementace. V tomto případě chyba vznikla malým posunutím funkce příslušnosti. Singletony jsou obecně odolnější vůči oscilacím. Ovšem vzniklá trvalá ustálená odchylka je mnohem větší a odstranit se dá jen korekcí příslušné funkce příslušnosti.

6.9 Fuzzy supervizor

Fuzzy supervizor byl vyvinut pro změnu parametrů PID regulátoru. Původně se zpravidla používal pro vytvoření PID regulátoru s nelineárním nastavováním parametrů K , T_I , T_D pro redukci překmitu nebo pro zrychlení přechodového děje. V tomto případě se vyhodnocuje ve fuzzy supervizoru hodnota odchylky a podle nastavené báze pravidel a tvaru funkcí příslušností je v každém kroku provedena úprava velikosti parametrů PID regulátoru. Typické pravidlo fuzzy supervizoru může mít tvar

$$\text{IF vstup}_1 \text{ is malý AND vstup}_2 \text{ is velký THEN parametr}_1 \text{ is střední} \quad (6.50)$$



Obr. 6.37:

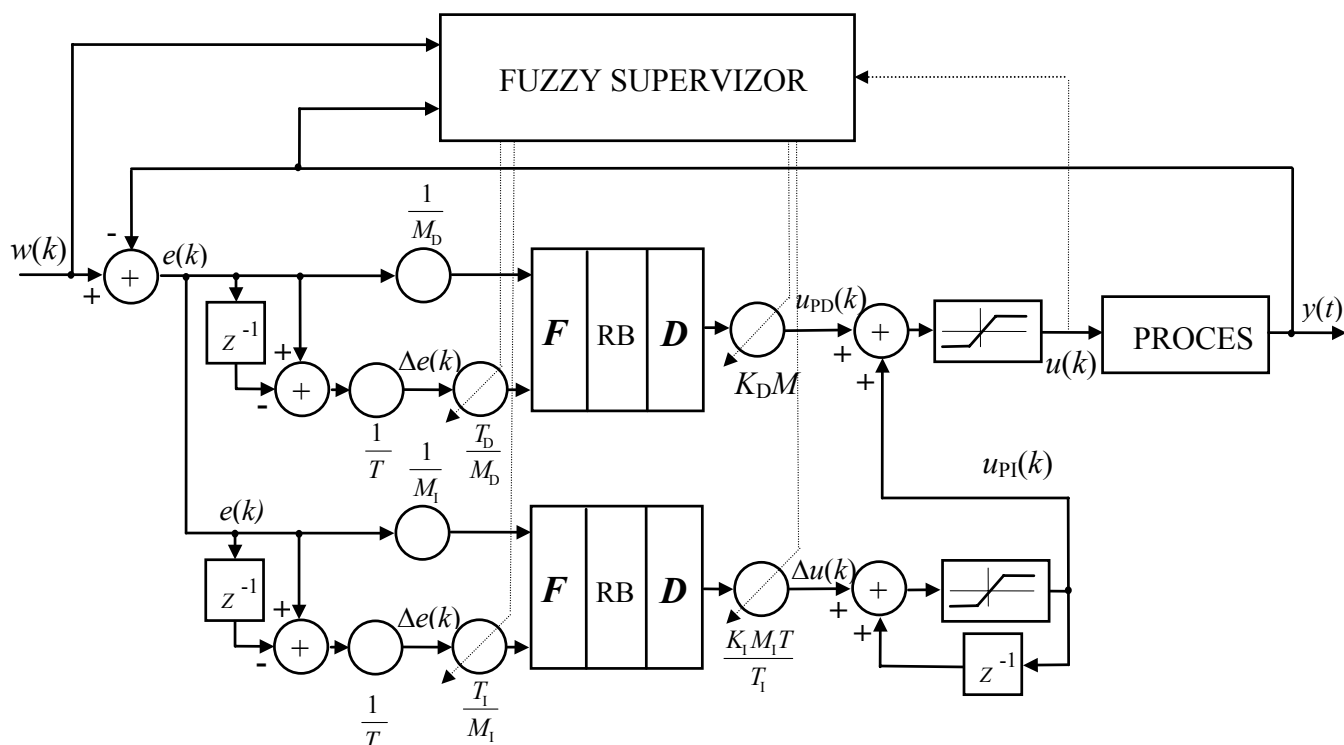
Regulační obvod s fuzzy supervizorem

Metodu fuzzy supervizoru lze snadno realizovat, protože moderní regulátory běžně umožňují externí průběžné nastavování parametrů. Použití fuzzy PID regulátoru s nelineárním nastavením by mohlo vést k podobným výsledkům. U klasického PID regulátoru může být tato strategie realizována např. pomocí pásmových algoritmů tak, že při velké hodnotě odchylky zvětšíme zesílení regulátoru a zmenšíme hodnotu integrační časové konstanty. Pokud se dostaneme do žádaného okolí pracovního bodu, zmenšíme zesílení a zvětšíme hodnotu integrační časové konstanty regulátoru.

Pokud je známo nastavení parametrů PID regulátoru pro okolí pracovních bodů, ve kterých se technologický proces může nacházet, lze snadno pomocí fuzzy supervizoru vybírat vhodné nastavení parametrů regulátoru **Obr. 6.37**. Rovněž pokud mají některé technologické části procesu poruchu či technologie musí pracovat při sníženém výkonu, pak všechny tyto stavy mohou měnit plynule parametry regulátoru. Použitý regulátor přitom může být fuzzy PI+PD (**Obr. 6.38**), případně dalších typů. Pro implementaci pak s výhodou můžeme použít fuzzy regulátor s normalizovaným tvarem universa, který lze realizovat v konkrétním řídicím systému jako standardní modul. Vhodnou strukturou fuzzy supervizoru z tohoto zapojení vytvoříme fuzzy adaptivní regulátor.

6.10 Fuzzy přepínač

Fuzzy přepínač s lokálními regulátory se nazývá Takagiho-Sugenův regulátor podle autorů odpovídajícího fuzzy modelu. Pravé strany pravidel u Takagiho-Sugenova modelu nejsou tvořeny fuzzy množinami, ale jsou obecně funkcí vstupních proměnných modelu $f_j(x_1, x_2, \dots, x_n)$ (6.51):

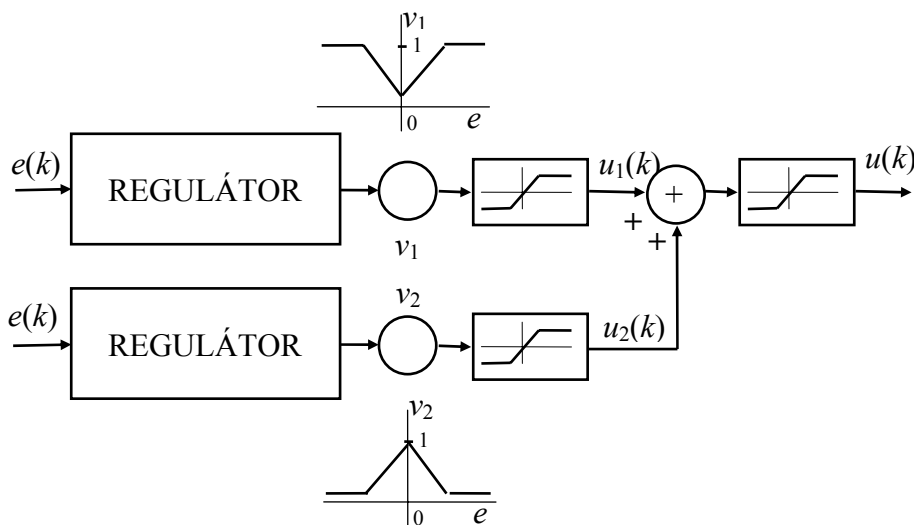


Obr. 6.38: Fuzzy adaptivní regulátor

$$\text{IF } x_1 \text{ is } A_{1j} \text{ AND } x_2 \text{ is } A_{2j} \text{ AND... AND } x_N \text{ is } A_{Nj} \text{ THEN } y_j = f_j(x_1, x_2, \dots, x_n) \quad (6.51)$$

kde $j = 1, 2, \dots, m$.

Realizace regulátoru podle Takagiho-Sugenoova modelu může být velmi jednoduchá. Na **Obr. 6.39** jsou použity dva regulátory, jejichž výstup je váhami v_1 a v_2 vážen. Okamžité hodnoty vah jsou závislé na okamžité hodnotě velikosti odchylky. Je zřejmé, že pro větší odchylky je použit první regulátor, pro okolí žádané hodnoty regulátor druhý. Podobně můžeme realizovat celou řadu paralelně pracujících regulátorů různých typů, každý může být nastaven optimálně na jeden z odpovídajících stavů technologie. Paralelní spolupráce více regulátorů je zde velmi vhodná, protože vážením lze zajistit beznárazové přepínání.



Obr. 6.39: Jednoduchá varianta realizace fuzzy přepínače

6.11 Fuzzy regulátor s více vstupy

Dosavadní praktické zkušenosti ukazují, že úspěšná nasazení fuzzy regulátorů byla realizována zejména u procesů s obtížně popsatelným chováním, které byly charakterizovány výskytem silně nelineárních závislostí mezi proměnnými veličinami jako jsou např. cementárny, tavírny skla, mlýny. V těchto aplikacích nebývá struktura fuzzy regulátoru založena jen na odchylce regulované veličiny a její diferenci, ale do fuzzy regulátoru vstupují další proměnné s cílem optimalizovat velikost akčního zásahu s ohledem na okamžitý stav celé technologie ve snaze dosáhnout výhodnějších průběhů regulovaných veličin. Z regulačního hlediska označujeme tyto regulátory jako fuzzy MISO (Multi Input Single Output) regulátory a v případě dvou a více výstupů jako fuzzy MIMO (Multi Input Multi Output) regulátory. Počet funkcí příslušnosti pro jednu proměnnou bývá mnohdy výrazně omezen, zpravidla na tři funkce příslušnosti. Protože báze pravidel s každou další vstupující proměnnou silně narůstá, jsou z této báze vybírána jen pravidla, která mají podstatný vliv na regulační pochody a tím je báze pravidel silně redukována. Jako příklad je uvedena regulace výkonu parního kotle, který dodává do parovodu páru o jistém tlaku a teplotě. Část pravidel by mohla být zapsána ve tvaru (jsou uvedena jen dvě pravidla z více možných)

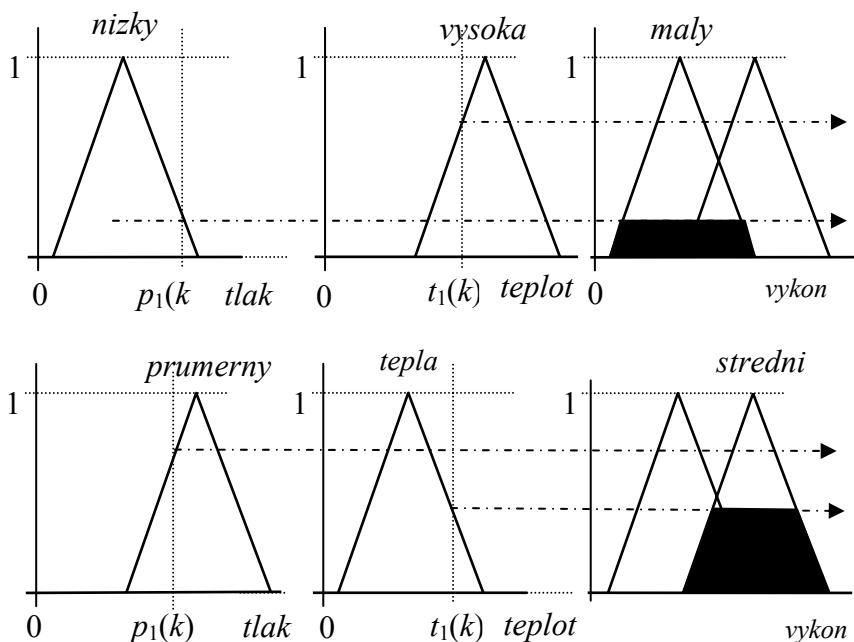
Komentář:

IF *tlak is nizky* AND *teplota is vysoka* THEN *vykon is maly*; když tlak je nízký a teplota je vysoká, je předpoklad, že se voda promění brzy v páru a proto snížíme výkon

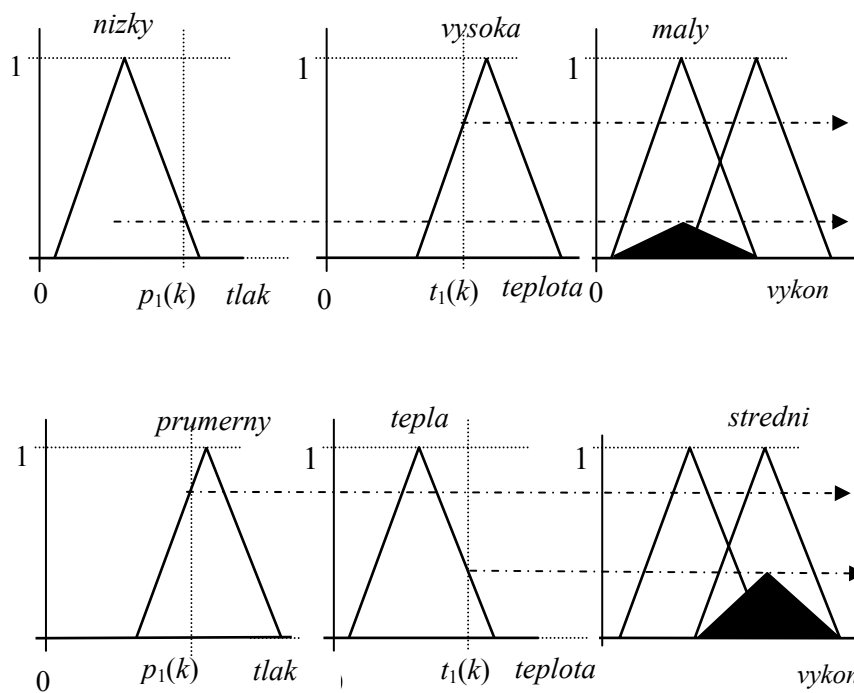
IF *tlak is prumerny* AND *teplota is tepla* THEN *vykon is stredni*; když tlak je průměrný a teplota je teplá, pak výkon necháme střední

Je zřejmé, že tvorba těchto pravidel může vycházet ze zkušeností obsluhy bloku, která po jisté době zácviku si pomáhá nějakými pomocnými pravidly. Právě zapsání těchto pravidel je úkolem technika seřizujícího regulátor. Dále musí být nastaveny rozsahy universa pro příslušné vstupní a výstupní veličiny. Složitější příklad je uveden v další části. Na **Obr. 6.40** je zakreslena fuzzy inference Min-Max a na **Obr. 6.41** Prod-Max pro dvě výše uvedená pravidla.

Inference je v tomto případě realizovaná odlišně od **Obr. 6.5** a je další možnou variantou inference Prod-Max.



Obr. 6.40: Inference Min-Max u regulátoru s dvěma vstupy a jedním výst.



Obr. 6.41: Inference Prod-Max u regulátoru s dvěma vstupy a jedním výst.

7 Neuronové sítě v řídicí technice

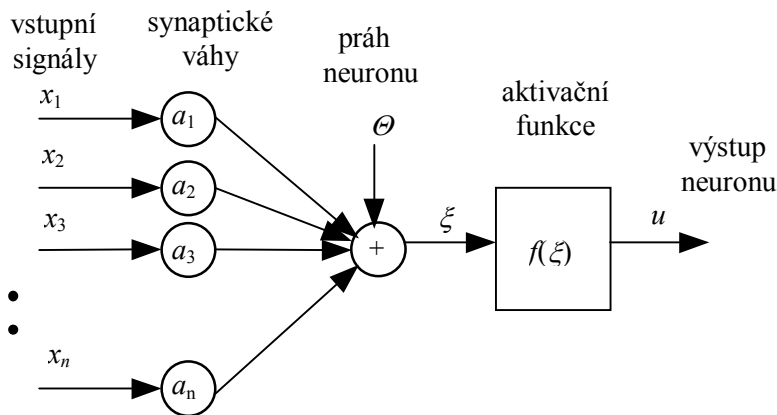
7.1 Úvod

Ve vývoji metod používajících umělou inteligenci hrají významnou roli umělé neuronové sítě, jejichž struktury se snaží napodobit biologické neuronové sítě. Název i model neuronu mají svůj původ v biologickém popisu neuronů v mozcích a tělech živých organismů, kde zjednodušeně řečeno, je jádro neuronu (soma) propojeno vstupními vlákny (dendrity) přes rozhraní (synapse) na výstupní vlákno (axon) dalšího neuronu. Podobně i umělou neuronovou síť ve své podstatě tvoří umělý neuron - dále pouze neuron, struktura neuronové sítě (vzájemné propojení neuronů) a metoda učení.

Neuron, jako základní stavební prvek umělé neuronové sítě je charakterizován (vnitřním) potenciálem, práhem a aktivitou svého výstupu. Výstupní signál neuronu podle **Obr. 7.1** je dán rovnicí (7.1).

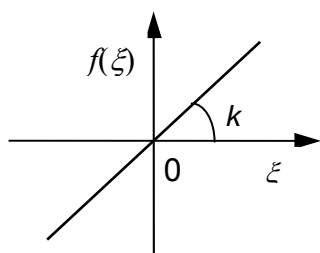
$$u = f(\xi) = \left(\sum_{i=1}^n x_i a_i + \Theta \right) = f \left(\sum_{i=0}^n x_i a_i \right) \quad (7.1)$$

Pro zjednodušení se volí práh výstupu $\Theta = x_0 a_0 = 1 \cdot a_0$

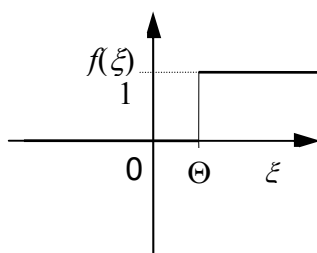


Obr. 7.1: Symbolické znázornění umělého neuronu

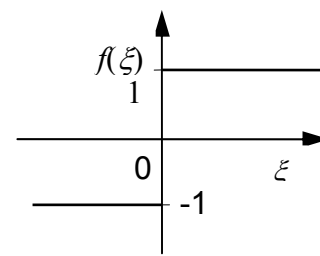
Jako aktivační funkce je používána celá řada funkcí v závislosti na použití v příslušné vrstvě neuronové sítě nebo na typu neuronové sítě **Obr. 7.2**. Základní a podstatnou vlastností neuronových sítí je jejich schopnost adaptace - tedy učení se na změněné podmínky. Proces hledání optimálního nastavení parametrů neuronové sítě se nazývá adaptace sítě. Pro adaptaci vícevrstvé neuronové sítě se často používá metoda učení typu Back-propagation (metoda zpětného šíření). Na **Obr. 7.3** je nakreslena vrstvená neuronová síť typu feed-forward (je realizována bez vnitřních zpětných vazeb). Standardní algoritmus pro adaptaci synaptických vah je založen na diferenci mezi aktuální a žádanou hodnotou výstupu. Podstatou je minimalizace chybové funkce:



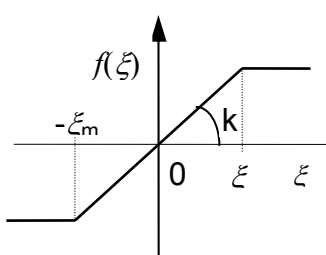
a) lineární funkce
 $f(\xi) = k\xi; k > 0$



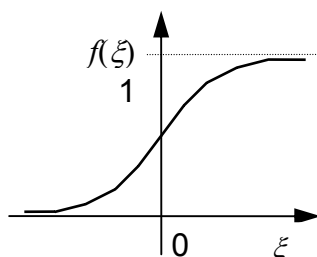
b) binární funkce- perceptron
 pro $\xi > \Theta, f(\xi) = 1$
 pro $\xi \leq \Theta, f(\xi) = 0$



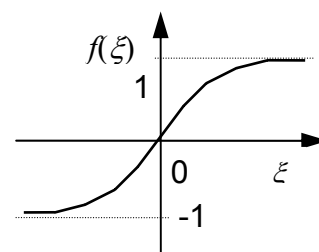
c) binární funkce
 $f(\xi) = \text{sign}(\xi)$



d) lineární funkce s omezením



e) sigmoida



f) posunutá sigmoida

$$f(\xi) = \frac{1}{1 + e^{-k\xi}}; k > 0 \quad f(\xi) = \frac{2}{1 + e^{-k\xi}} - 1; k > 0$$

Obr. 7.2: Aktivační funkce neuronu

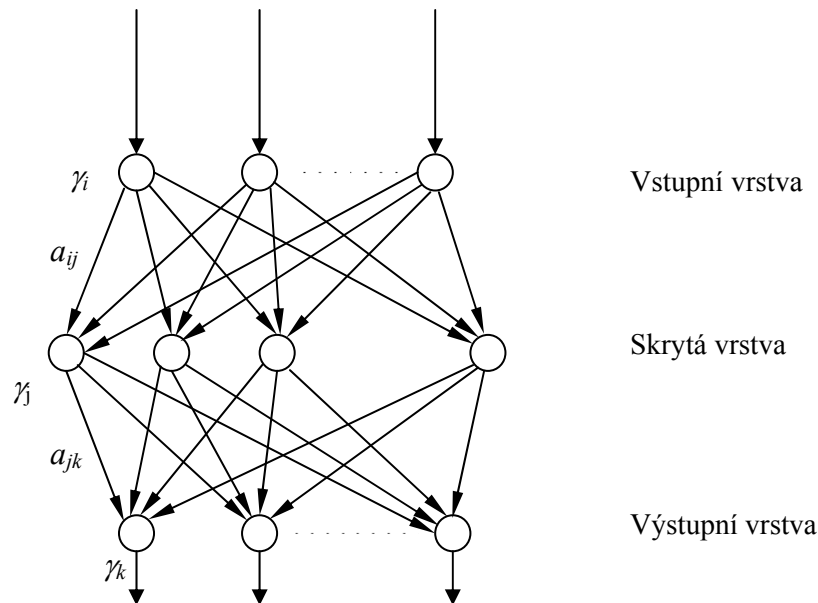
$$E = \frac{1}{2} \sum_{k=1}^m \sum_{q=1}^n (v_q(k) - \gamma_q(k))^2 \tag{7.2}$$

kde: $v_q(k)$ je aktuální hodnota q -tého výstupu neuronové sítě na vstup $r(k)$, $\gamma_q(k)$ je hodnota požadovaného q -tého výstupu sítě na vstup $r(k)$, m je počet vstupních vzorů $r(1), r(2), \dots, r(m)$, a n je počet výstupů neuronové sítě. Pro každou synaptickou váhu a_{ij} , je určena hodnota Δa_{ij} . V principu jde o gradientní metodu.

$$\Delta a_{ij}(k) = a_{ij}(k) - a_{ij}(k-1) \approx -\alpha \frac{\partial E}{\partial a_{ij}(k)} \tag{7.3}$$

kde α je učící konstanta, která má vliv na rychlost konvergence, i je index i -tého neuronu, j značí j -té spojení na i -tý neuron do nižší vrstvy. Pro omezení náhlého vlivu změn synaptických vah se zavádí momentum β . Rovnice (7.3) pak přejde do tvaru

$$a_{ij}(k) = a_{ij}(k-1) - \alpha \frac{\partial E}{\partial a_{ij}(k)} + \beta (a_{ij}(k-1) - a_{ij}(k-2)) \tag{7.4}$$



Obr. 7.3: Třívrstvá dopředná neuronová síť (feed-forward)

Algoritmus Back-Propagation BP a Marquart -Levenberg ML jsou nejčastěji používané pro učení dopředné vícevrstvé neuronové sítě, která je pro své vlastnosti nejpoužívanější v řídicí technice. Při aproximaci přenosové funkce se nejvíce osvědčuje síť s jednou skrytou vrstvou, kde ve vstupní vrstvě mají perceptrony lineární aktivační funkce. Ve skryté vrstvě je sigmoida nebo hyperbolický tangens, ty mají za následek, že je neuronová síť schopna aproximovat i nelineární funkci. Ve výstupní funkci může být posunutá sigmoida, nebo lineární funkce. Počet neuronů ve vstupní a výstupní vrstvě je dán počtem vstupů a výstupů sítě. Počet neuronů ve skryté vrstvě má být tak velký, aby byla síť schopna aproximovat daný průběh s námi požadovanou přesností. Právě tato volba je dosti obtížná a těžko se dá teoreticky přesně spočítat.

7.2 Off-line a On-line učení

Off-line učení je vhodné k naučení neuronového modelu před jeho použitím. Neuronový model se při něm učí na základě historických dat bez připojení k systému. Toto učení má výhodu v tom, že nejsme omezeni časem. Můžeme si ověřit jednotlivé varianty zapojení neuronových modelů. Navíc můžeme použít metody učení neuronové sítě známé z učení neuronových klasifikátorů a to i metody měnící topologii sítě (např. algoritmy měnící počet vnitřních neuronů).

Při off-line učení předložíme síti celý soubor historických dat jako učební vzory a na základě výsledků učení jsou algoritmem upraveny váhy v síti (příp. změněme topologii) podle dané metody učení (BP, ML). Takto naučenou síť potom použijeme pro identifikaci nebo pro návrh neuronového regulátoru nebo pro jejich kombinaci. Má-li regulátor být adaptivní musí se v učení pokračovat i během práce regulátoru. Toto učení potom nazýváme on-line učení.

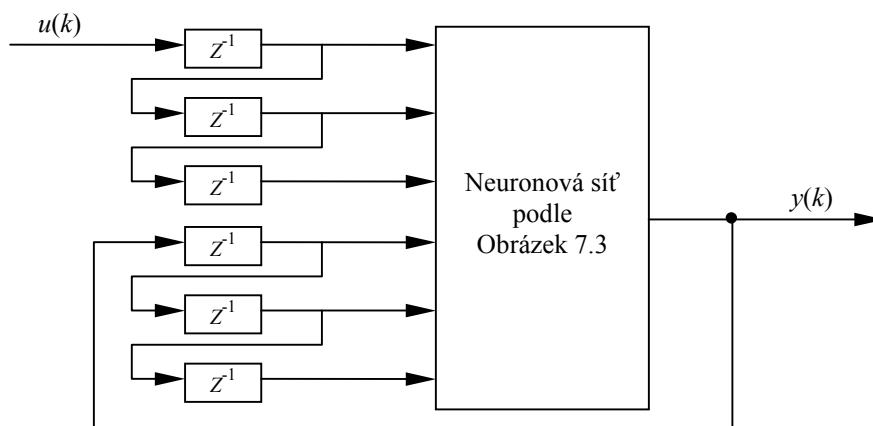
On-line učení

Při on-line učení je model připojen k systému a učí se jeho chování během pohybu systému. Předchozí průběh pohybu systému je „otisknut“ v aktuálním nastavení vah neuronového modelu a on-line učení toto nastavení v každém kroku modifikuje podle

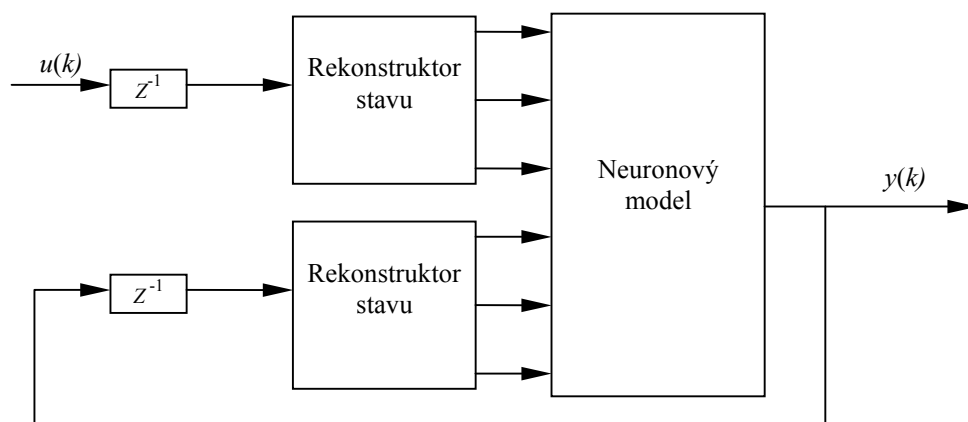
současného stavu. Z toho vyplývá, že se model učí pouze na základě současného stavu a předchozí stavy se tímto stavem postupně zapomínají. Rychlost zapomínání je závislá na konstantě učení α . To má za následek, že model zapomíná předchozí stavy a pokud se nemění dynamika procesu při změnách žádané hodnoty, model se naučí na působení poruchových veličin. Zapomíná dynamiku systému a jeho aktuální nastavení vah pak neodpovídá systému. V přírodě je zajímavý systém, který tento problém řeší. Je to regulace polohy očí. Bylo zjištěno, že i když se upřeně díváme na jeden bod, oči se nám stále pohybují. Systém se stále pohybuje a regulátor (model v něm) se má na čem učit. V technologických procesech je tento pohyb zpravidla nepřipustný (opotřebení akčních členů, nedodržení žádané hodnoty ...) a proto tento problém musíme řešit jinak.

7.3 Varianty zapojení modelu

Používá se řada variant zapojení modelu, které se liší především tvarem vstupního vektoru sítě.



Obr. 7.4: Neuronový model se zpožděnými vstupy



Obr. 7.5: Model s neuronovou sítí s rekonstruktory stavu

Rekonstruktor stavu je obvod který vypočítává (rekonstruuje) přibližně stav systému jako diferenci prvního, druhého a případně i dalšího řádu. Jen zřídka se lze setkat s jinými (většinou složitějšími) metodami rekonstrukce stavu.

Model vytváří příslušné difference jednotlivých vstupů ve váhové matici své vstupní vrstvy. Z hlediska kvantity vstupních informací jsou si tedy oba modely ekvivalentní. Mohou se lišit pouze v rychlosti učení a odolnosti proti šumu.

7.4 Neuronová síť jako jednoduchý neuronový regulátor typu PID

Výstupní vrstva neuronového regulátoru dává v kroku k akční zásah $u(k)$; odezva řízeného procesu však přichází v kroku $y(k+1)$. Pro adaptaci neuronové sítě však potřebujeme určit akční zásah $u_w(k)$, který na výstupu zajistí žádanou hodnotu $w = y_w(k+1)$. Hlavním problémem je, jak určit hodnotu $u(k) = u_w(k)$, protože odpovídající hodnota $u_w(k)$ není obecně známa. Jedna z možností, jak obejít tento problém, je určit tuto diferenci ze znalosti výstupu procesu s použitím metody Back-propagation:

$$E_1(k) = \frac{1}{2} \sum_{q=0}^n \left(y^{(q)}(k) - w^{(q)}(k) \right)^2 \quad (7.5)$$

kde $w^{(q)}(k)$ je požadovaná hodnota výstupu a její derivace, k krok výpočtu, $y^{(q)}(k)$ je výstupní veličina a její derivace a n je maximální řád uvažovaných derivací.

Pro výpočet chybové funkce musíme použít derivace (diference) výstupní veličiny. Výsledná chybová funkce se pak skládá z $n+1$ částí, z hodnoty výstupní veličiny a n jejích derivací. Každou z těchto částí je možné zdůraznit či potlačit individuální konstantou v_q , která má podstatný vliv na dynamiku přechodného děje a na stabilitu regulačního obvodu. Zavedením v_q dostaneme

$$E(k) = \frac{1}{2} \sum_{q=0}^n v_q \left(y^{(q)}(k) - w^{(q)}(k) \right)^2 \quad (7.6)$$

Pro výpočet $\frac{\partial E}{\partial a_{ij}}$ pro neuron v i -té vrstvě použijeme rovnici:

$$\frac{\partial E}{\partial a_{ij}} = \frac{\partial E}{\partial u_i} \frac{\partial u_i}{\partial \xi_i} \frac{\partial \xi_i}{\partial a_{ij}} \quad (7.7)$$

kde ξ_i je suma všech výstupů neuronů z j -té vrstvy, u_i výstup z i -té vrstvy neuronové sítě (výstup neuronu může být vyjádřen jako $u = f(\xi_i) = 1/(1+e^{-k\xi_i})$ kde k je strmost sigmoidy. Pro výstupní vrstvu (výstup neuronového regulátoru), použijeme rovnici

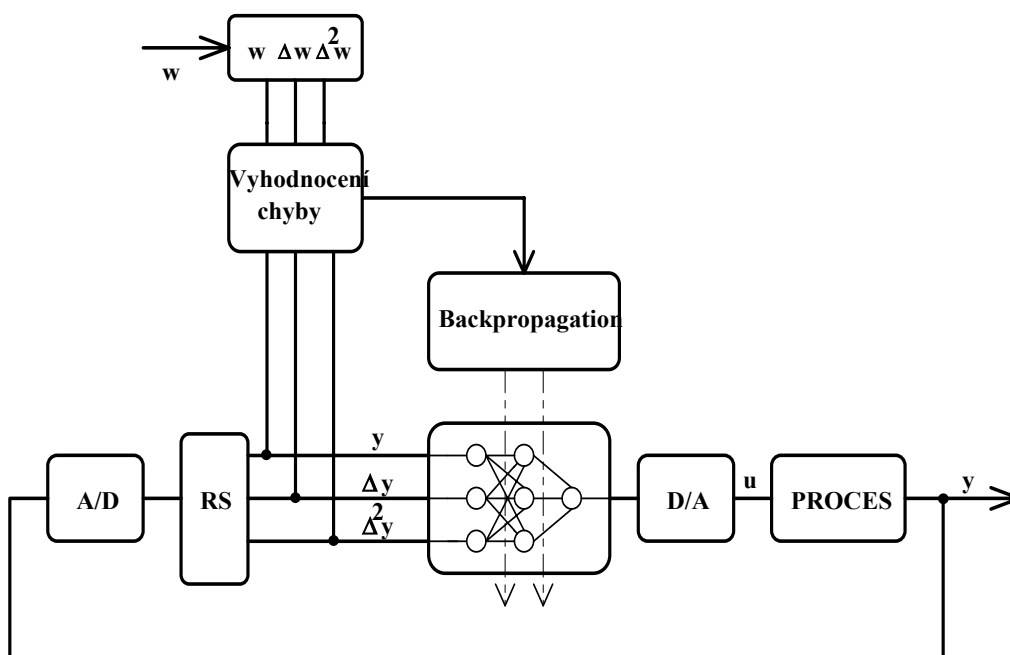
$$\frac{\partial E}{\partial a_{ij}} = \frac{\partial E}{\partial u} \frac{\partial u}{\partial \xi_i} \frac{\partial \xi_i}{\partial a_{ij}} \quad (7.8)$$

Rovnice (7.8) vyjadřuje vliv jednotlivých částí (výstup regulátoru, přenosová funkce (obecně nelineární) neuronu a výstupy neuronů z předchozích vrstev) na chybovou funkci. Klíčovým problémem je určit následující parciální derivaci:

$$\frac{\partial E}{\partial u} = \sum_{q=0}^n \frac{\partial E}{\partial y^{(q)}} \frac{\partial y^{(q)}}{\partial u} \approx \sum_{q=0}^n v_q [y^{(q)}(k) - w^{(q)}(k)] \frac{\partial y^{(q)}}{\partial u} \quad (7.9)$$

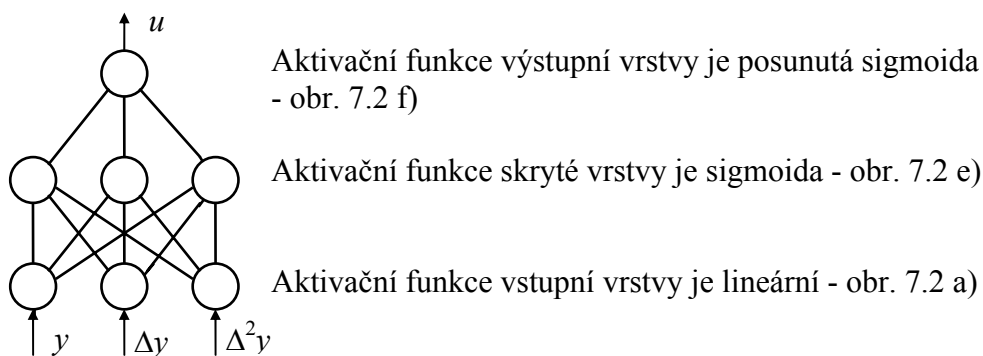
Pokud předpokládáme, že obecně přenosová funkce procesu není známá, pak nelze vypočítat parciální derivaci $\frac{\partial y^{(q)}}{\partial u}$ (citlivostní funkce). Pokud nemůžeme tuto funkci určit, citlivostní funkci systému nahrazujeme jejím znaménkem, které se pro většinu systémů považuje za kladné.

Neuronový regulátor je zapojen v regulačním obvodu podle **Obr. 7.6** (RS je rekonstruktor - zjištění diferencí výstupní veličiny). Neuronový regulátor byl testován na lineárních i nelineárních systémech. Ve vstupní vrstvě byla použita lineární funkce, ve skryté vrstvě sigmoida a ve výstupní vrstvě posunutá sigmoida. Nastavení parametrů neuronové sítě bylo $\alpha=0,05$, $\beta=0,05$, $\zeta=0,3$. Konstanty v_q byly nastavovány individuálně podle chování řízeného procesu.



Obr. 7.6: Řízení procesu pomocí neuronového regulátoru

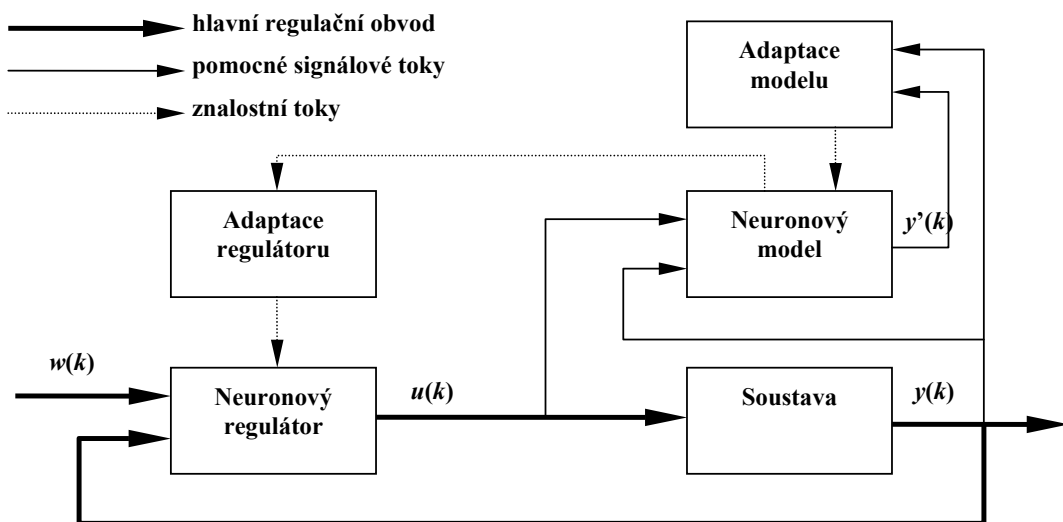
Řídicí algoritmus se skládá ze dvou kroků, v prvním kroku je vypočítán akční zásah, ve druhém, zpětném kroku jsou podle hodnoty chybové funkce upraveny velikosti synaptických vah. Je nutné podotknout, že při tomto způsobu realizace neuronového regulátoru odpadá speciální trénování neuronového regulátoru, který na reálných systémech zpravidla nemůžeme realizovat.



Obr. 7.7: Struktura jednoduchého neuronového regulátoru PID typu

7.5 Neuronové regulátory s modelem

Neuronové regulátory s modelem obsahují regulátor i model realizované pomocí neuronové sítě. Model procesu je nahrazen neuronovým modelem a regulátor neuronovým regulátorem. Adaptace modelu probíhá on-line učením neuronového modelu. Adaptace regulátoru probíhá na základě minimalizace regulační odchylky v následujícím kroku odhadnuté podle predikce výstupu soustavy pomocí neuronového modelu.



Obr. 7.8: Neuronový regulátor s modelem

Kriteriální funkce pro BP učení regulátoru je:

$$G(k) = \frac{1}{2} \cdot (w(k+1) - \hat{y}(k+1))^2 \quad (7.10)$$

Pro aplikaci algoritmu BP potřebujeme:

$$\frac{\partial G(k)}{\partial u(k)} = \frac{\partial G(k)}{\partial \hat{y}(k+1)} \cdot \frac{\partial \hat{y}(k+1)}{\partial u(k)} \quad (7.11)$$

Kde citlivostní funkci $\frac{\partial \hat{y}(k+1)}{\partial u(k)}$ získáme z neuronového modelu.

U tohoto regulátoru je následující časová následnost akcí v jednom kroku výpočtu akčního zásahu regulátoru:

- 1) Změření aktuálního výstupu ze soustavy
- 2) Výpočet akčního zásahu (aktivní režim neuronového regulátoru)
- 3) Vyslání akčního zásahu
- 4) Učení neuronového modelu
- 5) Výpočet predikce výstupu (aktivní režim neuronového modelu)

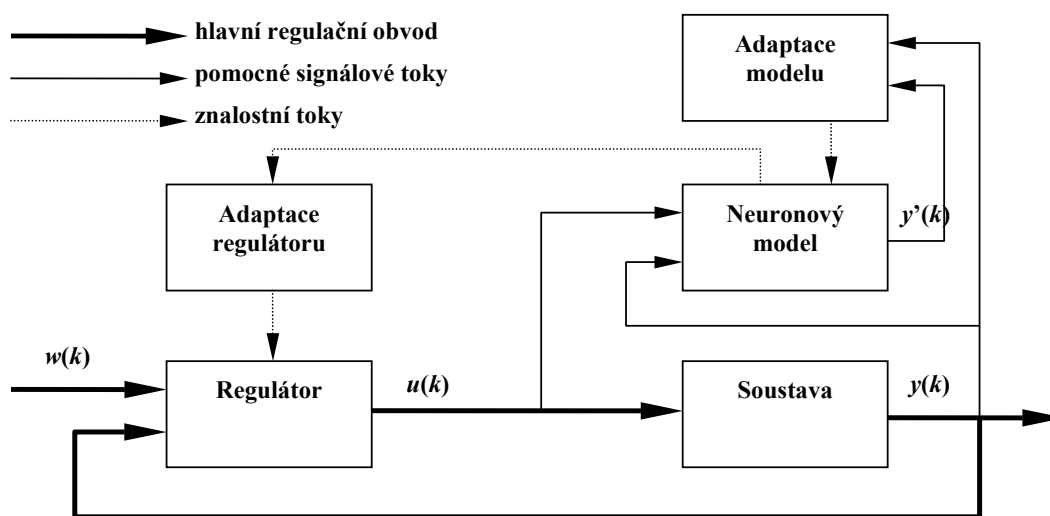
6) Analýza citlivostní funkce neuronového modelu (učení neuronového modelu bez úpravy vah)

7) Učení neuronového regulátoru

Akce 2) a 3) se provedou před akcí 7) z důvodu co nejkratší prodlevy mezi vzorkováním výstupu a vstupu. Akce 2) a 3) se mohou přesunout za akci 7) jen na počítači dostatečně rychlém vzhledem k periodě vzorkování.

7.6 Adaptivní regulátor s neuronovým modelem

Adaptivní regulátor s neuronovým modelem je obměnou předchozího regulátoru, ve kterém je použit klasický regulátor.



Obr. 7.9: Adaptivní regulátor s neuronovým modelem

Po adaptaci neuronového modelu se v krocích výpočtu z neuronového modelu určí diskrétní přenos.

Seznam použité literatury

- [1] ZIEGLER, J. G. – NICHOLS, N. B.: Optimum Settings for Automatic Controllers. In Proceedings of ASME, 1942, pp. 759 – 765.
- [2] VAVŘÍN, P. – JURA, P.: Systémy, procesy a signály II. Skriptum VUT. PC-DIR, Brno, 1996.
- [3] ASTRÖM, K. J. – WITTENMARK, B.: Computer-Controlled Systems. Prentice-Hall Inc, London, 1990.
- [4] PIVOŇKA, P.: Číslíková řídicí technika. Skriptum. VUT, Brno, 2004.
- [5] SCHLEGEL, M.: Exaktní revize Zieglerovy-Nicholsovy frekvenční metody. Automatizace, 43, (2000), č.12, s. 813 – 819.
- [6] KLÁN, P.: Moderní metody nastavení PID regulátorů, Část I: Procesy s přechodovou charakteristikou typu „S”. Automa (2000), č. 9, s. 54 – 57 .
- [7] KLÁN, P.: Moderní metody nastavení PID regulátorů, Část II: Integrovní procesy. Automa (2001), č. 1, s. 52 – 54.
- [8] KLÁN, P. a kol.: Adaptivní PID regulátory s monolitickými mikropočítači. ÚTIA, ČSAV, Praha, 1990.
- [9] BOBÁL, V. a kol.: Praktické aspekty samočinně se nastavujících regulátoru a implementace. VUT Brno, 1999.
- [10] SEBORG, D. E. – EDGAR, T. F. – MELLICHAMP, D. A.: Process Dynamics and Control, John Wiley and Sons, N.Y., 1989.
- [11] SYSEL, M.: Využití delta modelů pro řízení procesů. Disertační práce, UTB, Zlín, 2001.
- [12] SYSEL, M. – BOBÁL, V.: Moderní metody řízení- delta modely. Automa, 7, 2001, č. 12, s. 17-20.
- [13] HORÁČEK, P.: Systémy a modely. Skriptum. ČVUT, Praha, 2001.
- [14] PIVOŇKA, P.: Návrh a realizace standardních PID a PSD regulátorů. Automatizace, 41, 1998, č. 2, 4, 5, s. P11 – P19.
- [15] MALKI, H. a kol.: Fuzzy PID Control of a Flexible-Joint Robot Arm with Uncertainties from Time-Varying Loads. IEEE Transactions on Control Systems Technology, Vol.5, No. 3, May 1997, pp. 371–378.
- [16] KLÁN, P.: Metody zlepšení PI regulace. Automa, 7, 2001, č. 12, s. 4-10.
- [17] TAKAHASHI, Y. – CHAN, C. S. – AUSLANDER, D. M.: Parametereinstellung bei linearen DDC-Algorithmen. Regelungstechnik und Prozess-Datenverarbeitung, 19, 1971, pp. 237–244.
- [18] BENDIX, O.: Fuzzy versus PID. Elektrotechnik, 76, 1993, No. 3, pp. 8 – 13.
- [19] LAMBERT, L. a kol.: Was leistet ein Fuzzy-Regler. Elektronik, 1993, No. 24, pp. 74 – 82.
- [20] DRIANKOV, D. – HELLENDORRN, H. – REINFRANK, M.: An Introduction to Fuzzy Control. Springer-Verlag, 1993.
- [21] KOSKO, B.: Neural Networks and Fuzzy Systems. Prentice-Hall, Inc., 1992.

-
- [22] PIVOŇKA, P. Modelling, adaptive, neuro- and fuzzy-control of coal power plants. In proceedings IFAC symposium Control of power plants and power systems SIPOWER'95. Cancún, Mexico, 1995, pp. 207 – 212.
- [23] PIVOŇKA, P. – ŠÍDLO, M.: Fuzzy PID Controllers. BULLETIN for Studies and Exchanges on Fuzziness and its Applications, BUSEFAL, Toulouse, France, No. 74, 1998, pp. 93–97, ISSN 0296-3698.
- [24] PIVOŇKA, P.: Fuzzy PI/PD/PID regulátory. Automatizace, 41 (1998), č. 5–8, s. P20 – P30.
- [25] PIVOŇKA, P. – FINDURA, M.: Alternativní návrhy fuzzy regulátorů. Automatizace, 41 (1998). č. 10 – 12, 42 (1999), č. 1, s. P31 – P37.
- [26] PIVOŇKA, P. : Physical Background of Fuzzy PI and PD Controller. In proceedings of The Eighth International Fuzzy Systems Association World Congress Taipei, Taiwan, 1999, pp. 635-639.
- [27] PIVOŇKA, P.: Analysis and Design of Fuzzy PID Controller Based on Classical PID Controller Approach. Advances in Soft Computing, Physica Verlag, Springer, Heidelberg, 2000, pp. 186-199, ISBN 3-7908-1327-3.
- [28] PIVOŇKA, P. – ADAMČÍK, T.: On-Line Trained Neural Nets in Real-Process Control. Neural Network World, Vol.9. No. 1–2, 1999, pp. 75-89, ISSN 1210-0552.
- [29] VYCHODIL, H.: Neuronové regulátory. Teze disertační práce. VUT FEI, Brno, 2000.
- [30] ISERMANN, R.: Zur Anwendung der Fuzzy-Logik in der Regelungstechnik. Automatisierungstechnische Praxis, 1996, No. 11, pp. 24 –36.
- [31] PIVOŇKA, P. – ŽIŽKA, J.: Neural Controllers in Real-Process Control. Advanced Manufacturing Forum, Scitec Publications, Switzerland, vol. 1, 1996, pp. 255 –264.
- [32] B&R Documentation. Automation Studio, Automation Net, PVI, 1997–2002, <http://www.br-automation.com>.
- [33] ŠVANCARA, K. – PIVOŇKA, P.:The Real-Time Communication Between MATLAB and the Real Process Controlled by PLC. In the 7th International Research/Expert Conference "Trends in the Development of Machinery and Associated Technology" TMT 2003, Lloret de Mar, Barcelona, Spain, pp. 1077 – 1080, ISBN 9958–617–18–8
- [34] PIVOŇKA, P.:Fyzikální pohled na nastavování parametrů PID regulátoru metodou Zieglera a Nicholse. Automa, ISSN 1210–9592, 2003, roč. 9, č. 3, s. 70 – 75.
- [35] VEDRAL, J. – FISCHER, J.: Elektronické obvody pro měřicí techniku. Ediční středisko ČVUT, Praha, 1999.
- [36] MARŠÍK, J. – STREJC, V.: Application of identification-free algorithms for adaptive control. Automatica, 25, 1989, pp. 273-277.
- [37] KLÁN, P. – MARŠÍK, J. – GÖRNER, V. – VALÁŠEK, P.: Adaptivní PID regulátory s monolitickými mikropočítači. Praha, ÚTIA ČSAV, 1990.
- [38] ASTRÖM, K. J. – HÄGGLUND, T.: Automatic Tuning of PID Controllers. North Carolina, Instrument Society of America, Research Triangle Park, 1988.
- [39] ASTRÖM, K. J. – HÄGGLUND, T.: Automatic tuning of simple regulators with specifications on phase and amplitude margins. Automatica, 20, 1984, 645-651.

- [40] ASTRÖM, K. J. – HÄGGLUND, T.: PID Controllers: Theory, Design and Tuning. 2nd Edition, North Carolina, Instrument Society of America, Research Triangle Park, 1995.
- [41] LANDAU, I., D.: Adaptive control - the model reference approach. New York, Marcel Dekker, 1979.
- [42] IOANNOU, P., A.: Robust adaptive control. New Jersey, Englewood Cliffs, Prentice Hall, 1996.