**Soustava tří nádob**

Vychází se z kaskádního propojení tří nádob, kde každá z nádob lze popsat následující diferenciální rovnicí (1)

, (1)

kde S je příčná plocha hladiny nádoby [m2],

Sv je plocha výtoku z nádoby [m2],

Q(t) je přítok do nádoby [m3\*s-1],

v(t) je výtoková rychlost z nádoby [m\*s-1],

h(t) je hladina kapaliny v nádobě [m].

Výtokovou rychlost z nádoby popisuje Torriceliho vzorec (2)

(2)

Při kaskádním propojení více nádob se vztah (2) přepíše s ohledem na výšku hladiny kapaliny v sousední nádobě (3)

(3)

Znaménková funkce signum zajišťuje, že při záporném rozdílu hladin *h*1 a *h*2 se změní i směr rychlosti. Absolutní hodnota rozdílu hladin pod odmocninou zamezuje překročení definičního oboru odmocniny v .

Soustavu tří nádob popisuje soustava tří nelineárních diferenciálních rovnic (4)

Nejprve bude vytvořen model v Simulinku. První rovnici z (4) ukazuje obr. 1.



**Obr. 1: Model první nádoby v Simulinku.**

Model druhé nádoby je totožný s modelem první nádoby. U třetí nádoby je vynechán blok absolutní hodnoty, signum a chybí zde vstup hladiny v následující nádrži. Všechny tři modely jsou zamaskovány a parametrizovány jak ukazuje následující obrázek.



**Obr. 3: Model soustavy tří nádob v Simulinku.**



**Obr. 4: Průběh hladin v jednotlivých nádobách v čase.**

Soustavu rovnic (4) je třeba linearizovat v pracovním bodě. Budou provedeny dva způsoby linearizace. První způsob popisuje linearizaci pomocí Taylorova rozvoje v daném pracovním bodě (5)

(5)

Poněvadž je složité a pracné derivovat funkce signum a absolutní hodnoty, je třeba rozhodnout o směru rychlosti toku kapaliny mezi nádobami. Počáteční hladiny v jednotlivých nádobách mají sestupnou tendenci, tudíž směr rychlosti toku kapaliny bude vždy od první nádoby k poslední. Navíc tomu nasvědčuje i skutečnost, že kapalina vtéká do první nádoby a vytéká z poslední, třetí, nádoby. Rovnice (4) se tím pádem zjednoduší na tvar (6)

Tyto rovnice se linearizují v pracovním bodě pomocí Taylorova rozvoje (7)

Soustava diferenciálních rovnic po výpočtu Taylorova rozvoje

Po dosazení počátečního stavu (5) a parametrů soustavy se získá linearizovaný model (8)

V modelu bude potřeba stále odečítat počáteční hodnoty, zadané vztahem (5), protože model soustavy musí pracovat v okolí pracovního bodu (9).

Výstup soustavy poté bude (10):

Stavový popis soustavy tří nádob (11).

Následující obrázek ukazuje schéma v Simulinku po provedení linearizme v pracovním bodě



**Obr. 5: Linearizace Taylorovým rozvojem v pracovním bodě.**

Při porovnání průběhu výšky hladiny ve třetí nádobě u matematického modelu a u linearizovaného systému lze ukázat, že v pracovním bodě dané křivky leží na sobě. Čím je přítok vody do první hladiny *Q* dále od pracovního bodu *Q*0 = 2, tím se zvětšuje rozdíl mezi oběma modely, viz obr. 6.



**Obr. 6: Linearizace Taylorovým rozvojem v pracovním bodě.**

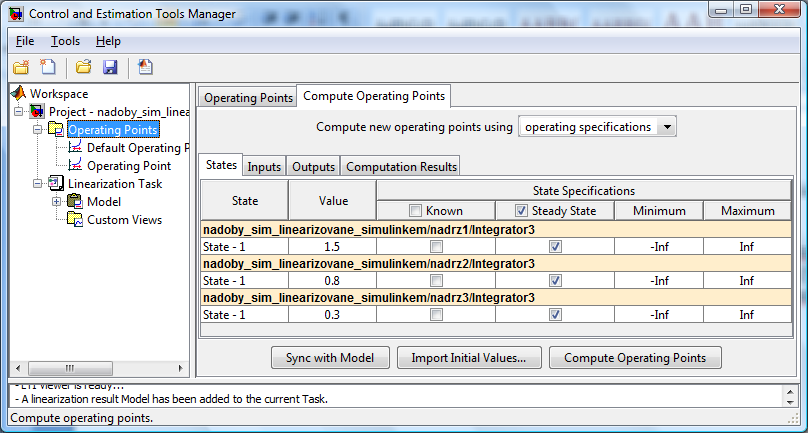
Druhý způsob linearizace se provede v programovém prostředí MATLAB&Simulink. Vyjde se ze zapojení na obr. 3.



**Obr. 7: Linearizace nástrojem z Control Systém toolboxu.**

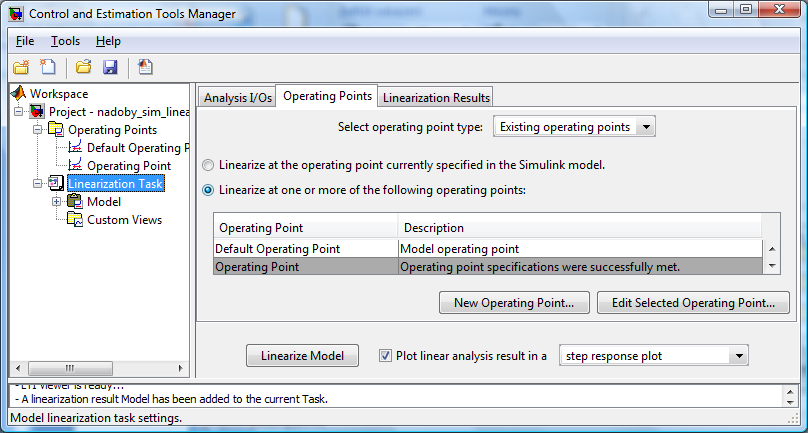
Stručný postup linearizme pomocí nástroje z Control Systém toolboxu je následující:

* Pustí se model (viz obr. 3) s daným pracovním bodem (5).
* Pravým kliknutím myši na linku Q se vybere z kontextové nabídky Linearization Points 🡪 Input Point. Na lince h3 se vybere Linearization Points 🡪 Output Point (viz obr. 7)
* Spustí se menu linearizace: menu Tools 🡪Control Design 🡪 Linear Analysis
* Ve stromové nabídce se vybere položka Operating Points 🡪 Compute Operating Points a nechá se zaškrtnutá položka Steady state (viz obr. 8).



**Obr. 8: Menu linearizace.**

* Stiskne se tlačítko Compute Operating Points. Výsledek jde vidět ve stromové nabídce Operating Point.
* Ve stromové nabídce Linearization Task v záložce Operating Points se vybere položka Linearize at one or more …🡪Operating Point a klikne se na tlačítko Linearize Model viz obr. 9.



**Obr. 9: Výpočet matic linearizovaného systému.**

* Ve stromové nabídce Model se objeví matice linearizovaného systému. Ty lze nyní exportovat do workspace Matlabu.

Výsledné matice linearizovaného systému (12) se samozřejmě liší od systému, linearizovaného taylorovým rozvojem (11).

Porovnáním obou linearizovaných modelů spolu s matematickým modelem (viz obr. 10) lze dospět k závěru, že pro tento případ je lepší použít systém, linearizovaný pomocí Taylorova rozvoje.



**Obr. 10: Porovnání jednotlivých linearizací v pracovním bodě.**

**Kvadraticky optimální regulace pro lineární systémy (LQ regulace)**

Kvadraticky optimální regulace lineárních systémů nebo taky zkráceně LQ (linear quadratic) regulace lze rozdělit na dvě základní úlohy. První z nich se zabývá optimálním přechodem ze stavu *x*0 do počátečního stavu 🡪 kvadraticky optimální regulace (optimální kompenzace poruch). Druhá úloha se zabývá kvadraticky optimálním sledováním, kdy se vyžaduje, aby výstup soustavy sledoval požadovanou (nenulovou) referenční trajektorii. V obou případech se předpokládá, že stav řízeného systému je plně měřitelný [1].

Vzhledem k stavovému popisu soustav tří nádob, kde je systém popsán v pracovním bodě (9) bude vhodné zvolit první způsob řízení, kdy se bude optimálně minimalizovat rozdíl hladiny mezi skutečnou hodnotou *h*3(t) a výchozí hodnotou *h*30. Systém popsaný rovnicemi (11)

má v pracovním bodě

podle vztahu (9)

minimalizované stavové veličiny a akční zásah je nulový.

Uvažujme kvadratické kritérium optimality

kde Q(t), t = 0, . . . ,N je posloupnost pozitivně semidefinitních matic a R(t), t = 0, . . . ,N−1

je posloupnost pozitivně definitních matic. N je horizont optimalizace.

Z tohoto kritéria po odvození dostaneme Riccatiho rovnici



Kde K(t) je časově proměnné Kalmanovo zesílení



Tuto algebraickou Riccatiho rovnici lze řešit pohodlně pomocí jednoho ze dvou příkazů v Matlabu.

Prvním z nich je příkaz

[P,L,K] = care(A,B,Q,R);

Kde A, B jsou stavové matice time invariantní soustavy,  
Q je regulární pozitivně semidefinitní váhovací matice stavů o patřičném rozměru, odpovídajícím počtu stavů systému,  
R je regulární pozitivně definitní váhovací matice řízení, odpovídající počtu vstupů sytému,  
K je kalmanovo zesílení,  
L jsou vlastní čísla matice dynamického systému se zavedenou stavovou zpětnou vazbou,  
P je výsledek řešení Riccatiho rovnice.

Příkaz care může mít i více parametrů, jako například váhovací matici S vážící součiny stavu a vstupu.

Druhým příkazem je příkaz

[K,S,e] = lqr(A,B,Q,R)

Kde K je kalmanovo zesílení,  
e jsou vlastní čísla matice dynamického systému se zavedenou stavovou zpětnou vazbou,  
S je výsledek řešení Riccatiho rovnice.

Vstupní parametry jsou totožné jako u předchozího příkazu.

Syntéza takto navrhovaného regulátoru se omezí na správné nastavení chovacích matic s ohledem na amplitudovou a fázovou bezpečnost.

Bude se hledat ustálené řešení Riccatiho rovnice.