

Univerzita Pardubice
Fakulta elektrotechniky a informatiky

Modelování a simulace dynamických systémů

Studijní opora

Ing. Daniel Honc, Ph.D.



Modelování a simulace dynamických systémů

Téma 1: Úvod do modelování a identifikace dynamických systémů

Studijní cíl

Seznámit studenty se základními pojmy a principy modelování, identifikace a simulace dynamických systémů.

Doba nutná k nastudování

2 hodiny

Klíčová slova

Modelování, identifikace, simulace, model dynamického systému, stavový popis, přenos

1 Základní pojmy a principy modelování a simulace

Při **modelování** vytváříme **matematický model** (obraz) zkoumaného reálného systému (originálu). Lze vytvářet také fyzické modely – například model zařízení v určitém měřítku, ale my se budeme zabývat pouze abstraktními modely vyjádřenými pomocí matematických rovnic. Existují dva krajní přístupy, jak lze postupovat. V případě **analytického přístupu** využíváme principu geometrické a fyzikální podobnosti a model konstruujeme například na základě bilance hmoty a/nebo energie. Teoreticky lze parametry modelu dohledat v tabulkách nebo odhadnout myšlenkovým experimentem, takže není nutné, aby reálný systém existoval a my jsme na něm prováděli experimenty. Tímto způsobem lze vytvořit model i fyzicky neexistujícího zařízení například ve fázi návrhu a konstrukce. Získáváme takzvaný model ve tvaru **bílé skříňky** (white box). Dalším krokem může být zpřesnění modelu na základě experimentů – přesnější určení parametrů modelu a případně změna struktury modelu, pokud chování modelu neodpovídá skutečnosti. Pokud jsou některé části modelu těžko popsitelné bilancemi a nemáme dostatečné informace nebo znalosti pro jejich popis, lze tyto části nahradit matematickými aproximacemi a hovoříme o takzvané **šedé skříňce** (gray box). V krajním případě lze pomocí **experimentálního přístupu** popsat veškeré chování matematickou aproximací s využitím experimentálních dat – využíváme pouze matematické podobnosti a vytváříme modely například ve tvaru diferenciálních nebo diferenčních rovnic (spojitých nebo diskrétních přenosů či stavových popisů) a hovoříme o takzvané **černé skříňce** (black box).

Tab. 1 – Srovnání analytického a experimentálního přístupu

	výhody	nevýhody
Analytický přístup	<ul style="list-style-type: none"> - model respektuje geometrickou a fyzikální podstatu systému – struktura modelu vyjde z bilancí - veličiny i parametry mají fyzikální význam - model popisuje systém s minimem parametrů v široké oblasti - teoreticky nepotřebujeme experimentální data 	<ul style="list-style-type: none"> - náročné na znalosti - může být příliš rozsáhlý, složitý - nemusíme jej vůbec umět vytvořit
Experimentální přístup	<ul style="list-style-type: none"> - méně náročné na znalosti - strukturu modelu je možné volit - algoritmus tvorby modelu je lépe automatizovatelný - lepší aproximace v oblasti dat 	<ul style="list-style-type: none"> - nutnost kvalitních experimentálních dat - strukturu modelu je nutné volit - platnost v omezené oblasti – extrapolace bývají horší

Proces modelování nám pomáhá porozumět chování systému, má význam při návrhu a konstrukci zařízení, provádění případových studií, testování situací, které nechceme nebo nelze na reálném zařízení provést, pro školení zaměstnanců a při výuce.

Systém je definován jako množina prvků a vazeb mezi nimi a mající jako celek určité vlastnosti. Systém může být pro zvýšení přehlednosti rozdělen na několik **subsystémů** se vzájemnými **vazbami**, čímž může být zavedena hierarchické struktura s různými úrovněmi modelu. **Okolí systému** je množina prvků, která nepatří do systému, má s ním vazby, ale jeho chování systém neovlivňuje. Struktura systému souvisí s vnitřním uspořádáním a příslušnými vazbami. Jednotlivým vazbám odpovídají vstupní, stavové a výstupní proměnné. V popisu se mohou vyskytovat závislé a nezávislé proměnné, parametry (konstanty v rámci jednotlivých simulací nebo jejich částí) a konstanty. Při tvorbě modelu přijímáme určité předpoklady a zavádíme různá zjednodušení. Předpokladem může být například, že se nemění teplota kapaliny, a tudíž její hustota a tím lze provést bilanci objemových průtoků místo hmotnostních průtoků. Zjednodušení souvisí často se zavedením takzvaných soustředěných parametrů místo rozložených – například nepředpokládáme, že je teplota kapaliny závislá na prostorových veličinách, ale je pouze funkcí času, čímž se z popisu pomocí parciálních diferenciálních rovnic dostaneme k popisu pomocí obyčejných diferenciálních rovnic. Dalším zjednodušením může být předpoklad, že se vlastnosti systému nemění v čase a místo nestacionárního (t-variantního) systému použijeme popis stacionární (t-invariantní) s konstantními parametry modelu. Dalším zjednodušením může být lineární aproximace popisu – použijeme lineární model místo nelineárního.

Simulaci chování modelu provádíme na počítači numerickým způsobem. U obyčejných diferenciálních rovnic se jedná o numerickou integraci derivovaných veličin za předpokladu znalosti počátečních podmínek. Zápis diferenciálních rovnic může být proveden buďto v grafickém programovacím jazyce (např. Simulink) nebo v textovém programovacím jazyce

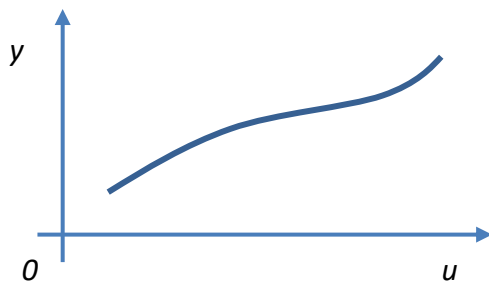
(např. skript nebo funkce MATLABu). Při simulaci časově diskrétního modelu se využívá postupného (rekurzivního) výpočtu soustavy diferenčních rovnic. Ověřením platnosti modelu (**verifikací**) rozumíme proces simulace chování modelu a jeho porovnání s experimentálními výsledky – používáme jiná data než ta, co byla použita pro dohledání struktury a parametrů modelu.

2 Formy matematických modelů

Pokud by byl výstup systému závislý pouze na vstupu, tak hovoříme o takzvaném **statickém systému** (popisu statického chování stabilního dynamického systému v ustáleném stavu)

$$y = f(u) \quad (1)$$

a lze znázornit statickou charakteristiku jako závislost výstupu systému na jeho vstupu. Zesílení systému je směrnice tečny ke statické charakteristice v daném pracovním bodu (u nelineárních systémů není konstantní, ale mění se s pracovním bodem).



Obr. 1 – Statická charakteristika nelineárního systému

Jestliže nejsou výstupy určeny pouze vstupy, ale závisí také na čase, hovoříme o dynamickém systému, který lze vektorově zapsat pomocí vektorové stavové a výstupní rovnice (**nelineární stavový popis**)

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) &= \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) \end{aligned} \quad (2)$$

$$\mathbf{u}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_{nu}(t) \end{bmatrix}, \mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{nx}(t) \end{bmatrix}, \mathbf{y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_{ny}(t) \end{bmatrix} \quad (3)$$

kde \mathbf{u} jsou veličiny vstupní, \mathbf{x} stavové a \mathbf{y} výstupní. Vektorová stavová rovnice uvádí závislost změn (derivací) stavových veličin na stavových veličinách a vstupech. Výstupní rovnice popisuje závislost výstupů na stavových veličinách a vstupech. Pro vyřešení těchto rovnic je zapotřebí znát počáteční podmínky – hodnoty stavových veličin v čase 0

$$\mathbf{x}(0) = \begin{bmatrix} x_1(0) \\ x_2(0) \\ \vdots \\ x_{nx}(0) \end{bmatrix} \quad (4)$$

Tento popis předpokládá, že se vlastnosti systému v čase nemění – jedná se o takzvaný stacionární (t-invariantní systém) – parametry modelu jsou konstantní. Rovnice (2) obsahuje obecně dvě nelineární funkce \mathbf{f} a \mathbf{g} . V případě lineárního popisu, lze tyto funkce vyjádřit maticově (**lineární stavový popis**) jako

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{aligned} \quad (5)$$

Rovnice (5) popisuje vícerozměrný systém (MIMO) s několika vstupy a výstupy. V případě jednorozměrného systému (SISO) jsou místo matic \mathbf{B} a \mathbf{C} vektory \mathbf{b} a \mathbf{c} a místo matice \mathbf{D} je skalár d

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) \\ y(t) &= \mathbf{c}\mathbf{x}(t) + du(t) \end{aligned} \quad (6)$$

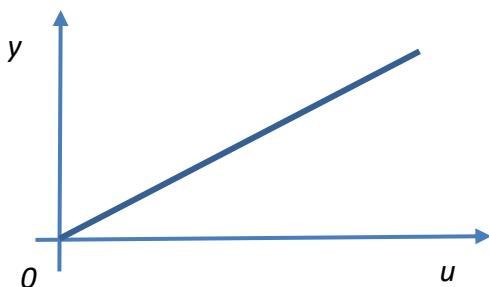
Pomocí Laplaceovy transformace, za předpokladu, že $d=0$ a nulových počátečních podmínek lze z popisu (5) získat přenosovou matici nebo z popisu (6) **spojitý obrazový přenos** (vstupně-výstupní popis)

$$G(s) = \frac{Y(s)}{U(s)} = \mathbf{c}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} = \frac{b_ms^m + \dots + b_1s + b_0}{a_ns^n + \dots + a_1s + a_0} \quad (7)$$

Stavový popis (5) a (6) nazýváme vnitřním popisem, protože obsahuje informace o vnitřních proměnných – stavových proměnných (stavech) systému. Jedná se o soustavu lineárních diferenciálních rovnic zapsanou maticově, jejichž počet je roven řádu systému. Matice \mathbf{A} je čtvercová matice o rozměru odpovídajícím řádu systému (pro soustavu první řádu se jedná o skalár a – rovnice není vektorová, ale skalární). Přenos (7) a odpovídající diferenciální rovnice řádu rovnému řádu systému vyjadřuje vstupně-výstupní (vnější popis) závislosti výstupů na vstupech systému. Zesílení systému je

$$k = \frac{b_0}{a_0} \quad (8)$$

a statické charakteristika je přímka se směrnici k procházející počátkem souřadného systému



Obr. 2 – Statická charakteristika lineárního systému

Doposud byly uvažovány časově spojité modely. Model (5) a (6) lze převést do časově diskrétního stavového popisu

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{M}\mathbf{x}(k) + \mathbf{N}\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) \end{aligned} \quad (9)$$

a

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{M}\mathbf{x}(k) + \mathbf{n}\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{c}\mathbf{x}(k) + \mathbf{d}\mathbf{u}(k) \end{aligned} \quad (10)$$

Pomocí Z-transformace, za předpokladu, že $d=0$ a nulových počátečních podmínek lze z popisu (9) získat diskrétní přenosovou matici nebo z popisu (10) **diskrétní obrazový přenos**

$$G(z) = \frac{Y(z)}{U(z)} = \mathbf{c}(z\mathbf{I} - \mathbf{M})^{-1}\mathbf{n} = \frac{b_1z^{-1} + b_2z^{-2} + \dots + b_nz^{-n}}{1 + a_1z^{-1} + a_2z^{-2} + \dots + a_nz^{-n}} \quad (11)$$

Zesílení systému (11) je

$$k = \frac{b_1 + b_2 + \dots + b_n}{1 + a_1 + a_2 + \dots + a_n} \quad (12)$$

Koeficienty a a b v modelu (7) a (11) se číselně liší – jednou se jedná o koeficienty diferenciální rovnice (spojitého přenosu) a podruhé o koeficienty diferenční rovnice (diskrétního přenosu).

3 Bilanční rovnice pro analytické modely

Při tvorbě analytických modelů budeme vycházet z bilancí energie, hmoty a z rovnováhy sil.

Bilanci energie (zákon zachování energie) za určitou jednotku času, lze zapsat jako

$$E_{\text{IN}} = E_{\text{OUT}} + E_{\text{AK}}, \quad (13)$$

kde E_{IN} je energie, která do systému vstoupila, J

E_{OUT} je energie, která ze systému vystoupila, J

E_{AK} je energie, která se v systému akumulovala, J .

Budeme pracovat se změnami energie za nekonečně malý časový úsek dt , čímž se rovnice (13) stane bilančí výkonů

$$P_{\text{IN}} = P_{\text{OUT}} + P_{\text{AK}}, \quad (14)$$

kde P_{IN} je výkon, který do systému vstoupil, W

P_{OUT} je výkon, který ze systému vystoupil, W

P_{AK} je výkon, který se v systému akumuloval, W .

Obdobným způsobem lze zapsat **Bilanci hmoty** (zákon zachování hmoty) za určitou jednotku času

$$m_{\text{IN}} = m_{\text{OUT}} + m_{\text{AK}}, \quad (15)$$

kde m_{IN} je hmota, která do systému vstoupila, *kg*

m_{OUT} je hmota, která ze systému vystoupila, *kg*

m_{AK} je hmota, která se v systému akumulovala, *kg*.

Opět budeme pracovat se změnami hmoty za nekonečně malý časový úsek dt , čímž se rovnice (15) stane bilancí hmotnostních průtoků

$$M_{\text{IN}} = M_{\text{OUT}} + \frac{dm}{dt}, \quad (16)$$

kde M_{IN} je vstupní hmotnostní průtok, *kg.s⁻¹*

M_{OUT} je výstupní hmotnostní průtok, *kg.s⁻¹*

$\frac{dm}{dt}$ je hmotnostní průtok, který se v systému akumuloval, *kg.s⁻¹*.

U modelů mechanických systémů budeme pro jednoduchost vycházet z **Newtonových zákonů**. Podle **prvního Newtonova zákona** platí „jestliže na těleso (hmotu) nepůsobí žádné vnější síly, nebo výslednice sil je 0, pak těleso setrvává v klidu nebo v rovnoměrném přímočarém pohybu“. **Druhý Newtonův zákon** zní „jestliže na těleso (hmotu) působí síla, pak se těleso pohybuje zrychlením, které je přímo úměrné působící síle a nepřímo úměrné hmotnosti tělesa“

$$F = ma, \quad (17)$$

kde F je síla způsobující pohyb, *N*

m hmotnost tělesa, *kg*

a zrychlení tělesa, *m.s⁻²*.

4 Použitá literatura

BALÁTĚ, J. 2004. *Automatické řízení*. Praha: BEN - technická literatura. ISBN 80-7300-148-9.

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

Seznam zkratk

SISO Single-Input Single-Output, jednorozměrný systém

MIMO Multiple-Input Multiple-Output, vícerozměrný systém

Rejstřík

analytický přístup, 1
balance energie, 5
balance hmoty, 5
diskrétní obrazový přenos, 5
druhý Newtonův zákon, 6
experimentální přístup, 1
lineární stavový popis, 4
matematický model, 1
modelování, 1
nelineární stavový popis, 3
Newtonovy zákony, 6
okolí systému, 2
první Newtonův zákon, 6
regulace, 1
simulace, 2
spojitý obrazový přenos, 4
statický systém, 3
subsystém, 2
systém, 2
vazba, 2
verifikace, 3

Modelování a simulace dynamických systémů

Téma 2: Model varné konvice a model průtokového ohříváče

Studijní cíl

Seznámit studenty s principy modelování tepelných systémů a vytvořit model varné konvice a model průtokového ohříváče.

Doba nutná k nastudování

2 hodiny

Klíčová slova

Model, tepelné systémy, varná konvice, průtokový ohříváč

1 Principy modelování tepelných systémů

Při modelování tepelných systémů budeme provádět bilance energie. Pokud tělesu (pevné látce nebo kapalině) o hmotnosti m přivedeme **teplo** (energií), tak zvýšíme jeho teplotu. Naopak pokud teplo odvedeme, jeho teplota se sníží. Tento proces lze popsat rovnicí

$$Q = mc\Delta T, \tag{1}$$

kde Q je teplo, J

m je hmotnost, kg

c je měrná tepelná kapacita, $J.kg^{-1}.K^{-1}$

ΔT je změna teploty, K .

Sdílení tepla probíhá vedením (kondukcí) v pevných látkách, prouděním (konvekcí) v pohybujících se kapalinách nebo plynech nebo sáláním (zářením, radiací) prostřednictvím elektromagnetického vlnění.

Pro **přestup tepla** mezi dvěma tělesy z pevné látky nebo mezi tělesem a kapalinou budeme uvažovat vztah

$$\frac{dQ}{dt} = \Phi = \alpha S(T_1 - T_2), \tag{2}$$

kde Φ je **tepelný tok** (výkon), W

α je přestupní koeficient, $W.m^{-2}.K^{-1}$

S je přestupní plocha, m^2

T_1 je teplota jednoho tělesa, K

T_2 je teplota druhého tělesa, K .

Přestupní koeficient budeme pro konkrétní případ určovat experimentálně.

Výkon přivedený nebo odvedený proudem kapaliny lze vyjádřit jako

$$\frac{dQ}{dt} = \Phi = McT, \quad (3)$$

kde Φ je tepelný tok (výkon), W

M je hmotnostní průtok kapaliny, $kg.s^{-1}$

c je měrná tepelná kapacita kapaliny, $J.kg^{-1}.K^{-1}$

T je teplota kapaliny, K .

V tělese bude docházet k **akumulaci energie** (tepla) podle vztahu

$$\frac{dQ}{dt} = \Phi = mc \frac{dT}{dt}, \quad (4)$$

kde Φ je tepelný tok (výkon), W

m je hmotnost tělesa, kg

c je měrná tepelná kapacita tělesa, $J.kg^{-1}.K^{-1}$

T je teplota tělesa, K .

Bilanci energie (zákon zachování energie) za určitou jednotku času, lze zapsat jako

$$E_{IN} = E_{OUT} + E_{AK}, \quad (5)$$

kde E_{IN} je energie, která do systému vstoupila, J

E_{OUT} je energie, která ze systému vystoupila, J

E_{AK} je energie, která se v systému akumulovala, J .

Budeme pracovat se změnami energie za nekonečně malý časový úsek dt , čímž se rovnice (5) stane bilancí výkonů

$$P_{IN} = P_{OUT} + P_{AK}, \quad (6)$$

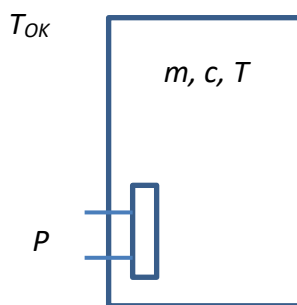
kde P_{IN} je výkon, který do systému vstoupil, W

P_{OUT} je výkon, který ze systému vystoupil, W

P_{AK} je výkon, který se v systému akumuloval, W .

2 Model varné konvice

Předpokládejme situaci, kdy elektrickým topením ohříváme kapalinu uvnitř varné konvice. Budeme uvažovat pouze tepelnou kapacitu kapaliny – zanedbáme tepelnou kapacitu topení (odporové spirály) i stěn varné konvice. V kapalině budeme uvažovat dokonalé míchání – zavedeme předpoklad systému se soustředěnými parametry – s jednou „charakteristickou“ teplotou kapaliny. Ve skutečnosti by teplota v okolí spirály byla vyšší než u stěn a podobně nahoře konvice by byla vyšší než u dna. Museli bychom, ale řešit parciální diferenciální rovnice pro rozložení teploty v prostoru (systém s rozloženými parametry) a čase. Předpokladem systému se soustředěnými parametry zůstane jedinou nezávislou proměnnou čas, závisle proměnnou teplota a systém popíšeme pomocí obyčejné diferenciální rovnice. Dále budeme uvažovat výměnu tepla s okolím – pokud bude teplota kapaliny vyšší než okolí, tak hovoříme o tepelných ztrátách. V opačném případě by se kapalina vlivem okolního prostředí ohřívala. Také předpokládáme, že hmotnost kapaliny je uvnitř konvice konstantní (nedochází k jejímu odpařování vlivem varu).



Obr. 1 – Schéma varné konvice

Bilanci výkonů (6) pro varnou konvici můžeme zapsat jako

$$P = \alpha S(T - T_{OK}) + mc \frac{dT}{dt}, \quad (7)$$

- kde P je výkon elektrického topení, W
 α je přestupní koeficient pro ztráty tepla, $W.m^{-2}.K^{-1}$
 S je přestupní plocha pro ztráty tepla, m^2
 T je teplota kapaliny ve varné konvici, $^{\circ}C$
 T_{OK} je teplota okolí, $^{\circ}C$
 m je hmotnost kapaliny ve varné konvici, kg
 c je měrná tepelná kapacita kapaliny, $J.kg^{-1}.K^{-1}$.

Člen $\alpha S(T - T_{OK})$ odpovídá ztrátovému výkonu a člen $mc \frac{dT}{dt}$ vyjadřuje akumulovaný výkon v kapalině.

Simulaci chování modelu – řešení rovnice (7) pro konkrétní počáteční teplotu kapaliny uvnitř konvice v čase 0 (počáteční podmínku diferenciální rovnice), teplotu okolí a výkon topení lze provést numericky integrací veličiny $\frac{dT}{dt}$ vyjádřené z rovnice (7)

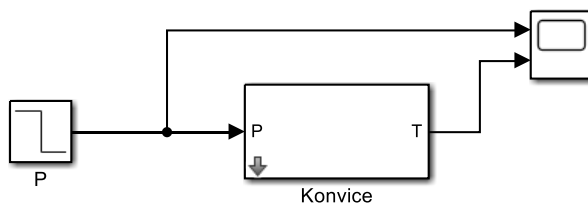
$$T = \int_0^t \frac{dT}{dt} dt = \int_0^t \frac{P - \alpha S(T - T_{OK})}{mc} dt \quad (8)$$

Parametry modelu jsou uvedeny v tabulce 1.

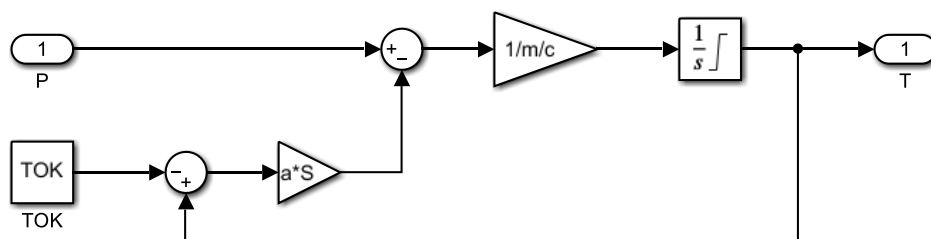
Tab. 1 – Parametry modelu varné konvice

Parametr	Značka	Hodnota	Jednotka
Přestupní koeficient	a	100	$W.m^{-2}.K^{-1}$
Přestupní plocha	S	0,08	m^2
Teplota okolí	T_{OK}	20	$^{\circ}C$
Hmotnost kapaliny	m	1	kg
Měrná tepelná kapacita kapaliny	c	4120	$J.kg^{-1}.K^{-1}$
Počáteční teplota kapaliny	T_0	16	$^{\circ}C$

Pro simulaci sestavíme model v Simulinku

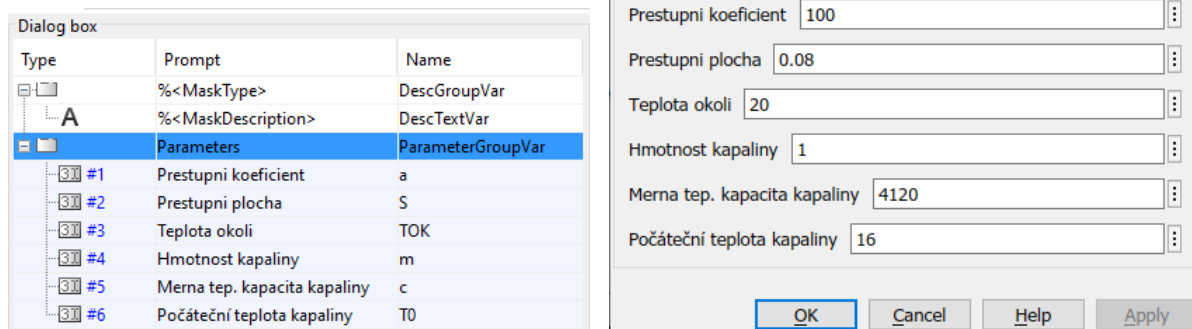


Obr. 2 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)



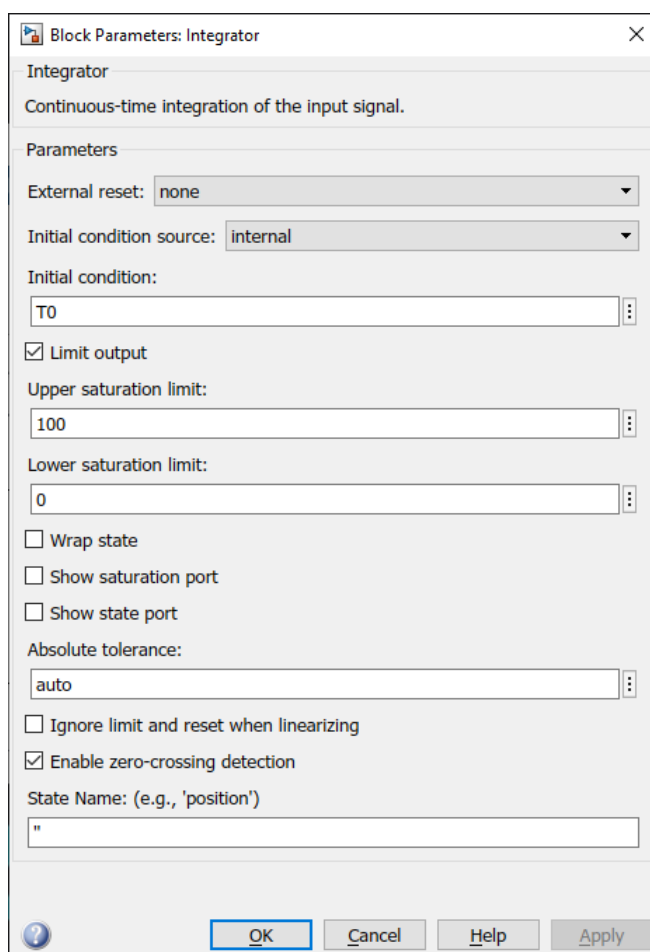
Obr. 3 – Simulační schéma subsystému konvice (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska subsystému



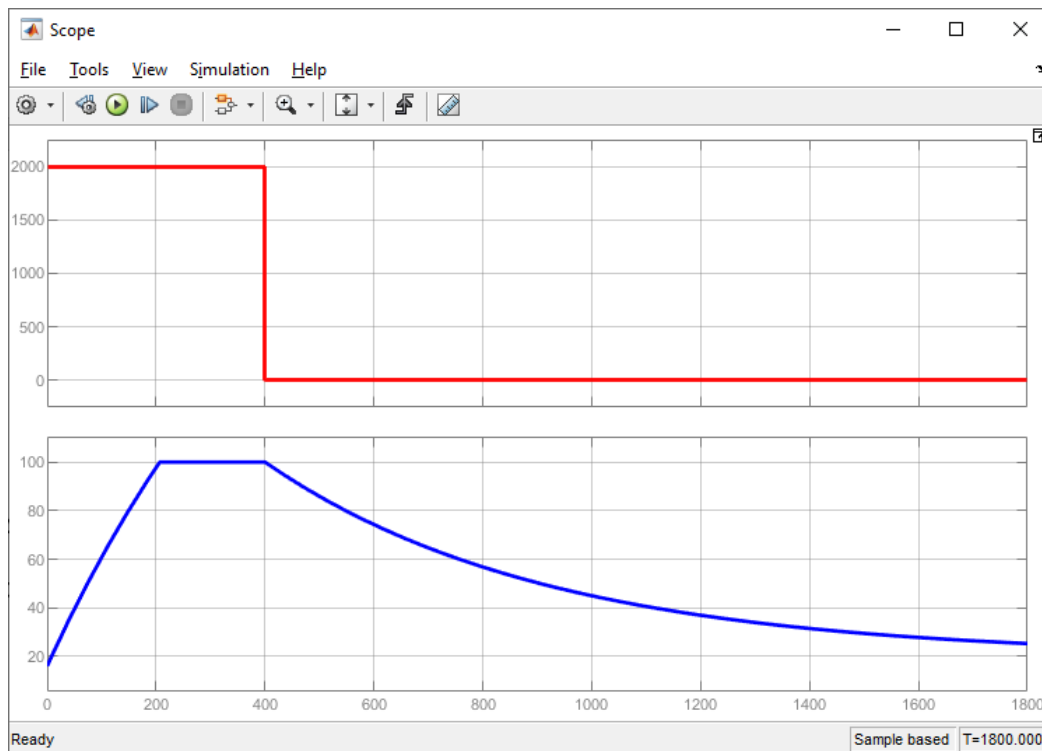
Obr. 4 – Maska a parametry modelu (vytvořeno v MATLAB R2021a)

Abychom mohli zadávat teplotu v čase 0 (na začátku simulace má kapalina v konvici teplotu 16 °C) je nutné v „Initial condition“ bloku integrátor zadat proměnnou T_0 . Pro to, aby teplota nepřekročila 100 °C a chování odpovídalo realitě, kdy kapalina vaří, je nutné nastavit ve vlastnostech bloku integrátor omezení integrace „Upper saturation limit“ na hodnotu 100.



Obr. 5 – Parametry bloku integrátor (vytvořeno v MATLAB R2021a)

Pro zvolený průběh výkonu topení je provedena simulace teploty kapaliny uvnitř varné konvice (v čase 400 s je proveden skok z 2000 W na 0 W – topení je vypnuto).

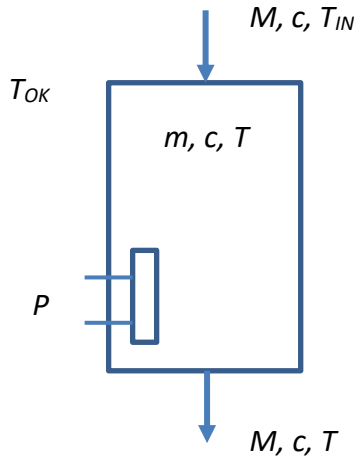


Obr. 6 – Simulace chování modelu varné konvice (vytvořeno v MATLAB R2021a)

Na obr. 6 je zobrazen výkon topení a průběh teploty kapaliny uvnitř varné konvice. Přibližně v čase 200 s dosáhne kapalina uvnitř konvice teploty 100 °C. Od času 400 s začíná kapalina v konvici chladnout – limitně by se teplota dostala na teplotu okolí 20 °C.

3 Model průtokového ohřivače

Model průtokového ohřivače vytvoříme rozšířením modelu varné konvice o vstupní a výstupní proud kapaliny o hmotnostním průtoku M . Předpokládáme, že na začátku simulace je ohřivač naplněn vodou a díky přepadu se objem kapaliny uvnitř ohřivače nemění. Pro jednoduchost předpokládáme, že se nemění hustota kapaliny a můžeme předpokládat konstantní hmotnost kapaliny uvnitř ohřivače. Na výstupu ohřivače je teplota T (stejná jako uvnitř ohřivače).



Obr. 7 – Schéma průtokového ohřivače

Bilanci výkonů (6) pro průtokový ohřivač lze zapsat jako

$$P + McT_{IN} = McT + \alpha S(T - T_{OK}) + mc \frac{dT}{dt}, \quad (9)$$

kde P je výkon elektrického topení, W

M je hmotnostní průtok kapaliny protékající ohřivačem, $kg \cdot s^{-1}$

c je měrná tepelná kapacita kapaliny, $J \cdot kg^{-1} \cdot K^{-1}$

T_{IN} je teplota vstupující kapaliny, K

T je teplota kapaliny v průtokovém ohřivači, $^{\circ}C$

α je přestupní koeficient pro ztráty tepla, $W \cdot m^{-2} \cdot K^{-1}$

S je přestupní plocha pro ztráty tepla, m^2

T_{OK} je teplota okolí, $^{\circ}C$

m je hmotnost kapaliny v průtokovém ohřivači, kg .

Člen $\alpha S(T - T_{OK})$ odpovídá ztrátovému výkonu a člen $mc \frac{dT}{dt}$ vyjadřuje akumulovaný výkon v kapalině. Pro $M = 0$, přejde model průtokového ohřivače na model varné konvice.

Simulaci chování modelu – řešení rovnice (9) pro konkrétní počáteční teplotu kapaliny uvnitř ohřivače v čase 0 (počáteční podmínku diferenciální rovnice), teplotu okolí, výkon topení, průtok kapaliny protékající ohřivačem a teplotu vstupující kapaliny lze provést numericky integrací veličiny $\frac{dT}{dt}$ vyjádřené z rovnice (9)

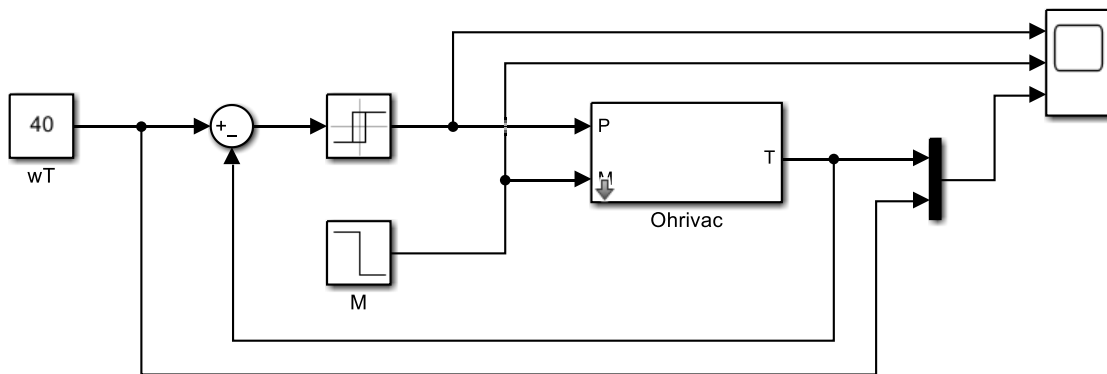
$$T = \int_0^t \frac{dT}{dt} dt = \int_0^t \frac{Mc(T_{IN} - T) + P - \alpha S(T - T_{OK})}{mc} dt \quad (10)$$

Parametry modelu jsou uvedeny v tabulce 1.

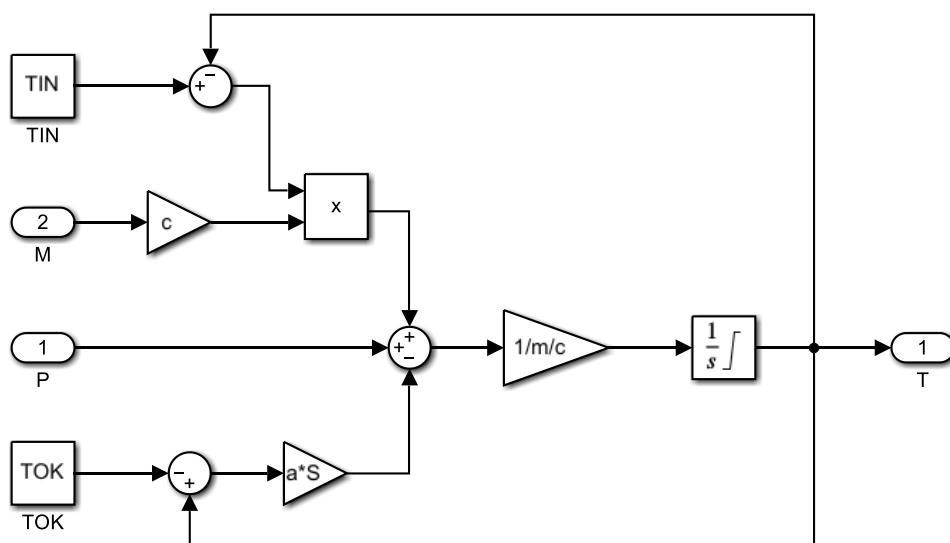
Tab. 2 – Parametry modelu průtokového ohřivače

Parametr	Značka	Hodnota	Jednotka
Přestupní koeficient	a	1	$W.m^{-2}.K^{-1}$
Přestupní plocha	S	0,5	m^2
Teplota okolí	T_{OK}	20	$^{\circ}C$
Hmotnost kapaliny	m	1	kg
Měrná tepelná kapacita kapaliny	c	4120	$J.kg^{-1}.K^{-1}$
Počáteční teplota kapaliny	T_0	16	$^{\circ}C$
Teplota vstupní kapaliny	T_{IN}	16	$^{\circ}C$

Budeme simulovat chování modelu v uzavřeném regulačním obvodu s dvupolohovým regulátorem. Pro simulaci sestavíme model v Simulinku

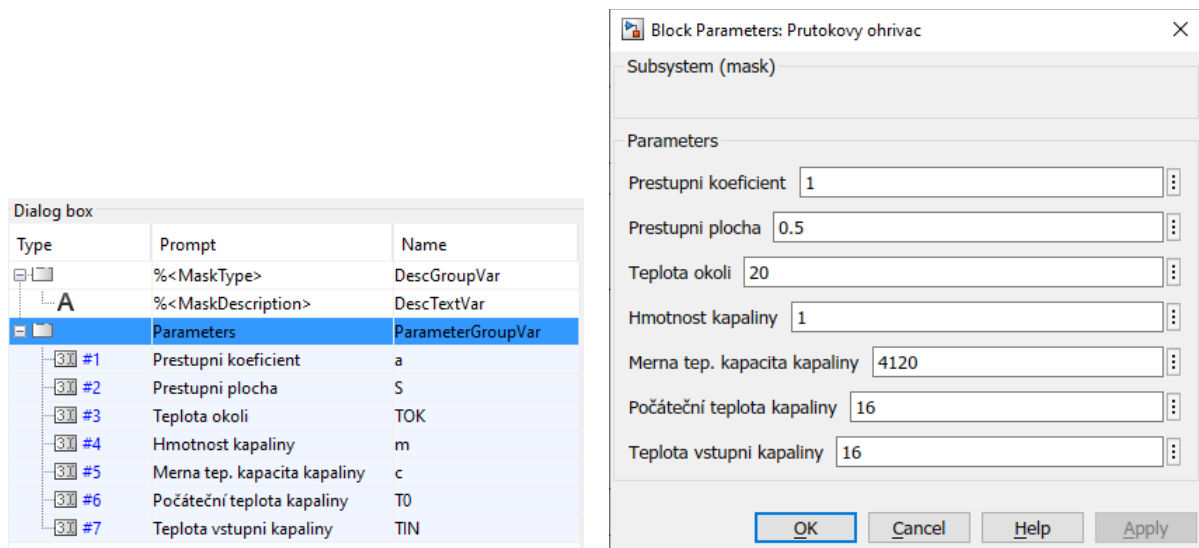


Obr. 8 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)



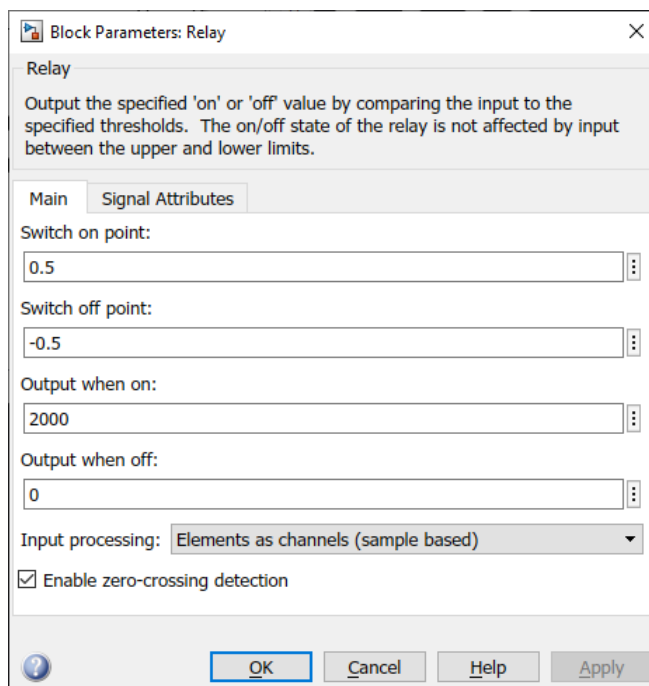
Obr. 9 – Simulační schéma subsystému ohřivače (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska subsystému – obr. 10.

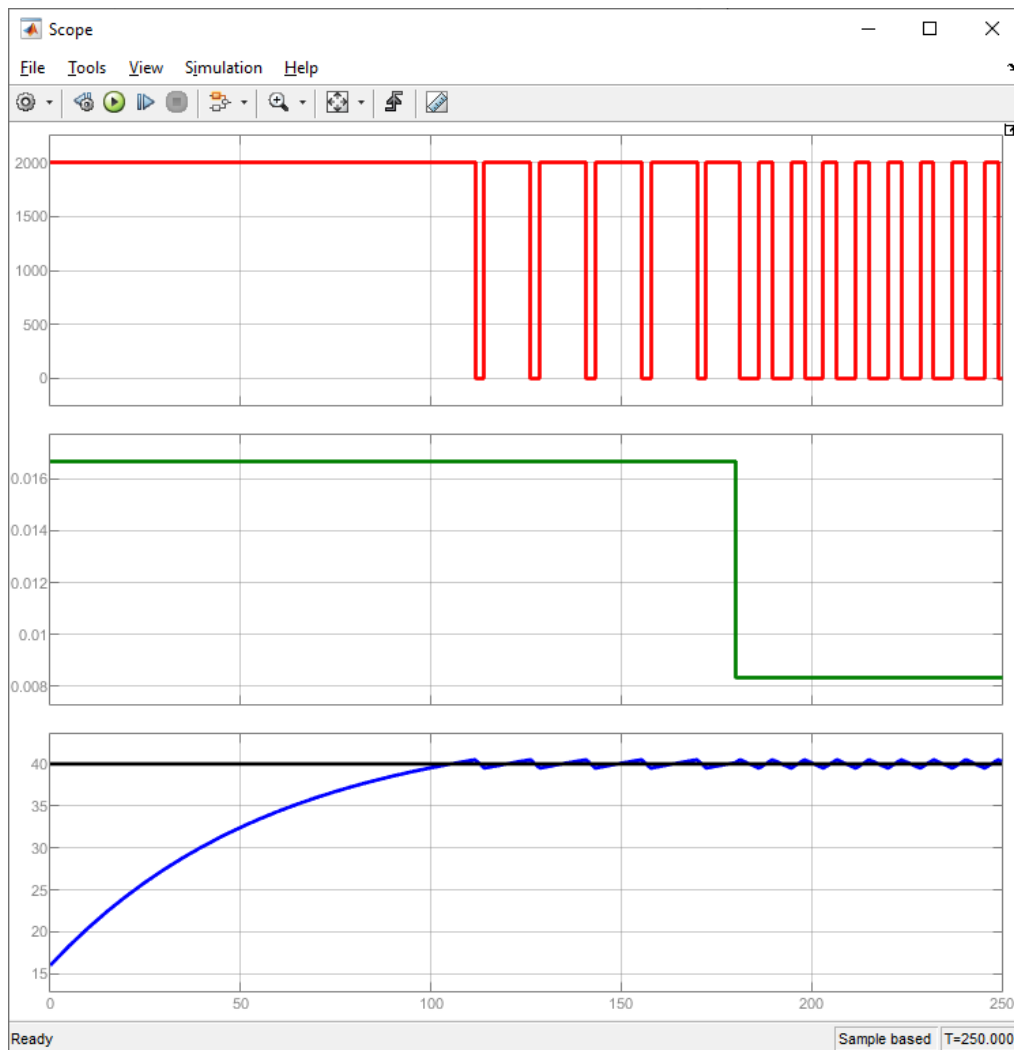


Obr. 10 – Maska a parametry modelu (vytvořeno v MATLAB R2021a)

Dvoupolohový regulátor (relé – blok Relay) zapíná a vypíná plný výkon 2000 W a aby nedocházelo v příliš rychlému přepínání akční veličiny, má nastavenou hysterezi +0,5 a -0,5 °C.



Obr. 11 – Parametry bloku relé (vytvořeno v MATLAB R2021a)



Obr. 12 – Simulace chování modelu průtokového ohřivače (vytvořeno v MATLAB R2021a)

Na obr. 12 je zobrazen výkon topení, teplota vstupní kapaliny a žádaná a regulovaná teplota uvnitř průtokového ohřivače. Přibližně po 100 s dosáhne kapalina uvnitř ohřivače žádané hodnoty – teploty 40 °C a relé začne vypínat a zapínat topení. V čase 180 s dojde ke skoku v průtoku kapaliny ohřivačem z 1/60 na 0,5/60 $kg.s^{-1}$. Z průběhu akční veličiny je vidět, že dodávaný výkon se sníží na polovinu. Doba sepnutí a rozepnutí topení je přibližně stejná a průměrný výkon topení tedy přibližně 1000 W.

4 Úkoly ke cvičení

1. V Simulinku vytvořte model varné konvice – použijte schéma podle obr. 2 a 3.
2. U subsystému konvice vytvořte masku s parametry podle obr. 4.
3. Proveďte simulaci chování modelu podle obr. 6.
4. V Simulinku vytvořte model průtokového ohřivače včetně zpětnovazebního řízení pomocí dvupolohového regulátoru – použijte schéma podle obr. 8 a 9.

5. U subsystému ohřívače vytvořte masku s parametry podle obr. 10.
6. Proveďte simulaci chování modelu podle obr. 12.

5 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

akumulace energie, 2
model průtokového ohřívače, 6
model varné konvice, 3

přestup tepla, 1
tepelný tok, 1
teplo, 1

Modelování a simulace dynamických systémů

Téma 3: Model domu

Studijní cíl

Seznámit studenty s tvorbou modelů tepelných systémů a vytvořit model domu.

Doba nutná k nastudování

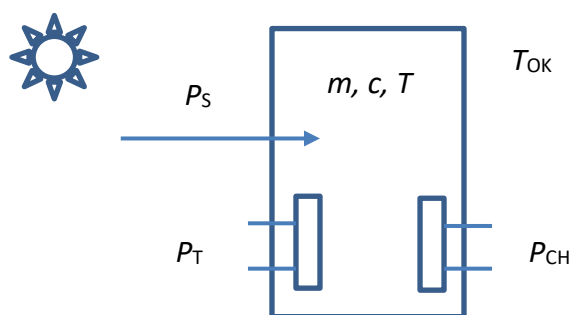
2 hodiny

Klíčová slova

Model, tepelné systémy, dům

1 Model domu

Vytvoříme jednoduchý tepelný model domu s elektrickým topením, elektrickým chlazením, vlivem okolní teploty a slunce. Budeme uvažovat tepelnou kapacitu „hmoty“ uvnitř domu zastupující tepelnou kapacitu zdí, podlahy, stropu, vybavení domu, topení i chlazení. Budeme uvažovat dokonalé míchání vzduchu – zavedeme takzvané soustředěné parametry s jednou „charakteristickou“ teplotou vzduchu. Ve skutečnosti by byla teplota vzduchu v okolí topení a chlazení jiná než u stěn a podobně u stropu by byla jiná než u podlahy. Museli bychom, ale řešit parciální diferenciální rovnice pro rozložení teploty v prostoru (rozložené parametry). Předpokladem jedné charakteristické teploty (soustředěné parametry) zůstane jedinou nezávislou proměnnou čas a systém popíšeme pomocí obyčejné diferenciální rovnice. Dále budeme uvažovat výměnu tepla s okolím – pokud bude teplota uvnitř domu vyšší než okolí, tak hovoříme o tepelných ztrátách. V opačném případě by se vnitřek domu vlivem okolního prostředí ohříval.



Obr. 1 – Schéma domu

Bilanci výkonů (6) v kapitole 2 pro model domu můžeme zapsat jako

$$P_T - P_{CH} + P_S = \alpha S(T - T_{OK}) + mc \frac{dT}{dt}, \quad (1)$$

- kde P_T je výkon elektrického topení, W
 P_{CH} je výkon elektrického chlazení, W
 P_S je výkon slunce vstupující do domu, W
 α je přestupní koeficient pro ztráty tepla, $W.m^{-2}.K^{-1}$
 S je přestupní plocha pro ztráty tepla, m^2
 T je teplota uvnitř domu, $^{\circ}C$
 T_{OK} je teplota okolí, $^{\circ}C$
 m je hmota uvnitř domu, kg
 c je měrná tepelná kapacita hmoty uvnitř domu, $W.m^{-2}.K^{-1}$.

Člen $\alpha S(T - T_{OK})$ odpovídá ztrátovému výkonu a člen $mc \frac{dT}{dt}$ vyjadřuje akumulovaný výkon uvnitř domu.

Simulaci chování modelu – řešení rovnice (1) pro konkrétní počáteční teplotu uvnitř domu v čase 0 (počáteční podmínka diferenciální rovnice), teplotu okolí, výkon topení, výkon chlazení a výkon slunce vstupující do domu lze provést numericky integrací veličiny $\frac{dT}{dt}$ vyjádřené z rovnice (1)

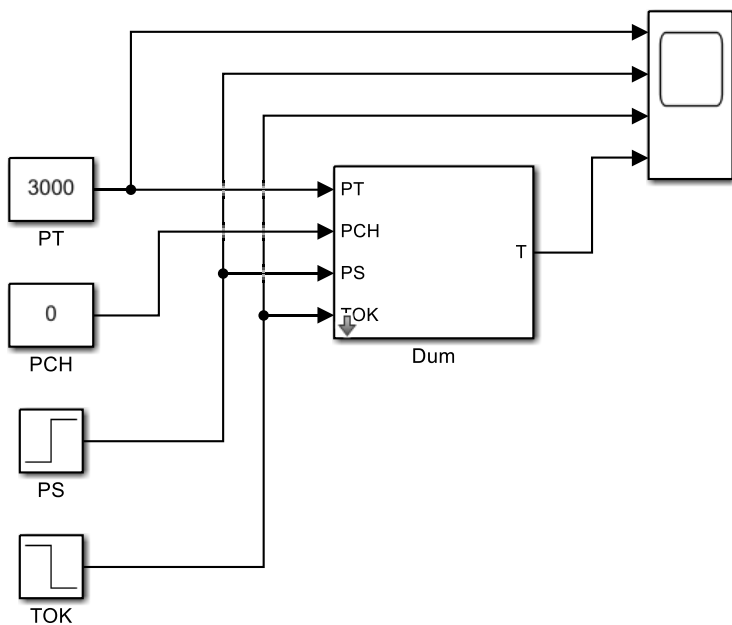
$$T = \int_0^t \frac{dT}{dt} dt = \int_0^t \frac{P_T - P_{CH} + P_S - \alpha S(T - T_{OK})}{mc} dt \quad (2)$$

Parametry modelu jsou uvedeny v tabulce 1.

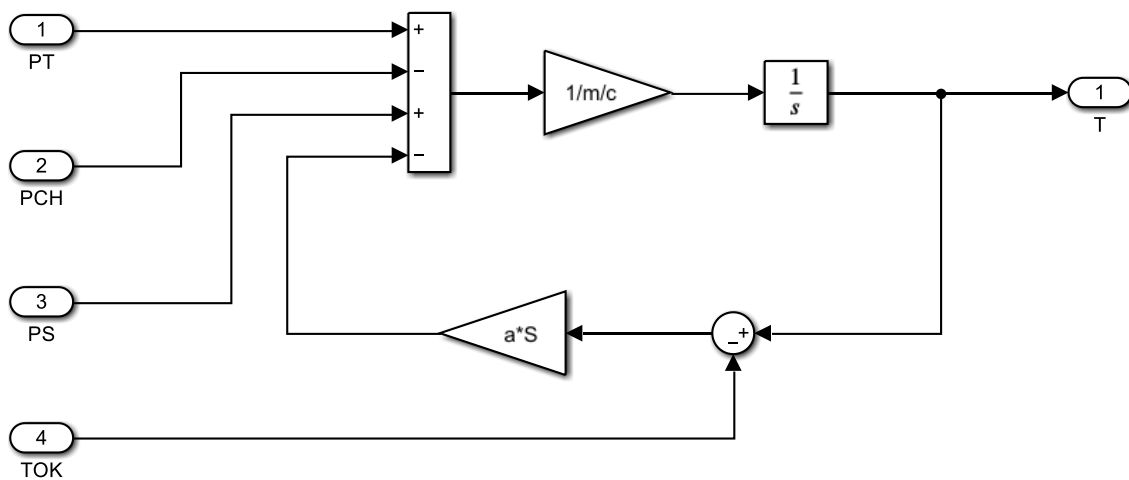
Tab. 1 – Parametry modelu domu

Parametr	Značka	Hodnota	Jednotka
Přestupní koeficient	α	2	$W.m^{-2}.K^{-1}$
Přestupní plocha	S	100	m^2
Teplota okolí	T_{OK}	10	$^{\circ}C$
Hmota uvnitř domu	m	50	kg
Měrná tepelná kapacita hmoty uvnitř domu	c	2000	$W.m^{-2}.K^{-1}$
Počáteční teplota uvnitř domu	T_0	20	$^{\circ}C$

Pro simulaci sestavíme model v Simulinku



Obr. 2 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)



Obr. 3 – Simulační schéma subsystému domu (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska subsystému

Dialog box		
Type	Prompt	Name
	%<MaskType>	DescGroupVar
A	%<MaskDescription>	DescTextVar
	Parameters	ParameterGroupVar
#1	Přestupní koeficient	a
#2	Přestupní plocha	S
#3	Hmotnost	m
#4	Merna tepelná kapacita	c
#5	Pocáteční teplota	T0

Block Parameters: Dum

Subsystem (mask)

Parameters

Přestupní koeficient

Přestupní plocha

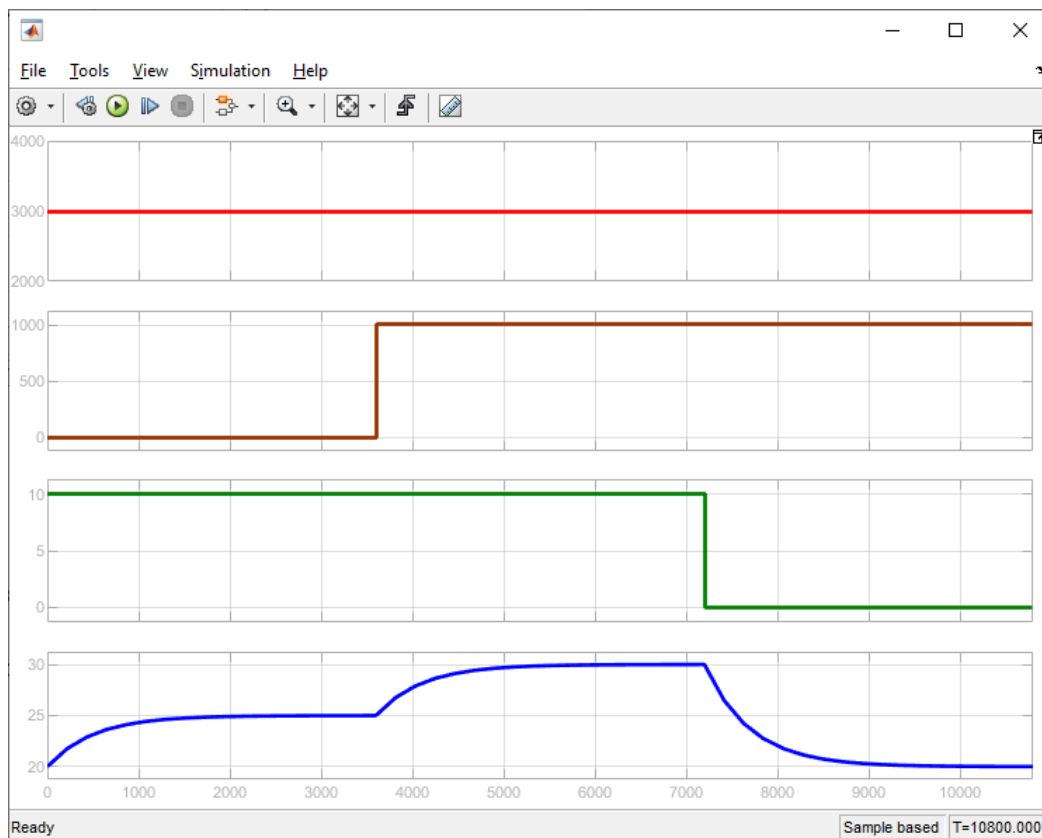
Hmotnost

Merna tepelná kapacita

Pocáteční teplota

Obr. 4 – Maska a parametry modelu (vytvořeno v MATLAB R2021a)

Pro zvolený výkon topení 3000 W je provedena simulace teploty uvnitř domu. Výkon chlazení je 0 W, výkon slunce vstupující do domu skokově vzroste v čase 3600 s z hodnoty 0 na 1000 W a okolní teplota skokově klesne v čase 7200 s z hodnoty 10 na 0 °C.



Obr. 5 – Simulace chování modelu domu (vytvořeno v MATLAB R2021a)

Na obrázku 5 je zobrazen výkon topení, výkon slunce vstupující do domu, okolní teplota a teplota uvnitř domu. Přibližně po hodině se teplota uvnitř domu ustálí na 25 °C, po skokové změně výkonu slunce vzroste na 30 °C a po skokové změně okolní teploty se vrátí zpátky na 25 °C.

2 Analýza ustálených stavů, zesílení a časové konstanty

Provedeme analýzu chování modelu domu z pohledu **ustáleného stavu**, kdy platí

$$P_T - P_{CH} + P_S = \alpha S(T - T_{OK}) \quad (3)$$

Pro ustálenou teplotu uvnitř domu platí

$$T = \frac{P_T - P_{CH} + P_S}{\alpha S} + T_{OK} \quad (4)$$

V našem konkrétním případě vyjde pro první ustálený stav na obr. 5 teplota uvnitř domu 25 °C. V ustáleném stavu je výkon topení roven ztrátovému výkonu a tento výkon udržuje 15 °C teplotního rozdílu uvnitř domu a v jeho okolí. Jinými slovy součin αS nám udává, jaký výkon musíme použít, aby byl rozdíl 1 °C uvnitř a v okolí domu. V našem případě je to 200 W.

Rovnici (1) lze převést do tvaru, ze kterého bude zřejmé **zesílení systému** a jeho **časová konstanta**

$$\frac{mc}{\alpha S} \frac{dT}{dt} + T = \frac{1}{\alpha S} (P_T - P_{CH} + P_S) + T_{OK} \quad (5)$$

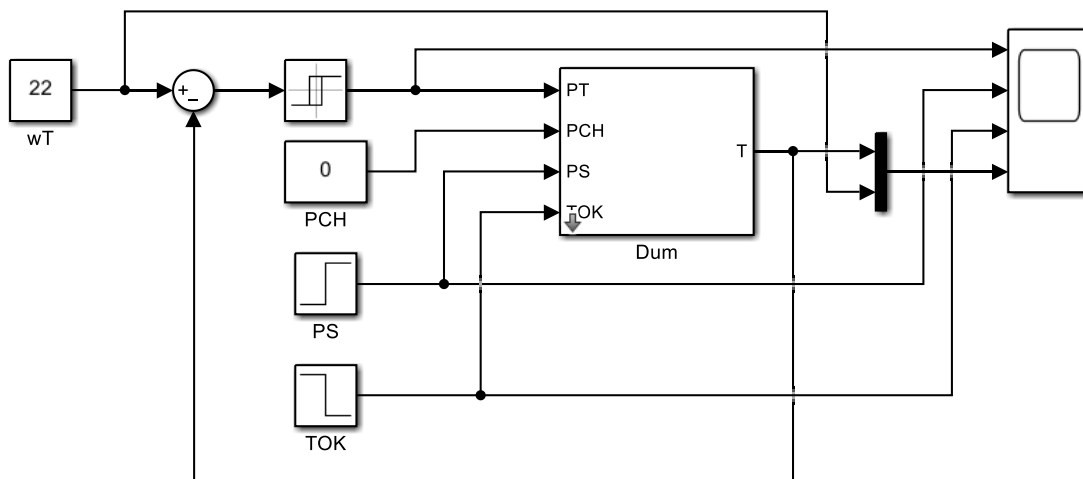
$$\tau \frac{dT}{dt} + T = \frac{1}{\alpha S} (P_T - P_{CH} + P_S) + T_{OK} \quad (6)$$

Časová konstanta vyjde 500 s a zesílení pro vstupy výkonu topení, chlazení a slunce 0,005 °C/W. Zesílení pro okolní teplotu je 1 – okolní teplota pouze posouvá teplotu uvnitř domu.

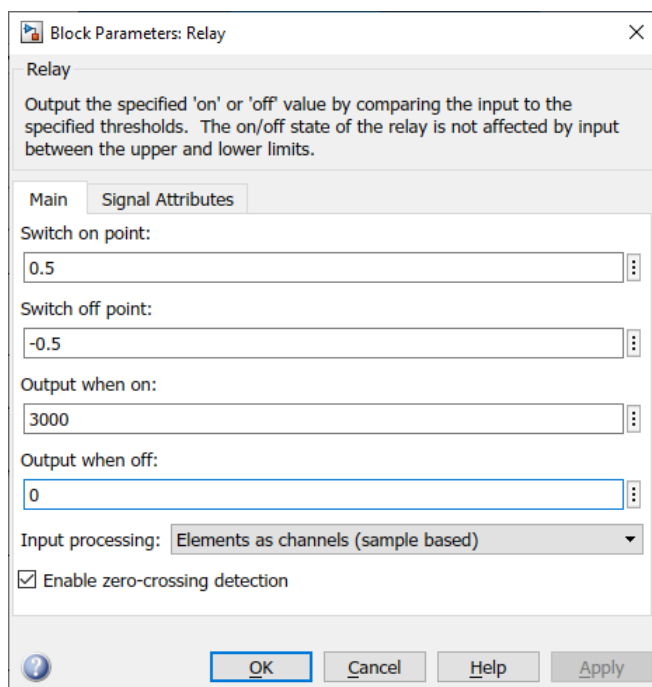
Jedná se o velmi zjednodušený model, který vystihuje hrubě chování skutečného domu, ale pro získání základní představy o vlivu vstupujících energií a okolní teploty na průběh teploty uvnitř domu je použitelný. Pokud bychom jej chtěli zpřesnit, lze doplnit dynamiku topení a chlazení, vývoj výkonu slunce v průběhu dne, vliv oblačnosti, vliv zastínění oken žaluziemi, vliv větrání, vliv pobytu osob v domě a podobně.

3 Simulace regulace teploty dvoupolohovým regulátorem

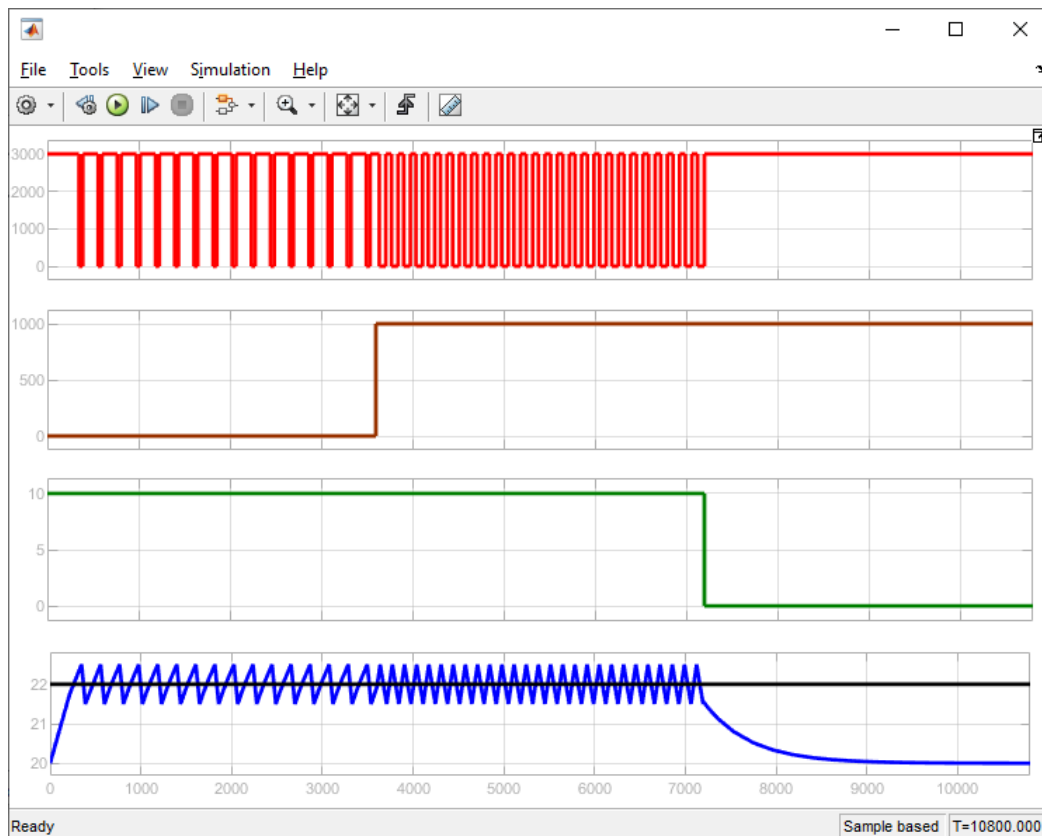
Pro stejné průběhy výkonu chlazení, slunce vstupujícího do domu a okolní teploty odsimulujeme průběh teploty uvnitř domu při použití dvoupolohového regulátoru spínajícího výkon topení 3000 W. Subsystem modelu domu zůstane stejný, ale změníme simulační schéma modelu a nastavíme parametry dvoupolohového regulátoru (relé – blok Relay). Stejně jakou u modelu průtokového ohřívače nastavíme hysterezi +0,5 a -0,5 °C.



Obr. 6 – Simulační schéma modelu s regulací (vytvořeno v MATLAB R2021a)



Obr. 7 – Parametry bloku relé (vytvořeno v MATLAB R2021a)



Obr. 8 – Simulace chování modelu domu s regulací (vytvořeno v MATLAB R2021a)

Na obrázku 8 je zobrazen výkon topení, výkon slunce vstupující do domu, okolní teplota a žádaná a regulovaná teplota uvnitř domu. Z průběhu výkonu topení je vidět, že pokud nesvítí slunce, musíme topit větším výkonem a pokud klesne okolní teplota o 10 °C, tak nám ani maximální výkon nestačí na dosažení žádané teploty uvnitř domu.

4 Úkoly ke cvičení

1. V Simulinku vytvořte model domu – použijte schéma podle obr. 2 a 3.
2. U subsystému vytvořte masku s parametry podle obr. 4.
3. Proveďte simulaci chování modelu podle obr. 5.
4. Modifikujte model a proveďte simulaci v případě regulace dvoupolohovým regulátorem podle schématu na obr. 6 a podle obr. 8.

5 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. Matlab a Simulink: úvod do používání. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

časová konstanta, 5
model domu, 1

ustálený stav, 5
zesílení, 5

Modelování a simulace dynamických systémů

Téma 4: Model nádrže a model nádrže s otvorem

Studijní cíl

Seznámit studenty s tvorbou modelů hydraulických systémů a vytvořit model nádrže a model nádrže s otvorem.

Doba nutná k nastudování

2 hodiny

Klíčová slova

Model, hydraulické systémy, nádrž, nádrž s otvorem

1 Principy modelování hydraulických systémů

Při modelování hydraulických systémů budeme provádět bilance hmoty. Pokud do systému hmotu přivedeme, tak zvýšíme zádrž hmoty v systému a zvýší se výška hladiny. Naopak pokud hmotu odvedeme, výška hladiny se sníží. Bilanci hmoty (zákon zachování hmoty) za určitou jednotku času, lze zapsat jako

$$m_{\text{IN}} = m_{\text{OUT}} + m_{\text{AK}}, \quad (1)$$

kde m_{IN} je hmota, která do systému vstoupila, *kg*

m_{OUT} je hmota, která ze systému vystoupila, *kg*

m_{AK} je hmota, která se v systému akumulovala, *kg*.

Budeme pracovat se změnami hmoty za nekonečně malý časový úsek dt , čímž se rovnice (1) stane bilancí hmotnostních průtoků

$$M_{\text{IN}} = M_{\text{OUT}} + \frac{dm}{dt}, \quad (2)$$

kde M_{IN} je vstupní hmotnostní průtok, *kg.s⁻¹*

M_{OUT} je výstupní hmotnostní průtok, *kg.s⁻¹*

$\frac{dm}{dt}$ je hmotnostní průtok, který se v systému akumuloval, *kg.s⁻¹*.

Za předpokladu, že se hustota kapaliny nemění (například z důvodu změny teploty nebo koncentrace), můžeme rovnici (2) vydělit hustotou kapaliny ρ a získat tak bilanci objemových průtoků

$$Q_{IN} = Q_{OUT} + \frac{dV}{dt}, \quad (3)$$

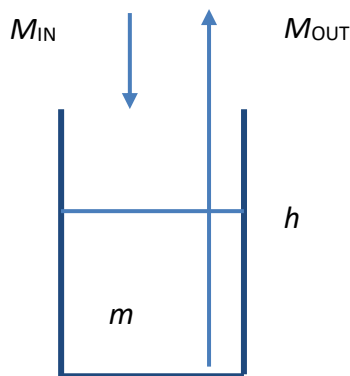
kde Q_{IN} je vstupní objemový průtok, $m^3 \cdot s^{-1}$

Q_{OUT} je výstupní objemový průtok, $m^3 \cdot s^{-1}$

$\frac{dV}{dt}$ je objemový průtok, který se v systému akumuloval, $m^3 \cdot s^{-1}$.

2 Model nádrže

Vytvoříme **model nádrže**, do které lze čerpat nebo ze které lze odčerpávat kapalinu.



Obr. 1 – Schéma nádrže

Pokud je průřez nádrže S konstantní (nemění se s výškou nádrže), můžeme akumulací člen v rovnici (3) vyjádřit pomocí průřezu nádrže a výšky hladiny

$$\frac{dV}{dt} = S \frac{dh}{dt} = Q_{IN} - Q_{OUT}, \quad (4)$$

kde S je průřez nádrže, m^2

h je výška hladiny v nádrži, m .

Simulaci chování modelu – řešení rovnice (4) pro konkrétní počáteční výšku hladiny uvnitř nádrže v čase 0 (počáteční podmínka diferenciální rovnice), vstupní a výstupní objemový průtok lze provést numericky integrací veličiny $\frac{dh}{dt}$ vyjádřené z rovnice (4)

$$h = \int_0^t \frac{dh}{dt} dt = \int_0^t \frac{Q_{IN} - Q_{OUT}}{S} dt \quad (5)$$

Rovnice vyjadřuje skutečnost, že pokud je $Q_{IN} = Q_{OUT}$, tak se hladina nemění, pokud je $Q_{IN} > Q_{OUT}$, hladina roste a naopak pokud $Q_{IN} < Q_{OUT}$, hladina klesá. Rychlost nárůstu a poklesu

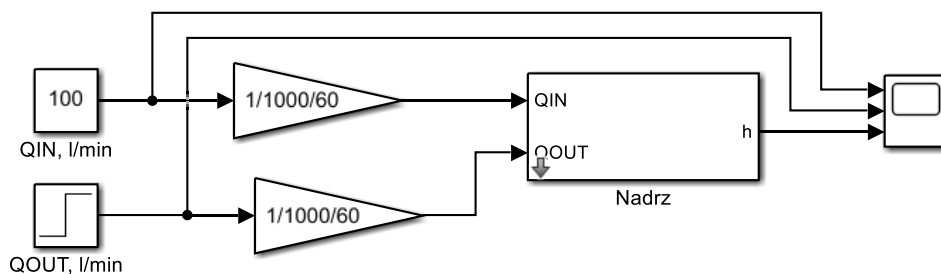
hladiny záleží na velikosti rozdílu průtoků, ale také nepřímo úměrně na průřezu nádrže. Budeme uvažovat nádrž s kruhovým průřezem – bude zadán průměr nádrže.

Parametry modelu jsou uvedeny v tabulce 1.

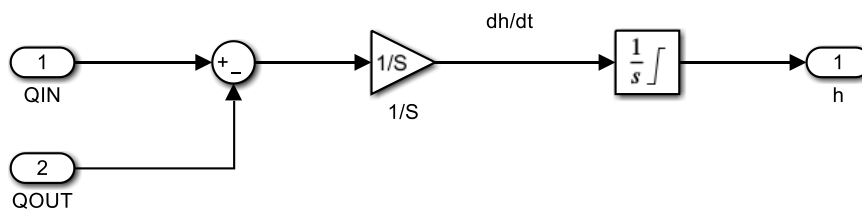
Tab. 1 – Parametry modelu nádrže

Parametr	Značka	Hodnota	Jednotka
Průměr nádrže	d	0,5	m
Počáteční výška hladiny	h_0	0	m
Výška nádrže	h_{MAX}	1,5	m

Pro simulaci sestavíme model v Simulinku

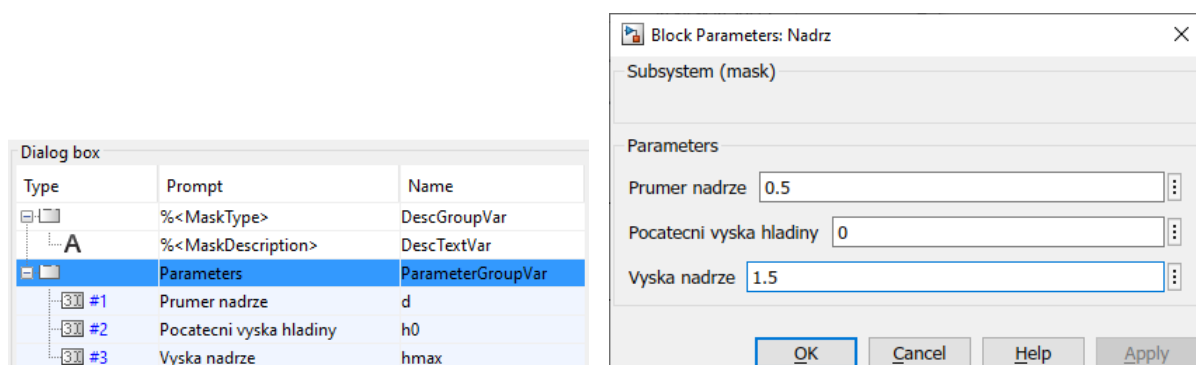


Obr. 2 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)



Obr. 3 – Simulační schéma subsystému nádrže (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska subsystému



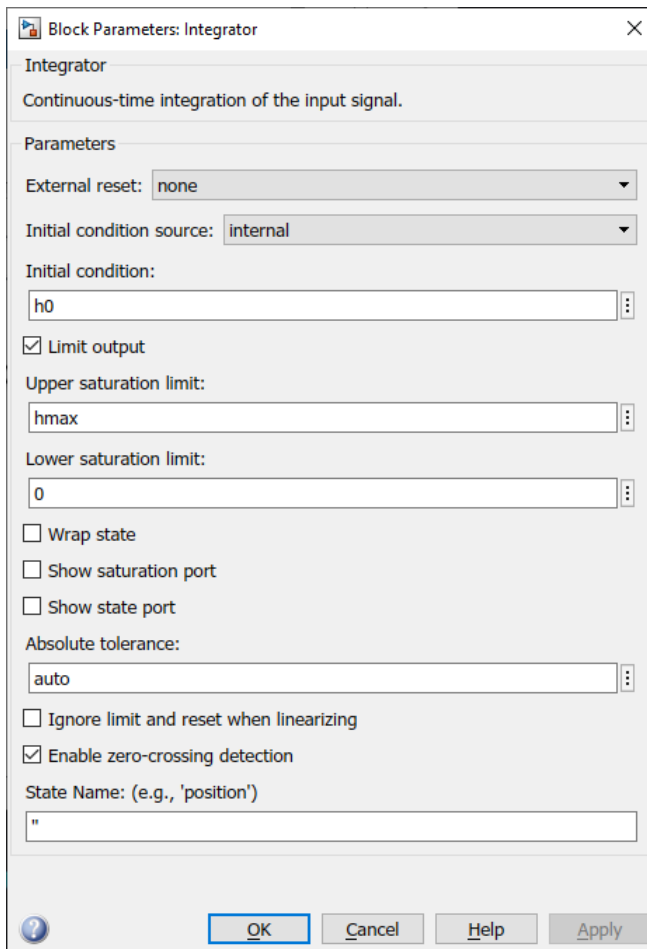
Obr. 4 – Maska a parametry modelu (vytvořeno v MATLAB R2021a)

V záložce Initialization je zadán přepočítání mezi průměrem a průřezem nádrže.

```
Initialization commands  
S=pi*d^2/4;
```

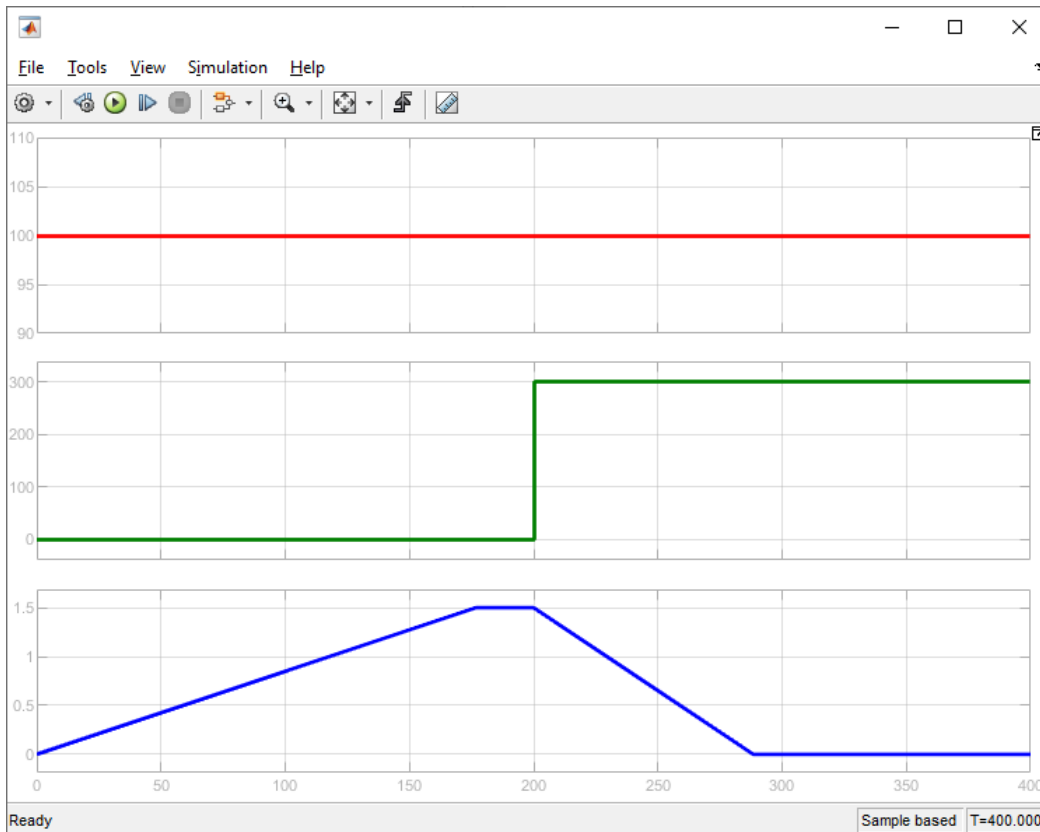
Obr. 5 – Inicializace subsystému (vytvořeno v MATLAB R2021a)

Aby chování modelu odpovídalo realitě, kdy je výška hladiny omezena výškou nádrže (poté začne kapalina přetékat přes okraj nádrže), je nutné nastavit ve vlastnostech integrátoru hladiny omezení integrace „Upper saturation limit“ na hodnotu h_{MAX} . Stejně tak nemůže hladina klesnout pod hodnotu 0 nastavenou v „Lower saturation limit“. Abychom mohli zadávat hladinu v čase 0 (výšku hladiny v nádrži na začátku simulace) je nutné v „Initial condition“ bloku integrátor zadat proměnnou h_0 .



Obr. 6 – Parametry bloku integrátor (vytvořeno v MATLAB R2021a)

Průtoky jsou zadávány v $l.min^{-1}$ a pomocí bloku Gain přepočítány na $m^3.s^{-1}$. Vstupní průtok je po celou dobu experimentu $100 l.min^{-1}$ a výstupní průtok skočí v čase 200 s z hodnoty 0 na $300 l.min^{-1}$.

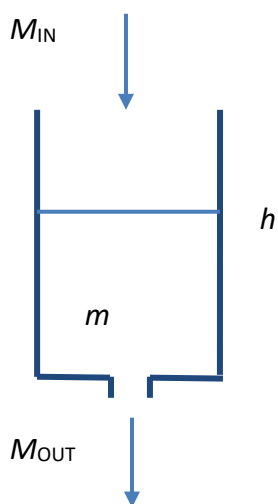


Obr. 7 – Simulace chování modelu nádrže (vytvoreno v MATLAB R2021a)

Na obrázku 7 je zobrazen vstupní průtok, výstupní průtok a výška hladiny v nádrži.

3 Model nádrže s otvorem

Vytvoříme **model nádrže s otvorem** ve dně, ze kterého bude kapalina samovolně vytékat.



Obr. 8 – Schéma nádrže s otvorem

Bilance objemových průtoků bude stejná, jako u modelu nádrže. Model se bude, ale lišit tím, že výstupní průtok nebude volitelnou proměnnou nebo parametrem, ale bude jej určovat výška hladiny v nádrži podle **Torricelliho rovnice**. Jedná se o speciální případ **Bernoulliho rovnice** – zákon zachování energií ustáleného proudění ideální kapaliny

$$E_{\text{pohybová}} + E_{\text{tlaková}} + E_{\text{polohová}} = \text{konst.} \quad (6)$$

Bilancujeme energii v místě hladiny a v místě výtoku kapaliny. Nad hladinou i pod výtokem je atmosférický tlak, a proto nebudeme tlakovou energii uvažovat. Průřez nádrže je mnohem větší než průřez výtokového otvoru, a proto můžeme pohybovou energii v místě hladiny zanedbat a budeme uvažovat pouze polohovou energii. Polohu výtokového otvoru umístíme do počátku souřadného systému výšky hladiny, takže lze říct, že veškerá polohová energie se přemění na kinetickou energii kapaliny v místě otvoru. Je to jako kdyby kapka vody (kapalina) padala volným pádem z výšky h a získala tím v gravitačním poli s gravitačním zrychlením g rychlost v .

$$\frac{1}{2}mv^2 = mgh \Rightarrow v = \sqrt{2gh}, \quad (7)$$

kde m je hmotnost kapaliny, kg
 v je rychlost kapaliny, $m \cdot s^{-1}$
 g je gravitační zrychlení, $m \cdot s^{-2}$
 h je výška hladiny v nádrži, $m \cdot s^{-1}$.

Pro výstupní objemový průtok platí vztah

$$Q_{\text{OUT}} = fv = f\sqrt{2gh}, \quad (8)$$

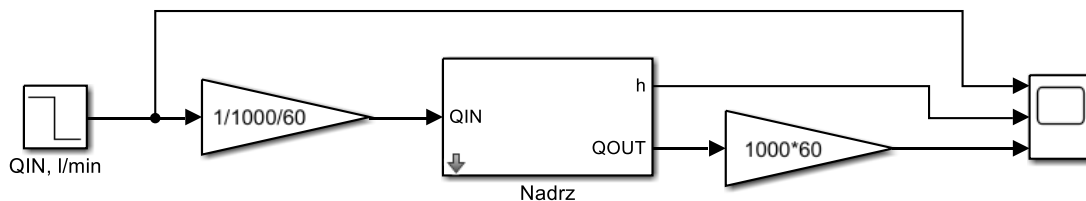
kde f je průřez výtokového otvoru, m^2 .

Parametry modelu jsou uvedeny v tabulce 1.

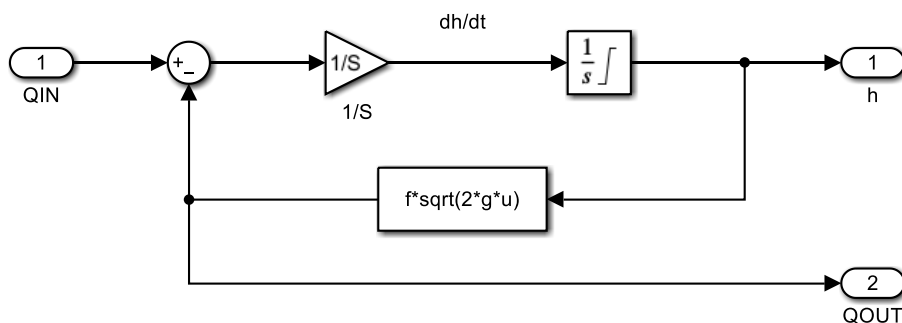
Tab. 2 – Parametry modelu nádrže s otvorem

Parametr	Značka	Hodnota	Jednotka
Průměr nádrže	d_N	1	m
Počáteční výška hladiny	h_0	0	m
Výška nádrže	h_{MAX}	1,5	m
Průměr výtokového otvoru	d_0	0,03	m

Pro simulaci sestavíme model v Simulinku



Obr. 9 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)



Obr. 10 – Simulační schéma subsystému nádrže s otvorem (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska subsystému

Dialog box		
Type	Prompt	Name
	%<MaskType>	DescGroupVar
A	%<MaskDescription>	DescTextVar
Parameters		ParameterGroupVar
#1	Prumer nadrze	dN
#2	Pocatecni vyska hladiny	h0
#3	Vyska nadrze	hmax
#4	Prumer vytokoveho otvoru	dO

Obr. 11 – Maska a parametry modelu (vytvořeno v MATLAB R2021a)

V záložce Initialization je zadán přepočít mezi průměrem a průřezem nádrže a výtokového otvoru a gravitační zrychlení.

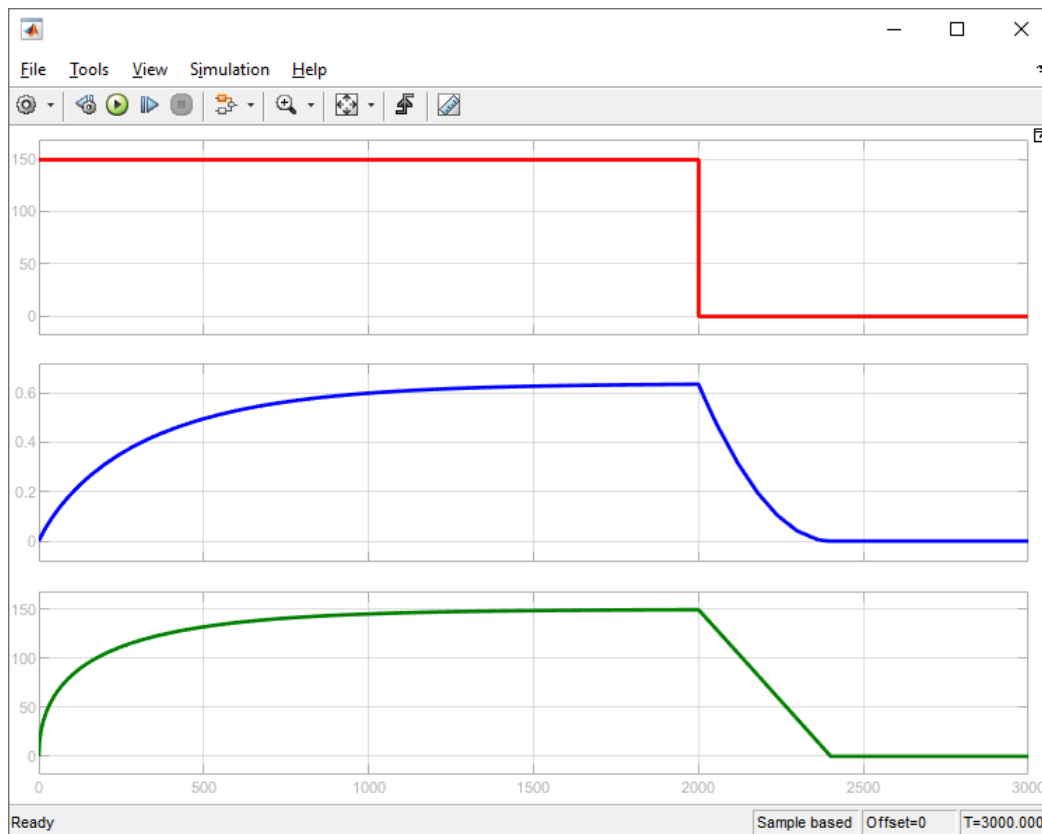
```

Initialization commands
S=pi*dN^2/4;
f=pi*dO^2/4;
g=9.81;

```

Obr. 12 – Inicializace subsystému (vytvořeno v MATLAB R2021a)

Integrátor je nastavený stejně jako u modelu nádrže v předchozí kapitole. Průtoky jsou zadávány v $l.min^{-1}$ a pomocí bloku Gain přepočítány na $m^3.s^{-1}$. Vstupní průtok skočí v čase 2000 s z hodnoty 150 na 0 $l.min^{-1}$.



Obr. 13 – Simulace chování modelu nádrže s otvorem (vytvořeno v MATLAB R2021a)

Na obrázku 13 je zobrazen vstupní průtok, výška hladiny v nádrži a výstupní průtok.

4 Úkoly ke cvičení

1. V Simulinku vytvořte model nádrže – použijte schéma podle obr. 2 a 3.
2. U subsystému vytvořte masku s parametry podle obr. 4.
3. Proveďte simulaci chování modelu podle obr. 7.
4. V Simulinku vytvořte model nádrže s otvorem – použijte schéma podle obr. 9 a 10.
5. U subsystému vytvořte masku s parametry podle obr. 11.
6. Proveďte simulaci chování modelu podle obr. 13.

5 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

Bernoulliho rovnice, 6
model nádrže, 2

model nádrže s otvorem, 5
Torricelliho rovnice, 6

Modelování a simulace dynamických systémů

Téma 5: Linearizace modelu nádrže s otvorem

Studijní cíl

Seznámit studenty se způsobem linearizace modelu nádrže s otvorem.

Doba nutná k nastudování

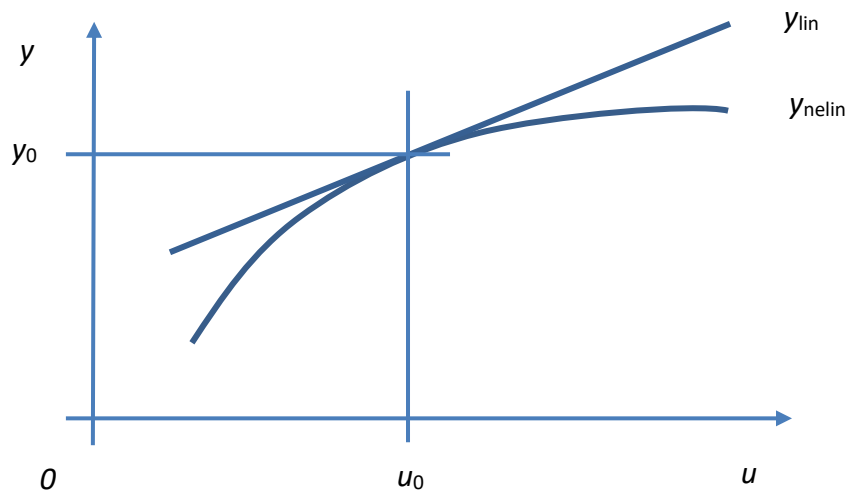
2 hodiny

Klíčová slova

Model, hydraulické systémy, nádrž s otvorem, linearizace modelu nádrže s otvorem

1 Linearizace nelineární funkce

Linearizaci lze provést následujícím způsobem. Ve zvoleném pracovním bodě vytvoříme tečnu k původní nelineární funkci y_{nelin} , čímž získáme lineární aproximaci y_{lin} .



Obr. 1 – Linearizace nelineární funkce

Matematicky lze zapsat lineární funkci jako

$$y_{\text{lin}} = ku + q, \tag{1}$$

kde k je směrnice tečny

q je posunutí tečny.

Směrnice tečny je derivace nelineární funkce y_{nelin} v bodě u_0, y_0

$$k = \left. \frac{dy_{\text{nelin}}}{du} \right|_{u_0, y_0} \quad (2)$$

Posunutí tečny vyjádříme dosazením bodu u_0, y_0 do rovnice (3)

$$q = y_0 - ku_0 \quad (3)$$

Dosazením vztahu (3) do rovnice (1) získáme

$$y_{\text{lin}} = ku + y_0 - ku_0 \quad (4)$$

Můžeme provést následující úpravu

$$\frac{y_{\text{lin}} - y_0}{\Delta y} = k \left(\frac{u - u_0}{\Delta u} \right), \quad (5)$$

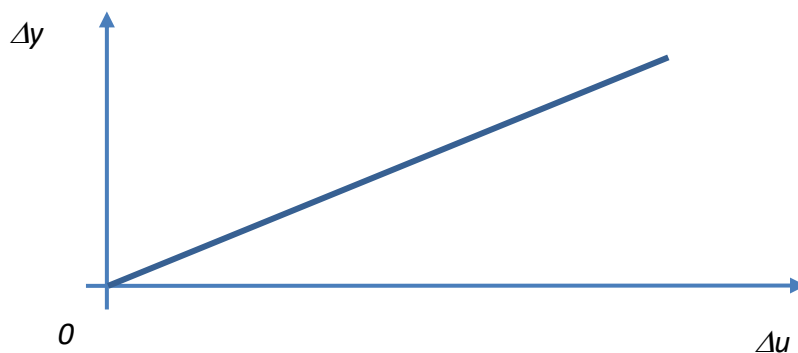
kde Δy je odchylka y od pracovního bodu y_0

Δu je odchylka u od pracovního bodu u_0 .

Výsledná lineární aproximace je

$$\Delta y = \left. \frac{dy_{\text{nelin}}}{du} \right|_{u_0, y_0} \Delta u \quad (6)$$

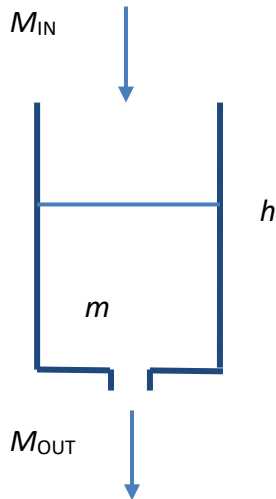
Nejenom, že jsme provedli linearizaci původní nelineární závislosti, ale také jsme transformovali osy souřadného systému (počátek souřadného systému jsme posunuli do pracovního bodu u_0, y_0). Budeme pracovat s „**odchylkovými**“ **veličinami** – odchylkami původních proměnných od pracovního bodu. Tím jsme získali tvar, který při převodu do obrazového přenosu splní podmínku nulových počátečních podmínek – v čase 0 je systém ustálený v bodě u_0, y_0 a odchylkové veličiny jsou nulové.



Obr. 2 – Lineární aproximace

2 Linearizace modelu nádrže s otvorem

Model nádrže s otvorem ve dně byl podrobně odvozen v kapitole 4. Zde je uveden pouze výsledný tvar modelu.



Obr. 3 – Schéma nádrže s otvorem

Za předpokladu, že se hustota kapaliny nemění (například z důvodu změny teploty nebo koncentrace) a pro konstantní průřez nádrže, můžeme zapsat bilanci objemových průtoků

$$Q_{IN} = Q_{OUT} + S \frac{dh}{dt}, \quad (7)$$

kde Q_{IN} je vstupní objemový průtok, $m^3 \cdot s^{-1}$

Q_{OUT} je výstupní objemový průtok, $m^3 \cdot s^{-1}$

S je průřez nádrže, m^2

h je výška hladiny v nádrži, m .

Pro výstupní objemový průtok platí vztah

$$Q_{OUT} = f \sqrt{2gh}, \quad (8)$$

kde f je průřez výtokového otvoru, m^2

g je gravitační zrychlení, $m \cdot s^{-2}$.

Rovnice (7) je lineární, ale rovnice (8) je díky odmocninové závislosti nelineární a pokud chceme vyjádřit obrazový přenos systému, musíme tento vztah linearizovat.

Postupem uvedeným v kapitole 1 získáme vztah

$$\Delta Q_{OUT} = \left. \frac{dQ_{OUT}}{dh} \right|_{h_0, Q_0} \Delta h \quad (9)$$

Derivaci odmocninové závislosti v rovnici (8) podle h , lze spočítat jako

$$\frac{dQ_{\text{OUT}}}{dh} = f\sqrt{2g} \frac{1}{2\sqrt{h}} = f\sqrt{2g} \frac{1}{2\sqrt{h}} \frac{\sqrt{h}}{\sqrt{h}} = \frac{f\sqrt{2gh}}{2h} = \frac{Q_{\text{OUT}}}{2h} \quad (10)$$

Derivace v pracovním bodě h_0, Q_0 je

$$\left. \frac{dQ_{\text{OUT}}}{dh} \right|_{h_0, Q_0} = \frac{Q_0}{2h_0} \quad (11)$$

V bodě linearizace (**pracovním bodu**, v ustáleném stavu) je vstupní průtok Q_{IN} i výstupní průtok Q_{OUT} roven průtoku Q_0 .

Rovnice (7) je sice lineární, ale provedeme způsobem ještě její převod do odchylových veličin. Od rovnice (7) odečteme průtok Q_0 , čímž získáme odchylové veličiny ΔQ_{IN} a ΔQ_{UOT}

$$\frac{Q_{\text{IN}} - Q_0}{\Delta Q_{\text{IN}}} = \frac{Q_{\text{OUT}} - Q_0}{\Delta Q_{\text{UOT}}} + S \frac{dh}{dt} \quad (12)$$

Pro výšku hladiny h platí vztah

$$h = h_0 + \Delta h \quad (13)$$

Pro její derivaci podle času platí

$$\frac{dh}{dt} = \frac{d(h_0 + \Delta h)}{dt} = \frac{d\Delta h}{dt} \quad (14)$$

Derivace původní i odchylové veličiny je stejná (h_0 je konstanta a její derivace je nula).

Výsledný **lineární model systému nádrže s otvorem** je

$$\begin{aligned} \Delta Q_{\text{IN}} &= \Delta Q_{\text{UOT}} + S \frac{d\Delta h}{dt} \\ \Delta Q_{\text{UOT}} &= \frac{Q_0}{2h_0} \Delta h \end{aligned} \quad (15)$$

Tyto dvě rovnice můžeme zapsat jako jednu rovnici

$$\Delta Q_{\text{IN}} = \frac{Q_0}{2h_0} \Delta h + S \frac{d\Delta h}{dt} \quad (16)$$

a upravit ji do tvaru, kde bude zřejmé zesílení k a časová konstanta systému τ

$$\underbrace{S \frac{2h_0}{Q_0}}_{\tau} \frac{d\Delta h}{dt} + \Delta h = \underbrace{\frac{2h_0}{Q_0}}_k \Delta Q_{\text{IN}} \quad (17)$$

Laplaceovou transformací získáme

$$\tau(s\Delta H(s) - \Delta h(0)) + \Delta H(s) = k\Delta Q_{\text{IN}}(s) \quad (18)$$

za předpokladu nulových počátečních podmínek $\Delta h(0) = 0$ je obrazový přenos systému

$$\frac{\Delta H(s)}{Q_{\text{IN}}(s)} = \frac{k}{\tau s + 1} \quad (19)$$

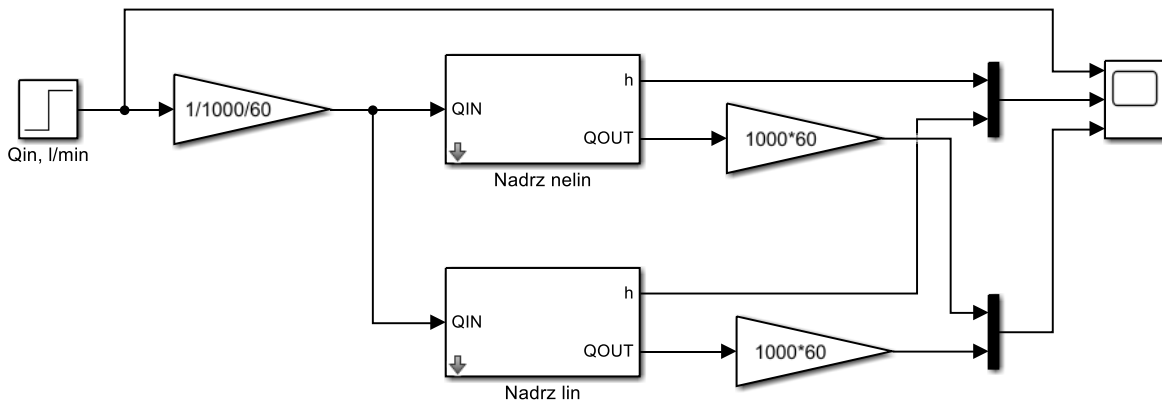
Pokud bychom chtěli vyjádřit závislost zesílení a časové konstanty na pracovním bodu ve smyslu ustálené výšky hladiny h_0 , musíme provést následující úpravu

$$k = \frac{2h_0}{Q_0} = \frac{2h_0}{f\sqrt{2gh_0}} = \frac{2h_0}{f\sqrt{2gh_0}\sqrt{h_0}} = \frac{2}{f\sqrt{2g}}\sqrt{h_0} \quad (20)$$

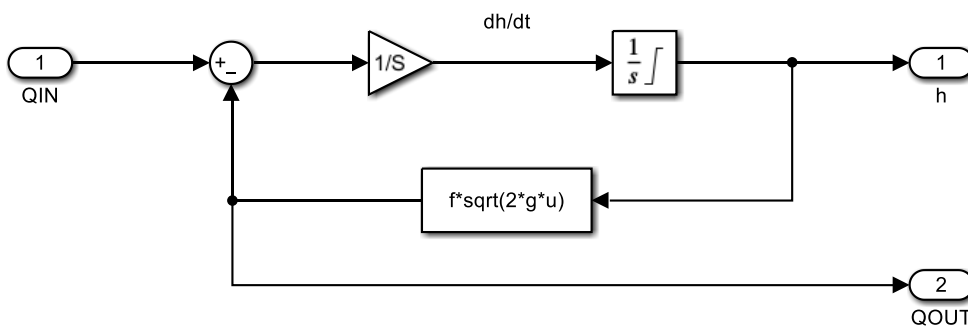
Zesílení a časová konstanta systému roste s výškou hladiny – ne lineárně, ale odmocninově.

Pro porovnání chování původního nelineárního modelu a linearizované aproximace provedeme následující simulaci. Parametry modelu jsou stejné, jak v kapitole 4.

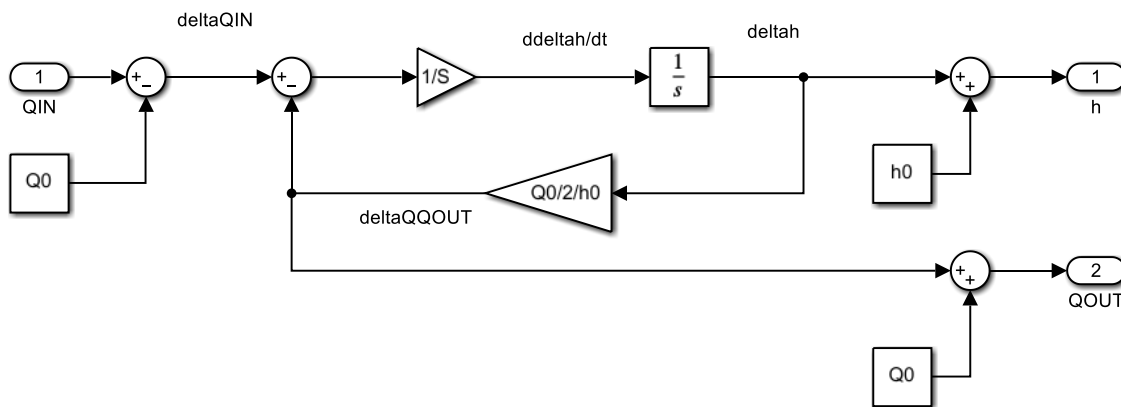
Pro simulaci sestavíme model v Simulinku



Obr. 4 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)

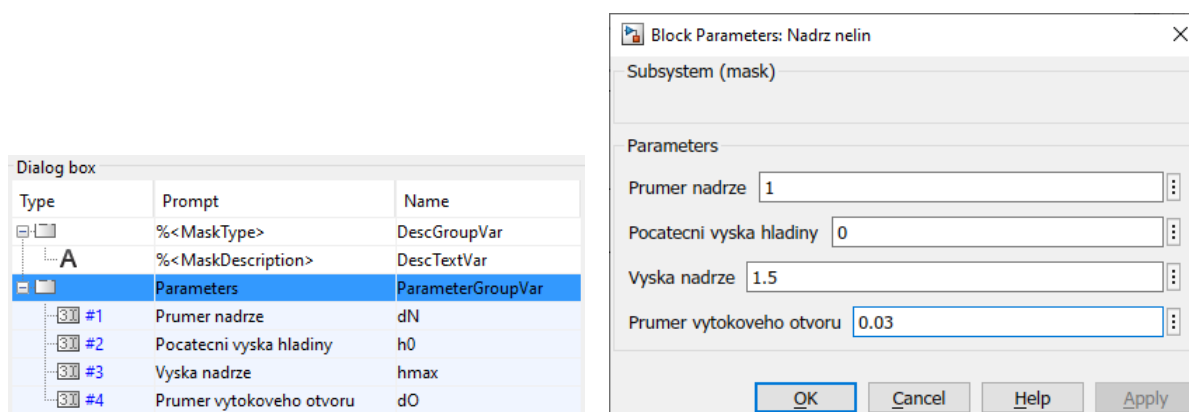


Obr. 5 – Schéma nelineárního modelu nádrže (vytvořeno v MATLAB R2021a)



Obr. 6 – Schéma lineárního modelu nádrže (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska nelineárního modelu



Obr. 7 – Maska a parametry nelineárního modelu (vytvořeno v MATLAB R2021a)

V záložce Initialization je zadán přepočít mezi průměrem a průřezem nádrže a výtakového otvoru a gravitační zrychlení.

```

Initialization commands
S=pi*dN^2/4;
f=pi*dO^2/4;
g=9.81;

```

Obr. 8 – Inicializace nelineárního subsystému (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska lineárního modelu

Dialog box		
Type	Prompt	Name
	%<MaskType>	DescGroupVar
A	%<MaskDescription>	DescTextVar
Parameters		ParameterGroupVar
#1	Prumer nadrze	dN
#2	Vyska nadrze	hmax
#3	Pracovni prtok	Q0
#4	Prumer výtokového otvoru	dO

Obr. 9 – Maska a parametry lineárního modelu (vytvořeno v MATLAB R2021a)

V záložce Initialization je zadán přepočítání mezi průměrem a průřezem nádrže a výtokového otvoru a gravitační zrychlení.

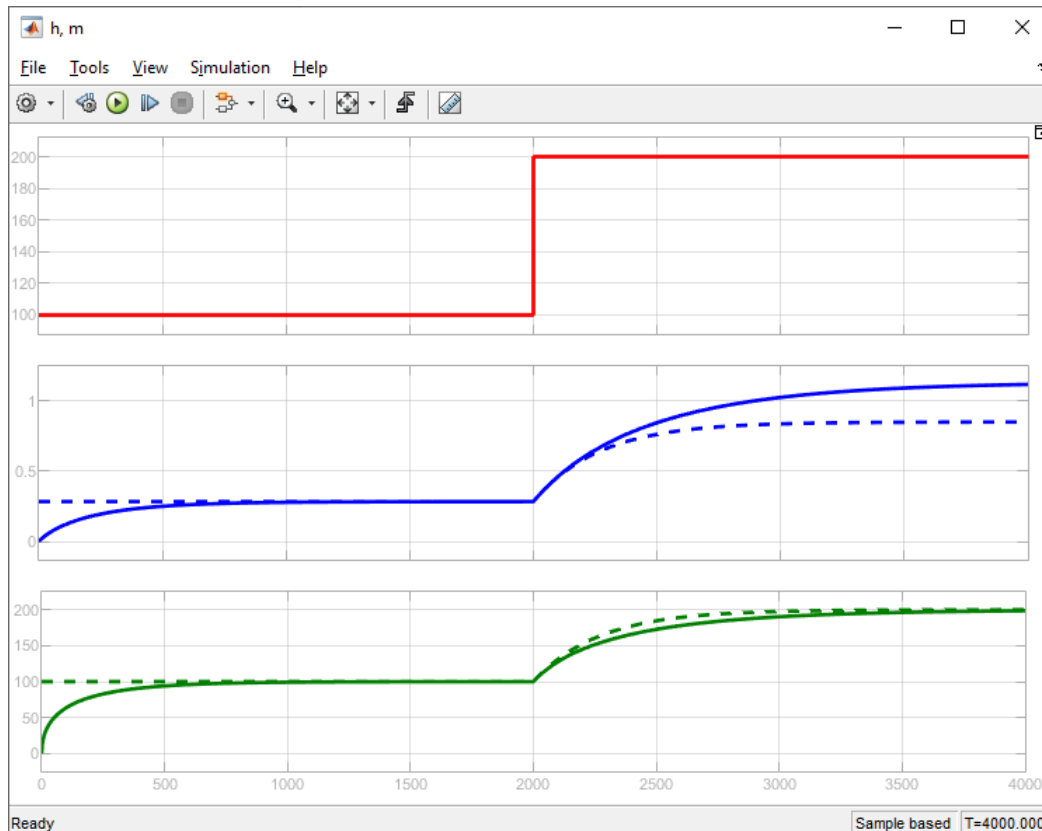
```

Initialization commands
S=pi*dN^2/4;
f=pi*dO^2/4;
g=9.81;
h0=Q0^2/f^2/2/g;

```

Obr. 10 – Inicializace lineárního subsystému (vytvořeno v MATLAB R2021a)

Průtoky jsou zadávány v $l.min^{-1}$ a pomocí bloku Gain přepočítány na $m^3.s^{-1}$. Vstupní průtok skočí v čase 2000 s z hodnoty 100 na 200 $l.min^{-1}$.



Obr. 11 – Simulace chování nelineárního a lineárního modelu nádrže (vytvořeno v MATLAB R2021a)

Na obrázku 11 je zobrazen vstupní průtok, výšky hladiny v nádržích a výstupní průtoky. Plnou čarou jsou zobrazeny výstupy nelineárního modelu, čárkovaně lineárního modelu. Výšky hladiny lineárního a nelineárního modelu jsou stejné pro první ustálený stav – pracovní bod pro průtok $100 \text{ l}\cdot\text{min}^{-1}$ je zvolen jako bod linearizace. Vlivem chyby aproximace nelineárního chování se po skokové změně průtoku na $200 \text{ l}\cdot\text{min}^{-1}$ liší průběhy hladin i výstupních průtoků. V ustáleném stavu se výstupní průtok shoduje se vstupním průtokem, ale hladiny se liší – nelineární model má pro vyšší hladinu vyšší zesílení, zatímco lineární model má zesílení odpovídající pracovnímu bodu pro průtok $100 \text{ l}\cdot\text{min}^{-1}$.

3 Analýza ustálených stavů, zesílení a časové konstanty

Pro podrobnější analýzu, jak se mění zesílení systému a časová konstanta s pracovním bodem linearizace, je vytvořen následující skript MATLABu

```
DN=1;
DO=0.03;
Q0=(0:10:200)/1000/60;

S=pi*DN^2/4;
f=pi*DO^2/4;
g=9.81;

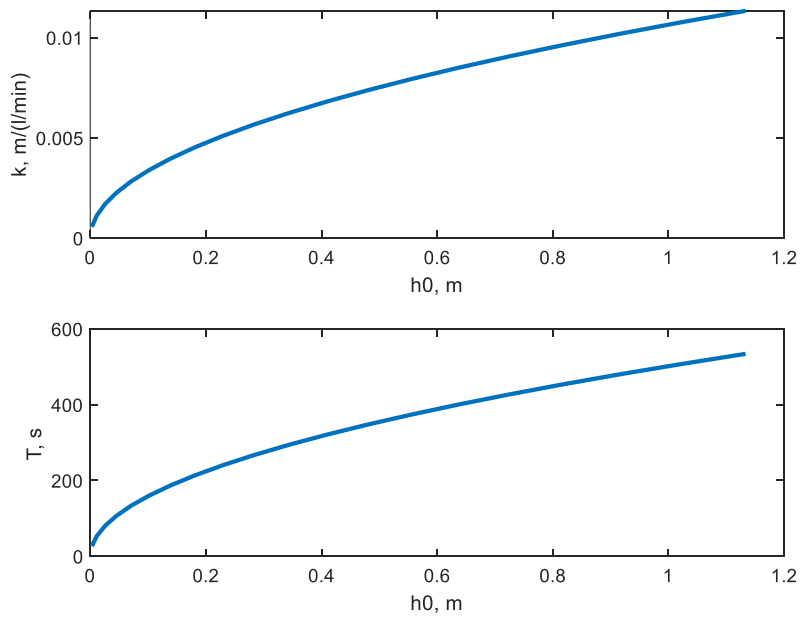
h0=Q0.^2/f^2/2/g;
k=2*h0./Q0;
T=S*k;

subplot(2,1,1), plot(h0,k/1000/60,'LineWidth',2), xlabel('h0, m'),
ylabel('k, m/(l/min)')
subplot(2,1,2), plot(h0,T,'LineWidth',2), xlabel('h0, m'), ylabel('T, s')

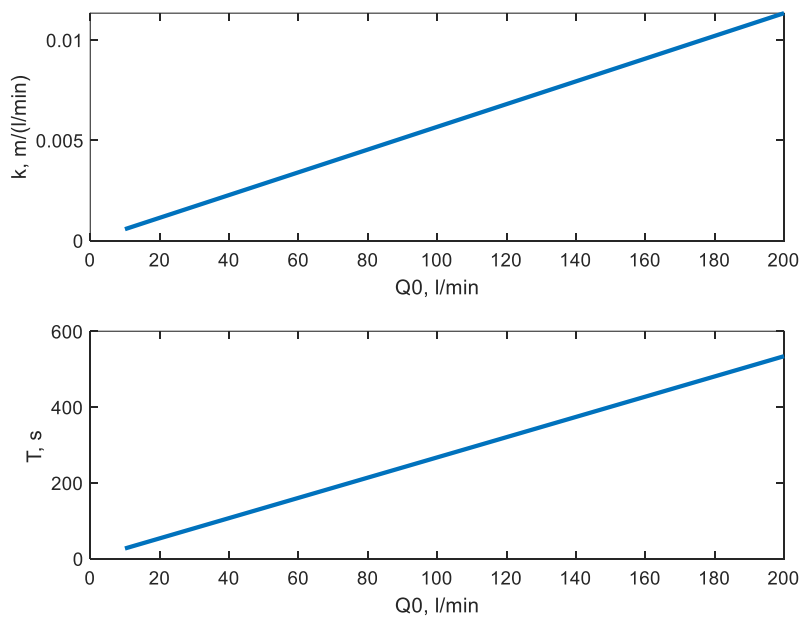
figure
subplot(2,1,1), plot(Q0*1000*60,k/1000/60,'LineWidth',2), xlabel('Q0,
l/min'), ylabel('k, m/(l/min)')
subplot(2,1,2), plot(Q0*1000*60,T,'LineWidth',2), xlabel('Q0, l/min'),
ylabel('T, s')

figure, plot(h0,Q0*1000*60,'LineWidth',2), xlabel('h0, m'), ylabel('Q0,
l/min')
```

Výsledkem jsou závislosti zesílení systému a časové konstanty na pracovním bodu linearizace (jako funkce hladiny h_0 a průtoku Q_0). Závislost parametrů na hladině je nelineární (odmocninová), jak bylo odvozeno v rovnici (20). Ovšem s průtokem se parametry mění lineárně díky mocninné závislosti hladiny na průtoku (inverzní funkci k průběhu na obrázku 14).

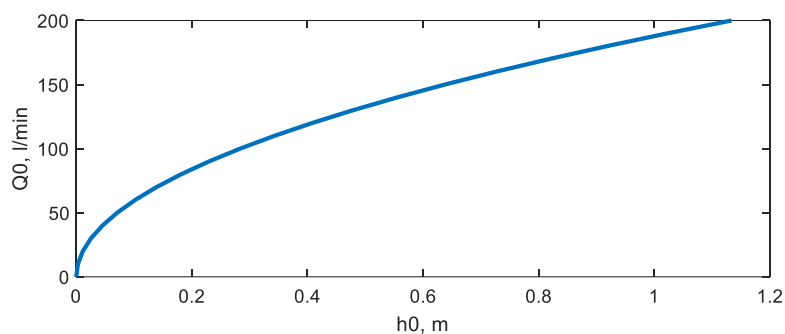


Obr. 12 – Parametry lineárního modelu v závislosti na výšce hladiny (vytvořeno v MATLAB R2021a)



Obr. 13 – Parametry lineárního modelu v závislosti na průtoku (vytvořeno v MATLAB R2021a)

Pokud vykreslíme závislost ustáleného průtoku na výšce hladiny, získáme následující graf



Obr. 14 – Závislost průtoku na hladině (vytvořeno v MATLAB R2021a)

4 Úkoly ke cvičení

1. V Simulinku vytvořte model pro porovnání chování nelineárního a lineárního modelu nádrže s otvorem – použijte schéma podle obr. 4, 5 a 6.
2. U obou subsystémů vytvořte masku s parametry podle obr. 7 a 9.
3. Proveďte simulaci chování modelu podle obr. 11.
4. Pomocí skriptu MATLABu vykreslete statickou charakteristiku nelineárního modelu nádrže s otvorem a závislosti zesílení a časové konstanty lineárního modelu na pracovním bodě.

5 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

linearizace, 1

lineární model nádrže s otvorem, 4

odchylkové veličiny, 2

pracovní bod, 4

Modelování a simulace dynamických systémů

Téma 6: Model dvou nádrží pod sebou a model dvou nádrží vedle sebe

Studijní cíl

Seznámit studenty s tvorbou modelů hydraulických systémů a vytvořit model dvou nádrží pod sebou a model dvou nádrží vedle sebe.

Doba nutná k nastudování

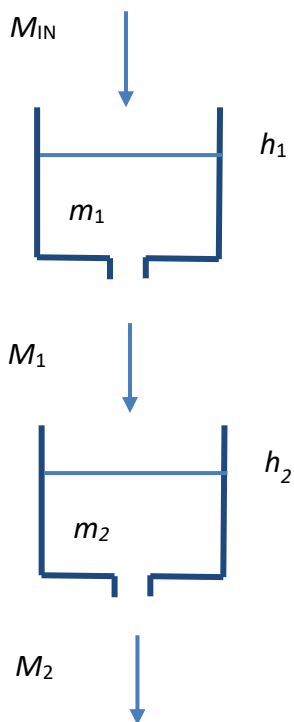
2 hodiny

Klíčová slova

Model, hydraulické systémy, dvě nádrže pod sebou, dvě nádrže vedle sebe

1 Model dvou nádrží pod sebou

Při tvorbě **modelu dvou nádrží pod sebou** (nádrží spojených do série) vyjdeme modelu nádrže s otvorem ve dně, který byl odvozen v kapitole 4 a nádrže umístíme pod sebe.



Obr. 1 – Schéma dvou nádrží pod sebou

Kapalina bude přitékat pouze do první nádrže. Budeme bilancovat zvlášť první a druhou nádrž jako dva samostatné subsystémy. Bilancí hmotnostních průtoků získáme dvě diferenciální rovnice

$$\begin{aligned} M_{\text{IN}} &= M_1 + \frac{dm_1}{dt} \\ M_1 &= M_2 + \frac{dm_2}{dt} \end{aligned} \quad (1)$$

kde M_{IN} je vstupní hmotnostní průtok do první nádrže, $\text{kg}\cdot\text{s}^{-1}$

M_1 je výstupní hmotnostní průtok z první nádrže, $\text{kg}\cdot\text{s}^{-1}$

M_2 je výstupní hmotnostní průtok z druhé nádrže, $\text{kg}\cdot\text{s}^{-1}$

$\frac{dm_1}{dt}$ je hmotnostní průtok, který se akumuloval v první nádrži, $\text{kg}\cdot\text{s}^{-1}$

$\frac{dm_2}{dt}$ je hmotnostní průtok, který se akumuloval ve druhé nádrži, $\text{kg}\cdot\text{s}^{-1}$.

Za předpokladu, že se hustota kapaliny nemění, můžeme rovnice (1) vydělit hustotou kapaliny ρ a získat tak bilanci objemových průtoků

$$\begin{aligned} Q_{\text{IN}} &= Q_1 + \frac{dV_1}{dt} \\ Q_1 &= Q_2 + \frac{dV_2}{dt} \end{aligned} \quad (2)$$

kde Q_{IN} je vstupní objemový průtok do první nádrže, $\text{m}^3\cdot\text{s}^{-1}$

Q_1 je výstupní objemový průtok z první nádrže, $\text{m}^3\cdot\text{s}^{-1}$

Q_2 je výstupní objemový průtok z druhé nádrže, $\text{m}^3\cdot\text{s}^{-1}$

$\frac{dV_1}{dt}$ je objemový průtok, který se akumuloval v první nádrži, $\text{m}^3\cdot\text{s}^{-1}$

$\frac{dV_2}{dt}$ je objemový průtok, který se akumuloval ve druhé nádrži, $\text{m}^3\cdot\text{s}^{-1}$.

Pokud jsou průřezy nádrží S_1 a S_2 konstantní (nemění se s výškou nádrží), můžeme akumulací člen vyjádřit pomocí průřezu nádrže a výšky hladiny

$$\begin{aligned} \frac{dV_1}{dt} &= S_1 \frac{dh_1}{dt} = Q_{\text{IN}} - Q_1 \\ \frac{dV_2}{dt} &= S_2 \frac{dh_2}{dt} = Q_1 - Q_2 \end{aligned} \quad (3)$$

kde S_1 je průřez první nádrže, m^2

S_2 je průřez druhé nádrže, m^2

h_1 je výška hladiny v první nádrži, m

h_2 je výška hladiny ve druhé nádrži, m .

Pro výstupní objemové průtoky z obou nádrží platí

$$Q_1 = f_1 \sqrt{2gh_1}$$

$$Q_2 = f_2 \sqrt{2gh_2} \tag{4}$$

kde f_1 je průřez výtokového otvoru první nádrže, m^2

f_2 je průřez výtokového otvoru druhé nádrže, m^2

g je gravitační zrychlení, $m.s^{-2}$.

Simulaci chování modelu – řešení rovnic (3) pro konkrétní počáteční výšky hladiny uvnitř nádrží v čase 0 (počáteční podmínky obou diferenciálních rovnic) a vstupní objemový průtok do první nádrže lze provést numericky integrací veličin $\frac{dh_1}{dt}$ a $\frac{dh_2}{dt}$ vyjádřených z rovnic (3)

$$h_1 = \int_0^t \frac{dh_1}{dt} dt = \int_0^t \frac{Q_{IN} - Q_1}{S_1} dt$$

$$h_2 = \int_0^t \frac{dh_2}{dt} dt = \int_0^t \frac{Q_1 - Q_2}{S_2} dt \tag{5}$$

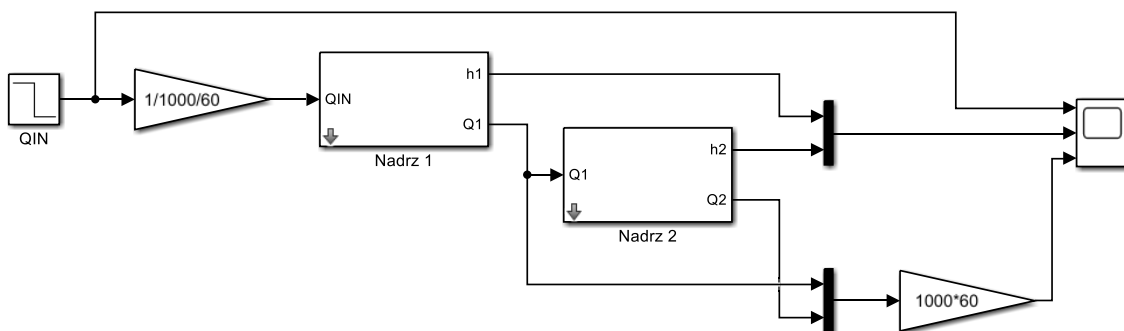
Opět budeme uvažovat nádrže s kruhovým průřezem – bude zadán průměr nádrží.

Parametry modelu jsou uvedeny v tabulce 1 – parametry jsou shodné pro obě nádrže.

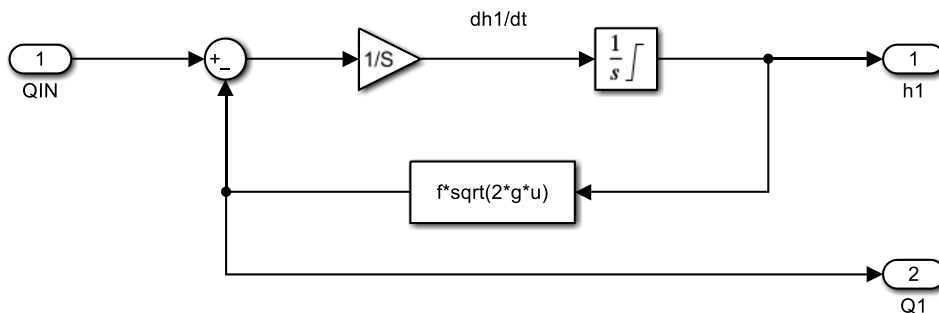
Tab. 1 – Parametry modelu dvou nádrží pod sebou

Parametr	Značka	Hodnota	Jednotka
Průměr nádrže	d_N	1	m
Počáteční výška hladiny	h_0	0	m
Výška nádrže	h_{MAX}	1,5	m
Průměr výtokového otvoru	d_o	0,03	m

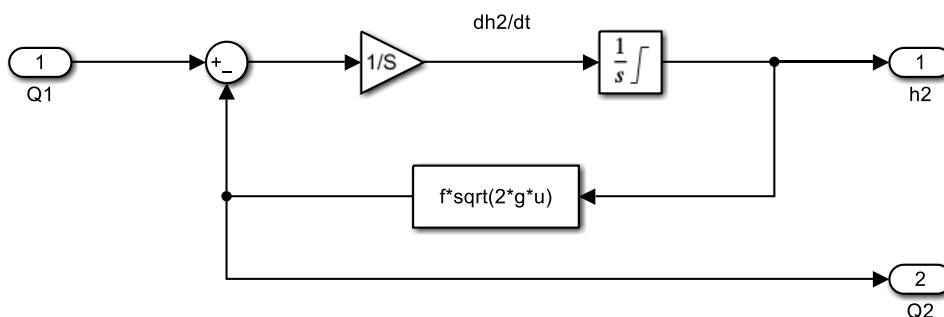
Pro simulaci sestavíme model v Simulinku



Obr. 2 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)

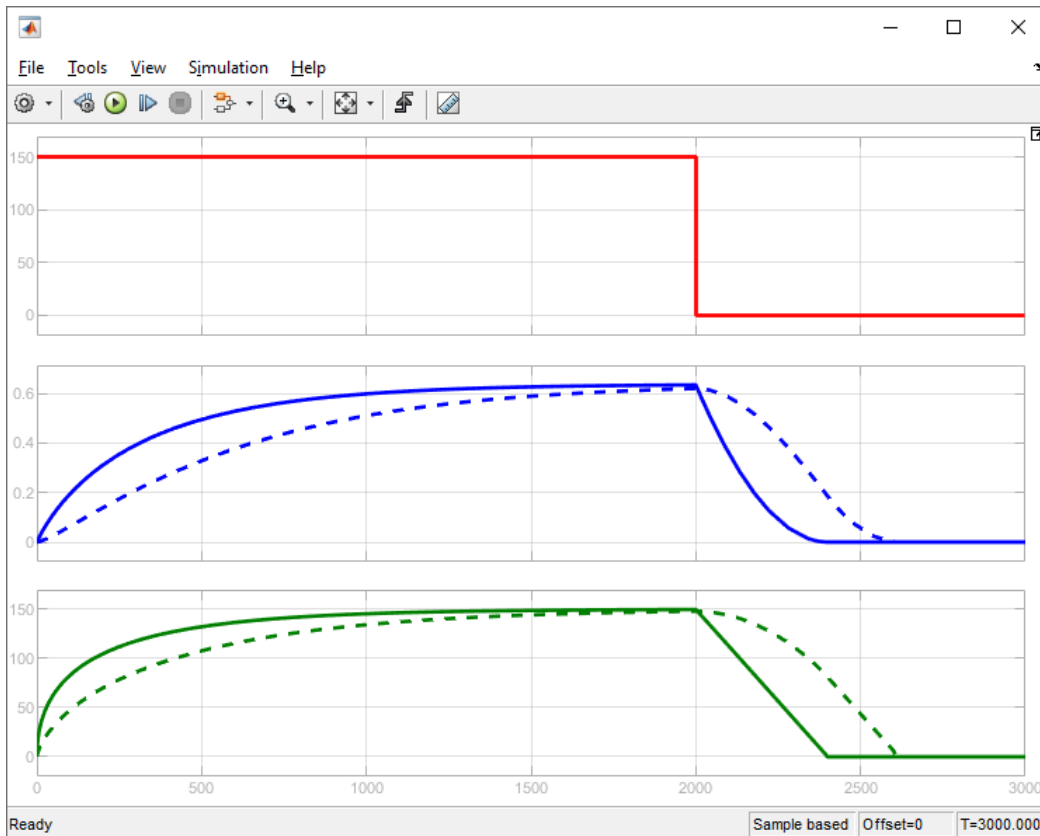


Obr. 3 – Simulační schéma subsystému první nádrže (vytvořeno v MATLAB R2021a)



Obr. 4 – Simulační schéma subsystému druhé nádrže (vytvořeno v MATLAB R2021a)

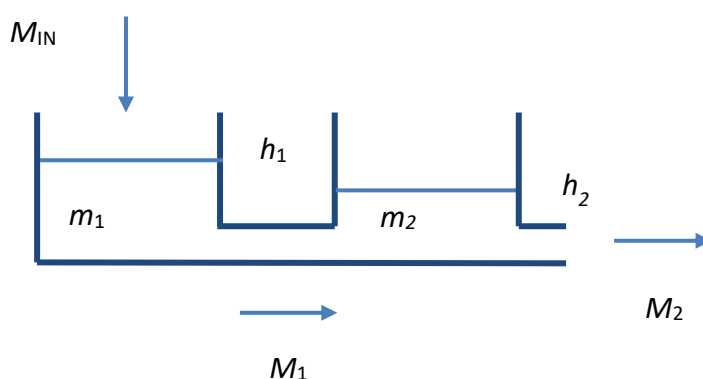
Masky subsystému (parametry i kód v záložce Initialization) jsou shodné pro obě nádrže a jsou stejné jako u modelu nádrže v kapitole 4. Také integrátory jsou nastaveny stejně jako v kapitole 4. Průtoky jsou zadávány v $l.min^{-1}$ a pomocí bloku Gain přepočítány na $m^3.s^{-1}$. Vstupní průtok skočí v čase 2000 s z hodnoty 150 na 0 $l.min^{-1}$. Na obrázku 5 je zobrazen vstupní průtok, výšky hladiny v nádržích a výstupní průtoky (plnou čarou jsou zobrazeny veličiny odpovídající první nádrži, čárkovaně jsou zobrazeny veličiny pro druhou nádrž). Z pohledu řízení se jedná o nelineární systém s dynamikou druhého řádu. Zatímco hladinu pro jednu nádrž nelze P regulátorem rozkmitat a kmity vzniknou teprve při použití PI regulátoru a nelze docílit nestability. U dvou nádrží lze rozkmitat hladinu ve druhé nádrži již použitím P regulátoru a s PI regulátorem lze dosáhnout nestabilního chování. Pro tři nádrže pod sebou by bylo možné soustavu destabilizovat již při použití P regulátoru s dostatečně velkým zesílením. Jedná se o **systém bez zpětného ovlivňování** – hladina ve druhé nádrži neovlivňuje hladinu v nádrži první.



Obr. 5 – Simulace chování modelu dvou nádrží pod sebou (vytvořeno v MATLAB R2021a)

2 Model dvou nádrží vedle sebe

Při tvorbě **modelu dvou nádrží vedle sebe** (spojených paralelně) opět vyjdeme z modelu nádrže s otvorem odvozeným v kapitole 4.



Obr. 6 – Schéma dvou nádrží vedle sebe

Bilanční rovnice budou stejné jako u dvou nádrží pod sebou s tím rozdílem, že pro průtoky mezi nádržemi (M_1 a Q_1) platí jiný vztah – velikost průtoku neurčuje pouze hladina h_1 , ale rozdíl hladin $h_1 - h_2$. Může se stát, že bude h_2 vyšší jak h_1 . V takovém případě by pod odmocninou bylo záporné číslo a výsledkem by bylo číslo komplexní. Proto musíme uvažovat absolutní

hodnotu rozdílu, což nám dá velikost průtoku a výsledek vynásobíme znaménkovou funkcí *sign* rozdílu, což určí jeho směr

$$Q_1 = \text{sign}(h_1 - h_2) f_1 \sqrt{2g|h_1 - h_2|} \quad (6)$$

Kladná hodnota průtoku znamená, že kapalina mezi nádržemi protéká z první nádrže do druhé. Záporná hodnota by znamenala opačný směr – že kapalina protéká z druhé nádrže do první. K takové situaci může v našem případě dojít pouze vlivem počátečních podmínek – hladin v čase 0, kdyby na začátku simulace byla hladina ve druhé nádrži větší než v první. Tento stav může trvat pouze přechodně. Obecně může přitékat kapalina i do druhé nádrže a z první nádrže může rovnou vytékat. Pokud by byl vstupní průtok do druhé nádrže větší než do první, protékala by mezi nádržemi kapalina z druhé nádrže do první i v ustáleném stavu.

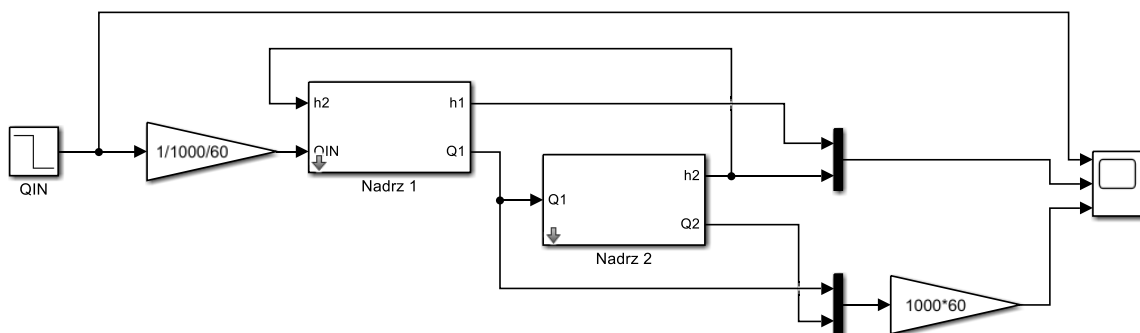
Opět budeme uvažovat nádrže s kruhovým průřezem – bude zadán průměr nádrží.

Parametry modelu jsou uvedeny v tabulce 1 – parametry jsou shodné pro obě nádrže.

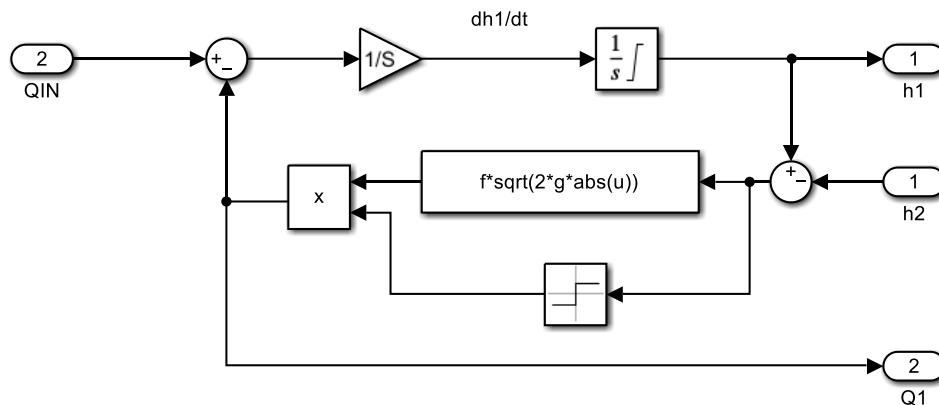
Tab. 2 – Parametry modelu dvou nádrží vedle sebe

Parametr	Značka	Hodnota	Jednotka
Průměr nádrže	d_N	1	m
Počáteční výška hladiny	h_0	0	m
Výška nádrže	h_{MAX}	1,5	m
Průměr výtokového otvoru	d_O	0,03	m

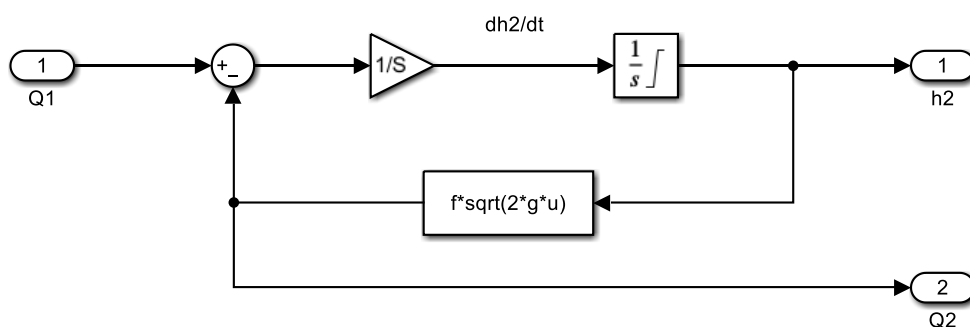
Pro simulaci sestavíme model v Simulinku



Obr. 7 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)

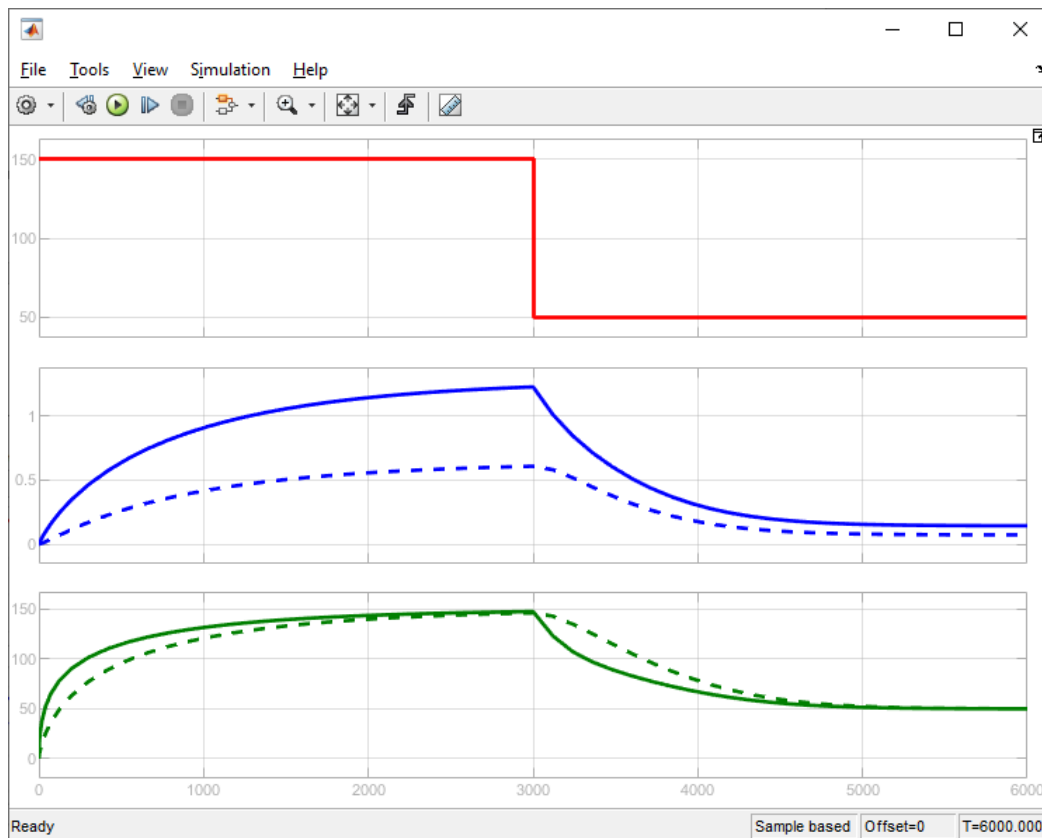


Obr. 8 – Simulační schéma subsystému první nádrže (vytvořeno v MATLAB R2021a)



Obr. 9 – Simulační schéma subsystému druhé nádrže (vytvořeno v MATLAB R2021a)

Masky subsystému (parametry i kód v záložce Initialization) i nastavení integrátorů jsou shodné pro obě nádrže a jsou stejné jako v předchozí kapitole. Průtoky jsou zadávány v $l \cdot min^{-1}$ a pomocí bloku Gain přepočítány na $m^3 \cdot s^{-1}$. Vstupní průtok skočí v čase 3000 s z hodnoty 150 na $50 l \cdot min^{-1}$. Na obrázku 10 je zobrazen vstupní průtok, výšky hladiny v nádržích a výstupní průtoky (plnou čarou jsou zobrazeny veličiny odpovídající první nádrži, čárkovaně jsou zobrazeny veličiny pro druhou nádrž). Jedná se o **system se zpětným ovlivňováním** – hladina ve druhé nádrži ovlivňuje hladinu v nádrži první. Proto musí být hladina druhé nádrže použita jako vstupní veličina nádrže první. Na základě obou hladin je počítán průtok mezi nádržemi. Z pohledu řízení je tato varianta příznivější než dvě nádrže pod sebou, neboť se jedná o takzvaný systém pseudoprvního řádu (polynom v čitateli přenosu je prvního řádu a polynom ve jmenovateli je řádu druhého – rozdíl mezi řádem jmenovatele a čitatele je 1). Proto nelze hladinu druhé nádrže rozkmitat použitím P regulátoru a s PI regulátorem nelze systém ani destabilizovat.



Obr. 10 – Simulace chování modelu dvou nádrží vedle sebe (vytvořeno v MATLAB R2021a)

3 Úkoly ke cvičení

1. V Simulinku vytvořte model dvou nádrží pod sebou – použijte schéma podle obr. 2, 3 a 4.
2. U subsystému vytvořte masku s parametry z tab. 1.
3. Proveďte simulaci chování modelu podle obr. 5.
4. V Simulinku vytvořte model dvou nádrží pod sebou – použijte schéma podle obr. 7, 8 a 9.
5. U subsystému vytvořte masku s parametry z tab. 2.
6. Proveďte simulaci chování modelu podle obr. 10.

4 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

model dvou nádrží pod sebou, 1
model dvou nádrží vedle sebe, 5

system bez zpětného ovlivňování, 4
system se zpětným ovlivňováním, 7

Modelování a simulace dynamických systémů

Téma 7: Model RC článku a model sériového RLC obvodu

Studijní cíl

Seznámit studenty s tvorbou modelů elektrických systémů a vytvořit model RC článku a model sériového RLC obvodu.

Doba nutná k nastudování

2 hodiny

Klíčová slova

Model, elektrické systémy, RC článek, sériový RLC obvod

1 Principy modelování elektrických systémů

Při modelování elektrických systémů budeme provádět bilance energie.

Bilanci energie (zákon zachování energie) za určitou jednotku času, lze zapsat jako

$$E_{\text{IN}} = E_{\text{OUT}} + E_{\text{AK}}, \quad (1)$$

kde E_{IN} je energie, která do systému vstoupila, J

E_{OUT} je energie, která ze systému vystoupila, J

E_{AK} je energie, která se v systému akumulovala, J .

Budeme pracovat se změnami energie za nekonečně malý časový úsek dt , čímž se rovnice (1) stane bilancí výkonů

$$P_{\text{IN}} = P_{\text{OUT}} + P_{\text{AK}}, \quad (2)$$

kde P_{IN} je výkon, který do systému vstoupil, W

P_{OUT} je výkon, který ze systému vystoupil, W

P_{AK} je výkon, který se v systému akumuloval, W .

Podle Ohmova zákona je napětí na rezistoru

$$u_R = Ri, \quad (3)$$

kde R je odpor rezistoru, Ω

i je proud, A.

Pro napětí na kondenzátoru platí vztah

$$u_C = \frac{1}{C} \int_0^t i dt \quad (4)$$

kde C je kapacita kondenzátoru, F.

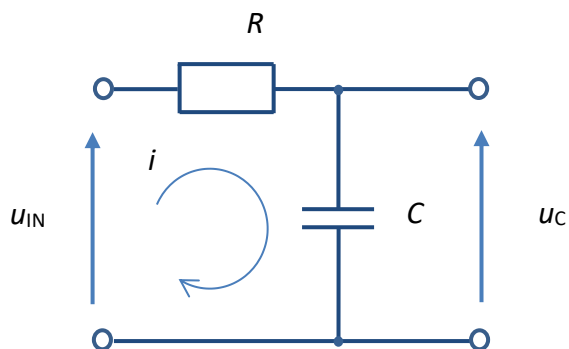
Pro napětí na cívce platí vztah

$$u_L = L \frac{di}{dt} \quad (5)$$

kde L je indukčnost cívky, H.

2 Model RC článku

Vytvoříme **model RC článku** s ideálním rezistorem a ideálním kondenzátorem



Obr. 1 – Schéma RC článku

Bilanci výkonů (2) pro RC článek můžeme zapsat jako

$$u_{IN}i = u_Ri + u_Ci, \quad (6)$$

kde u_{IN} je vstupní napětí, V

u_R je napětí na rezistoru, V

u_C je napětí na kondenzátoru, V

i je proud protékající obvodem, A.

Protože se jedná o sériový obvod, je proud v celém obvodu i . Rovnici (6) můžeme proudem vydělit a získáme bilanci napětí

$$u_{IN} = u_R + u_C \quad (7)$$

Po derivaci rovnice (4) lze vyjádřit proud protékající kondenzátorem (v našem případě celým obvodem)

$$i = C \frac{du_C}{dt} \quad (8)$$

Dosažením (8) do rovnice (3) a poté za u_R do rovnice (7) získáme model RC článku

$$RC \frac{du_C}{dt} + u_C = \frac{1}{k} u_{IN} \quad (9)$$

Součin RC je časová konstanta systému τ . zesílení systému k je 1, což znamená, že se v ustáleném stavu na napětí kondenzátoru dostaneme na hodnotu napájecího napětí.

Simulaci chování modelu – řešení rovnice (9) pro konkrétní počáteční napětí na kondenzátoru v čase 0 (počáteční podmínku diferenciální rovnice) a vstupní napětí lze provést numericky integrací veličiny $\frac{du_C}{dt}$ vyjádřené z rovnice (9)

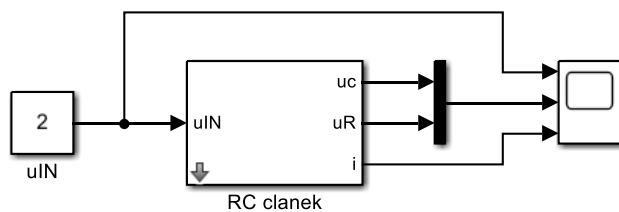
$$u_C = \int_0^t \frac{du_C}{dt} dt = \int_0^t \frac{u_{IN} - u_C}{RC} dt \quad (10)$$

Parametry modelu jsou uvedeny v tabulce 1.

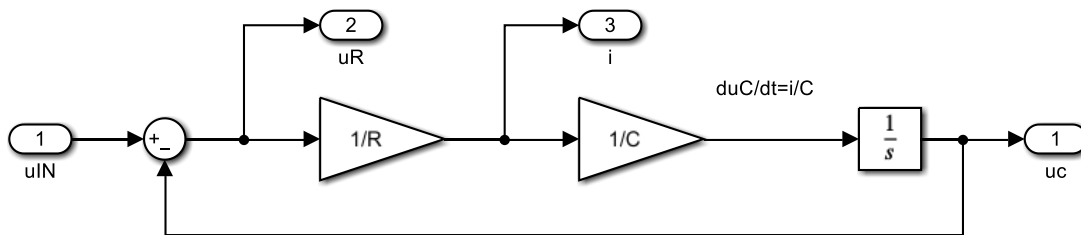
Tab. 1 – Parametry modelu RC článku

Parametr	Značka	Hodnota	Jednotka
Odpor rezistoru	R	$1 \cdot 10^3$	Ω
Kapacita kondenzátoru	C	$1 \cdot 10^{-3}$	F
Počáteční napětí na kondenzátoru	u_{c0}	0	V

Pro simulaci sestavíme model v Simulinku

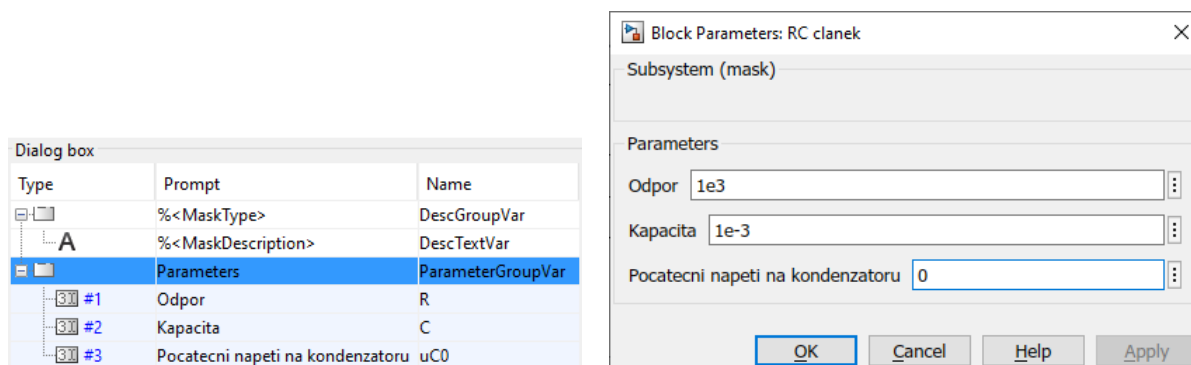


Obr. 2 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)



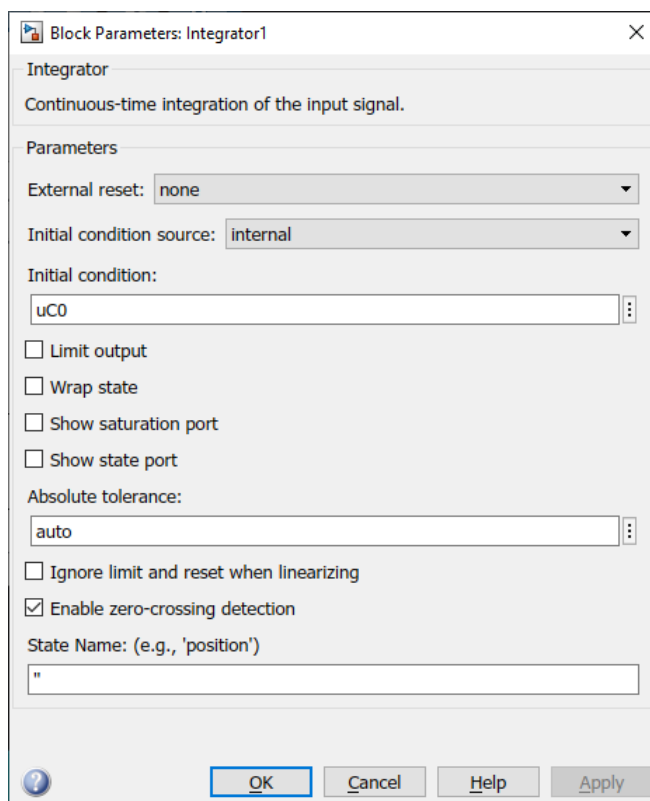
Obr. 3 – Simulační schéma subsystému RC článku (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska subsystému



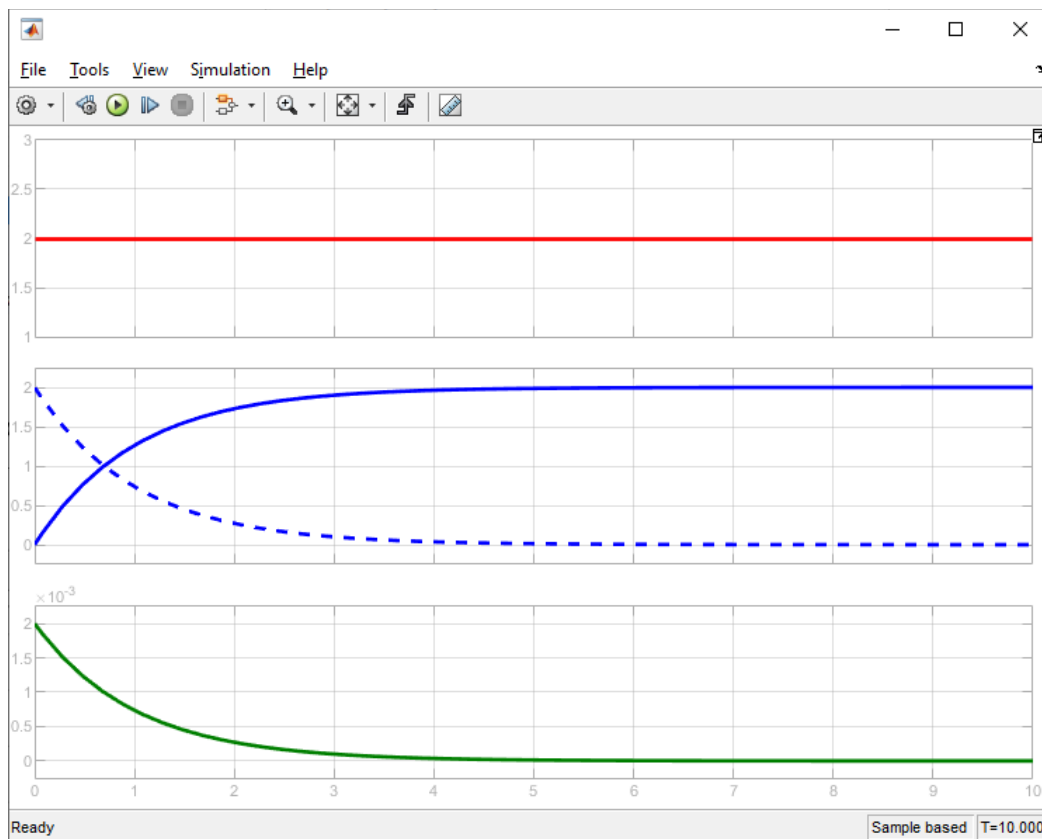
Obr. 4 – Maska a parametry modelu (vytvořeno v MATLAB R2021a)

Abychom mohli zadávat napětí na kondenzátoru v čase 0 je nutné v „Initial condition“ bloku integrátoru zadat proměnnou u_{C0} .



Obr. 5 – Parametry bloku integrátor (vytvořeno v MATLAB R2021a)

Pro vstupní napětí 2 V je provedena simulace modelu.

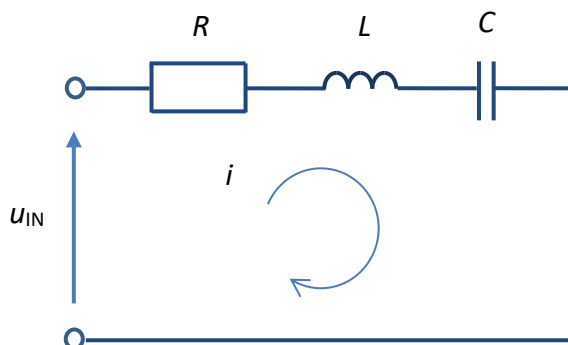


Obr. 6 – Simulace chování modelu RC článku (vytvoreno v MATLAB R2021a)

Na obrázku 6 je zobrazeno vstupní napětí, napětí na kondenzátoru (plnou čarou) a napětí na odporu (čárkovaně) a proud protékající obvodem. Jedná se o nabíjení kondenzátoru. Kdybychom zadali nenulové počáteční napětí na kondenzátoru a vstupní napětí nula, získali bychom průběh vybíjení kondenzátoru.

3 Model sériového RLC obvodu

Vytvoříme **model sériového RLC obvodu** s ideálním rezistorem, ideálním kondenzátorem a ideální cívkou



Obr. 7 – Schéma sériového RLC obvodu

Bilanci výkonů (2) pro sériový RLC obvod můžeme zapsat jako

$$u_{IN}i = u_R i + u_L i + u_C i, \quad (11)$$

kde u_{IN} je vstupní napětí, V
 u_R je napětí na rezistoru, V
 u_L je napětí na cívce, V
 u_C je napětí na kondenzátoru, V
 i je proud protékající obvodem, A .

Protože se jedná o sériový obvod, je proud v celém obvodu i . Rovnici (11) můžeme proudem vydělit a získáme bilanci napětí

$$u_{IN} = u_R + u_L + u_C \quad (12)$$

Za u_R dosadíme vztah (3), za u_L vztah (5) a u_C vztah (4) a získáme model sériového RLC obvodu

$$u_{IN} = Ri + L \frac{di}{dt} + \frac{1}{C} \int_0^t i dt \quad (13)$$

Jedná se o integro-diferenciální rovnici. Simulaci chování modelu – řešení rovnice (13) pro konkrétní počáteční napětí na kondenzátoru v čase 0 a proud v čase nula (počáteční podmínky integro-diferenciální rovnice) a vstupní napětí lze provést numericky integrací veličiny $\frac{di}{dt}$ vyjádřené z rovnice (13)

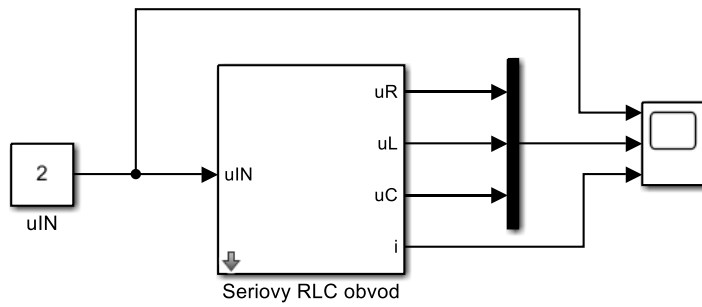
$$i = \int_0^t \frac{di}{dt} dt = \int_0^t \frac{u_{IN} - Ri - \frac{1}{C} \int_0^t i dt}{L} dt \quad (14)$$

Parametry modelu jsou uvedeny v tabulce 1.

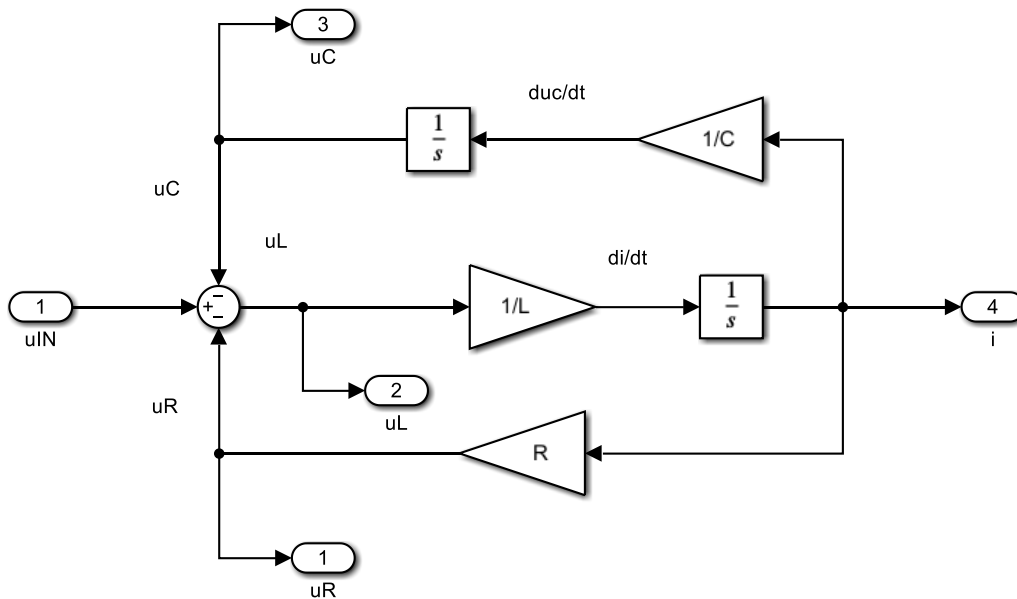
Tab. 2 – Parametry modelu sériového RLC obvodu

Parametr	Značka	Hodnota	Jednotka
Odpor rezistoru	R	$1 \cdot 10^3$	Ω
Indukčnost cívky	L	1	H
Kapacita kondenzátoru	C	$1 \cdot 10^{-6}$	F
Počáteční napětí na kondenzátoru	u_{c0}	0	V
Počáteční proud	i_0	0	A

Pro simulaci sestavíme model v Simulinku



Obr. 8 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)



Obr. 9 – Simulační schéma subsystému sériového RLC obvodu (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry modelu je vytvořena maska subsystému

Type	Prompt	Name
	%<MaskType>	DescGroupVar
A	%<MaskDescription>	DescTextVar
Parameters		ParameterGroupVar
#1	Odpor	R
#2	Indukčnost	L
#3	Kapacita	C
#4	Počáteční napětí na kondenzátoru	uC0
#5	Počáteční proud	i0

Block Parameters: Seriovy RLC obvod

Subsystem (mask)

Parameters

Odpor

Indukčnost

Kapacita

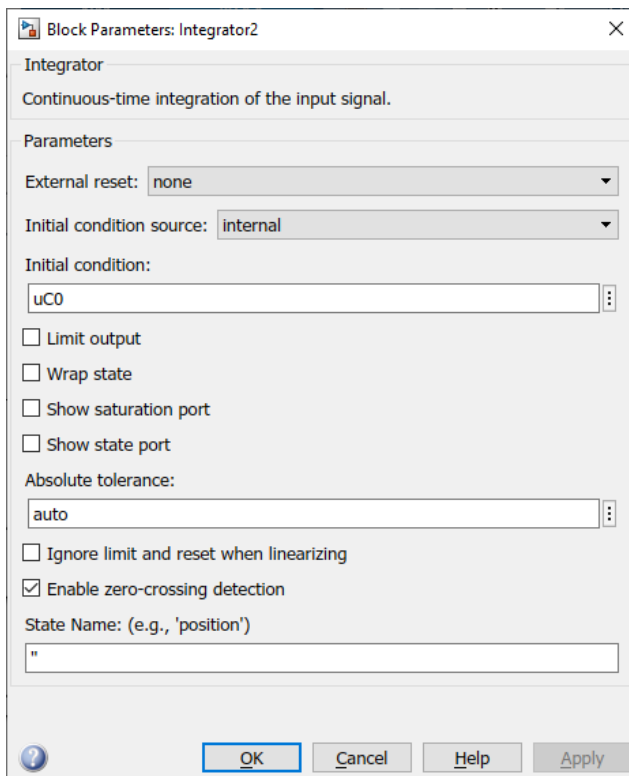
Počáteční napětí na kondenzátoru

Počáteční proud

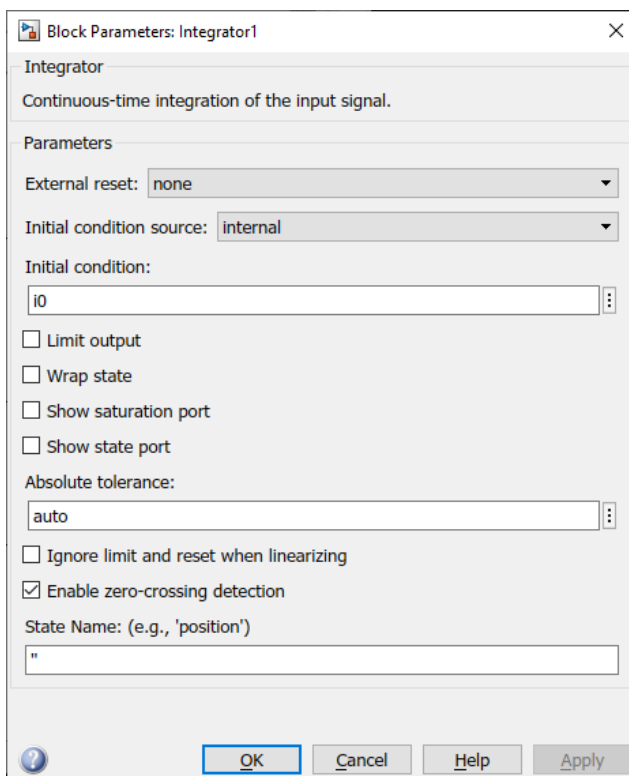
OK Cancel Help Apply

Obr. 10 – Maska a parametry modelu (vytvořeno v MATLAB R2021a)

Abychom mohli zadávat napětí na kondenzátoru a proud v čase 0 je nutné v „Initial condition“ bloku integrátor napětí na kondenzátoru zadat proměnnou u_{C0} a v bloku integrátor proudu zadat proměnnou i_0 .

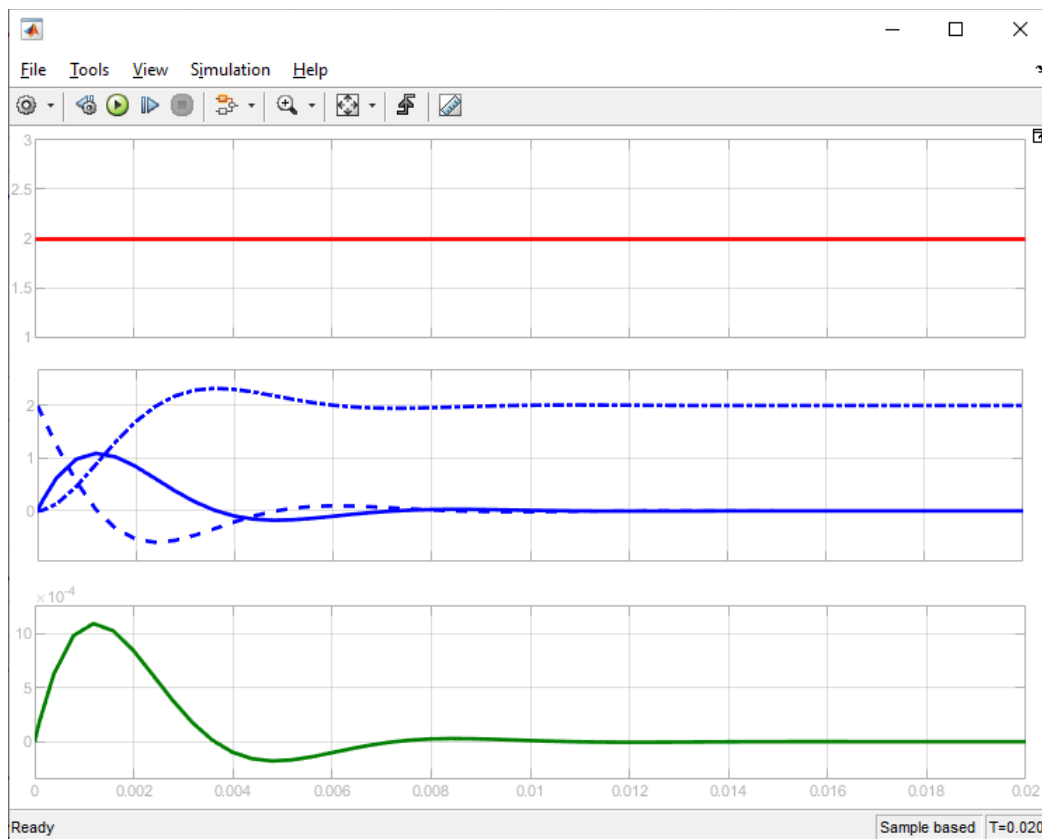


Obr. 11 – Parametry bloku integrátor napětí na kondenzátoru (vytvořeno v MATLAB R2021a)



Obr. 12 – Parametry bloku integrátor proudu (vytvořeno v MATLAB R2021a)

Pro vstupní napětí 2 V je provedena simulace modelu.



Obr. 13 – Simulace chování modelu sériového RLC obvodu (vytvořeno v MATLAB R2021a)

Na obrázku 13 je zobrazeno vstupní napětí, napětí na odporu (plnou čarou), napětí na cívce (čárkovaně) a napětí na kondenzátoru (čerchovaně) a proud protékající obvodem. Je vidět, že v ustáleném stavu se napětí na kondenzátoru rovná hodnotě vstupního napětí. Protože je proud nulový, tak je napětí na odporu nula. V ustáleném stavu se proud nemění, a proto je napětí na cívce také nulové. Hodnota proudu během přechodového děje změnila znaménko – proud procházel po určitou dobu opačným směrem.

4 Úkoly ke cvičení

1. V Simulinku vytvořte model RC článku – použijte schéma podle obr. 2 a 3.
2. U subsystému vytvořte masku s parametry podle obr. 4.
3. Proveďte simulaci chování modelu podle obr. 6.
4. V Simulinku vytvořte model sériového RLC obvodu – použijte schéma podle obr. 8 a 9.
5. U subsystému vytvořte masku s parametry podle obr. 10.
6. Proveďte simulaci chování modelu podle obr. 13.

5 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

model RC článku, 2

model sériového RLC obvodu, 5

Modelování a simulace dynamických systémů

Téma 8: Model soustavy kulička na tyči

Studijní cíl

Seznámit studenty s tvorbou modelů mechanických systémů a vytvořit model laboratorní soustavy kulička na tyči včetně ošetření situace, kdy kulička dosáhne krajních poloh (dorazů).

Doba nutná k nastudování

2 hodiny

Klíčová slova

Model, mechanické systémy, kulička na tyči

1 Principy modelování mechanických systémů

Při modelování mechanických systémů budeme vycházet z **Newtonových zákonů**. Podle **prvního Newtonova zákona** platí „jestliže na těleso (hmotu) nepůsobí žádné vnější síly, nebo výslednice sil je 0, pak těleso setrvává v klidu nebo v rovnoměrném přímočarém pohybu“. **Druhý Newtonův zákon** zní „jestliže na těleso (hmotu) působí síla, pak se těleso pohybuje zrychlením, které je přímo úměrné působící síle a nepřímo úměrné hmotnosti tělesa“

$$F = ma, \tag{1}$$

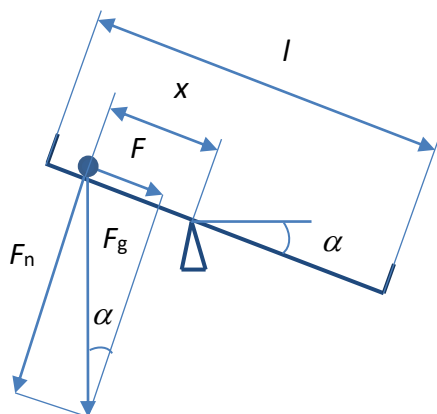
kde F je síla způsobující pohyb, N

m hmotnost tělesa, kg

a zrychlení tělesa, $m.s^{-2}$.

2 Model kuličky na tyči

Vytvoříme **model laboratorní soustavy kulička na tyči** - nakloněná lineární dráha, po které se pohybuje kulička



Obr. 1 – Schéma soustavy kulička na tyči a rozklad sil

Dráha o délce l je nakloněná o úhel α oproti vodorovné poloze, poloha kuličky x se měří od středu bodu otáčení. Na kuličku působí gravitační síla F_g , která se podle úhlu naklonění rozkládá do normálového a tečného směru na síly F_n a F . Síla F způsobuje pohyb kuličky.

$$F = F_g \sin(\alpha), \quad (2)$$

kde F je síla způsobující pohyb kuličky, N
 F_g je gravitační síla působící na kuličku, N
 a je naklonění dráhy kuličky, $^\circ$.

Gravitační síla F_g je

$$F_g = mg, \quad (3)$$

kde m je hmotnost kuličky, kg
 g je gravitační zrychlení, $m \cdot s^{-2}$.

Podle druhého Newtonova zákona platí, že se kulička pohybuje se zrychlením

$$a = \frac{F}{m} = g \sin(\alpha) = \frac{dv}{dt} = \frac{d^2x}{dt^2} \quad (4)$$

kde v je rychlost kuličky, $m \cdot s^{-1}$
 x je poloha kuličky, m .

Nezáleží na hmotnosti kuličky, ale pouze na naklonění dráhy.

Simulaci chování modelu – řešení rovnice (4) pro konkrétní počáteční polohu a rychlost kuličky v čase 0 (počáteční podmínky diferenciální rovnice) a úhel naklonění dráhy lze provést numericky dvojitou integrací veličiny $\frac{d^2x}{dt^2}$ vyjádřené z rovnice (4)

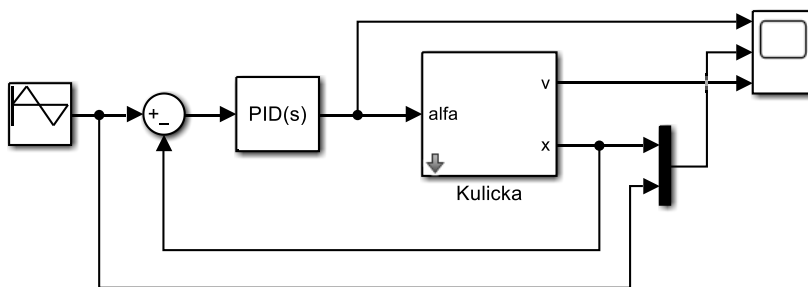
$$x = \int_0^t \int_0^t \frac{d^2x}{dt^2} dt^2 = \sin(\alpha) \int_0^t \underbrace{\left(\int_0^t g dt \right)}_v dt \quad (5)$$

Parametry modelu jsou uvedeny v tabulce 1.

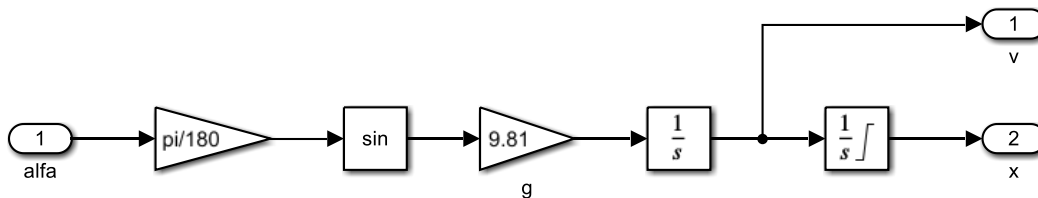
Tab. 1 – Parametry modelu kuličky na tyči

Parametr	Značka	Hodnota	Jednotka
Délka dráhy	l	0,4	m

Pro simulaci sestavíme model v Simulinku. Protože se jedná o nestabilní systém (dvojitý integrátor), budeme simulovat chování v uzavřeném regulačním obvodu s PD regulátorem.

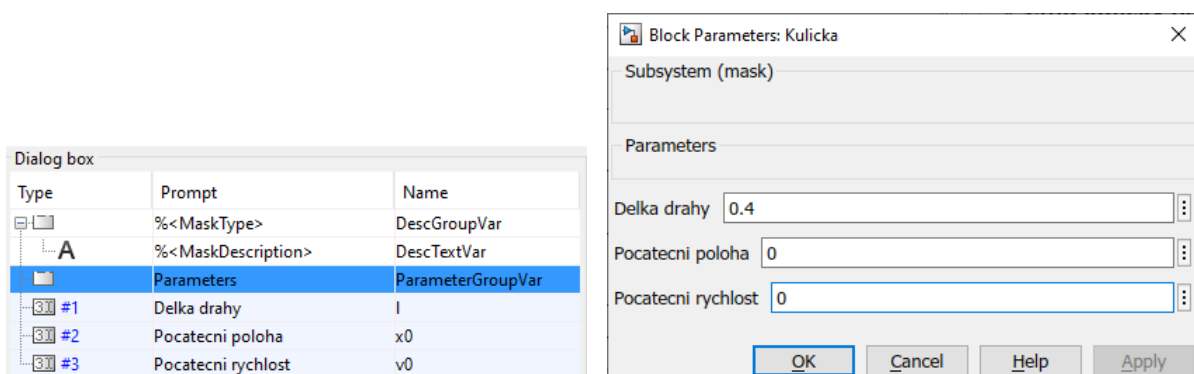


Obr. 2 – Simulační schéma modelu (vytvořeno v MATLAB R2021a)



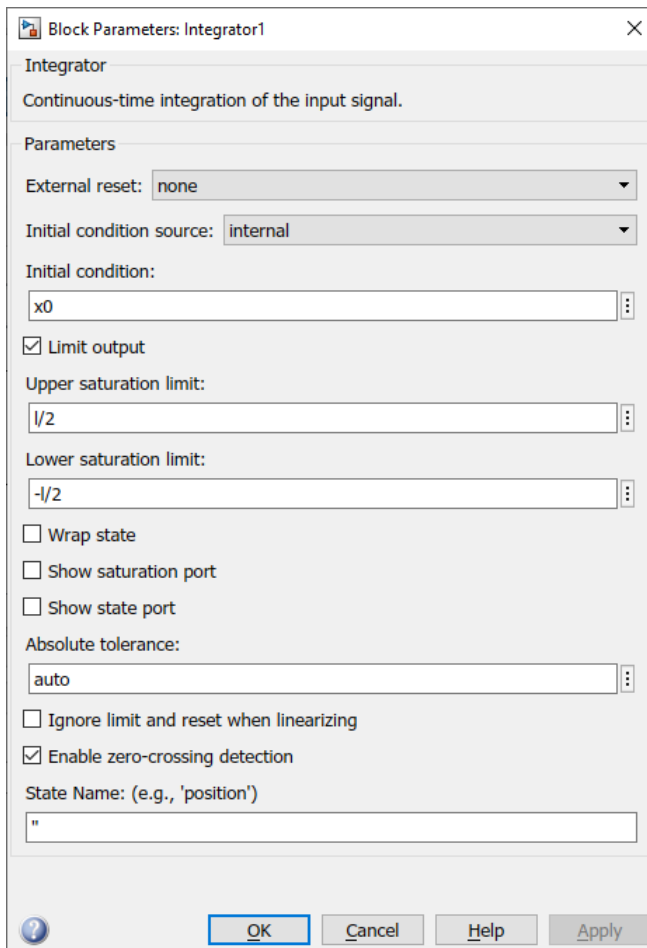
Obr. 3 – Simulační schéma subsystému kuličky na tyči (vytvořeno v MATLAB R2021a)

Pro možnost zadat parametry počáteční podmínky je vytvořena maska subsystému

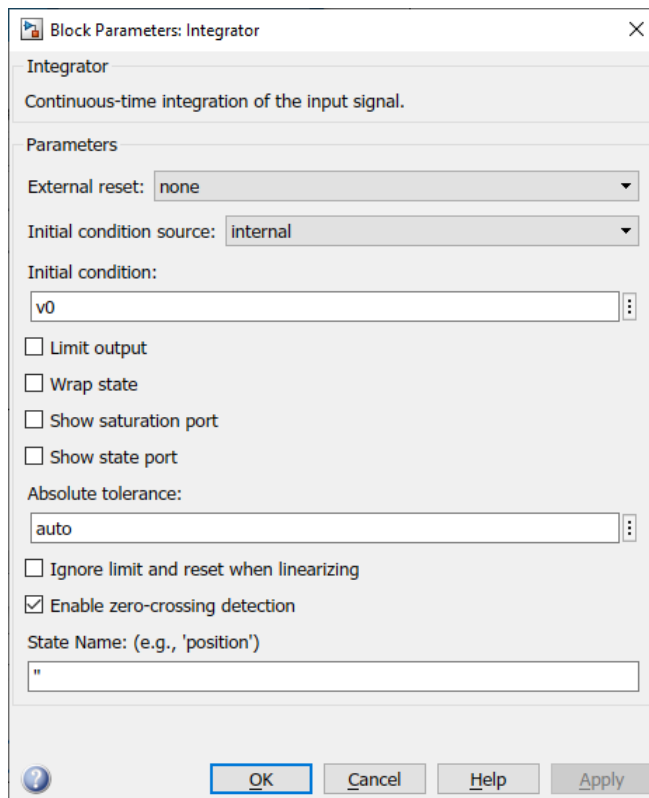


Obr. 4 – Maska a parametry modelu (vytvořeno v MATLAB R2021a)

Abychom mohli zadávat délku dráhy a polohu kuličky v čase 0 (na začátku simulace) je nutné v bloku integrátor polohy zadat omezení integrace „Upper saturation limit“ na hodnotu $l/2$, „Lower saturation limit“ na hodnotu $-l/2$ a „Initial condition“ na hodnotu x_0 . Abychom mohli zadávat rychlost kuličky v čase 0 musíme v bloku integrátor rychlosti zadat „Initial condition“ na hodnotu v_0 .

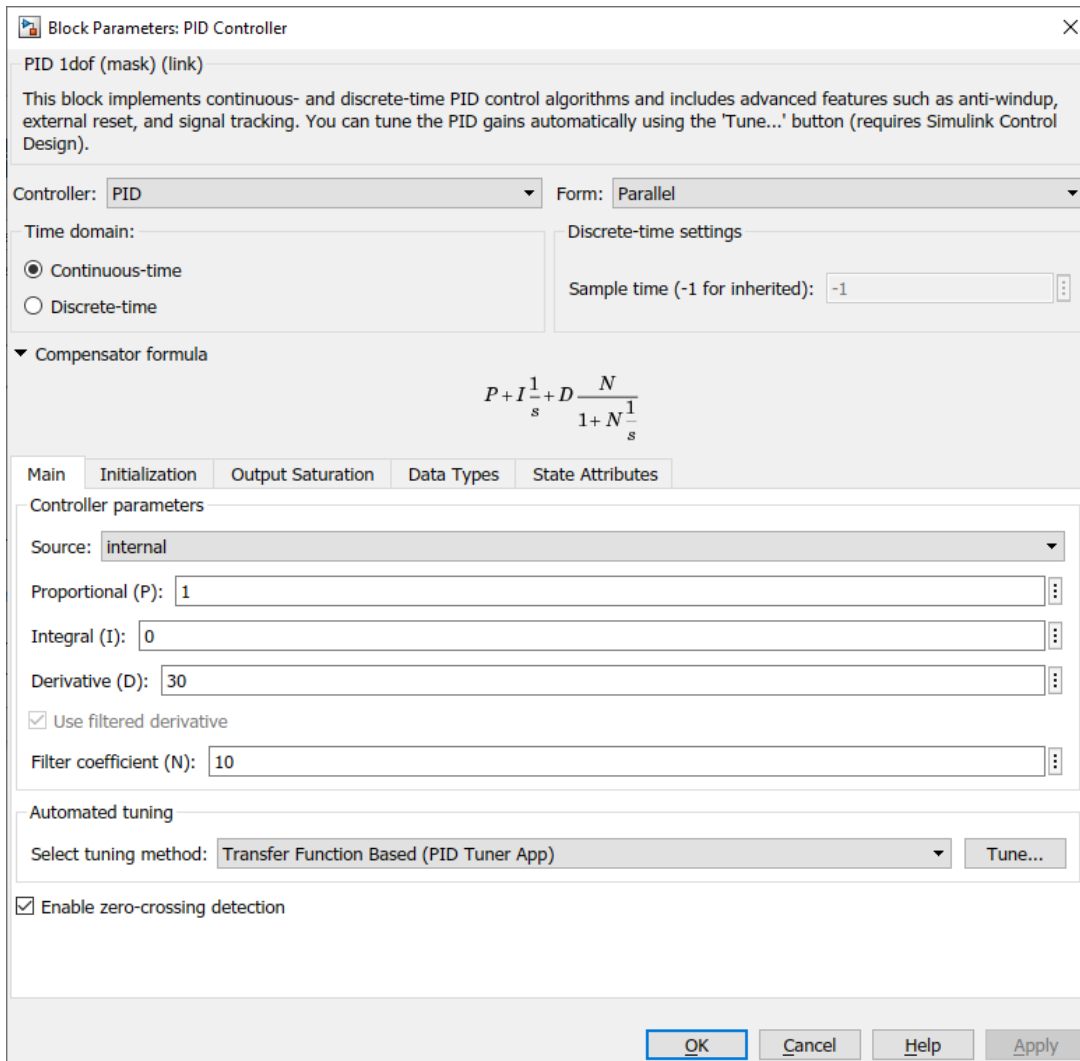


Obr. 5 – Parametry bloku integrátor polohy (vytvořeno v MATLAB R2021a)



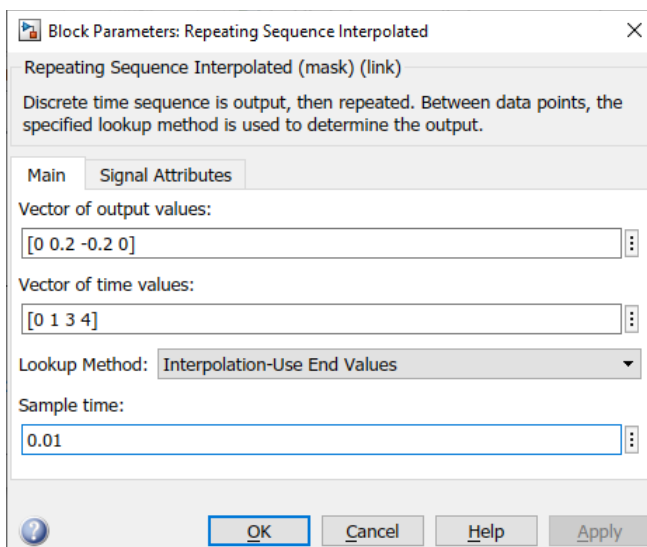
Obr. 6 – Parametry bloku integrátor rychlosti (vytvořeno v MATLAB R2021a)

Dále je nutné nastavit parametry PD regulátoru v bloku PID Controller. Je použita pouze proporcionální a derivační složka, která je filtrována filtrem typu dolní propust (proporcionální systém prvního řádu se zesílením 1 a časovou konstantou $1/N$ s).

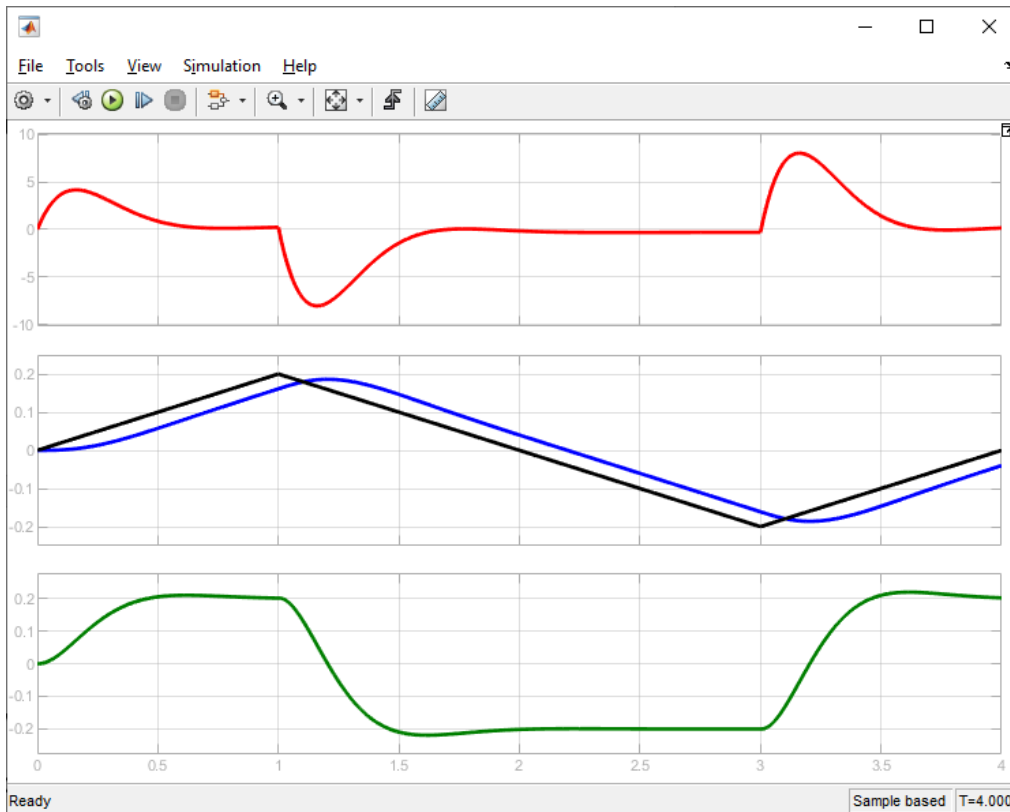


Obr. 7 – Parametry bloku PID regulátor (vytvořeno v MATLAB R2021a)

Pro generování žádané hodnoty použijeme blok Repeating Sequence Interpolated a zadáme v něm pilový průběh periodou 3 s a amplitudou 0,2.



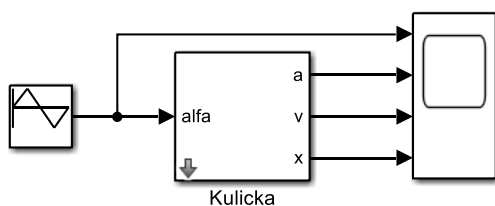
Obr. 8 – Parametry bloku pro generování žádané hodnoty (vytvořeno v MATLAB R2021a)



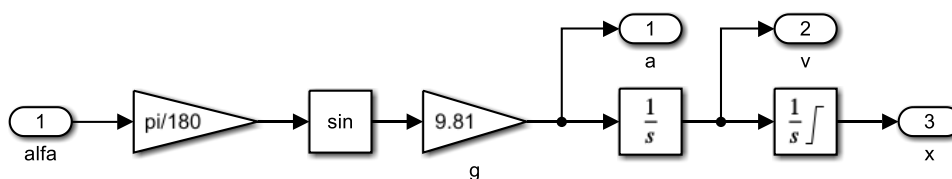
Obr. 9 – Simulace chování modelu kuličky na tyči (vytvořeno v MATLAB R2021a)

Na obrázku 9 je zobrazen úhel naklonění dráhy kuličky, žádaná hodnota polohy kuličky (černou čarou) a regulovaná poloha kuličky (modře) a rychlost kuličky.

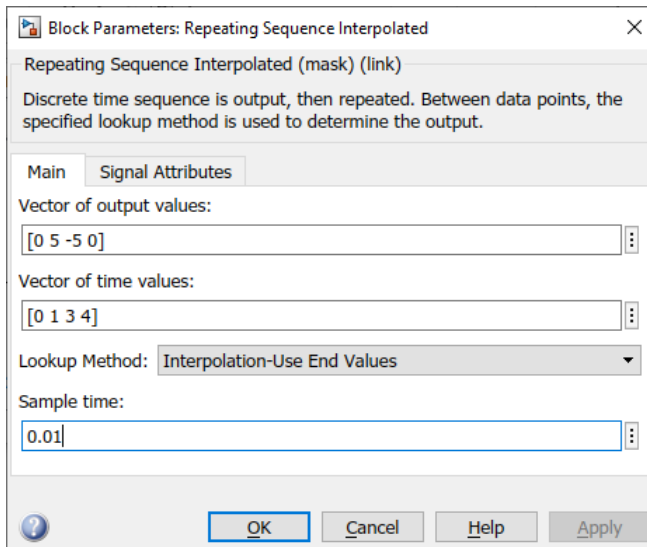
Mohlo by se zdát, že je vše v pořádku, ale pokud bychom nepoužili zpětnovazební regulátor, všimli bychom si, že se model při dosažení krajní polohy nechová korektně. Místo žádané hodnoty budeme rovnou nastavovat úhel naklonění dráhy kuličky v rozmezí od -5 do 5° . Do subsystému kuličky na tyči umístíme další výstupní port s informací o zrychlení a .



Obr. 10 – Simulační schéma modelu bez regulátoru (vytvořeno v MATLAB R2021a)

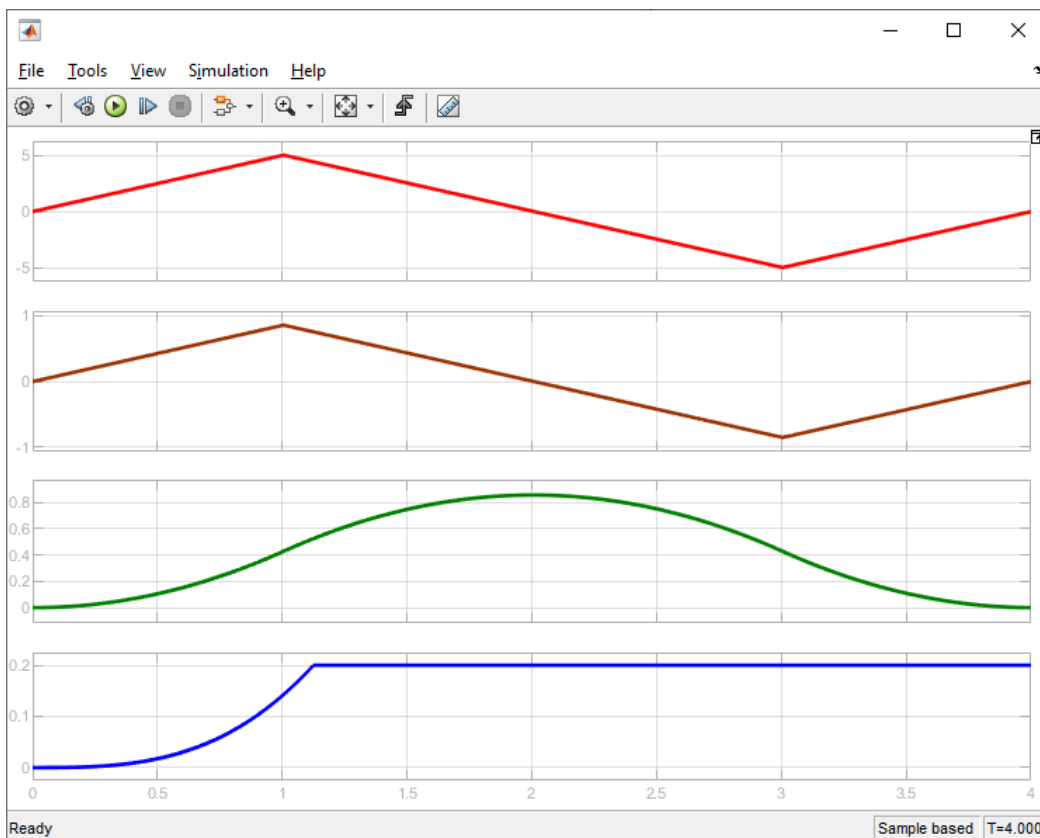


Obr. 11 – Simulační schéma subsystému kuličky na tyči (vytvořeno v MATLAB R2021a)



Obr. 12 – Parametry bloku pro generování úhlu naklonění dráhy kuličky (vytvoreno v MATLAB R2021a)

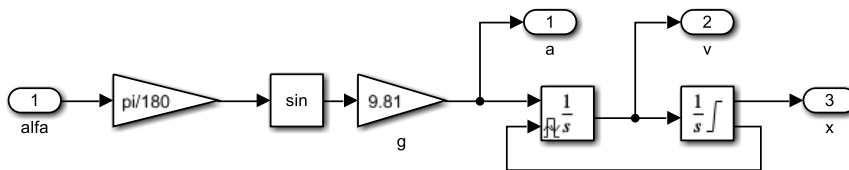
Výsledek simulace je následující



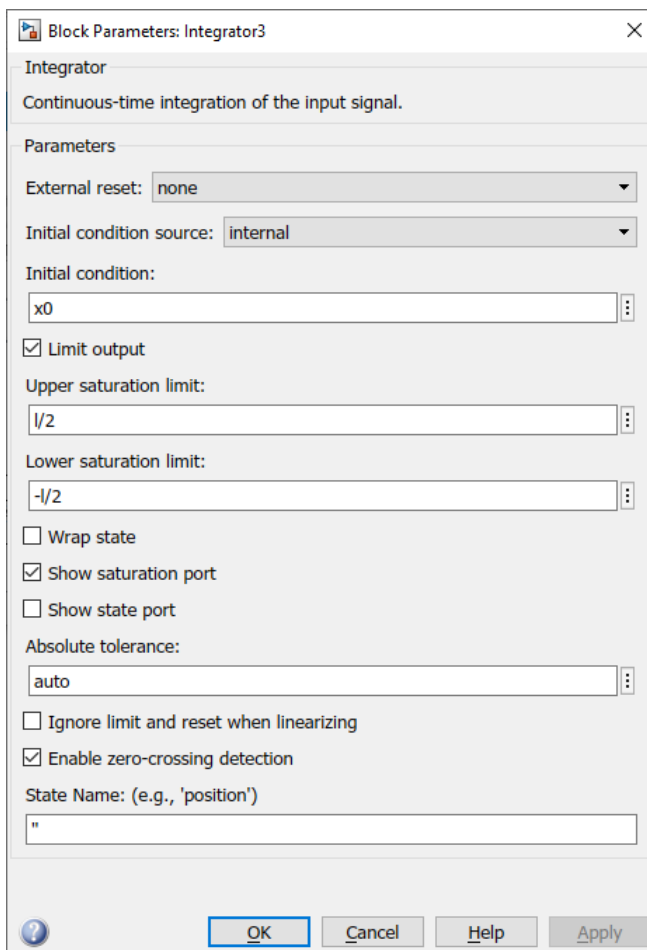
Obr. 13 – Simulace chování modelu kulička na tyči bez regulátoru (vytvoreno v MATLAB R2021a)

Na obrázku 13 je zobrazen úhel naklonění dráhy kuličky, zrychlení, rychlost a poloha kuličky. Jakmile kulička dojde do krajní polohy, tak se sice poloha omezí, ale kulička zůstane u dorazu a už se nevrátí ani když se dráha natočí na druhou stranu. Dalším problémem je, že je poloha kuličky omezená, ale rychlost nadále stoupá (měla by být nulová – kulička by měla zastavit).

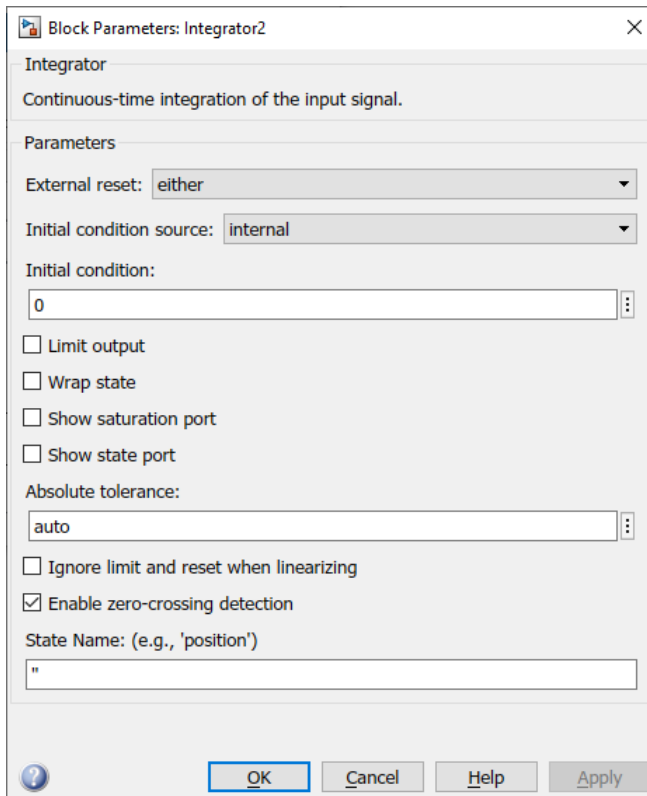
Je to způsobeno tím, že zrychlení je určeno úhlem natočení, rychlost je jeho integrací a model se chová z pozice zrychlení a rychlosti, jako kdyby kulička na doraz vůbec nenarazila. Řešení není úplně jednoduché. Musíme zajistit, že se při dosažení krajní polohy zastaví rychlost. To lze udělat tak, že u bloku integrátor polohy necháme zobrazit, že došlo k omezení volbou „Show saturation port“. Tento signál přivedeme do bloku integrátor rychlosti a při dosažení krajní polohy necháme rychlost zastavit – u „External reset“ zvolíme „either“, aby se rychlost resetovala jak při náběžné, tak při sestupné hraně, zvolíme zdroj počáteční podmínky „Initial condition source“ jako „Internal“ a zadáme „Initial condition“ na hodnotu 0. Propojíme saturation port bloku integrátor polohy se vstupem external reset bloku integrátor rychlosti.



Obr. 14 – Simulační schéma subsystému kuličky na tyči s resetem rychlosti (vytvoreno v MATLAB R2021a)

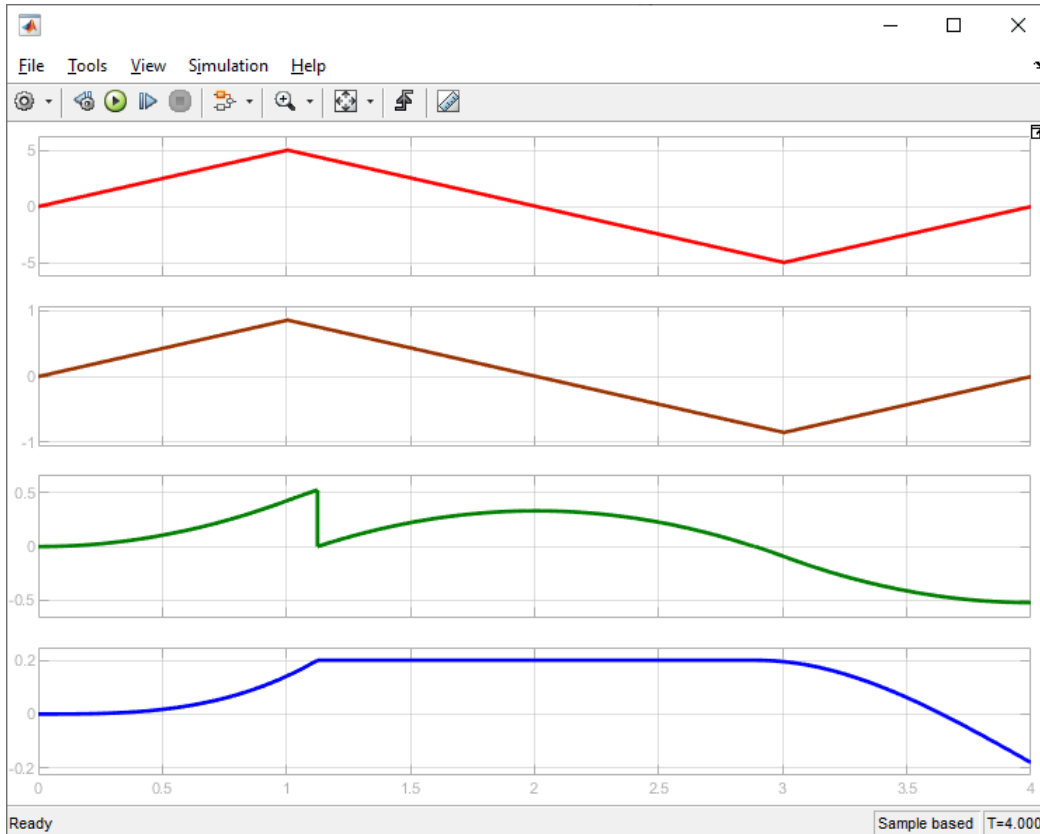


Obr. 15 – Parametry bloku integrátor polohy (vytvoreno v MATLAB R2021a)



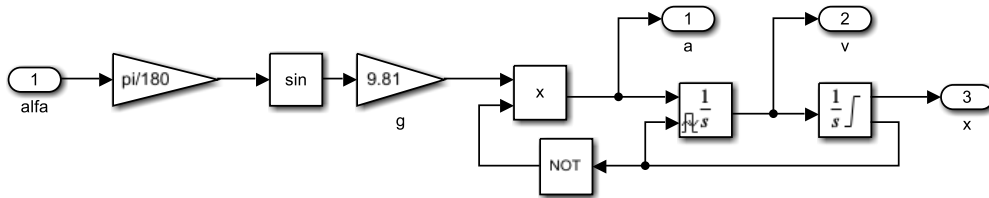
Obr. 16 – Parametry bloku integrátor rychlosti (vytvořeno v MATLAB R2021a)

Výsledek simulace je



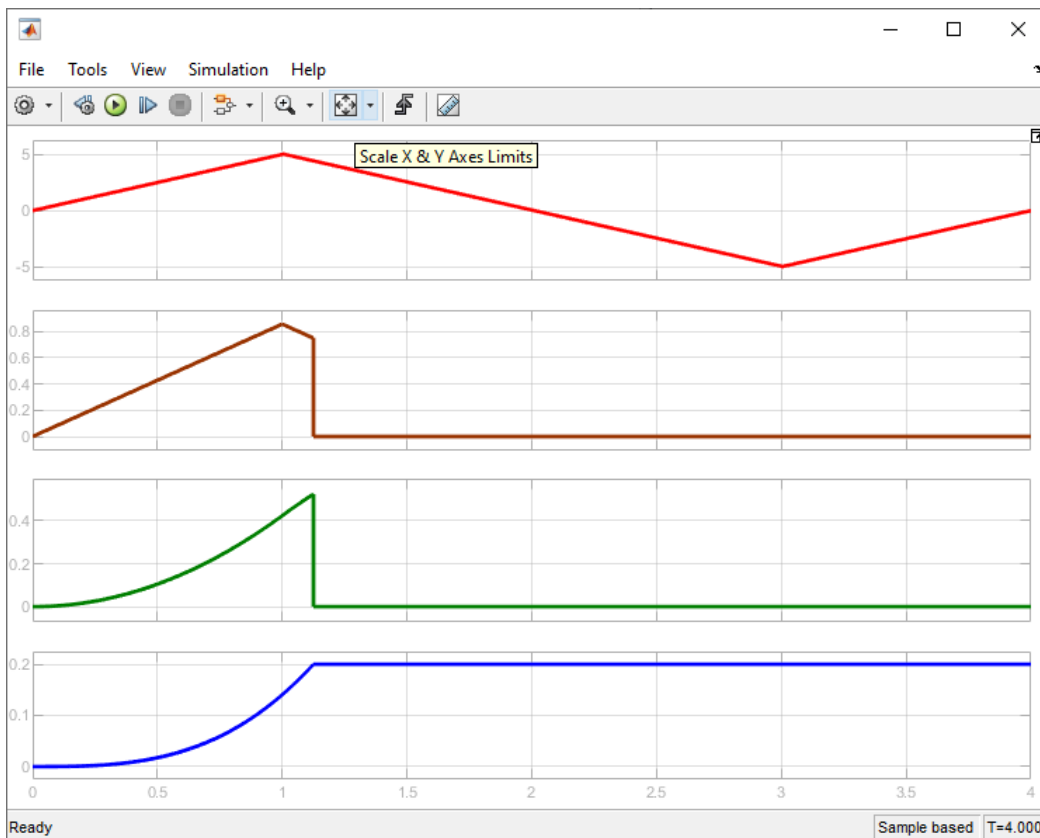
Obr. 17 – Simulace chování modelu kulička na tyči bez regulátoru s resetem rychlosti (vytvořeno v MATLAB R2021a)

Kulička se zastaví, ale okamžitě se zase začne rozjíždět. V tomto okamžiku je potřeba „vypnout“ gravitaci – nastavit zrychlení 0, aby nedocházelo k nárůstu rychlosti. Toho lze dosáhnout tak, že signál zrychlení vynásobíme negovaným signálem o tom, že kulička narazila na doraz – negovaný signál ze saturation port.



Obr. 18 – Simulační schéma subsystému kuličky na tyči s resetem rychlosti a zrychlení (vytvořeno v MATLAB R2021a)

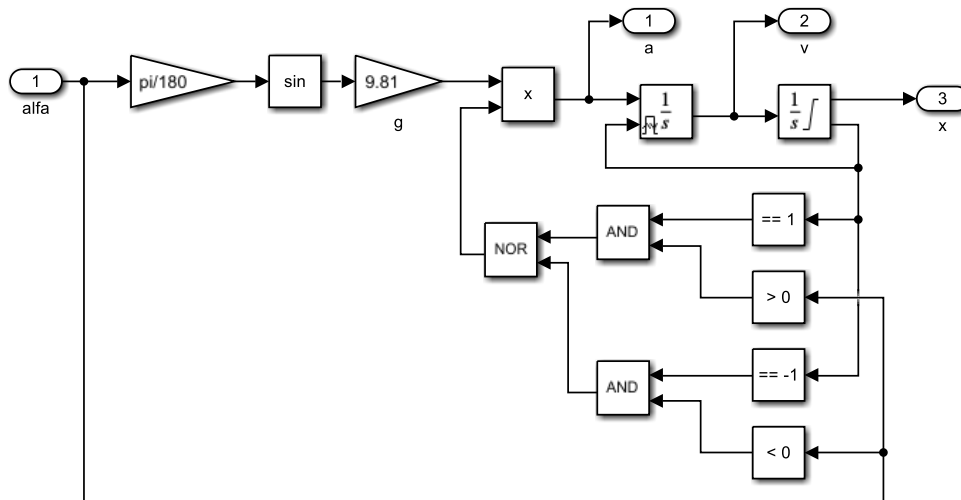
Výsledek simulace je



Obr. 19 – Simulace chování modelu kulička na tyči bez regulátoru s resetem rychlosti a zrychlení (vytvořeno v MATLAB R2021a)

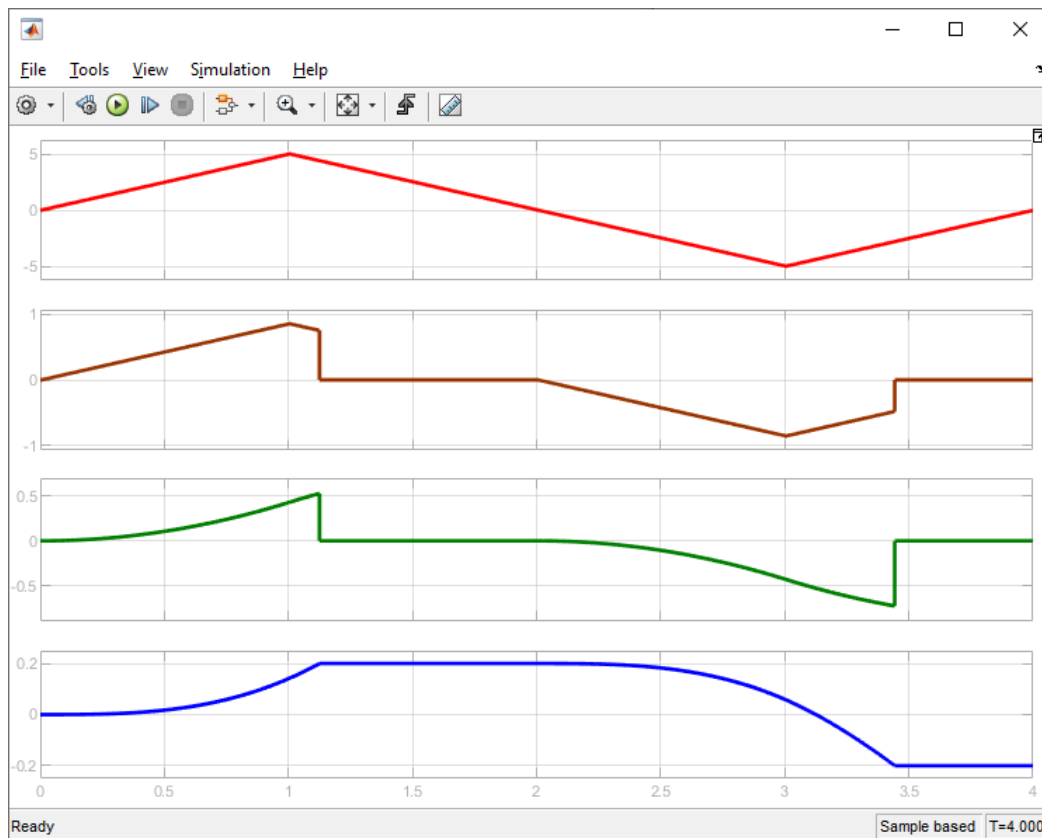
Rychlost spadne na nulu, ale bohužel kulička se už od dorazu „neodlepí“ ani když se dráha natočí na druhou stranu. Abychom postihli toto chování, musíme vytvořit logický obvod, pro „vypínání a opětovné zapínání“ gravitace. Slovy lze funkci popsat následujícím způsobem. Pokud je kulička na pravém dorazu, tak nuluj gravitaci, dokud se nezačne dráha naklánět doleva dolů a pokud je kulička na levém dorazu, tak nuluj gravitaci, dokud se nezačne dráha naklánět doprava dolů. Takovouto logiku lze zajistit následujícím zapojením logických bloků,

ve kterých se testuje, na jaký doraz kulička narazila podmínkou signálu saturace (1 znamená pravý, -1 znamená levý doraz). Druhá podmínka vstupující do bloku AND testuje, kam je natočená dráha kuličky (větší, jak nula znamená, že je natočená směrem dolů doprava, menší jak nula směrem dolů doleva). Výsledkem bloku AND je signál pro nulování gravitace (1 nuluj, 0 nenuluj). Při splnění obou podmínek musí dojít k nulování gravitace a abychom mohli použít nulování násobením, je použit blok negace NOR.



Obr. 20 – Simulační schéma subsystemu kuličky na tyči s resetem rychlosti a zrychlení pomocí logického obvodu (vytvořeno v MATLAB R2021a)

Výsledek simulace na dalším obrázku je už v pořádku – kulička se při dosažení dorazu zastaví (má nulovou rychlost a zrychlení). Teprve při natočení dráhy kuličky na druhou stranu se kulička rozjede a na druhém dorazu se opět zastaví. Jedná se o ukázkou kombinace spojitě simulace řízené logickými podmínkami.



Obr. 21 – Simulace chování modelu kulička na tyči bez regulátoru s resetem rychlosti a zrychlení pomocí logického obvodu (vytvořeno v MATLAB R2021a)

3 Úkoly ke cvičení

1. V Simulinku vytvořte model kuličky na tyči – použijte schéma podle obr. 2 a 3.
2. U subsystému vytvořte masku s parametry podle obr. 4.
3. Proveďte simulaci chování modelu podle obr. 9.
4. Postupně modifikujte simulační schéma, tak aby simulace odpovídala reálnému chování i v případě, kdy kulička dosáhne krajní polohy (drazu) – použijte schéma podle obr. 10, 11, 14, 18 a 20 dokud nedostanete výsledek podle obr. 21.

4 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

druhý Newtonův zákon, 1
model kuličky na tyči, 1

Newtonovy zákony, 1
první Newtonův zákon, 1

Modelování a simulace dynamických systémů

Téma 9: Experimentální identifikace – úvod

Studijní cíl

Seznámit studenty s tvorbou modelů experimentálním přístupem s využitím experimentálních dat. Budou ukázány modely základních typů spojitých proporcionálních, integračních a derivačních dynamických systémů, jejich statické a přechodové charakteristiky a bude demonstrován jednoduchý postup, jak odhadnout parametry pro proporcionální systém 1. řádu.

Doba nutná k nastudování

2 hodiny

Klíčová slova

Model, experimentální identifikace, diferenciální rovnice, spojitý přenos

1 Typy dynamických systémů

Dynamické systémy můžeme rozdělit na proporcionální, integrační a derivační. V následujícím textu jsou uvedeny základní dynamické systémy, spojitě diferenciální rovnice pro jejich popis a spojitě obrazové přenosy. Jsou také uvedeny póly systému – kořeny charakteristického polynomu (jmenovatele obrazového přenosu) – řešení charakteristické rovnice – body singularity, pro které s by u přenosu došlo k dělení nulou.

1.1 Proporcionální systémy

Proporcionální systémy mají nenulový ustálený stav. Ustalují se – pro nenulový vstup se ustálí na nenulovém výstupu) a patří mezi ně následující systémy:

Proporcionální člen bez setrvačnosti (nultého řádu, zesilovač bez dynamiky)

$$y = k_1 u, \quad G(s) = k_1 \quad (1)$$

k_1 je zesílení. Systém nemá pól.

Proporcionální člen se setrvačností 1. řádu (aperiodický, jednodukapacitní)

$$T_1 \frac{dy}{dt} + y = k_1 u, \quad G(s) = \frac{k_1}{T_1 s + 1} \quad (2)$$

T_1 je časová konstanta. Systém má jeden reálný pól $s = -\frac{1}{T_1}$.

Proporcionální člen se setrvačností 2. řádu

$$T_0^2 \frac{d^2 y}{dt^2} + 2\xi_0 T_0 \frac{dy}{dt} + y = k_1 u, \quad G(s) = \frac{k_1}{T_0^2 s^2 + 2\xi_0 T_0 s + 1} \quad (3)$$

T_0 je časová konstanta a ξ_0 je relativní tlumení.

Pro $0 < \xi_0 < 1$ nelze jmenovatele přenosu v rovnici (3) rozložit na kořenové činitele a jedná se o **kmitavý člen 2. řádu**. Kdybychom počítali (řešili charakteristickou rovnici systému – kvadratickou rovnici $T_0^2 s^2 + 2\xi_0 T_0 s + 1 = 0$), vyšly by nám dva komplexně sdružené kořeny – systém má dva komplexně sdružené póly s_1 a s_2 .

Pro $\xi_0 = 1$ lze (3) zapsat jako **aperiodický člen 2. řádu se dvěma stejnými kapacitami**

$$T_0^2 \frac{d^2 y}{dt^2} + 2T_0 \frac{dy}{dt} + y = k_1 u, \quad G(s) = \frac{k_1}{T_0^2 s^2 + 2T_0 s + 1} = \frac{k_1}{(T_0 s + 1)^2} \quad (4)$$

T_0 je dvojnásobná časová konstanta. Systém má jeden dvojnásobný reálný pól $s_1 = s_2 = -\frac{1}{T_0}$.

Toto je případ sériového spojení dvou proporcionálních jednodokapacitních systémů o stejných časových konstantách.

Pro $\xi_0 > 1$ lze (3) zapsat jako **aperiodický člen 2. řádu se dvěma různými kapacitami**

$$T_1 T_2 \frac{d^2 y}{dt^2} + (T_1 + T_2) \frac{dy}{dt} + y = k_1 u, \quad G(s) = \frac{k_1}{T_1 T_2 s^2 + (T_1 + T_2) s + 1} = \frac{k_1}{(T_1 s + 1)(T_2 s + 1)} \quad (5)$$

T_1 a T_2 jsou různé časové konstanty. Systém má dva reálné póly $s_1 = -\frac{1}{T_1}$ a $s_2 = -\frac{1}{T_2}$.

Toto je případ sériového spojení dvou proporcionálních jednodokapacitních systémů o různých časových konstantách.

Mezi proporcionální členy patří také **člen dopravního zpoždění**

$$y(t) = u(t - T_d), \quad G(s) = e^{-T_d s} \quad (6)$$

kde T_d je dopravní zpoždění (způsobené například transportem hmoty).

1.2 Integrační systémy

U **integračních členů** (astatických systémů) neexistuje ustálený stav. Patří mezi ně například:

Integrační člen bez setrvačnosti (nultého řádu, integrátor bez dynamiky)

$$y = k_1 \int_0^t u(\tau) d\tau, \quad G(s) = \frac{k_1}{s} \quad (7)$$

k_1 je zesílení integrátoru. Systém má jeden pól $s = 0$, který leží v počátku souřadného systému Gaussovy komplexní roviny.

Integrační člen se setrvačností 1. řádu (integrátor s jednou kapacitou)

$$T_1 \frac{dy}{dt} + y = k_1 \int_0^t u(\tau) d\tau, \quad G(s) = \frac{k_1}{s(T_1 s + 1)} \quad (8)$$

System má dva póly: $s_1 = 0$ leží v počátku souřadného systému Gaussovy komplexní roviny a $s_2 = -\frac{1}{T_1}$. Jedná se o sériové spojení integrátoru a proporcionalního jednodukapacitního systému.

1.3 Derivační systémy

U **derivačních členů** existuje ustálený stav, ale je nulový. Patří mezi ně například:

Derivační člen bez setrvačnosti (nultého řádu, derivátor bez dynamiky)

$$y = k_1 \frac{du}{dt}, \quad G(s) = k_1 s \quad (9)$$

Stehně jako proporcionalní člen bez setrvačnosti nemá pól.

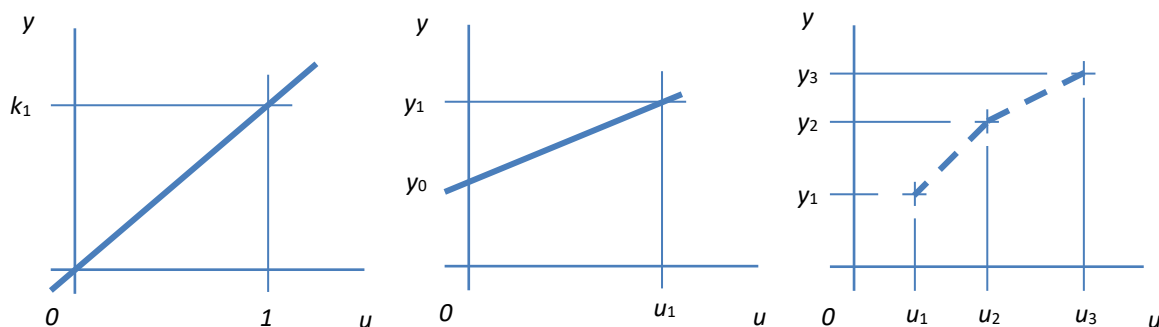
Derivační člen se setrvačností prvního řádu (derivátor s jednou kapacitou)

$$T_1 \frac{dy}{dt} + y = k_1 \frac{du}{dt}, \quad G(s) = \frac{k_1 s}{T_1 s + 1} \quad (10)$$

System má jeden reálný pól $s = -\frac{1}{T_1}$.

2 Statické charakteristiky

Statické charakteristiky existují u proporcionalních systémů. U derivačních systémů má statická charakteristika speciální tvar – ustálený stav je nulový pro všechny hodnoty vstupu. Statická charakteristika je grafické znázornění závislosti ustálených výstupů na vstupech systému. Pokud možno, tak se proměřuje celá pracovní oblast soustavy (pro přípustný rozsah vstupů, stavů a výstupů).



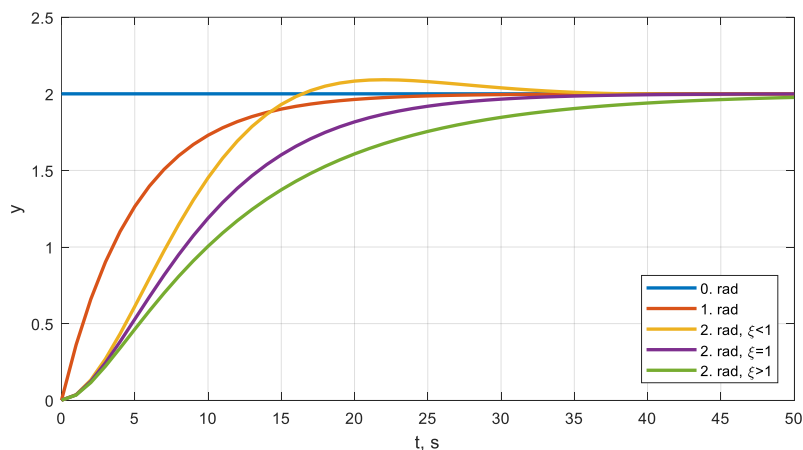
Obr. 1 – Statické charakteristiky

U lineálních soustav, kde prochází statická charakteristika počátkem souřadného systému je statickou charakteristikou přímka se směrnici k_1 viz. první graf na obr. 1 – charakteristiky proporcionalních systému v kapitole 1.1. Často se stává, že pro nulový vstup je výstup soustavy nenulový, avšak stále se jedná o lineární soustavu – například u modelu domu pro nulový výkon topení získáme teplotu v domě rovnou okolní teplotě. V tomto případě je zesílení systému rovno $k_1 \approx \frac{\Delta y}{\Delta x} = \frac{y_1 - y_0}{u_1}$ viz druhý graf na obr. 1. Statická charakteristika je často

nelineární – zesílení soustavy odpovídá směrnici tečny ke statické charakteristice v daném pracovním bodu. Pokud máme k dispozici dvojici hodnot u a y v odpovídajících ustálených stavech, můžeme je vynést do grafu a zesílení určit aproximačně jako směrnice přímek spojujících dva sousední body $k_1 \approx \frac{\Delta y}{\Delta x}$. Pro první dvojici platí $k_1 \approx \frac{y_2 - y_1}{u_2 - u_1}$ a pro druhou dvojici $k_1 \approx \frac{y_3 - y_2}{u_3 - u_2}$ viz třetí graf na obr. 1

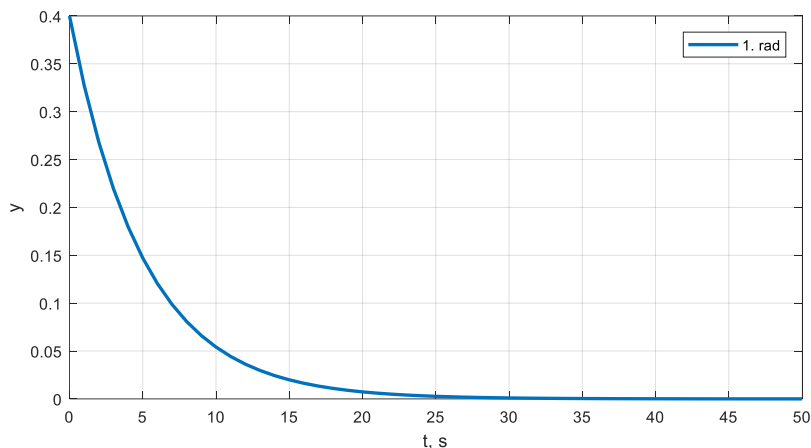
3 Přechodové charakteristiky

Přechodová charakteristika je grafické znázornění odezvy systému na jednotkový Heavisidův skok definovaný jako: $u(t) = 0$ pro $t < 0$ a $u(t) = 1$ pro $t \geq 0$. V následujícím textu uvedeme typické tvary přechodových charakteristik pro proporcionálních systémy uvedené v kapitole 1.1, derivační systémy z kapitoly 1.3 a integrační systémy z kapitoly 1.2. Přechodové charakteristiky proporcionálních členů se setrvačností nultého, prvního a druhého řádu jsou na obr. 2, derivačního členu se setrvačností prvního řádu na obr. 3 a integračních členů nultého a prvního řádu na obr. 4. Jsou použity tyto parametry modelů: $k_1 = 2$, $T_1 = 5$ s, $T_0 = 5$ s, $\zeta = \{0,7; 1; 1,3\}$.

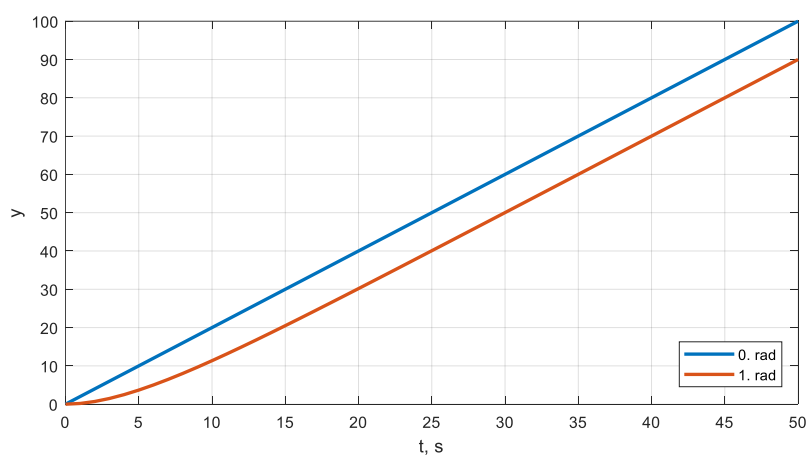


Obr. 2 – Přechodové charakteristiky proporcionálních členů (vytvoreno v MATLAB R2021a)

Zesílení soustavy je určeno hodnotou ustáleného výstupu (vstup má hodnotu 1). Pro soustavu prvního řádu by byla směrnice tečny v počátku rovna poměru časové konstanty a zesílení systému. Směrnice tečny v počátku je u vyšších řádů nulová.



Obr. 3 – Přejchodová charakteristika derivačního členu (vytvořeno v MATLAB R2021a)



Obr. 4 – Přejchodové charakteristiky integračních členů (vytvořeno v MATLAB R2021a)

U integračních členů je zesílení soustavy směrnici lineární části přechodové charakteristiky.

Prakticky bychom nevolili vstup ve tvaru jednotkového skoku, ale v rámci pracovní oblasti bychom skočili například z 30 % vstupu na 60 %. Tím bychom vyloučili vliv častých nelinearit na začátku statické charakteristiky a také bychom použili skok „rozumné“ velikosti. Pro vykreslení přechodové charakteristiky bychom museli výsledek normovat – odečíst hodnoty prvního ustáleného vstupu a změřené hodnoty podělit velikostí změny na vstupu, čímž by výsledek odpovídal reakci na jednotkový skok. Někdy se dělají menší skoky v různých pracovních oblastech pro posouzení nelinearity soustavy, případně se porovnává reakce při skoku nahoru a dolů – u některých nelineárních soustav se průběhy liší.

Impulzní charakteristiky nejsou prakticky měřitelné, protože nelze generovat vstupní signál ve tvaru Diracova impulzu, ale můžeme je z přechodových charakteristik vypočítat jako jejich derivaci podle vstupu, případně změřit odezvu na impulz nenulové tloušťky a co největší výšky a získat tak přibližný tvar impulzní charakteristiky.

Změřit lze frekvenční charakteristiky a příslušné parametry odhadnou z nich – podle útlumu můžeme rozhodnout o řádu systému (první řád 20 dB/dek, druhý řád 40 dB/dek) a zlomové frekvence odpovídají časovým konstantám.

4 Experimentální identifikace

Při tvorbě modelu experimentálním přístupem musíme nejprve zvolit strukturu modelu poté dohledat (identifikovat) jeho neznámé parametry. Pro volbu struktury je vhodné změřit přechodovou charakteristiku a podle jejího tvaru se rozhodnout o typu systému a jeho řádu. Zesílení se z přechodových charakteristik odečítá poměrně dobře. Časové konstanty musíme odhadnout iteračně nebo je nalézt na základě minimalizace zvoleného kritéria (bude ukázáno v další kapitole).

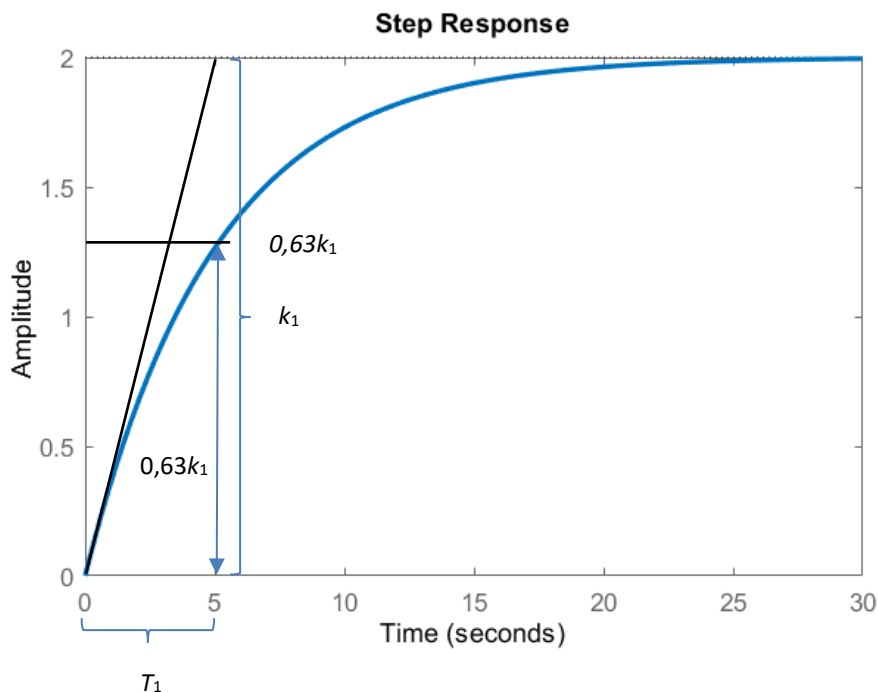
Pro proporcionální systém 1. řádu můžeme použít následující postup. Z rovnice (2) vychází derivace výstupu

$$\frac{dy}{dt} = \frac{k_1 u - y}{T_1} \quad (11)$$

V čase $t=0$ je derivace výstupu (směrnice tečny v počátku) rovna

$$\frac{dy(0)}{dt} = \frac{k_1 \overbrace{u(0)}^1 - \overbrace{y(0)}^0}{T_1} = \frac{k_1}{T_1} \quad (12)$$

Graficky je postup znázorněn na obr. 4. Zkonstruujeme tečnu v počátku a čas, kdy protíná tečna ustálenou hodnotu výstupu (hodnotu zesílení), odpovídá časové konstantě soustavy.



Obr. 5 – Přechodová charakteristika proporcionální soustavy 1. řádu (vytvořeno v MATLAB R2021a)

Další možností je odečíst čas, kdy hodnota na přechodové charakteristice dosáhla 63 % ustálené hodnoty a tento čas odpovídá časové konstantě. Za čas roven pěti časovým konstantám dosáhne výstup 99 % ustálené hodnoty (je prakticky ustálený).

Toto vychází z přechodové funkce, která má pro proporcionální soustavu 1. řádu tvar

$$y(t) = k_1 \left(1 - e^{-t/T_1}\right) \quad (13)$$

a výstup soustavy za čas T_1 a $5T_1$ je

$$y(T_1) = k_1 \left(1 - e^{-T_1/T_1}\right) = 0,63k_1 \quad (14)$$

$$y(5T_1) = k_1 \left(1 - e^{-5T_1/T_1}\right) = 0,99k_1$$

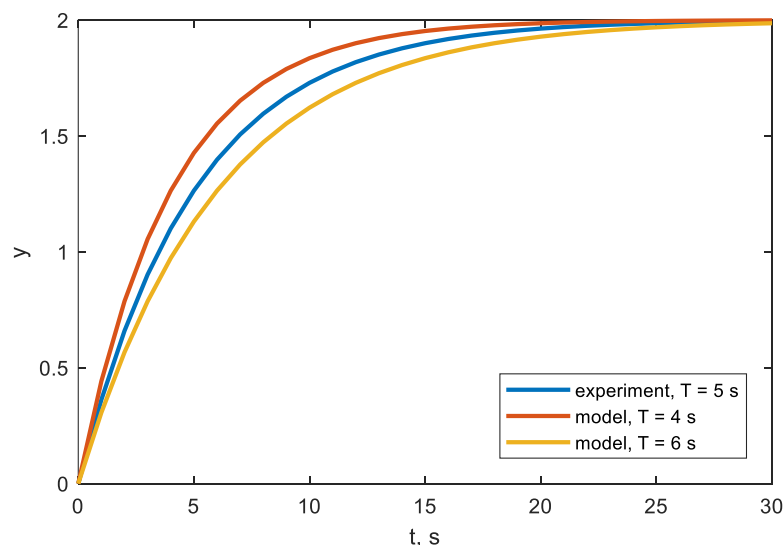
Podobné vztahy a postupy lze v literatuře nalézt i pro další typy modelů – např. (Balátě, 2004) nebo (Noskievič, 1999).

V MATLABu lze vypočítat odezvu modelu a porovnat ji se změřenou odezvou a iterací parametrů se snažit dosáhnout co nejlepší shody. Postup budeme ilustrovat následujícím příkladem. Místo měření vytvoříme v MATLABu objekt typu spojitý přenos se zesílením 2 a časovou konstantou 5 s a vypočítáme v časech po jedné sekundě od 0 do 30 s přechodovou charakteristiku systému

```
S=tf(2,[5 1]);
t=0:30;
[y,t]=step(S,t);
```

Parametry bychom ve skutečnosti neznali, tak bychom musel odhadnout hodnotu zesílení 2 a časové konstanty dvou modelů např. 4 a 6 s, vypočítáme jejich přechodové charakteristiky a porovnáme je se „změřenou“ přechodovou charakteristikou.

```
M1=tf(2,[4 1]);
[yM1,t]=step(M1,t);
M2=tf(2,[6 1]);
[yM2,t]=step(M2,t);
plot(t,y,t,yM1,t,yM2,'LineWidth',2), xlabel('t, s'), ylabel('y')
legend('experiment, T = 5 s','model, T = 4 s','model, T = 6 s')
```



Obr. 6 – Přechodová charakteristika – změřená a modely (vytvoreno v MATLAB R2021a)

Z grafu na obr. 6 vidíme, že oba modely aproximují dobře zesílení soustavy, model s časovou konstantou 4 s je moc rychlý, a naopak model s časovou konstantou 6 s je pomalý. Ideální aproximace bude „někde“ mezi těmito hodnotami.

Tento postup lze libovolně modifikovat pro další modely proporcionálních, integračních a derivačních dynamických systémů. K posouzení kvality aproximace (shody mezi změřenou a simulovanou odezvou) může posloužit **kritérium** shody například sumu absolutních hodnot nebo kvadrátů odchylek mezi změřenými a vypočítanými výstupy. Často je tato suma dělená počtem hodnot, aby byla hodnota kritéria porovnatelná mezi odezvami s různým počtem vzorků. Je použito vektorové operace, kdy součin transponovaného sloupcového vektoru a původního sloupcového vektoru odchylek výstupů experimentu a modelu počítá sumu kvadrátů odchylek.

$$K1 = (y - yM1)' * (y - yM1) / \text{length}(y)$$
$$K2 = (y - yM2)' * (y - yM2) / \text{length}(y)$$

Výsledkem jsou hodnoty kritéria $K_1 = 0,0072$ a $K_2 = 0,0059$. Vidíme, že model s časovou konstantou 6 s dává o něco lepší výsledek než model s časovou konstantou 4 s.

5 Úkoly ke cvičení

1. V MATLABu vytvořte modely a vypočítejte jejich přechodové charakteristiky stejné jako jsou uvedeny na obr. 2, 3 a 4.
2. Zkuste aproximovat proporcionální soustavu 2. řádu modelem soustavy 1. řádu. Vypočítejte kritérium dané aproximace.

6 Použitá literatura

BALÁTĚ, J. 2004. *Automatické řízení*. Praha: BEN - technická literatura. ISBN 80-7300-148-9.

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

aperiodický člen 2. řádu se dvěma různými kapacitami, 2
aperiodický člen 2. řádu se dvěma stejnými kapacitami, 2
člen dopravního zpoždění, 2
derivační člen, 3
derivační člen bez setrvačnosti, 3
derivační člen se setrvačností prvního řádu, 3
experimentální identifikace, 6
integrační člen, 2
integrační člen bez setrvačnosti, 2

integrační člen se setrvačností 1. řádu, 2
kmitavý člen 2. řádu, 2
kritérium, 8
proporcionální člen bez setrvačnosti, 1
proporcionální člen se setrvačností 1. řádu, 1
proporcionální člen se setrvačností 2. řádu, 2
proporcionální systém, 1
přechodová charakteristika, 4
statická charakteristika, 3

Modelování a simulace dynamických systémů

Téma 10: Experimentální identifikace s optimalizací

Studijní cíl

Seznámit studenty s tvorbou modelů experimentálním přístupem s využitím experimentálních dat a numerické optimalizace kritéria vyjadřujícího kvalitu aproximace.

Doba nutná k nastudování

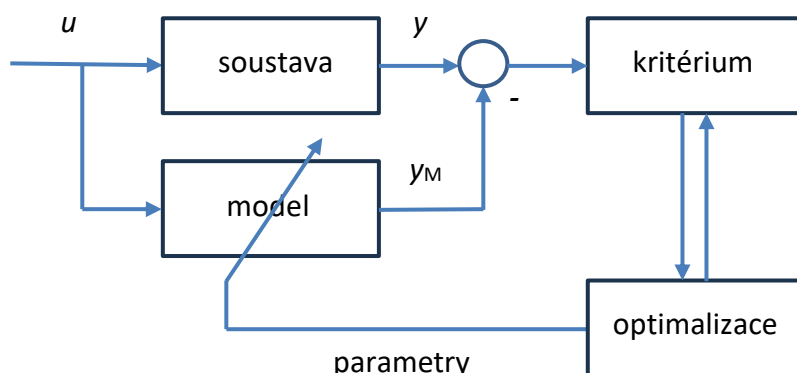
2 hodiny

Klíčová slova

Model, experimentální identifikace, diferenciální rovnice, spojitý přenos, kritérium, optimalizace

1 Princip experimentální identifikace s optimalizací

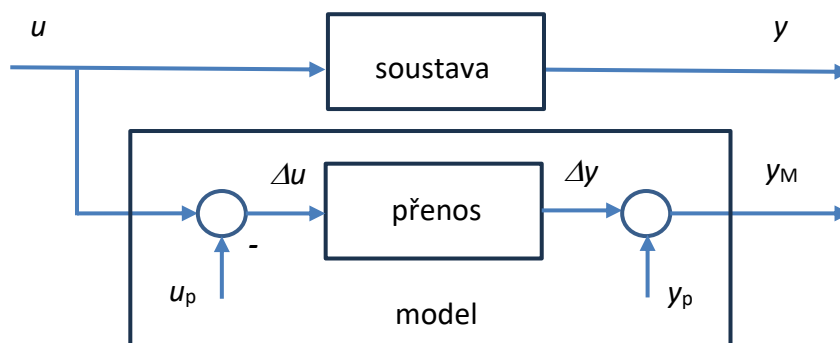
V kapitole 9 byla uvedena problematika experimentální identifikace. V této kapitole si ukážeme, jak lze pomocí optimalizace hledat optimální parametry modelu, které zajistí nejlepší shodu modelu a reálné soustavy. K posouzení kvality modelu bylo v kapitole 9 použito kritérium ve tvaru sumy kvadrátů odchylek mezi změřenými výstupy a výstupy počítanými modelem. Princip experimentální identifikace s optimalizací lze naznačit následujícím schématem



Obr. 1 – Princip experimentální identifikace s optimalizací

Nad simulací modelu soustavy je spuštěna optimalizace, která mění parametry modelu tak, aby kritérium dosáhlo svého minima. Kritérium je ve tvaru sumy kvadrátů odchylek mezi výstupy soustavy a výstupy modelu pro stejný vstup. Pro začátek přechodového děje je nutné znát počáteční podmínky (hodnoty stavových veličin – stavy soustavy v čase 0, počáteční podmínky obyčejné diferenciální rovnice). Pokud bychom je neznali, tak na nich po určité době nebude záležet, ale začátek experimentu bude jiný u soustavy a u modelu.

Na začátku experimentu necháme soustavu ustábit a hodnoty vstupu a ustáleného výstupu použijeme jako pracovní bod u_p, y_p . Vstupy do přenosu Δu budou odchylky vstupu u od pracovního bodu vstupu u_p . Tím zajistíme předpoklad nulových počátečních podmínek, což vychází z definice přenosu. Jinak bychom museli použít stavový popis a v něm zadat nenulové počáteční podmínky. Výstupy přenosu mají význam odchylek výstupu od pracovního bodu Δy . Abychom získali výstup modelu y_M , musíme k výstupu přenosu Δy přičíst hodnotu pracovního bodu výstupu y_p .



Obr. 2 – Model ve tvaru přenosu s konceptem pracovního bodu

Princip pracovního bodu byl popsán a využit v kapitole 5 při linearizaci modelu nádrže s otvorem. Pracovní bod má význam, jako oblast, ve které bude aproximace chování soustavy nejpřesnější – v pracovním bodě je odchylka mezi výstupem soustavy a modelu nulová. Jak se budeme od pracovního bodu vzdalovat, tak se vlivem nelinearity soustavy bude chování modelu a soustavy rozcházet. Jak přesná bude lineární aproximace záleží na míře nelinearity soustavy.

2 Soustavy a data pro experimentální identifikaci

Na reálné soustavě bychom data změřili. V rámci pracovní oblasti bychom zrealizovali experiment – navrhli a použili vhodné průběhy vstupu a změřili odezvu soustavy – průběh výstupu. Vstup nemůže být konstantní – tím bychom určili maximálně zesílení soustavy. Musí obsahovat minimálně jednu skokovou změnu, abychom zaznamenali přechodový děj a mohli určit časovou konstantu modelu. Na druhou stranu nesmí být vstup moc rychlý, aby na něj soustava stačila reagovat. Pro začátek je vhodným tvarem vybuzujícího signálu skoková změna případně několik skokových změn. Skriptem MATLABu vygenerujeme a uložíme data, která

následně použijeme pro experimentální identifikaci. Místo reálných soustav použijeme tyto modely (podrobněji jsou popsány v předcházející kapitole):

Soustava P₁ - proporcionální soustava 1. řádu

$$T_1 \frac{dy}{dt} + y = k_1 u, \quad G(s) = \frac{k_1}{T_1 s + 1} \quad (1)$$

$k_1 = 2$ je zesílení, $T_1 = 5$ s je časová konstanta.

Model proporcionální soustavy 2. řádu lze obecně zapsat jako

$$T_0^2 \frac{d^2 y}{dt^2} + 2\xi_0 T_0 \frac{dy}{dt} + y = k_1 u, \quad G(s) = \frac{k_1}{T_0^2 s^2 + 2\xi_0 T_0 s + 1} \quad (2)$$

T_0 je časová konstanta a ξ_0 je relativní tlumení.

Soustava P_{2a} - pro $0 < \xi_0 < 1$ vede (2) na kmitavou soustavu 2. řádu ($\xi_0 = 0,7$).

Soustava P_{2b} - Pro $\xi_0 = 1$ vede (2) na aperiodickou soustavu 2. řádu se dvěma stejnými kapacitami. Model lze zapsat jako

$$T_0^2 \frac{d^2 y}{dt^2} + 2T_0 \frac{dy}{dt} + y = k_1 u, \quad G(s) = \frac{k_1}{T_0^2 s^2 + 2T_0 s + 1} = \frac{k_1}{(T_0 s + 1)^2} \quad (3)$$

T_0 je dvojnásobná časová konstanta.

Soustava P_{2c} - Pro $\xi_0 > 1$ vede (2) na aperiodickou soustavu 2. řádu se dvěma různými kapacitami ($\xi_0 = 1,3$). Model lze zapsat jako

$$T_1 T_2 \frac{d^2 y}{dt^2} + (T_1 + T_2) \frac{dy}{dt} + y = k_1 u, \quad G(s) = \frac{k_1}{T_1 T_2 s^2 + (T_1 + T_2) s + 1} = \frac{k_1}{(T_1 s + 1)(T_2 s + 1)} \quad (4)$$

T_1 a T_2 jsou různé časové konstanty.

Soustava I₁ - integrační soustava se setrvačností 1. řádu (integrátor s jednou kapacitou)

$$T_1 \frac{dy}{dt} + y = k_1 \int_0^t u(\tau) d\tau, \quad G(s) = \frac{k_1}{s(T_1 s + 1)} \quad (5)$$

Pracovní bod všech modelů byl zvolen $u_p = 4$, $y_p = 2$. Na výstupy modelů byl přičten šum se směrodatnou odchylkou 0,05. Tím získáme reálnější chování a při identifikaci nedostaneme úplně stejné parametry jako jsme použili pro simulaci. Data jsou uložena do souboru `ident_optim_data.mat`.

```
clear
k1=2;
T1=5;
T0=5;
ksi0=[0.7; 1; 1.3];

P1=tf(k1, [T1 1]);
P2a=tf(k1, [T0^2 2*ksi0(1)*T0 1]);
P2b=tf(k1, [T0^2 2*ksi0(2)*T0 1]);
P2c=tf(k1, [T0^2 2*ksi0(3)*T0 1]);
I1=tf(k1, [T1 1 0]);

t=(0:1:70)';
```

```

up=4; yp=2;
u=[zeros(10,1); ones(20,1); -ones(20,1); zeros(21,1)]+up;
N=length(u); smer_odch=0.05;
yP1=lsim(P1,u-up,t,'zoh')+randn(N,1)*smer_odch+yp;
yP2a=lsim(P2a,u-up,t,'zoh')+randn(N,1)*smer_odch+yp;
yP2b=lsim(P2b,u-up,t,'zoh')+randn(N,1)*smer_odch+yp;
yP2c=lsim(P2c,u-up,t,'zoh')+randn(N,1)*smer_odch+yp;
yI1=lsim(I1,u-up,t,'zoh')+randn(N,1)*smer_odch+yp;

save('ident_optim_data.mat', 't', 'u', 'yP1', 'yP2a', 'yP2b', 'yP2c',
'yI1');

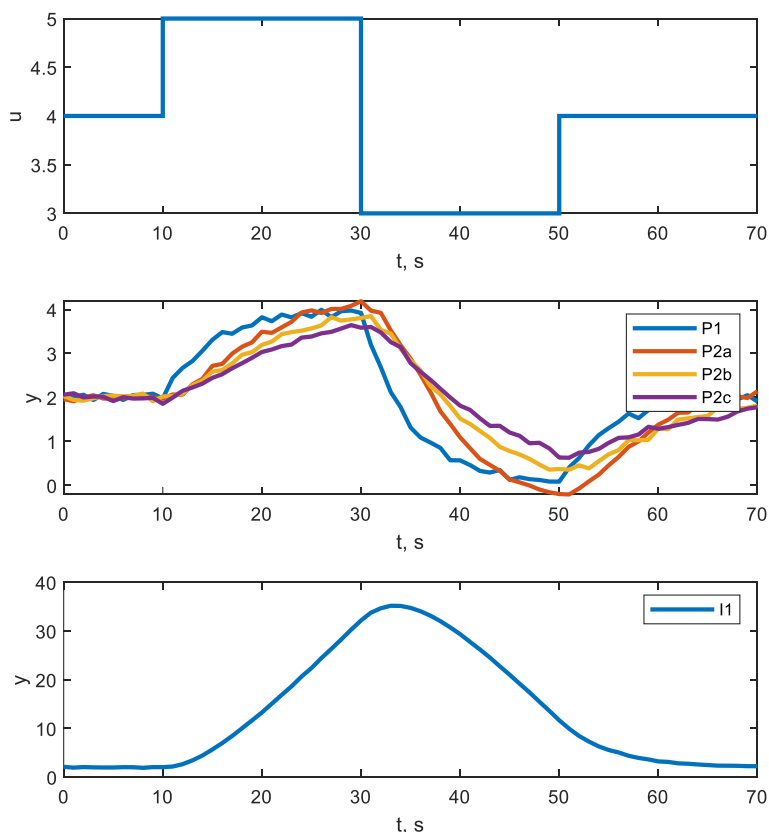
```

Ještě, než začneme zpětně hledat „neznámé“ parametry modelů jednotlivých soustav, tak načteme data ze souboru `ident_optim_data.mat` a zobrazíme je.

```

load ident_optim_data
subplot(3,1,1), stairs(t,u,'LineWidth',2), xlabel('t, s'), ylabel('u')
subplot(3,1,2), plot(t,yP1,t,yP2a,t,yP2b,t,yP2c,'LineWidth',2), xlabel('t, s'), ylabel('y')
legend('1. rad', '2. rad, \xi=0.7', '2. rad, \xi=1', '2. rad, \xi=1.3')
subplot(3,1,3), plot(t,yI1,'LineWidth',2), xlabel('t, s'), ylabel('y')
legend('1. rad')

```



Obr. 3 – Data pro experimentální identifikaci (vytvořeno v MATLAB R2021a)

Horní graf zobrazuje průběh vstupu všech soustav, v prostředním a dolním grafu jsou „měřené“ výstupy. Prostřední graf zobrazuje výstupy proporcionálních soustav a v dolním grafu je výstup integrační soustavy.

3 Aplikace experimentální identifikace s optimalizací

Začneme identifikací (odhadem) neznámých parametrů modelu **soustavy P₁** – proporcionální soustavy 1. řádu – model (1). Abychom mohli použít funkci *fminsearch* **vířerovměrné nelineární minimalizace** (simplexová metoda), je potřeba vytvořit funkci *hledej1*, která bude pro zadané parametry a vstup počítat kritérium a uložit ji do souboru *hledej1.m*. Ve funkci používáme funkci *lsim*, která vrací výstup modelu M pro daný průběh vstupu. Parametr 'zoh' je použit, aby funkce neinterpolovala průběh vstupu, ale použila skokové změny.

```
function K=hledej1(X,t,u,y)
```

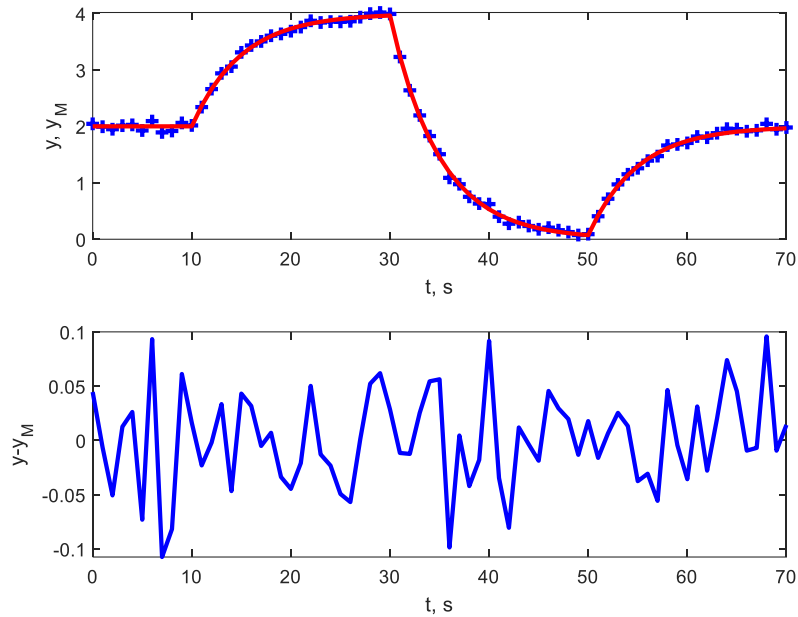
```
k1=X(1);  
T=X(2);  
M=tf(k1,[T 1]);  
yM=lsim(M,u,t,'zoh');  
K=(y-yM)'*(y-yM);
```

Z obr. 3 odečteme pracovní bod, zadefinujeme počáteční podmínky parametrů pro optimalizaci a spustíme funkci *fminsearch*

```
up=4; yp=2;  
X0=[1 1];  
X=fminsearch('hledej1',X0,[],t,u-up,yP1-yp);
```

Do proměnné X nám funkce *fminsearch* vrátí optimální hodnoty parametrů modelu. Abychom vykreslili odezvy modelu společně s experimentálními daty a vypočítali hodnotu kritéria, musíme opět vytvořit model a vypočítat odezvu. Do horního grafu zakreslíme experimentální data jako křížky a výstup modelu plnou čarou. Ve spodním grafu zobrazíme odchylky mezi „měřeným“ výstupem a počítaným výstupem modelu.

```
k1=X(1);  
T=X(2);  
M=tf(k1,[T 1]);  
yM=lsim(M,u-up,t,'zoh')+yp;  
K=(yP1-yM)'*(yP1-yM);  
subplot(2,1,1), plot(t,yP1,'+b',t,yM,'r','LineWidth',2), xlabel('t, s'),  
ylabel('y, y_M')  
subplot(2,1,2), plot(t,yP1-yM,'b','LineWidth',2), xlabel('t, s'),  
ylabel('y-y_M')
```



Obr. 4 – Výstup modelu soustavy P_1 (vytvořeno v MATLAB R2021a)

Zesílení soustavy vyšlo $k_1 = 1,999$ a časová konstanta $T_1 = 4,972$ s. Hodnota kritéria vyjde $K = 0,137$.

Podobným způsobem provedeme odhad neznámých parametrů zbývajících modelů.

Model **soustavy P_{2a}** je model proporcionální kmitavé soustavy 2. řádu – model (2) pro $\xi_0 = 0,7$. Musíme vytvořit funkci *hledej2*, která bude pro zadané parametry a vstup počítat kritérium a uložit ji do souboru *hledej2.m*.

```
function K=hledej2(X,t,u,y)

k1=X(1);
T0=X(2);
ksi0=X(3);
M=tf(k1,[T0^2 2*ksi0*T0 1]);
yM=lsim(M,u,t,'zoh');
K=(y-yM)'*(y-yM);
```

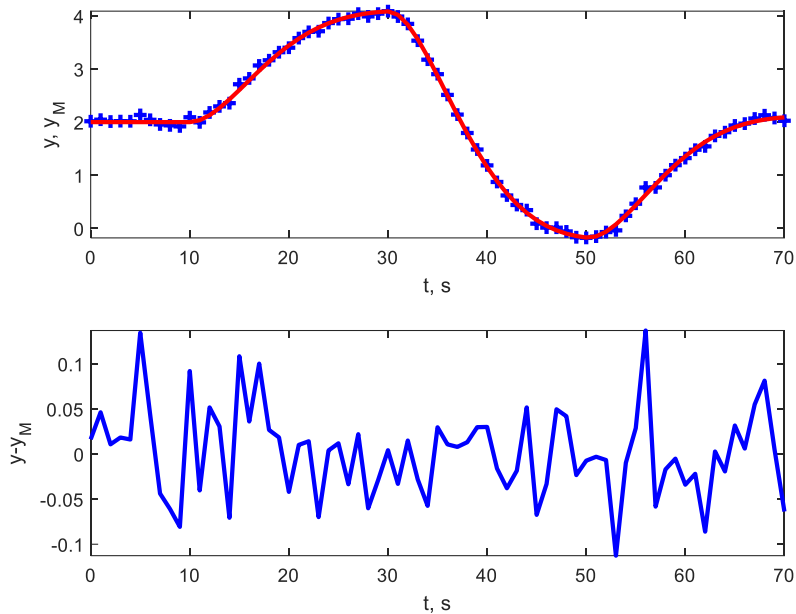
Zdefinujeme počáteční podmínky parametrů pro optimalizaci a spustíme funkci *fminsearch*

```
up=4; yp=2;
X0=[1 1 1];
X=fminsearch('hledej2',X0,[],t,u-up,yP2a-yp);
```

Musíme opět znovu vytvořit model a vypočítat odezvu.

```
k1=X(1);
T0=X(2);
ksi0=X(3);
M=tf(k1,[T0^2 2*ksi0*T0 1]);
yM=lsim(M,u-up,t,'zoh')+yp;
K=(yP2a-yM)'*(yP2a-yM);
subplot(2,1,1), plot(t,yP2a,'+b',t,yM,'r','LineWidth',2), xlabel('t, s'),
ylabel('y, y_M')
```

```
subplot(2,1,2), plot(t,yP2a-yM, 'b', 'LineWidth',2), xlabel('t, s'),
ylabel('y-y_M')
```



Obr. 5 – Výstup modelu soustavy P_{2a} (vytvoreno v MATLAB R2021a)

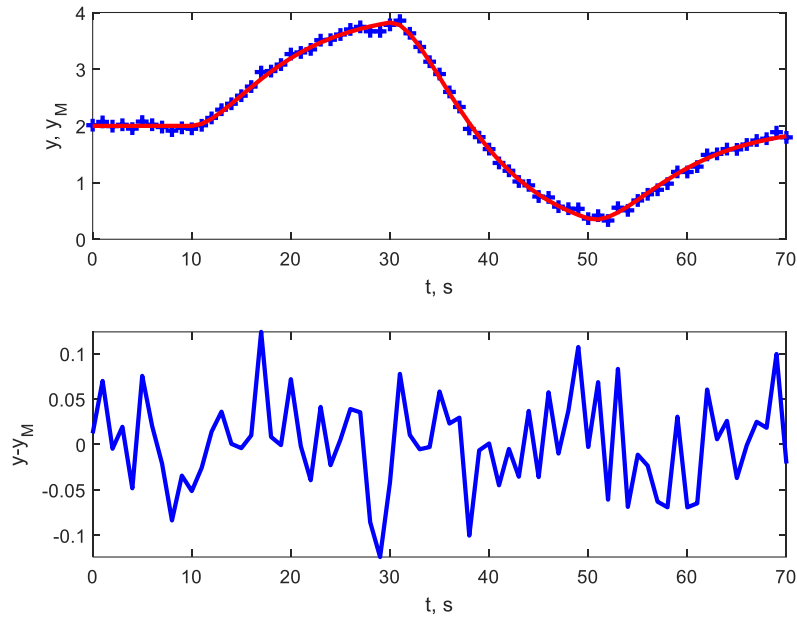
Zesílení soustavy vyšlo $k_1 = 1,9992$, časová konstanta $T_0 = 5,0449$ s a relativní tlumení a $\xi_0 = 0,6964$. Hodnota kritéria vyjde $K = 0,1752$.

Abychom získali model **soustavy P_{2b}** – aperiodické soustavy 2. řádu se dvěma stejnými kapacitami – model (3), stačí změnit parametr ve volání funkce *fminsearch*

```
x=fminsearch('hledej2',x0,[],t,u-up,yP2b-yp);
```

Stejným způsobem musíme modifikovat skript při výpočtu kritéria a kreslení grafu

```
K=(yP2b-yM)'*(yP2b-yM);
subplot(2,1,1), plot(t,yP2b, '+b',t,yM, 'r', 'LineWidth',2), xlabel('t, s'),
ylabel('y, y_M')
subplot(2,1,2), plot(t,yP2b-yM, 'b', 'LineWidth',2), xlabel('t, s'),
ylabel('y-y_M')
```



Obr. 6 – Výstup modelu soustavy P_{2b} (vytvořeno v MATLAB R2021a)

Zesílení soustavy vyšlo $k_1 = 2,0217$, časová konstanta $T_0 = 4.9389$ s a relativní tlumení a $\xi_0 = 1,0265$. Hodnota kritéria vyjde $K = 0,1788$.

Pokud bychom věděli, že se jedná o aperiodickou soustavu 2. řádu se dvěma stejnými kapacitami ($\xi_0 = 1$), mohli bychom parametr v optimalizaci vynechat a hledat pouze zesílení a časovou konstantu. To lze udělat modifikací počátečních podmínek pro optimalizaci

$\mathbf{x0} = [1 \ 1];$

a kódu funkce *hledej2* a hlavního skriptu

```
k1=X(1);
T0=X(2);
ksi0=1;
```

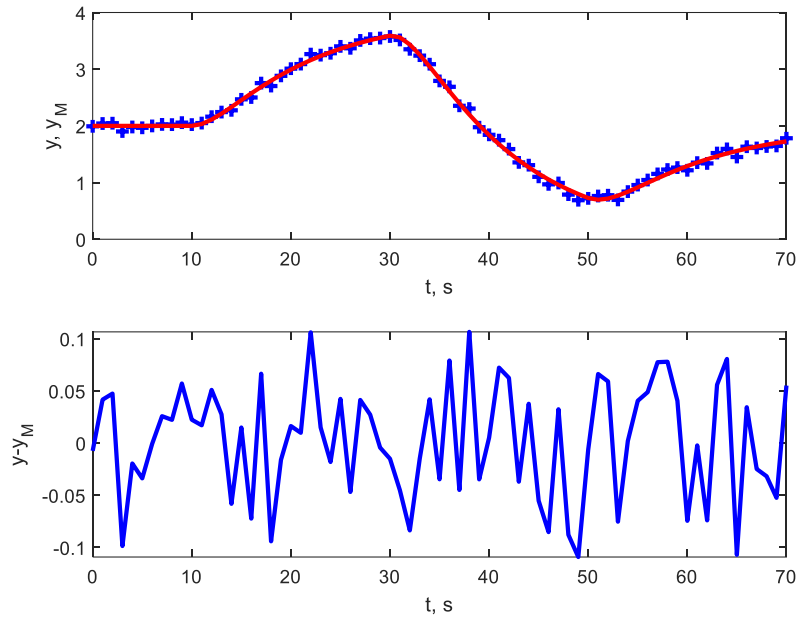
Zesílení soustavy potom vyjde $k_1 = 1,9999$ a časová konstanta $T_0 = 5,0005$ s. Hodnota kritéria vyjde $K = 0,1812$. Hodnota kritéria je o něco vyšší, protože model má o jeden parametr méně a není tolik pružný.

U modelu **soustavy P_{2c}** – aperiodické soustavy 2. řádu se dvěma různými kapacitami – model (4), $\xi_0 = 1,3$, stačí opět změnit parametr ve volání funkce *fminsearch*

```
X=fminsearch('hledej2',X0,[ ],t,u-up,yP2c-yp);
```

a modifikovat skript při výpočtu kritéria a kreslení grafu

```
K=(yP2c-yM)'*(yP2c-yM);
subplot(2,1,1), plot(t,yP2c,'+b',t,yM,'r','LineWidth',2), xlabel('t, s'),
ylabel('y, y_M')
subplot(2,1,2), plot(t,yP2c-yM,'b','LineWidth',2), xlabel('t, s'),
ylabel('y-y_M')
```

Obr. 7 – Výstup modelu soustavy P_{2c} (vytvořeno v MATLAB R2021a)

Zesílení soustavy vyšlo $k_1 = 1,9652$, časová konstanta $T_0 = 5,0668$ s a relativní tlumení a $\xi_0 = 1,2653$. Hodnota kritéria vyjde $K = 0,2132$.

Pokud bychom věděli, že se jedná o aperiodickou soustavu 2. řádu se dvěma různými kapacitami, mohli bychom identifikovat rovnou zesílení a dvě časové konstanty. To lze udělat modifikací kódu funkce *hledej2* a hlavního skriptu

```
k1=X(1);
T1=X(2);
T2=X(3);
M=tf(k1, [T1*T2 T1+T2 1]);
```

Zesílení soustavy potom vyjde $k_1 = 1,9652$, časová konstanta $T_1 = 2,4830$ s a časová konstanta $T_2 = 10,3395$ s. Hodnota kritéria vyjde $K = 0,2132$. Hodnota kritéria je stejná, protože se jedná o stejný model.

Poslední model je model **soustavy I₁** – integrační soustavy se setrvačností 1. řádu – model (5). Musíme vytvořit novou funkci *hledej1i* a uložit ji do souboru *hledej1i.m*

```
function K=hledej1i(X,t,u,y)
```

```
k1=X(1);
T1=X(2);
M=tf(k1, [T1 1 0]);
yM=lsim(M,u,t,'zoh');
K=(y-yM)'*(y-yM);
```

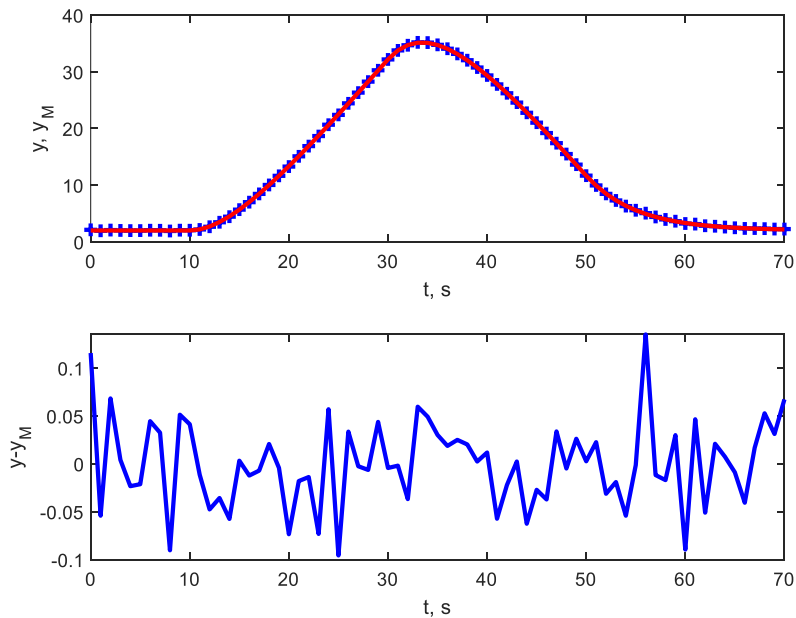
a modifikovat skript při výpočtu kritéria a kreslení grafu

```
X0=[1 1];
X=fminsearch('hledej1i',X0,[],t,u-up,yI1-yp);
k1=X(1);
T1=X(2);
```

```

M=tf(k1,[T1 1 0]);
K=(yI1-yM) '* (yI1-yM);
subplot(2,1,1), plot(t,yI1,'+b',t,yM,'r','LineWidth',2), xlabel('t, s'),
ylabel('y, y_M')
subplot(2,1,2), plot(t,yI1-yM,'b','LineWidth',2), xlabel('t, s'),
ylabel('y-y_M')

```



Obr. 8 – Výstup modelu soustavy I_1 (vytvořeno v MATLAB R2021a)

Zesílení soustavy vyšlo $k_1 = 2,001$ a časová konstanta $T_1 = 5,0144$ s. Hodnota kritéria vyjde $K = 0,1413$.

4 Úkoly ke cvičení

1. Vygenerujte „experimentální“ data pro soustavy P_1 , P_{2a} , P_{2b} , P_{2c} a I_1 podle obr. 3.
2. Identifikujte neznámé parametry modelů soustav P_1 , P_{2a} , P_{2b} , P_{2c} a I_1 a výsledky porovnejte s výsledky uvedenými v kapitole 3.

5 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

aperiodická soustava 2. řádu se dvěma
stejnými kapacitami, 3
aperiodický soustava 2. řádu se dvěma
různými kapacitami, 3

integrační soustava 1. řádu, 3
proporcionální kmitavá soustava 2. řádu, 3
proporcionální soustava 1. řádu, 3
vířerovměrná nelineární minimalizace, 5

Modelování a simulace dynamických systémů

Téma 11: Experimentální identifikace s využitím MNČ – úvod

Studijní cíl

Seznámit studenty s tvorbou časově diskretních modelů experimentálním přístupem s využitím experimentálních dat za použití metody nejmenších čtverců. Bude odvozeno analytické řešení úlohy pro soustavu 1. a 2. řádu a ukázán příklad použití.

Doba nutná k nastudování

2 hodiny

Klíčová slova

Model, experimentální identifikace, diferenční rovnice, diskretní přenos

1 Princip experimentální identifikace s využitím MNČ

V kapitolách 9 a 10 a byl vysvětlen princip experimentální identifikace a aplikace numerické optimalizace. Zaměřili jsme se na časově spojitě dynamické systémy popsané obyčejnou diferenciální rovnicí nebo spojitým obrazovým přenosem. Stejný postup bychom mohli aplikovat i na časově diskretní modely. Prakticky je, ale výhodnější identifikovat parametry spojitých systémů a ty následně převést do diskretní podoby. Spojité parametry jsou totiž jednodušeji interpretovatelné (mají význam jako zesílení, časové konstanty, relativní tlumení a podobně). U diskretních modelů jsou parametry koeficienty diferenčních rovnic a jejich interpretace není tak jednoduchá. Určitá nevýhoda experimentální identifikace s optimalizací spočívá v tom, že se jedná o nelineární problém, a tudíž je nutné řešit úlohu numericky. Hodně záleží na tvaru kriteriální funkce, zda najde optimalizace globální minimum, nebo zda uvízne v lokálním minimu. Také není garantována rychlost optimalizace – opět záleží na tvaru kriteriální funkce a na použité optimalizační metodě. Výhodou je, že můžeme při řešení respektovat omezení parametrů – aplikovat optimalizační metody s omezeními.

Naopak u experimentální identifikace parametrů diskretních modelů lze pomocí aplikace metody nejmenších čtverců (MNČ) odvodit analytické řešení úlohy. Na rozdíl od numerického řešení ovšem nelze pracovat s omezeními. To by vedlo pět na numerické řešení – na problém kvadratického programování.

2 Odvození experimentální identifikace s využitím MNC

Jelikož se jedná o experimentální identifikaci na základě změřených dat musíme opět nejprve navrhnout **vybuzovací signál** a změřit odezvu soustavy. Signál by měl pokrýt pracovní oblast, ve které chceme model a jeho parametry identifikovat. Nesmí být moc rychlý, aby nedošlo k profiltrování dynamikou soustavy a ani moc pomalý, protože potom by převládla informace o ustáleném stavu a přechodový děj by ztratil váhu. Předpokládejme, že máme změřené následující dvojice vstupu a výstupu soustavy

Tab. 1 – Experimentální data

k	$u(k)$	$y(k)$
1	$u(1)$	$y(1)$
2	$u(2)$	$y(2)$
3	$u(3)$	$y(3)$
\vdots	\vdots	\vdots
N	$u(N)$	$y(N)$

Data jsou vzorkovaná periodou T_s . Časové okamžiky, ve kterých je tvarování vstupu (tvarovačem nultého řádu) a vzorkování výstupu prováděno jsou označeny proměnnou k (diskrétní čas intervalů vzorkování). Z praktických důvodů začneme indexem 1, aby se k shodovalo s indexem prvků vektoru MATLABu. Počet vzorků je N .

2.1 Odvození pro soustavu 1. řádu

Model soustavy (diferenční rovnice 1. řádu) má tvar

$$y(k) + a_1 y(k-1) = b_1 u(k-1) \quad (1)$$

Model obsahuje dva neznámé parametry a_1 a b_1 . Například z přechodové charakteristiky by nám stačily dva body mimo počátek, abychom je určili. Z přechodové charakteristiky bychom mohli určit zesílení a potom by nám stačil už jenom jeden bod. My k tomu přistoupíme trochu jinak. Z rovnice (1) vyjádříme jednokrokovou predikci výstupu

$$\hat{y}(k) = -a_1 y(k-1) + b_1 u(k-1) \quad (2)$$

Stříška nad y znamená, že se nejedná o změřenou hodnotu, ale o hodnotu vypočítanou modelem (predikovanou). Pro $k = 1$ neznáme hodnoty $u(0)$ a $y(0)$, takže můžeme začít predikovat až od $k = 2$. Vyjádříme ještě predikci pro $k = 3$.

$$\hat{y}(2) = -a_1 y(1) + b_1 u(1) \quad (3)$$

$$\hat{y}(3) = -a_1 y(2) + b_1 u(2)$$

Rovnice (3) představují **jednokrokový odhad** výstupu soustavy v čase $k = 2$ a 3 (intervalu vzorkování). Kdybychom znali parametry a_1 a b_1 , tak bychom tyto odhady mohli vypočítat ze znalosti změřených dat $u(1)$, $u(2)$, $y(1)$ a $y(2)$. My to, ale otočíme a ze znalosti $y(2)$ a $y(3)$ vypočítáme neznámé parametry – budeme předpokládat nulovou chybu odhadu – to že jsou odhady rovné změřeným hodnotám

$$y(2) = -a_1 y(1) + b_1 u(1) \quad (4)$$

$$y(3) = -a_1 y(2) + b_1 u(2)$$

Jedná se o soustavu dvou rovnic o dvou neznámých a lze ji vyřešit například použitím maticových operací. Maticově lze soustavu rovnic (4) zapsat jako

$$\underbrace{\begin{bmatrix} y(2) \\ y(3) \end{bmatrix}}_{\mathbf{Y}} = \underbrace{\begin{bmatrix} -y(1) & u(1) \\ -y(2) & u(2) \end{bmatrix}}_{\mathbf{F}} \underbrace{\begin{bmatrix} a_1 \\ b_1 \end{bmatrix}}_{\mathbf{X}} \quad (5)$$

Řešení rovnice (5) získáme jejím vynásobením maticí \mathbf{F}^{-1} (maticí inverzní k matici \mathbf{F}) zleva

$$\mathbf{F}^{-1}\mathbf{Y} = \underbrace{\mathbf{F}^{-1}\mathbf{F}}_{\mathbf{I}}\mathbf{X} \quad (6)$$

Z čehož vyplývá řešení

$$\mathbf{X} = \mathbf{F}^{-1}\mathbf{Y} \quad (7)$$

Znamená to naplnit matici \mathbf{F} a vektor \mathbf{Y} a provést maticové operace v rovnici (7).

Z praktického pohledu to není úplně ideální řešení, protože data jsou vzdálená pouze jeden interval vzorkování a protože obsahují šum, byly by odhady parametrů velmi nepřesné. Využijeme tedy všechna experimentální data – data z tab. 1 a zapíšeme následující soustavu jednokrokových predikcí

$$\hat{y}(2) = -a_1 y(1) + b_1 u(1) \quad (8)$$

$$\hat{y}(3) = -a_1 y(2) + b_1 u(2)$$

⋮

$$\hat{y}(N) = -a_1 y(N-1) + b_1 u(N-1)$$

Soustavu rovnic (8) můžeme zapsat maticově jako

$$\underbrace{\begin{bmatrix} \hat{y}(2) \\ \hat{y}(3) \\ \vdots \\ \hat{y}(N) \end{bmatrix}}_{\hat{\mathbf{Y}}} = \underbrace{\begin{bmatrix} -y(1) & u(1) \\ -y(2) & u(2) \\ \vdots & \vdots \\ -y(N-1) & u(N-1) \end{bmatrix}}_{\mathbf{F}} \underbrace{\begin{bmatrix} a_1 \\ b_1 \end{bmatrix}}_{\mathbf{X}} \quad (9)$$

Někdy se tato rovnice označuje jako **jednokrokový prediktor**. Nelze udělat stejnou úvahu, jako se soustavou rovnic (4) a neznámé koeficienty vypočítat pomocí rovnice (7), protože matice \mathbf{F} není čtvercová – nelze provést její inverzi. Jedná se o přeurčený systém – „máme víc rovnic než neznámých“. Můžeme ovšem udělat odhad parametrů ve smyslu optimálních predikcí. Budeme se snažit nalézt takové parametry a_1 a b_1 , pro které se jednokrokové predikce, co nevíce přiblíží ke změřeným datům. Pro tyto účely zadefinujeme **kritérium** ve tvaru

$$K = \sum_{k=2}^N (\hat{y}(k) - y(k))^2 \quad (10)$$

Jako kritérium jsme zvolili sumu kvadrátů odchylek jednokrokových predikcí výstupů modelu a změřených výstupů. Budeme hledat takové neznámé parametry modelu, které nám dají minimální hodnotu kritéria K . Jedná o minimalizaci sumy čtverců chyb modelu v matematice označovanou jako **metodu nejmenších čtverců** (MNČ). Mohli bychom místo kvadrátu zvolit

absolutní hodnotu, ale kvadratické kritérium je hladké stejně jako kvadratická funkce a minimum můžeme nalézt pomocí derivace. Začneme sumovat od indexu $k = 2$.

Pokud zapíšeme vektor výstupů soustavy \mathbf{Y} jako

$$\mathbf{Y} = \begin{bmatrix} y(2) \\ y(3) \\ \vdots \\ y(N) \end{bmatrix}, \quad (11)$$

Můžeme kritérium (2) zapsat maticově jako

$$K = (\hat{\mathbf{Y}} - \mathbf{Y})^T (\hat{\mathbf{Y}} - \mathbf{Y}) \quad (12)$$

Dosazením rovnice (9) do kritéria (12) získáme

$$K = (\mathbf{F}\mathbf{X} - \mathbf{Y})^T (\mathbf{F}\mathbf{X} - \mathbf{Y}) \quad (13)$$

Rovnice (13) obsahuje pouze data a neznámé parametry. Lze provést její derivaci podle vektoru \mathbf{X} , položit ji rovnu nule a získat výsledný vztah pro výpočet neznámých parametrů

$$\mathbf{X} = (\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T \mathbf{Y} \quad (14)$$

Podrobněji bude odvození tohoto vztahu ukázáno v kapitole 12. Praktický postup je stejný jako v případě rovnice (7). Musíme naplnit matici \mathbf{F} a vektor \mathbf{Y} a provést příslušné maticové operace (14). Člen $(\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T$ je označován jako pseudoinverze matice \mathbf{F} .

Hledané (optimální) parametry modelu jsou prvky vektoru \mathbf{X} (a_1 je první prvek a b_1 druhý).

Diskrétní přenos soustavy 1. řádu (časově diskrétní model) můžeme zapsat jako

$$F(z) = \frac{Y(z)}{U(z)} = \frac{b_1 z^{-1}}{1 + a_1 z^{-1}} = \frac{b_1}{z + a_1} \quad (15)$$

Pomocí rovnice (9) můžeme vypočítat vektor jednokrokových predikcí a pomocí rovnice (12) hodnotu kritéria K . Hodnotu kritéria můžeme vypočítat rovnou z rovnice (13).

2.2 Odvození pro soustavu 2. řádu

Model soustavy (diferenční rovnice 2. řádu) má tvar

$$y(k) + a_1 y(k-1) + a_2 y(k-2) = b_1 u(k-1) + b_2 u(k-2) \quad (16)$$

Model obsahuje čtyři neznámé parametry a_1 , a_2 , b_1 a b_2 . Z rovnice (16) vyjádříme jednokrokovou predikci výstupu

$$\hat{y}(k) = -a_1 y(k-1) - a_2 y(k-2) + b_1 u(k-1) + b_2 u(k-2) \quad (17)$$

Pro $k = 1$ ani $k = 2$ neznáme hodnoty $u(-1)$, $y(-1)$, $u(0)$ a $y(0)$ takže můžeme začít predikovat až od kroku $k = 3$. Opět bychom mohli sestavit 4 predikční rovnice a z nich určit 4 neznámé parametry. Stejně jakou u soustavy 1. řádu provedeme optimální odhad parametrů vzhledem ke kritériu. Jednokrokové predikce výstupu soustavy od $k = 3$ až do N jsou

$$\hat{y}(3) = -a_1 y(2) - a_2 y(1) + b_1 u(2) + b_2 u(1) \quad (18)$$

$$\hat{y}(4) = -a_1 y(3) - a_2 y(2) + b_1 u(3) + b_2 u(2)$$

⋮

$$\hat{y}(N) = -a_1 y(N-1) - a_2 y(N-2) + b_1 u(N-1) + b_2 u(N-2)$$

Soustavu rovnic (18) můžeme zapsat maticově jako

$$\underbrace{\begin{bmatrix} \hat{y}(3) \\ \hat{y}(4) \\ \vdots \\ \hat{y}(N) \end{bmatrix}}_{\mathbf{Y}} = \underbrace{\begin{bmatrix} -y(2) & -y(1) & u(2) & u(1) \\ -y(3) & -y(2) & u(3) & u(2) \\ \vdots & \vdots & \vdots & \vdots \\ -y(N-1) & -y(N-2) & u(N-1) & u(N-2) \end{bmatrix}}_{\mathbf{F}} \underbrace{\begin{bmatrix} a_1 \\ a_2 \\ b_1 \\ b_2 \end{bmatrix}}_{\mathbf{X}} \quad (19)$$

Použijeme stejné kritérium jako v předchozím případě jen s tím rozdílem, že budeme začínat od indexu $k = 3$

$$K = \sum_{k=3}^N (\hat{y}(k) - y(k))^2 \quad (20)$$

Vektor výstupů soustavy \mathbf{Y} zapíšeme jako

$$\mathbf{Y} = \begin{bmatrix} y(3) \\ y(4) \\ \vdots \\ y(N) \end{bmatrix}, \quad (21)$$

Maticový tvar kritéria i řešení je stejné jako v předchozím případě. Opět musíme naplnit matici \mathbf{F} a vektor \mathbf{Y} a provést příslušné maticové operace podle rovnice (14).

Hledané (optimální) parametry modelu jsou prvky vektoru \mathbf{X} (postupně a_1 , a_2 , b_1 a b_2).

Diskrétní přenos soustavy 2. řádu (časově diskrétní model) můžeme zapsat jako

$$F(z) = \frac{Y(z)}{U(z)} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} = \frac{b_1 z + b_2}{z^2 + a_1 z + a_2} \quad (22)$$

Pomocí rovnice (9) můžeme vypočítat vektor jednokrokových predikcí a pomocí rovnice (12) hodnotu kritéria K . Hodnotu kritéria můžeme vypočítat rovnou z rovnice (13).

3 Aplikace experimentální identifikace s využitím MNČ

3.1 Aplikace pro soustavu 1. řádu

Místo měření vytvoříme v MATLABu objekt typu spojité přenos a vypočítáme odezvu soustavy se zesílením 2 a časovou konstantou 5 s na daný vstupní signál. Na výstup modelu soustavy byl přičten šum se směrodatnou odchylkou 0,02.

```
S=tf(2,[5 1]);
Ts=1;
u=[zeros(5,1); ones(15,1); zeros(16,1)];
N=length(u);
t=(0:(N-1))*Ts;
y=lsim(S,u,t)+0.02*randn(N,1);
```

Naplníme matici \mathbf{F} a vektor \mathbf{Y} provedeme výpočet parametrů soustavy

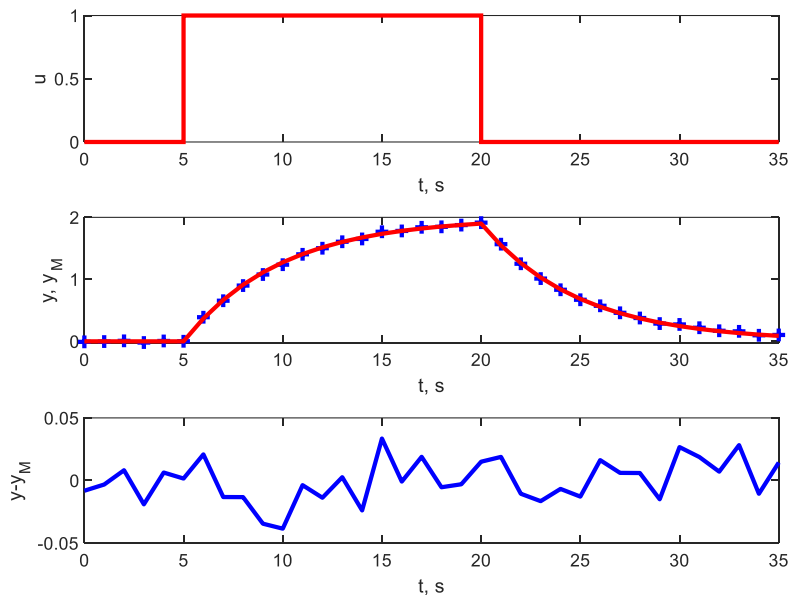
```
F=[-y(1:N-1) u(1:N-1)];
Y=y(2:N);
X=inv(F'*F)*F'*Y;
```


Můžeme vypočítat vektor jednokrokových predikcí a hodnotu kritéria $K = 0,0181$

```
Y0=F*X;
K=(Y0-Y)'*(Y0-Y);
```

Vytvoříme diskretní model, vypočítáme jeho odezvu na vstupní signál a zobrazíme ji společně s „experimentálními“ daty (křížky) a zobrazíme odchylku mezi změřeným a vypočítaným výstupem.

```
M=tf(X(2),[1 X(1)],Ts);
yM=lsim(M,u,t,'zoh');
subplot(3,1,1), stairs(t,u,'r','LineWidth',2), xlabel('t, s'), ylabel('u')
subplot(3,1,2), plot(t,y,'+b',t,yM,'r','LineWidth',2), xlabel('t, s'),
ylabel('y, y_M')
subplot(3,1,3), plot(t,y-yM,'b','LineWidth',2), xlabel('t, s'), ylabel('y-
y_M')
```



Obr. 1 – Identifikace s využitím MNČ pro soustavu 1. řádu (vytvořeno v MATLAB R2021a)

Diskretní přenos modelu je

$$F(z) = \frac{0,3613z^{-1}}{1 - 0,8193z^{-1}} \quad (23)$$

Kdybychom převedli spojitou soustavu do časově diskretního tvaru s periodou vzorkování T_s příkazem $D=c2d(s, Ts)$, získali bychom diskretní přenos „nominálního modelu“

$$F(z) = \frac{0,3625z^{-1}}{1 - 0,8187z^{-1}} \quad (24)$$

3.2 Aplikace pro soustavu 2. řádu

Opět místo měření vytvoříme v MATLABu objekt typu spojitý přenos a vypočítáme odezvu soustavy se zesílením 2 a dvojnásobnou časovou konstantou 3 s na daný vstupní signál. Na

výstup modelu soustavy byl opět přičten šum se směrodatnou odchylkou 0,02. Musíme zmodifikovat příslušný řádek kódu

```
s=tf(2,[9 6 1]);
```

Naplníme matici **F** a vektor **Y**

```
F=[-y(2:N-1) -y(1:N-2) u(2:N-1) u(1:N-2)];  
Y=y(3:N);
```

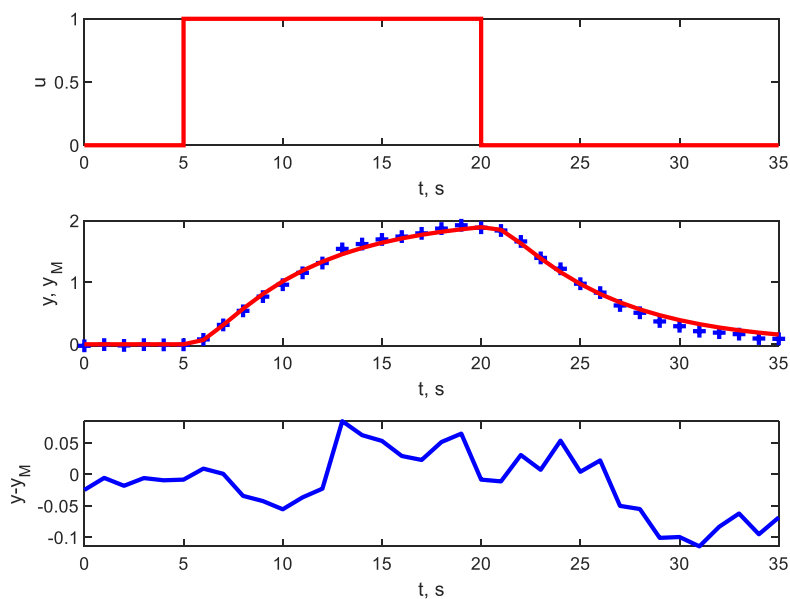
a provedeme výpočet parametrů soustavy – příkazy jsou stejné jako v předchozím příkladu

Můžeme vypočítat vektor jednokrokových predikcí a hodnotu kritéria $K = 0,0416$

Vytvoříme diskretní model

```
M=tf(X(3:4)',[1 X(1:2)'],Ts);
```

a vypočítáme jeho odezvu na vstupní signál a zobrazíme ji společně s „experimentálními“ daty – příkazy jsou stejné jako v předchozím příkladu



Obr. 2 – Identifikace s využitím MNČ pro soustavu 2. řádu (vytvořeno v MATLAB R2021a)

Diskretní přenos modelu je

$$F(z) = \frac{0,1215z^{-1} + 0,121z^{-2}}{1 - 1,119z^{-1} + 0,2331z^{-2}} \quad (25)$$

Kdybychom převedli spojitou soustavu do časově diskretního tvaru s periodou vzorkování T_s příkazem $D=c2d(s, Ts)$, získali bychom diskretní přenos „nominálního modelu“

$$F(z) = \frac{0,08925z^{-1} + 0,07146z^{-2}}{1 - 1,433z^{-1} + 0,5134z^{-2}} \quad (26)$$

4 Úkoly ke cvičení

1. Proveďte simulace z kapitoly 3 - vygenerujte „experimentální“ data pro soustavu 1. a 2. řádu, identifikujte neznámé parametry modelů ve tvaru diskrétních přenosů a výsledky porovnejte s výsledky na obr. 1 a 2.
2. Porovnejte přechodové charakteristiky identifikovaných modelů s časově spojitými i diskrétními „nominálními“ modely.

5 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

jednokrokový odhad, 2
jednokrokový prediktor, 3
kritérium, 3

metoda nejmenších čtverců, 3
vybuzovací signál, 2

Modelování a simulace dynamických systémů

Téma 12: Experimentální identifikace s využitím MNČ – obecné řešení

Studijní cíl

Seznámit studenty s tvorbou časově diskretních modelů experimentálním přístupem s využitím experimentálních dat za použití metody nejmenších čtverců. Bude odvozeno obecné řešení úlohy a metoda bude demonstrována na několika příkladech.

Doba nutná k nastudování

2 hodiny

Klíčová slova

Model, experimentální identifikace, diferenční rovnice, diskretní přenos

1 Obecné řešení experimentální identifikace s využitím MNČ

V kapitole 11 a byl vysvětlen princip experimentální identifikace s využitím metody nejmenších čtverců (MNČ) pro soustavu 1. a 2. řádu. V této kapitole bude odvozeno obecné řešení metody pro model libovolného řádu n .

Opět předpokládejme, že máme změřené následující dvojice vstupu a výstupu soustavy

Tab. 1 – Experimentální data

k	$u(k)$	$y(k)$
1	$u(1)$	$y(1)$
2	$u(2)$	$y(2)$
3	$u(3)$	$y(3)$
\vdots	\vdots	\vdots
N	$u(N)$	$y(N)$

Data jsou vzorkovaná periodou T_s .

Model soustavy (diferenční rovnice n . řádu) má tvar

$$y(k) + a_1 y(k-1) + \dots + a_n y(k-n) = b_1 u(k-1) + \dots + b_n u(k-n) \quad (1)$$

Model obsahuje $2n$ neznámých parametrů a_1 až a_n a b_1 až b_n . Z rovnice (1) vyjádříme jednokrokovou predikci výstupu

$$\hat{y}(k) = -a_1 y(k-1) - \dots - a_n y(k-n) + b_1 u(k-1) + \dots + b_n u(k-n) \quad (2)$$

Můžeme začít predikovat až od $k = n+1$ (nejstarší data máme $y(1)$ a $u(1)$).

Stejně jako v kapitole 11 provedeme optimální odhad parametrů. **Jednokrokové odhady** (predikce) výstupu soustavy od $k = n+1$ až do N jsou

$$\begin{aligned} \hat{y}(n+1) &= -a_1 y(n) - \dots - a_n y(1) + b_1 u(n) + \dots + b_n u(1) \\ \hat{y}(n+2) &= -a_1 y(n+1) - \dots - a_n y(2) + b_1 u(n+1) + \dots + b_n u(2) \\ &\vdots \\ \hat{y}(N) &= -a_1 y(N-1) - \dots - a_n y(N-n) + b_1 u(N-1) + \dots + b_n u(N-n) \end{aligned} \quad (3)$$

Soustavu rovnic (3) – **jednokrokový prediktor** můžeme zapsat maticově jako

$$\underbrace{\begin{bmatrix} \hat{y}(n+1) \\ \hat{y}(n+2) \\ \vdots \\ \hat{y}(N) \end{bmatrix}}_{\hat{\mathbf{Y}}} = \underbrace{\begin{bmatrix} -y(n) & \dots & -y(1) & u(n) & \dots & u(1) \\ -y(n+1) & \dots & -y(2) & u(n+1) & \dots & u(2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -y(N-1) & \dots & -y(N-n) & u(N-1) & \dots & u(N-n) \end{bmatrix}}_{\mathbf{F}} \underbrace{\begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_n \end{bmatrix}}_{\mathbf{X}} \quad (4)$$

Použijeme kvadratické **kritérium** (sumu kvadrátů odchylek predikovaných a změřených výstupů) a začneme sumovat od indexu $k = n+1$

$$K = \sum_{k=n+1}^N (\hat{y}(k) - y(k))^2 \quad (5)$$

Vektor výstupů soustavy \mathbf{Y} zapíšeme jako

$$\mathbf{Y} = \begin{bmatrix} y(n+1) \\ y(n+2) \\ \vdots \\ y(N) \end{bmatrix}, \quad (6)$$

Maticový tvar kritéria (5) je

$$K = (\hat{\mathbf{Y}} - \mathbf{Y})^T (\hat{\mathbf{Y}} - \mathbf{Y}) \quad (7)$$

Dosazením rovnice (4) do kritéria (7) získáme

$$K = (\mathbf{F}\mathbf{X} - \mathbf{Y})^T (\mathbf{F}\mathbf{X} - \mathbf{Y}) \quad (8)$$

Provedeme roznásobení obou členů kritéria

$$K = (\mathbf{X}^T \mathbf{F}^T - \mathbf{Y}^T) (\mathbf{F}\mathbf{X} - \mathbf{Y}) = \mathbf{X}^T \mathbf{F}^T \mathbf{F}\mathbf{X} - \mathbf{X}^T \mathbf{F}^T \mathbf{Y} - \mathbf{Y}^T \mathbf{F}\mathbf{X} + \mathbf{Y}^T \mathbf{Y} = \mathbf{X}^T \underbrace{\mathbf{F}^T \mathbf{F}}_{\mathbf{H}} \mathbf{X} - \underbrace{2\mathbf{Y}^T \mathbf{F}}_{2\mathbf{g}^T} \mathbf{X} + \underbrace{\mathbf{Y}^T \mathbf{Y}}_k \quad (9)$$

Protože je kritérium skalár, tak se člen $\mathbf{X}^T \mathbf{F}^T \mathbf{Y} = \mathbf{Y}^T \mathbf{F}\mathbf{X}$. Kritérium má tvar **kvadratické formy**, kde \mathbf{H} je **hessián** a \mathbf{g} **gradient**.

Minimalizaci můžeme provést následujícím způsobem. Vypočítáme derivaci kritéria podle vektoru \mathbf{X} (podle jednotlivých parametrů vektoru \mathbf{X}) a výslednou soustavu rovnic položíme rovnu nule (nulovému vektoru o velikosti rovné počtu neznámých parametrů)

$$\frac{dK}{d\mathbf{X}} = 2\mathbf{H}\mathbf{X} + 2\mathbf{g} = \mathbf{0} \quad (10)$$

Řešením rovnice (10) je vektor neznámých parametrů modelu soustavy n . řádu

$$\mathbf{X} = -\mathbf{H}^{-1}\mathbf{g} = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{Y} \quad (11)$$

Pro získání řešení musíme naplnit matici \mathbf{F} a vektor \mathbf{Y} daty a provést příslušné maticové operace podle rovnice (11).

Hledané (optimální) parametry modelu jsou prvky vektoru \mathbf{X} naplněné, jak je uvedeno v rovnici (4).

Diskrétní přenos soustavy n . řádu (časově diskrétní model) můžeme zapsat jako

$$F(z) = \frac{Y(z)}{U(z)} = \frac{b_1z^{-1} + \dots + b_nz^{-n}}{1 + a_1z^{-1} + \dots + a_nz^{-n}} = \frac{b_1z + \dots + b_n}{z^n + a_1z + \dots + a_n} \quad (12)$$

Pomocí rovnice (4) můžeme vypočítat vektor jednokrokových predikcí a pomocí rovnice (7) hodnotu kritéria K . Hodnotu kritéria můžeme vypočítat rovnou z rovnice (8).

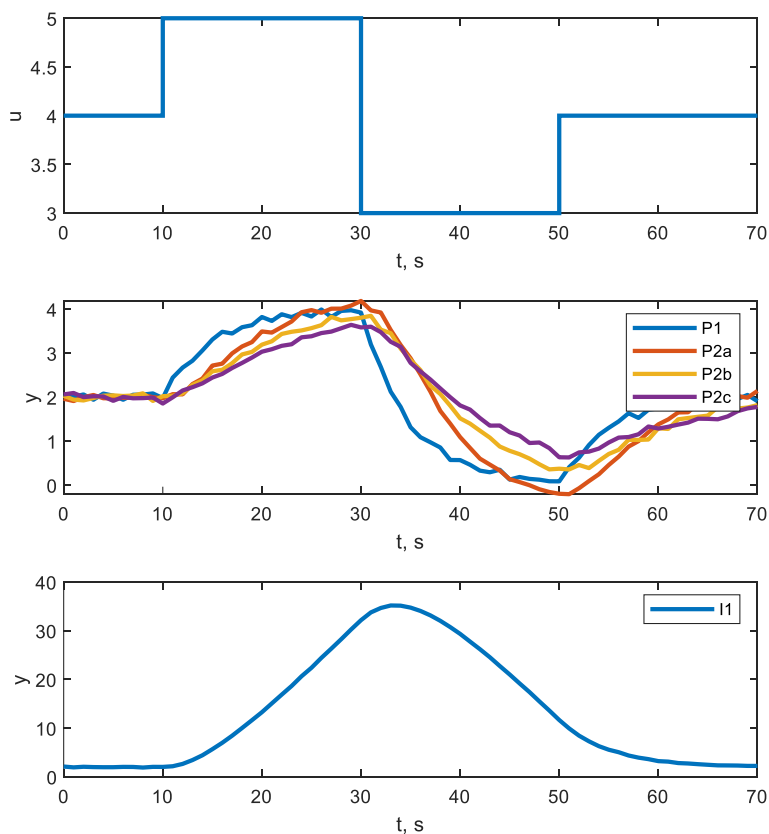
2 Aplikace experimentální identifikace s využitím MNC

Použijeme stejné modely soustav jako v kapitole 10.

```
k1=2;
T1=5;
T0=5;
ksi0=[0.7; 1; 1.3];
P1=tf(k1, [T1 1]);
P2a=tf(k1, [T0^2 2*ksi0(1)*T0 1]);
P2b=tf(k1, [T0^2 2*ksi0(2)*T0 1]);
P2c=tf(k1, [T0^2 2*ksi0(3)*T0 1]);
I1=tf(k1, [T1 1 0]);
```

Data lze buďto vygenerovat skriptem nebo načíst data uložená v souboru `ident_optim_data.mat`

```
load ident_optim_data
subplot(3,1,1), stairs(t,u, 'LineWidth',2), xlabel('t, s'), ylabel('u')
subplot(3,1,2), plot(t,yP1,t,yP2a,t,yP2b,t,yP2c, 'LineWidth',2), xlabel('t, s'), ylabel('y')
legend('1. rad', '2. rad, \xi=0.7', '2. rad, \xi=1', '2. rad, \xi=1.3')
subplot(3,1,3), plot(t,yI1, 'LineWidth',2), xlabel('t, s'), ylabel('y')
legend('1. rad')
```



Obr. 1 – Data pro experimentální identifikaci (vytvořeno v MATLAB R2021a)

Horní graf zobrazuje průběh vstupu všech soustav, v prostředním a dolním grafu jsou „měřené“ výstupy. Prostřední graf zobrazuje výstupy proporcionálních soustav a v dolním grafu je výstup integrační soustavy.

Vytvoříme funkci, která z experimentálních dat t , u a y a pro zvolený řád systému n vrátí diskretní přenos modelu M a hodnotu kritéria K (chyby jednokrokových odhadů)

```
function [M,K]=MNCn_fun(t,u,y,n)

Fy=[]; Fu=[];
for i=1:n
    Fy=[-y(i:end-n+i-1) Fy];
    Fu=[u(i:end-n+i-1) Fu];
end
F=[Fy Fu];
Y=y(n+1:end);
X=inv(F'*F)*F'*Y;
Yo=F*X;
K=(Yo-Y)'*(Yo-Y);
M=tf(X(n+1:end)', [1 X(1:n)'], t(2)-t(1));
```

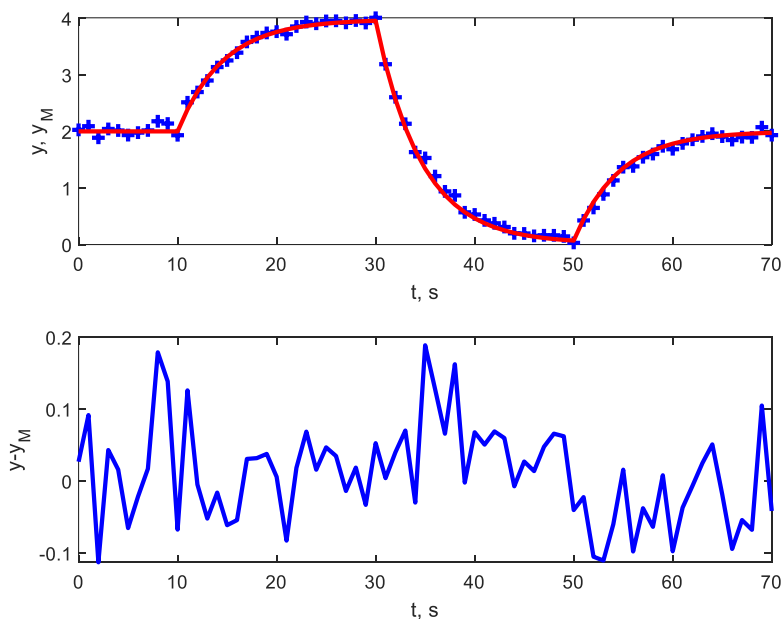
Plnění matice F a vektor Y je provedeno obecně pro libovolný řád, výpočet parametrů soustavy, hodnoty kritéria i diskretního modelu je stejný jako v kapitole 11.

V hlavním skriptu budeme volat funkci $[M,K]=MNCn_fun(t,u,y,n)$.

Následujícím skriptem najdeme model **soustavy P₁** – model proporcionální soustavy 1. řádu – model (1) v kapitole 10.

```
load ident_optim_data
up=4; yp=2;

S=P1; y=yP1; n=1;
[M,K]=MNCn_fun(t,u-up,y-yp,n)
yM=lsim(M,u-up,t,'zoh')+yp;
subplot(2,1,1), plot(t,yP1,'+b',t,yM,'r','LineWidth',2), xlabel('t, s'),
ylabel('y, y_M')
subplot(2,1,2), plot(t,yP1-yM,'b','LineWidth',2), xlabel('t, s'),
ylabel('y-y_M')
```



Obr. 2 – Výstup modelu soustavy P₁ (vytvoreno v MATLAB R2021a)

Na obrázku 2 je zobrazen výstup modelu soustavy na daný vstupní signál společně s „experimentálními“ daty (křížky) a odchylka měřeného a vypočítaného výstupu.

Hodnota kritéria vyjde $K = 0,4204$.

Diskrétní přenos modelu je

$$F(z) = \frac{0,3883z^{-1}}{1 - 0,8039z^{-1}} \quad (13)$$

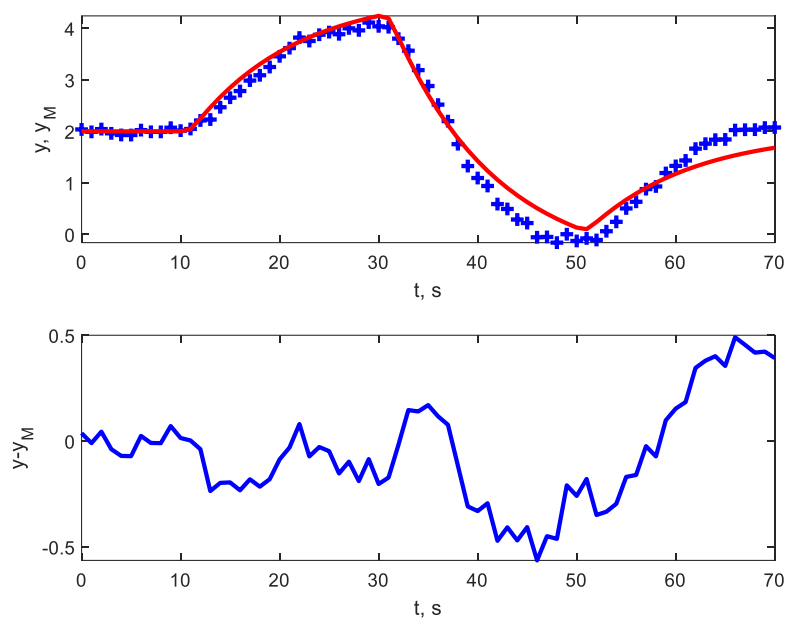
Kdybychom převedli spojitou soustavu do časově diskrétního tvaru s použitou periodou vzorkování příkazem $D=c2d(s, t(2)-t(1))$, získali bychom diskrétní přenos „nominálního modelu“

$$F(z) = \frac{0,3625z^{-1}}{1 - 0,8187z^{-1}} \quad (14)$$

Pokud bychom vypočítali stejné kritérium jako v kapitole 10 – sumu kvadrátů odchylek mezi vypočítanými a „měřenými“ výstupy příkazem $K=(y-y_M)'*(y-y_M)$, dostaneme $K = 0,3388$.

Dále najdeme model **soustavy P_{2a}** – model proporcionální kmitavé soustavy 2. řádu – model (2) v kapitole 10 pro $\xi_0 = 0,7$. Skript zůstane stejný – změníme pouze řádek

S=P2a; y=yP2a; n=2;



Obr. 3 – Výstup modelu soustavy P_{2a} (vytvořeno v MATLAB R2021a)

Hodnota kritéria vyjde $K = 0,6324$.

Diskrétní přenos modelu je

$$F(z) = \frac{0,04346z^{-1} + 0,04346z^{-2}}{1 - 1,075z^{-1} + 0,151z^{-2}} \quad (15)$$

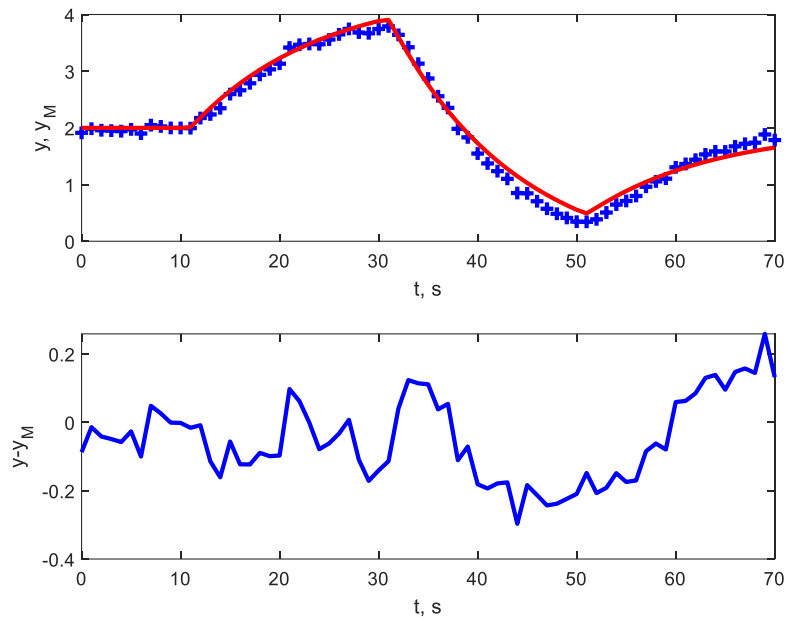
Diskrétní přenos „nominálního modelu“ je

$$F(z) = \frac{0,03639z^{-1} + 0,03315z^{-2}}{1 - 1,721z^{-1} + 0,7558z^{-2}} \quad (16)$$

Kritérium ve smyslu sumy kvadrátů odchylek mezi vypočítanými a „měřenými výstupy“ $K = 4,4864$.

Podobně najdeme model **soustavy P_{2b}** – model aperiodické soustavy 2. řádu se dvěma stejnými kapacitami – model (3) v kapitole 10, pro $\xi_0 = 1$. Změníme řádek skriptu

S=P2b; y=yP2b; n=2;



Obr. 4 – Výstup modelu soustavy P_{2b} (vytvořeno v MATLAB R2021a)

Hodnota kritéria vyjde $K = 0,3039$.

Diskrétní přenos modelu je

$$F(z) = \frac{0,01134z^{-1} + 0,1601z^{-2}}{1 - 0,987z^{-1} + 0,05721z^{-2}} \quad (17)$$

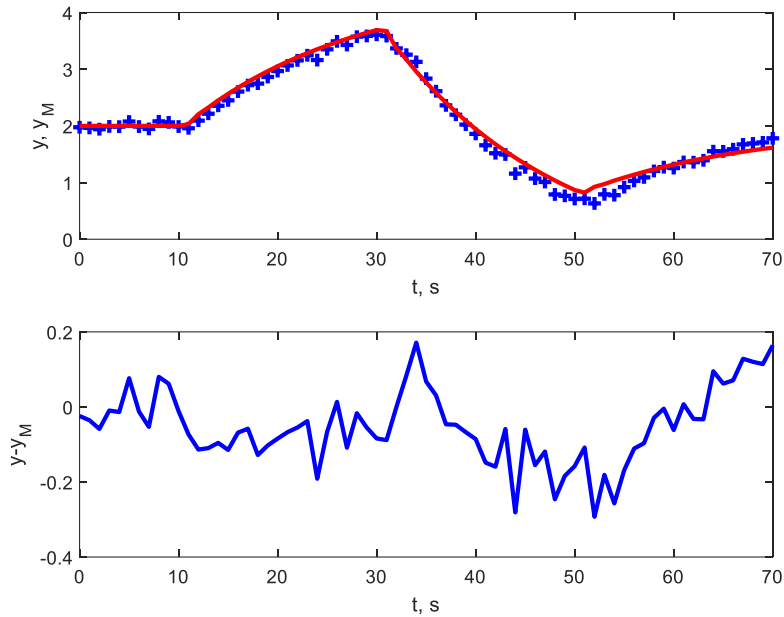
Diskrétní přenos „nominálního modelu“ je

$$F(z) = \frac{0,03505z^{-1} + 0,03067z^{-2}}{1 - 1,637z^{-1} + 0,6703z^{-2}} \quad (18)$$

Kritérium ve smyslu sumy kvadrátů odchylek mezi vypočítanými a „měřenými výstupy“ $K = 1,2141$.

Najdeme také model **soustavy P_{2c}** - model aperiodické soustavy 2. řádu se dvěma různými kapacitami – model (4) v kapitole 10 pro $\xi_0 = 1,3$. Změníme řádek skriptu

S=P2c; y=yP2c; n=2;



Obr. 5 – Výstup modelu soustavy P_{2c} (vytvořeno v MATLAB R2021a)

Hodnota kritéria vyjde $K = 0,3304$.

Diskrétní přenos modelu je

$$F(z) = \frac{0,03161z^{-1} + 0,1553z^{-2}}{1 - 0,6159z^{-1} + 0,3095z^{-2}} \quad (19)$$

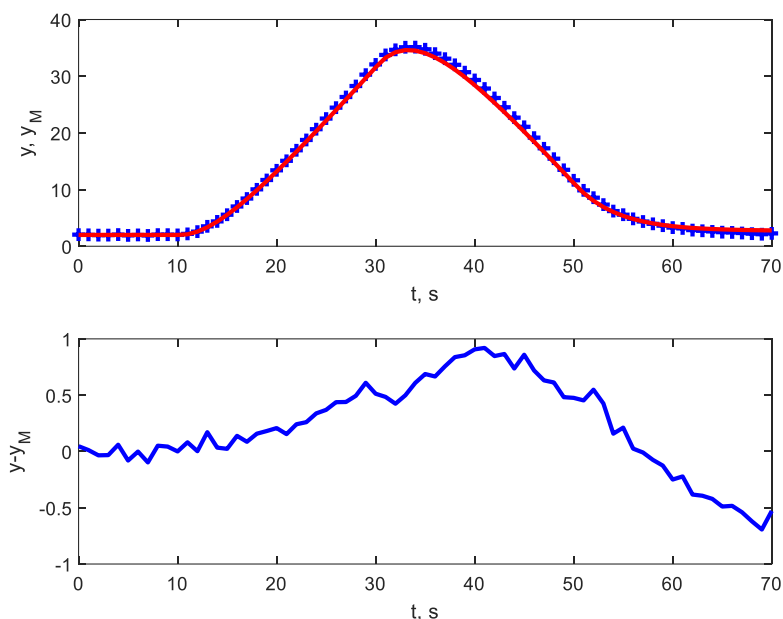
Diskrétní přenos „nominálního modelu“ je

$$F(z) = \frac{0,03377z^{-1} + 0,0284z^{-2}}{1 - 1,563z^{-1} + 0,5945z^{-2}} \quad (20)$$

Kritérium ve smyslu sumy kvadrátů odchylek mezi vypočítanými a „měřenými výstupy“ $K = 0,8910$.

Poslední model **soustavy I₁** je model integrační soustavy se setrvačností 1. řádu – model (5) v kapitole 10. Změníme řádek skriptu

S=I1; y=yI1; n=2;



Obr. 6 – Výstup modelu soustavy I_1 (vytvořeno v MATLAB R2021a)

Hodnota kritéria vyjde $K = 1,0446$.

Diskrétní přenos modelu je

$$F(z) = \frac{0,1025z^{-1} + 0,2904z^{-2}}{1 - 1,794z^{-1} + 0,7936z^{-2}} \quad (21)$$

Diskrétní přenos „nominálního modelu“ je

$$F(z) = \frac{0,1873z^{-1} + 0,1752z^{-2}}{1 - 1,819z^{-1} + 0,8187z^{-2}} \quad (22)$$

Kritérium ve smyslu sumy kvadrátů odchylek mezi vypočítanými a „měřenými výstupy“ $K = 15,2575$.

Z výše uvedených výsledků je zřejmé, že kritérium ve smyslu sumy kvadrátů odchylek mezi vypočítanými a „měřenými výstupy“ časově spojitých modelů dosahuje nižších hodnot než u diskretních modelů. Je to dáno tím, že parametry diskretních modelů jsou optimální pro kritérium sumy kvadrátů chyb jednokrokových predikcí, zatímco u spojitých modelů bylo minimalizováno přímo kritérium vycházející z odchylek vypočítaného výstupu modelu a změřeného výstupu.

3 Úkoly ke cvičení

1. Provedte simulace z kapitoly 2 – použijte „experimentální“ data z kapitoly 10, identifikujte neznámé parametry modelů z kapitoly 10 ve tvaru diskretních přenosů a výsledky porovnejte s výsledky na obr. 2 až 6.

2. Porovnejte přechodové charakteristiky identifikovaných modelů s časově spojitými i diskrétními „nominálními modely“.
3. Porovnejte hodnoty kritérií časově diskrétních modelů s kritérii spojitých modelů z kapitoly 10.

4 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

gradient, 2

hessián, 2

jednokrokový odhad, 2

jednokrokový prediktor, 2

kritérium, 2

kvadratická forma, 2

Modelování a simulace dynamických systémů

Téma 13: Simulace dynamického systému pomocí skriptu MATLABu a v Excelu

Studijní cíl

Seznámit studenty s možností simulace časově diskrétních dynamických systémů ve formě diferenčních rovnic (diskrétních obrazových přenosů) pomocí skriptu MATLABu a v Excelu.

Doba nutná k nastudování

2 hodiny

Klíčová slova

Simulace, diferenční rovnice, diskrétní přenos

1 Principy simulace diskrétních dynamických systémů

Pro simulaci „analytických“ modelů spojitých dynamických systémů jsme používali nástroj MATLABu Simulink. Vyjádřili jsme nejvyšší derivaci sledované veličiny a postupnou integrací (použitím bloku integrátor) jsme došli až k průběhu nederivované veličiny jako výsledku simulace. Pro řešení jsme museli znát počáteční podmínky diferenciální rovnice (počáteční hodnoty jednotlivých integrátorů). Další možností by bylo nadefinovat diferenciální rovnici jako funkci MATLABu a použít řešitele diferenciálních rovnic – například funkci *ode45*. Při numerické experimentální identifikaci jsme pro simulace používali funkci *lsim*, která nám vypočítala průběh výstupu modelu pro daný průběh vstupu. Model může být zadán jako objekt MATLABu definovaný jako spojitý nebo diskrétní přenos či stavový popis. V této kapitole bude ukázán postup, jak si můžeme řešení diferenciální rovnice po převedení na rovnici diferenční naprogramovat sami pomocí skriptu MATLABu. Podobným způsobem můžeme provádět simulaci dynamických systémů v jakémkoliv programovacím jazyce – na konci této kapitoly bude ukázána aplikace v Excelu.

Časově spojitý model budeme diskretizovat pro danou periodu vzorování T_s , čímž získáme model ve tvaru diskrétního stavového popisu nebo diferenční rovnice. V cyklu budeme rekurzivně počítat výstup soustavy $y(k)$ pro vstup $u(k-1)$ a stav $x(k-1)$. U vstupně-výstupního popisu je výstup lineární kombinací minulých vstupů a výstupů a jejich počet je určen řádem systému. V případě simulace uzavřeného regulačního obvodu bude vstupem žádaná hodnota $w(k)$, ze znalosti modelu se vypočte výstup soustavy $y(k)$, regulační odchylka $e(k) = w(k) - y(k)$ a podle diferenční rovnice regulátoru akční zásah $u(k)$.

1.1 Diskretizace modelu soustavy

Model soustavy n . řádu můžeme vyjádřit pomocí **diskrétního stavového popisu** (vnitřní popis)

$$\mathbf{x}(k) = \mathbf{M}\mathbf{x}(k-1) + \mathbf{n}u(k-1), \quad (1)$$

$$y(k) = \mathbf{c}\mathbf{x}(k)$$

nebo pomocí **diferenční rovnice** (vnější popis, vstupně-výstupní)

$$y(k) + a_1y(k-1) + \dots + a_ny(k-n) = b_1u(k-1) + \dots + b_nu(k-n) \quad (2)$$

Při převodu z časově spojitého do diskrétního popisu musíme kromě periody vzorování T_s zavést předpoklad na chování vstupu mezi vzorkovacími intervaly. Pokud uvažujeme, že se vstup mezi intervaly vzorkování nemění (zůstává konstantní, mění se skokově pouze v okamžiku vzorkování), hovoříme o takzvaném předpokladu tvarovače nultého řádu.

Pro jednoduchost si ukážeme výpočet koeficientů diferenční rovnice pro proporcionální soustavu 1. řádu popsanou diferenciální rovnicí

$$T_1 \frac{dy}{dt} + y = k_1 u \quad (3)$$

Přechodová funkce soustavy 1. řádu má tvar

$$y(t) = k_1 \left(1 - e^{-\frac{t}{T_1}}\right) \quad (4)$$

V prvním intervalu vzorkování po skokové změně dostaneme po dosažení hodnoty periody vzorkování T_s za čas t výstup

$$y(T_s) = k_1 \left(1 - e^{-\frac{T_s}{T_1}}\right) \quad (5)$$

Z diferenční rovnice soustavy 1. řádu

$$y(k) + a_1y(k-1) = b_1u(k-1) \quad (6)$$

vyjde výstup v prvním vzorkovacím intervalu

$$y(1) = -a_1y(0) + b_1u(0) \quad (7)$$

a protože se jedná o odezvu na jednotkový skok za nulových počátečních podmínek, tak $y(0) = 0$ a $u(0) = 1$ a

$$y(1) = b_1 \quad (8)$$

Z rovnice (5) vyjde

$$y(1) = k_1 \left(1 - e^{-\frac{T_s}{T_1}}\right) \quad (9)$$

Koeficient b_1 má tedy hodnotu

$$b_1 = k_1 \left(1 - e^{-\frac{T_s}{T_1}}\right) \quad (10)$$

Pro výpočet koeficientu a_1 použijeme vztah pro ustálený stav

$$y(\infty) + a_1y(\infty) = b_1u(\infty) \quad (11)$$

Pro zesílení soustavy platí

$$k_1 = \frac{y(\infty)}{u(\infty)} = \frac{b_1}{1 + a_1} \quad (12)$$

Z rovnice (12) lze vyjádřit

$$b_1 = k_1(1 + a_1) \quad (13)$$

a proto na základě rovnice (10) je

$$a_1 = -e^{-\frac{T_s}{T_1}} \quad (14)$$

1.2 Diskretizace regulátoru

Integrál a derivace v rovnici spojitého PID regulátoru

$$u(t) = k_p \left[e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau + T_D \frac{de(t)}{dt} \right] \quad (15)$$

nahradíme numerickým výpočtem (součtem ploch obdélníků u integrálu a podílem prvních diferencí u derivace)

$$\int_0^t e(\tau) d\tau \approx T_s \sum_{i=1}^k e(i), \quad \frac{de(t)}{dt} \approx \frac{e(k) - e(k-1)}{T_s} \quad (16)$$

Polohový tvar číslicového PID regulátoru je

$$u(k) = k_p \left[e(k) + \frac{T_s}{T_I} \sum_{i=1}^k e(i) + T_D \frac{e(k) - e(k-1)}{T_s} \right] \quad (17)$$

Z praktických důvodů je v případě použití I složky vhodnější **přírůstkový tvar číslicového PID regulátoru**, u kterého lze jednoduše realizovat anti-windup

$$u(k) = u(k-1) + \underbrace{k_p \left(1 + \frac{T_s}{T_I} + \frac{T_D}{T_s} \right)}_{q_0} e(k) + \underbrace{k_p \left(-1 - \frac{2T_D}{T_s} \right)}_{q_1} e(k-1) + \underbrace{k_p \frac{T_D}{T_s}}_{q_2} e(k-2) \quad (18)$$

$\Delta u(k-1)$

2 Simulace pomocí skriptu MATLABu

2.1 Simulace soustavy

Následující skript vypočítá odezvu proporcionální soustavy 1. řádu na zadaný vstupní signál. Zesílení soustavy $k_1 = 2$, časová konstanta $T_1 = 5$ s a perioda vzorkování $T_s = 1$ s.

```
clear
k1=2;
T1=5;
Ts=1;
u=[zeros(10,1); ones(20,1); -ones(20,1); zeros(21,1)];
N=length(u);
y=zeros(N,1);
t=(0:(N-1))*Ts;
a1=-exp(-Ts/T1);
```



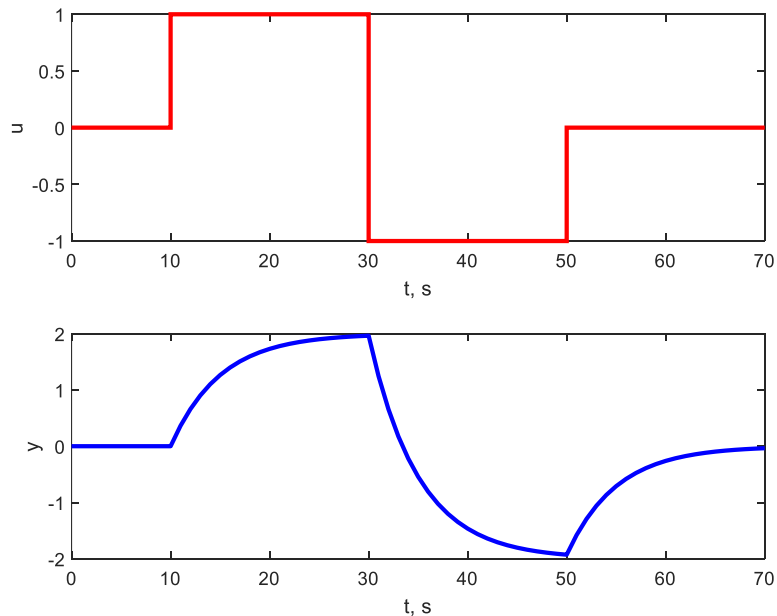
```

b1=k1*(1+a1);

for k=2:N
    y(k)=-a1*y(k-1)+b1*u(k-1);
end

subplot(2,1,1), stairs(t,u,'r','LineWidth',2), xlabel('t, s'), ylabel('u')
subplot(2,1,2), plot(t,y,'b','LineWidth',2), xlabel('t, s'), ylabel('y')

```



Obr. 1 – Odezva soustavy (vytvoreno v MATLAB R2021a)

2.2 Simulace řízení dvupolohovým regulátorem

Předešlý script zmodifikujeme, aby provedl simulaci řízení s dvupolohovým regulátorem, který spíná akční zásah -1 a 1.

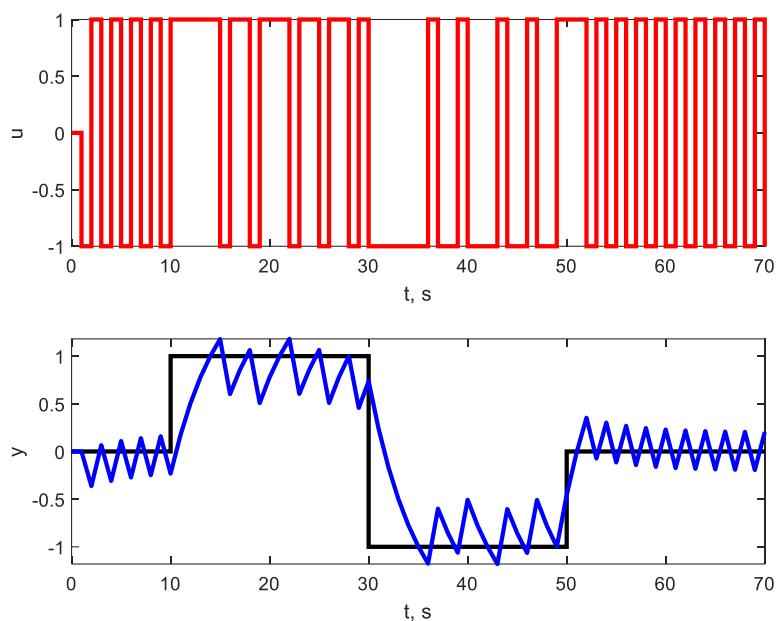
```

w=[zeros(10,1); ones(20,1); -ones(20,1); zeros(21,1)];
umax=1; umin=-1;
N=length(w);
y=zeros(N,1); e=y; u=y;

for k=2:N
    y(k)=-a1*y(k-1)+b1*u(k-1);
    e(k)=w(k)-y(k);
    if e(k)>0
        u(k)=umax;
    else
        u(k)=umin;
    end
end

subplot(2,1,1), stairs(t,u,'r','LineWidth',2), xlabel('t, s'), ylabel('u')
subplot(2,1,2), stairs(t,w,'k','LineWidth',2), hold,
plot(t,y,'b','LineWidth',2), xlabel('t, s'), ylabel('y')

```



Obr. 2 – Řízení dvoupolohovým regulátorem (vytvořeno v MATLAB R2021a)

2.3 Simulace řízení PID regulátorem

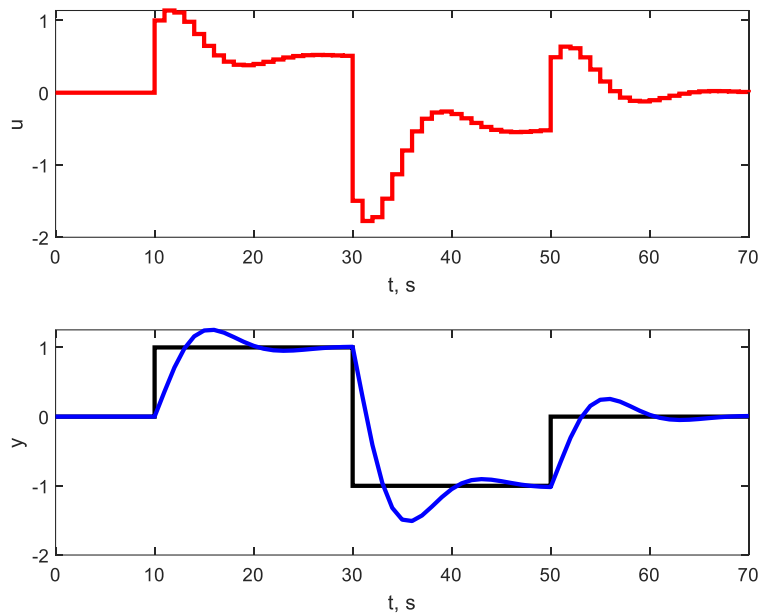
Poslední modifikací zaměníme dvoupolohový regulátor za číslicovou verzi PI regulátoru se zesílením $k_p = 0,5$ a integrační časovou konstantou $T_i = 1$ s. Perioda vzorkování 1 s je na hranici použitelnosti vzhledem k dynamice soustavy. Kdybychom zvolili kratší periodu, přiblížili bychom se chování spojitému PID regulátoru.

```

kp=0.5;
Ti=1;
Td=0;
q0=kp*(1+Ts/Ti+Td/Ts);
q1=-kp*(1+2*Td/Ts);
q2=kp*Td/Ts;

for k=3:N
    y(k)=-a1*y(k-1)+b1*u(k-1);
    e(k)=w(k)-y(k);
    u(k)=u(k-1)+q0*e(k)+q1*e(k-1)+q2*e(k-2);
end

```



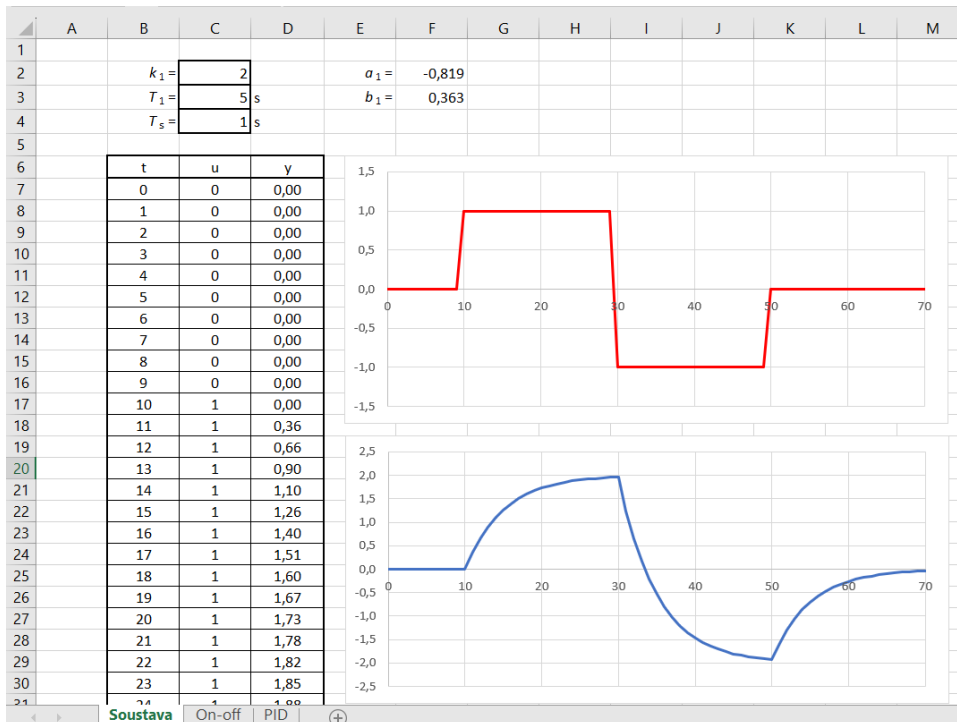
Obr. 3 – Řízení PID regulátorem (vytvořeno v MATLAB R2021a)

3 Simulace v Excelu

Stejně simulace jako v předešlé kapitole provedeme také v Excelu. Data budou v tabulce tvořit sloupce, parametry zadáme do buněk, aby došlo při jejich změně k přepočítání. Pro zobrazení všech veličin jsou použity XY-bodové grafy.

3.1 Simulace soustavy

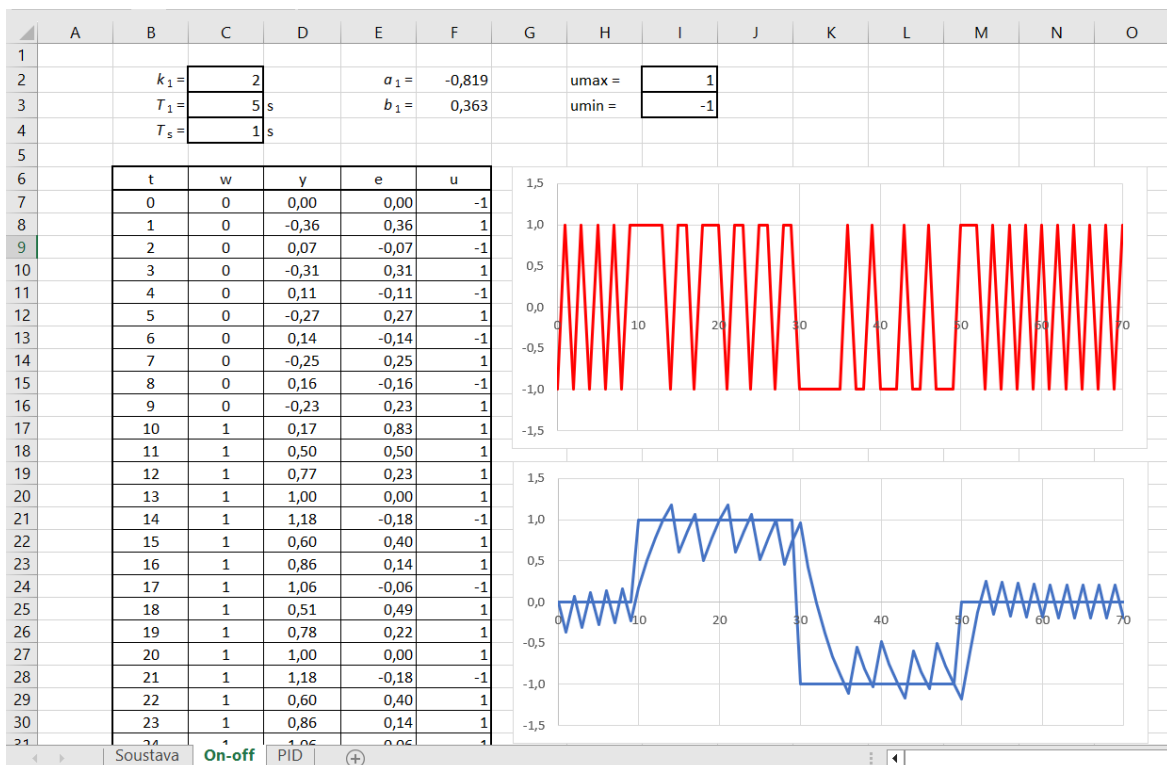
V buňkách F2 a F3 jsou vzorečky pro přepočet parametrů diferenční rovnice $=-EXP(-C4/C3)$ a $=C2*(1+F2)$. V buňce D7 je 0 (počáteční podmínka výstupu soustavy). V buňce D8 je vzoreček pro výpočet výstupu soustavy $=-F52*D7+F53*C7$. Ten je poté rozkopírován do celého sloupce.



Obr. 4 – Odezva soustavy (vytvořeno v Excel - Microsoft Office 365)

3.2 Simulace řízení dvupolohovým regulátorem

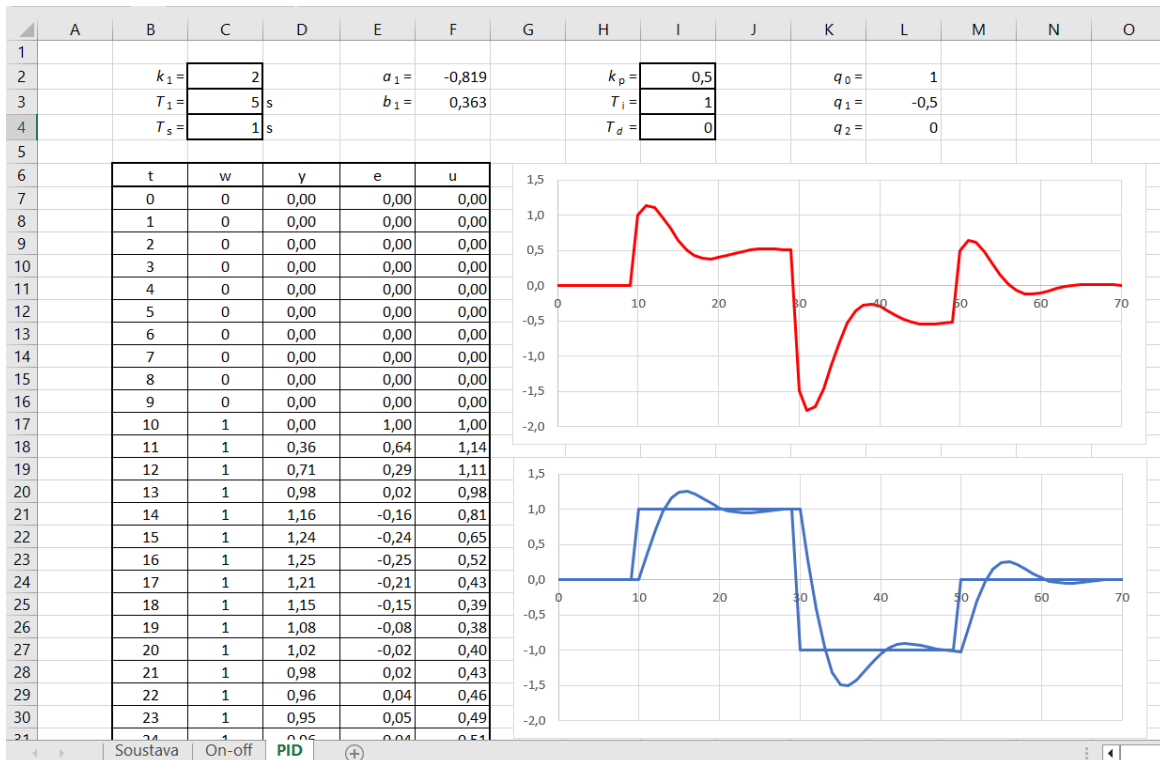
Ve sloupcích dat jsme místo vstupu soustavy použili žádanou hodnotu, v buňce E7 jsme vypočítali regulační odchylku vzorečkem $=C7-D7$ a v buňce F7 akční zásah vzorečkem $=KDYŽ(E7>0; \$I\$2; \$I\$3)$. Opět jsme oba vzorečky rozkopírovali do celého sloupce.



Obr. 5 – Řízení dvupolohovým regulátorem (vytvořeno v Excel - Microsoft Office 365)

3.3 Simulace řízení PID regulátorem

V buňkách L2, L3 a L4 jsou přepočtené parametry PID regulátoru na hodnoty koeficientů diferenční rovnice přírůstkového tvaru regulátoru $=k_1(1+\frac{q_1}{T_i}+\frac{q_2}{T_i^2})$, $=k_1(1+2*\frac{q_1}{T_i}+\frac{q_2}{T_i^2})$ a $=-k_1*\frac{q_1}{T_i}$. Abychom měli k dispozici dvě minulé regulační odchylky (obě jsou nula), začíná vzoreček pro výpočet akčního zásahu až v buňce F9 =F8+L2*E9+L3*E8+L4*E7. Opět je nutné vzoreček rozkopírovat do celého sloupce.



Obr. 6 – Řízení PID regulátorem (vytvoreno v Excel - Microsoft Office 365)

4 Úkoly ke cvičení

1. Postupem uvedeným v kapitole 2 vytvořte skript MATLABu pro simulaci výstupu soustavy, řízení dvupolohovým regulátorem a řízení číslicovým PID regulátorem. Výsledky porovnejte s průběhy na obr. 1, 2 a 3.
2. Postupem uvedeným v kapitole 3 odsimulujte v Excelu stejné úlohy jako v kapitole 1. Výsledky porovnejte s průběhy na obr. 4, 5 a 6.

5 Použitá literatura

NOSKIEVIČ, P. 1999. *Modelování a identifikace systémů*. Ostrava: Montanex. ISBN 80-7225-030-2.

DUŠEK, F., HONC, D. 2005. *Matlab a Simulink: úvod do používání*. Pardubice: Univerzita Pardubice. ISBN 80-7194-776-8.

Rejstřík

diferenční rovnice, 2

diskrétní stavový popis, 2

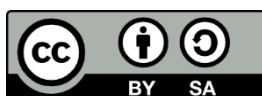
polohový tvar číslicového PID regulátoru, 3

přírůstkový tvar číslicového PID regulátoru,

3

Vytvořeno v rámci projektu **Studijní program Automatizace (SPAUT)**
na **Univerzitě Pardubice**, reg. č. NPO_UPCE_MSMT-16591/2022.

Toto dílo podléhá licenci Creative Commons BY 4.0. Pro zobrazení licenčních podmínek navštivte <https://creativecommons.org/licenses/by-sa/4.0/>.



Financováno
Evropskou unií
NextGenerationEU



Národní
plán
obnovy

MS
MIT
MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY