

GUI APLIKACE PRO VÝUKU AUTOMATIZACE

J. Škutová

VŠB-Technická univerzita Ostrava, Fakulta strojní

Abstrakt

V rámci projektu ESF byla vytvořena GUI aplikace pro výuku předmětu **Základy automatizace**. Cílem aplikace je umožnit studentům vykreslit základní charakteristiky lineárních invariantních systémů pro vybraný systém, který je možné definovat buď mnohočlenem, nebo výběrem konkrétního přenosu systému.

1 Úvod

Oblast analýzy dynamických systémů je jednou z částí oboru automatizace. Popis dynamických vlastností spojitého systému lze popsat v oblasti komplexní proměnné obrazovým přenosem, v oblasti časové pak přechodovou, impulsní charakteristikou a v oblasti kmitočtové jsou spojitý systém popsány kmitočtovými charakteristikami. S analýzou dynamických systémů mají být v rámci výuky fakultního předmětu **Základy automatizace** seznámeni studenti fakulty strojní. Cílem výuky je uvedení studentů do problematiky automatizace, konkrétně experimentální identifikace systémů. Studenti jsou seznámeni s jednotlivými základními typy systémů, jejich charakteristikami a na konkrétních příkladech pak vlivy základních parametrů (zesílení, integrační a derivační časové konstanty a zpoždění) na průběhy přechodové, impulsní a kmitočtové charakteristiky.

V rámci projektu ESF, který podporuje výuku formou e-learningu, vznikla interaktivní aplikace, která má za následek přehlednější, rychlejší a zajímavější formu výuky analýzy dynamických systémů.

2 Analýza dynamických systémů

Přechodová funkce systému je jeho odezva na vstup Heavisideova jednotkového skoku $\eta(t)$. Značí se $h(t)$ a její časová závislost se nazývá přechodová charakteristika. Přechodovou funkci lze určit z analyticky z obrazového přenosu systému $G(s)$, což je vyjádřeno vztahem

$$h(t) = L^{-1}\{H(s)\} = L^{-1}\left\{\frac{1}{s}G(s)\right\} \quad (1)$$

kde $H(s)$ je obraz přechodové funkce. Pro konkrétní příklad proporcionalního systému se setrvačností 2. řádu, kde přenos systému je dán

$$G(s) = \frac{k}{(T_1s + 1)(T_2s + 1)} \quad (2)$$

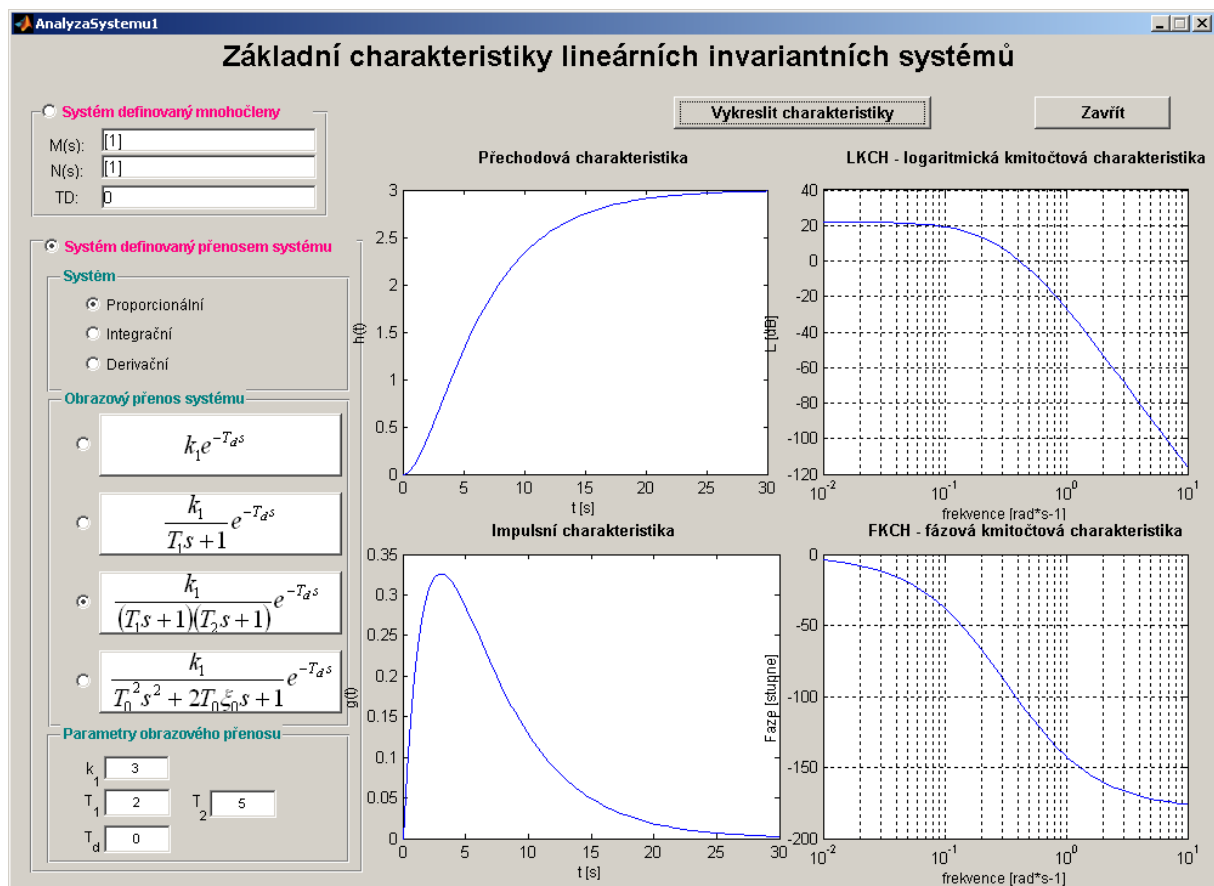
pak přechodová charakteristika s konkrétními parametry obrazového přenosu systému $k = 3$, $T_1 = 2$, $T_2 = 5$ je dána grafem (obrázek 1).

Impulsní funkce systému je jeho odezva na vstup ve tvaru Diracova impulsu $\delta(t)$. Označuje se $g(t)$ a její časová závislost je impulsní charakteristika. Impulsní funkce je možné vyjádřit analyticky z obrazového přenosu systému $G(s)$ takto

$$g(t) = L^{-1}\{G(s)\} \quad (3)$$

Pro výše uvedené parametry obrazového přenosu systému je rovněž impulsní charakteristika dána grafem (obrázek 1).

Logaritmické kmitočtové charakteristiky (LKCH) vyjadřují grafické znázornění kmitočtového přenosu v logaritmických souřadnicích. LKCH předávají informace o amplitudě a fázi dané kmitočtové složky. Nejčastěji se využívají v oblasti elektronických obvodů, konkrétně při filtraci signálu, práce se zvukem. Kmitočtové charakteristiky se získají odezvou soustavy na harmonicky proměnlivý vstupní signál konstantní amplitudy. Měření se opakuje s různými kruhovými frekvencemi u vstupního signálu. Na obrázku 1 jsou zobrazeny logaritmické kmitočtové charakteristiky pro vybraný proporcionalní systém se setrvačností 2. Řádu dle výše uvedených konkrétních parametrů.



Obrázek 1: GUI aplikace pro zobrazení základních charakteristik systémů

3 Grafická uživatelská aplikace pro analýzu dynamických systémů

Program Matlab byl zvolen pro realizaci aplikace z oblasti Analýzy systémů. Předpokladem aplikace byla jednoduchost, názornost a vizuální podpora pro výuku předmětu Základy automatizace. Uživatel aplikace si zvolí typ systému, jehož základní charakteristiky chce vykreslit. Systém je vždy dán obrazovým přenosem $G(s)$ a jeho parametry.

$$G(s) = \frac{b_m s^m + \dots + b_1 s + b_0}{a_n s^n + \dots + a_1 s + a_0} \cdot e^{-T_d s} = \frac{M(s)}{N(s)} e^{-T_d s} \quad (4)$$

V případě, že uživatel aplikace neumí určit typ systému dle zadaného přenosu, pak je možné zvolit první variantu zadávání systému, tj. „Systém definovaný mnohočleny“. Zde se systém zadává pomocí mnohočlenů čitatele $M(s)$ a jmenovatele $N(s)$ obrazového přenosu ve tvaru běžně zadávaného řádkového vektoru v prostředí Matlab. V případě, že systém nemá nulové dopravní zpoždění, pak je nutné změnit hodnotu T_d jako skalární veličinu.

Druhou možností, jak zadat systém, je vybrat typ systému (proporcionální, integrační, derivační) a dále zvolit konkrétní obrazový přenos, který student porovná se zadaným přenosem systému a následně vkládá jednotlivé jeho parametry dolů v levé části okna aplikace. Poté tlačítkem *Vykreslit charakteristiky* zobrazí pro konkrétní přenos přechodovou, impulsní a kmitočtové charakteristiky. Ve výuce pak dochází ke konfrontaci souvislostí mezi typem systému, typem přenosu, zadanými parametry a konkrétními průběhy zobrazených charakteristik.

4 Tvorba grafické uživatelské aplikace

Příspěvek má za cíl seznámit zejména s pozadím aplikace, tj. zobrazením jednotlivých přenosů jako matematických vzorců v aplikaci, řízeným ovládním prepínačů a zobrazováním správných přenosů dle volby typu systému (proporcionální, integrační, derivační), zobrazením potřebných polí

pro zadávání parametrů obrazových přenosů a dalšími prvky zajišťující komfort a bezpečnost zadávaných nebo zobrazovaných dat v grafické uživatelské aplikaci.

4.1 Vkládání matematických vzorců do grafické uživatelské aplikace

Uživatelská aplikace umožní uživateli vložit text prostřednictvím textového nebo editačního pole, vkládat ovládací prvky a zobrazovat grafy. V případě, že textem není běžný ascii text, ale např. matematický vzorec jako textový popis pro uživatele aplikace, není možné pro zobrazení matematického vzorce využít běžného textového pole. Vhodnějším způsobem je uvádět vzorec jako obrazovou informaci, která je dána rastrovým obrázkem. Zobrazení matematického vzorce lze tedy aplikovat pouze na ovládací prvky tlačítka („Push Button“) nebo přepínač („Toggle Button“).

Matematický vzorec je vhodné vytvořit v programu, který podporuje tvorbu vzorců. V případě, že matematických vzorců bude v aplikaci použito více, pak pro jejich zobrazení je vhodné použít maximální potřebnou šířku a výšku, respektive maximální počet pixelů v horizontálním a vertikálním směru. V programu podporujícího práci s rastrovou grafikou, pak do předem stanoveného rozměru vložit kopii vzorce (vyříznutá část snímku aktivního okna obrazovky). Vzorec uložit jako soubor ve vybraném grafickém formátu, např. BMP, JPEG, PNG, TIFF a dalších. Soubor pak načíst funkcí

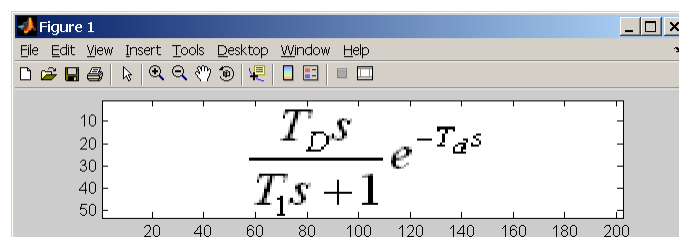
```
>> vzorec=imread('D:\..\vzorecP1radu','BMP');
```

do proměnné *vzorec*, která bude obsahovat v případě černobílého obrázku dvourozměrnou matici, v případě barevného obrázku trojrozměrnou matici.

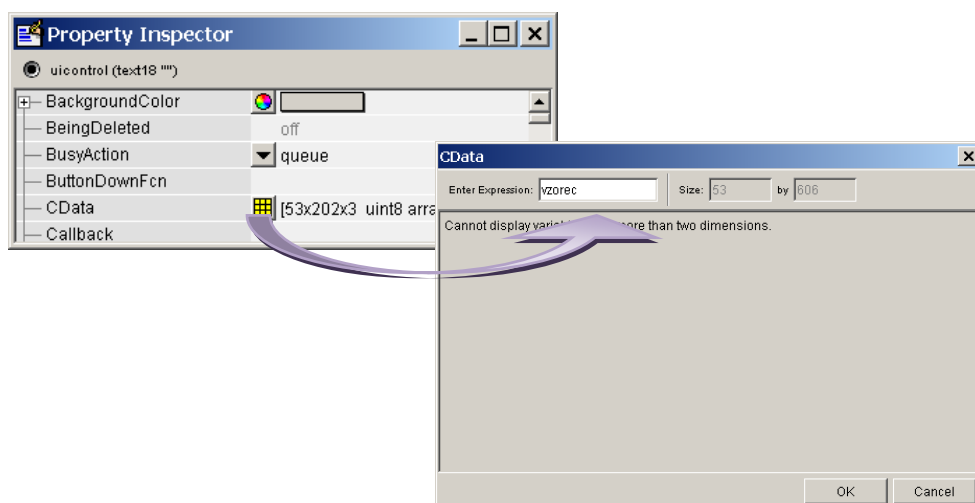
Správně přiřazený obrázek do proměnné lze ověřit funkcí

```
>> image(vzorec);
```

zobrazený v okně „Figure 1“ (obrázek 2), kde je mimo jiné na osách grafu zobrazen počet pixelů vzorce. Po ověření je možné přiřadit matici *vzorec* k příslušnému ovládacímu prvku v okně GUI aplikace.



Obrázek 2: Zobrazení matice *vzorec* ve formátu rastrového obrázku



Obrázek 3: Nastavení vlastnosti CData ovládacího prvku “Push Button”

Pro vybraný ovládací prvek v grafické uživatelské aplikaci zobrazit okno “Property Inspector” a vlastnost “CData” změnit vložit matici vzorec (obrázek 3). Obrázek se zobrazí na ovládacím prvku až po spuštění grafické uživatelské aplikace. Odstranit data u vlastnosti CData je možné vložením jakékoliv proměnné definované prázdnou hodnotou.

4.2 Řízené ovládání přepínačů

Dynamický systém může uživatel této GUI aplikace definovat dvěma způsoby:

1. Mnohočlenem čitatele obrazového přenosu $M(s)$, mnohočlenem jmenovatele obrazového přenosu $N(s)$ a skalární hodnotou představující časovou konstantu dopravního zpoždění T_d .
2. Volbou typu systému a příslušného obrazového přenosu pomocí zobrazených vzorců v aplikaci a nastavením konkrétních parametrů zesílení a časových konstant.

Uživatel zvolí pouze jednu možnost a ostatní přepínače a editační pole jsou mu znepřístupněny pro zadávání. V okamžiku, kdy je tedy zvolena možnost první („Systém definovaný mnohočleny“), uživatel nemůže používat přepínače pro volbu systému nebo obrazového přenosu systému (obrázek 1).

Vzhledem k tomu, že proporcionalní systém se setrvačností 1. řádu má jiný vzorec než integrační systém se setrvačností 1. řádu, pak při výběru daného systému (proporcionalní, integrační nebo derivační) se také změní vzorec. Zde bylo využito nastavení hodnoty **Off** pro vlastnost „Visible“ u ovládacích prvků, které zobrazují vzorce jiných než vybraného systému. Jednoduchým řešením se tak zmenší prostor nezbytný k zadávání parametrů systému a tím se zvětší prostor pro zobrazení grafů v pravé části aplikace.

Ve verzi programu Matlab 7.0.4, v níž byla tato aplikace vytvořena, není možnost umístit přepínače do skupiny tak, jak je to běžné v jiných programech. To znamená, že při volbě jednoho přepínače se ostatní přepínače dané skupiny vypnou. V současnosti každý navrhovatel GUI aplikace musí ošetřit funkčnost přepínačů v rámci m-souboru.

4.3 Zobrazení požadovaného počtu editačních polí

Obdobně dochází k zobrazení nebo skrytí editačních polí, které jsou nutné pro zadávání všech parametrů vybraného systému. V případě systému bez setrvačnosti uživatel zadává pouze hodnoty zesílení a dopravního zpoždění. U systému se setrvačností 1. řádu nebo 2. řádu se zvyšuje počet zadaných parametrů o časové konstanty T_1 případně T_2 . V podstatě aplikace využívá maximálního počtu čtyř editačních polí, pak pro systémy bez setrvačnosti využívá minimálně dvou editačních polí. Problém správného sestavení vstupních dat pro vykreslení přechodové, impulsní a frekvenčních charakteristik (obrázek 4) je vyřešen na úrovni programového kódu m-souboru.

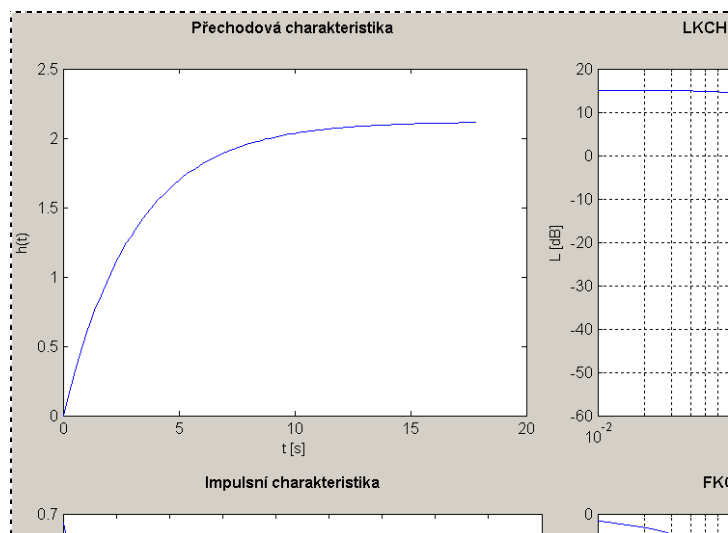
```
570 -     if strcmp(get(handles.ETd1, 'Visible'), 'on')==1
571 -         par4=str2num(get(handles.ETd1, 'String'));
572 -         if isempty(par4)==1 msgbox('Zadejte parametr obrazového přenosu ve čtvrtém editačním poli !', 'Ch
573 -
574 -     end
575 -
576 -
577 -     % Určení přenosu systému pomocí volby typu přenosu a zadáním: k, T1, T2, Td.
578 -     if handles.TypSystemu==1
579 -         % proporcionalní systém
580 -         if handles.TypPrenosu==1
581 -             M=par1;
582 -             N=1;
583 -             Td=par2;
584 -         elseif handles.TypPrenosu==2
585 -             M=par1;
586 -             N=[par2 1];
587 -             Td=par4;
588 -         elseif handles.TypPrenosu==3
589 -             M=par1;
590 -             N=[par2*par3 par2+par3 1];
591 -             Td=par4;
592 -         elseif handles.TypPrenosu==4
593 -             M=par1;
594 -             N=[par2*par2 2*par2*par3 1];
595 -             Td=par4;
596 -         else
597 -             msgbox('Při určení typu přenosu systému došlo k chybě procedury. Kontaktujte správce aplikace.',
```

Obrázek 4: Ukázka M-souboru pro grafickou uživatelskou aplikaci.

Názvy editačních polí byly navrženy z hlediska jednoznačnosti obecnými názvy: *par1*, *par2*, *par3* a *par4*, i když se do nich vkládají parametry k , T_1 , T_2 , T_0 , ξ a T_d . Není nutné mít více než čtyři editační pole, proto mají obecné názvy a zobrazí se podle volby obrazového přenosu pouze příslušné textové pole (popisky).

4.4 Požadovaná změna zobrazení charakteristiky v celé oblasti grafu

Pro vykreslení přechodové funkce je použita funkce *step* a impulsní charakteristika je vypočtena funkcí *impulse*. Obě funkce vrací hodnoty nezávislé proměnné (čas t) a závislé proměnné pro přechodovou charakteristiku $h(t)$ a pro impulsní charakteristiku $g(t)$.



Obrázek 5: Ukázka zobrazení přechodové charakteristiky s funkcí *step* bez úprav časové osy

Při určitých kombinacích vstupních parametrů funkcí *step* nebo *impulse* jsou návratové hodnoty času těchto funkcí dány nezaokrouhlenými hodnotami, což se projeví v grafu zobrazením části průběhu charakteristiky. Obecně tyto funkce určí dobu výpočtu přechodové nebo impulsní charakteristiky dle svých vnitřních algoritmů, které zajistí vykreslení charakteristik po ustálení překmitu nebo aperiodického průběhu.

Jednou z možností, jak vykreslit charakteristiky v celém rozsahu času, je upravit maximální hodnotu osy x dle maximální hodnoty nezávislé proměnné, kterou vrací funkce *step* nebo *impulse*. Pak bude graf vykreslen od levého do pravého okraje grafu, ale měřítko v ose nezávislé proměnné t nebude zobrazeno ve vhodném měřítku a graf by pak byl ukončen hodnotou např. 17.858 (pro charakteristiku na obrázku 5). Vhodnějším řešením se jeví provést výpočet funkce *step* nebo *impulse* na nejbližší vyšší zaokrouhlenou hodnotu. Pak pro zaokrouhlení byla použita funkce *ceil*, která zaokrouhluje kladnou hodnotu času výpočtu na nejbližší celé číslo směrem ke kladnému nekonečnu. Pak úprava algoritmu výpočtu přechodové funkce v m -souboru dané grafické uživatelské aplikace je uvedena na obrázku 6.

```

sys=tf(M,N,'ioDelay',Td); % tvorba přenosu systému z mnohočlenu M a N

axes(handles.APrechodova);
cla;

% Vypocet prechodove charakteristiky včetně vykreslení
[y,t]=step(sys);
[m,n] = size(t);
tfinal=ceil(t(m,n)/10)*10;
[y,t]=step(sys,tfinal)
plot(t,y);
xlabel('t [s]','FontSize',9)
ylabel('h(t)','FontSize',9)

% Vypocet impulsní charakteristiky včetně vykreslení
axes(handles.AImpulsní);

```

Obrázek 6: Část m -souboru GUI aplikace pro výpočet hodnot přechodové charakteristiky

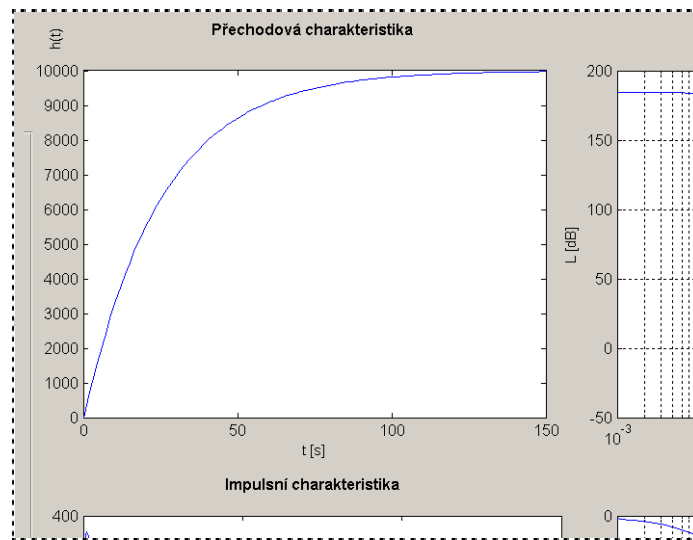
Vstupními parametry pro výpočet přechodové nebo impulsní funkce je přenos systému, který je dán proměnnou `sys`, která je objektem obrazového přenosu (aplikací funkce `tf`) a která je použita jako vstupní parametr funkce `step` nebo `impz`.

4.5 Zamezení zobrazení popisu osy přes jiné objekty

V případě, že pro zobrazení grafu je dále vložen popis os, umístění těchto popisů se nastavuje pro každý objekt `axes` v rámci grafické uživatelské aplikace automaticky. U grafu s měřítkem, jejichž popis je šířkou textu pro popisky velký, pak i popis osy `y` je vsazen příliš daleko od osy `y` a může zasahovat do jiné oblasti, v níž jiné ovládací prvky GUI aplikace (obrázek 1). Jedním z možných řešení je zmenšit oblast pro prvky `axes` tak, aby při různých testovacích hodnotách na ose `y` nedocházelo k zobrazení popisu osy `y` přes jiné ovládací prvky.

V případě, že studenti potřebují pro zjištění ustálených hodnot, časových konstant systému co největší zobrazení grafu v aplikaci, je vhodné tedy maximalizovat prostor pro zobrazení grafu a minimalizovat volný prostor v GUI aplikaci.

Jednou z možností je ošetření zobrazování popisů os tak, aby nezasahovaly do jiné části určené pro zobrazení dalších ovládacích prvků v grafické aplikaci (obrázek 7). Zde bylo využito vlastnosti `Position`, která vrátí hodnoty pozice počátku ovládacího prvku a jeho šířky a výšky pro ovládací prvek `axes` nebo jeho pozice v ose `x`, `y` a `z` pro popis osy `y` ovládacího prvku `axes`.



Obrázek 7: Ukázka zobrazení grafu s posunem popisu osy `y`

K posunu popisu osy `y` dochází pouze tehdy, když je vzdálenost od osy větší než je maximálně možný prostor, který je zjištěn přepočtem mezi koncem plochy pro zadávání parametrů s názvem `Panel4` a počátkem grafu v horizontálním směru (obrázek 8).

```
% Vypocet prechodove charakteristiky včetně vykreslení
[y,t]=step(sys);
[m,n] = size(t);
tfinal=ceil(t(m,n)/10)*10;
[y,t]=step(sys,tfinal);
plot(t,y);
xlabel('t [s]','FontSize',9);
hy=ylabel('h(t)','FontSize',9);
posYLabel=get(hy,'Position');
posAxes=get(gca,'Position'); % pozice Axes v okně (default v Character)
posPanel=get(handles.Panel4,'Position');
novaPoziceX=-tfinal/posAxes(1,3)*(posAxes(1,1)-posPanel(1,1)-posPanel(1,3)-4);
novaPoziceY=1.1*(posYLabel(1,2)*2);
if posYLabel(1,1)<novaPoziceX
    set(hy,'Position',[novaPoziceX novaPoziceY posYLabel(1,3)])
end
```

Obrázek 8: Část m-souboru GUI aplikace pro posun popisu osy `Y`

5 Závěr

Grafická uživatelská aplikace podporuje oblast analýzy dynamických systémů, která je určena pro studenty druhého ročníku celofakultního studia. Jejím záměrem je seznámení studentů s touto oblastí a nabídnou jim kvalitní podpůrný prostředek pro získání potřebných znalostí.

Tvorba grafických uživatelských aplikací je záležitostí zamezení neočekávaných uživatelských zásahů v rámci jejich ovládání, ošetření chyb zadávání hodnot uživatelů a zobrazení hodnot v grafech, tak, aby dále uživatelé neměli rozporuplné otázky nad tím, co je vlastně zobrazenou křivkou myšleno, tj. vkládat grafy v rozumném měřítku, s úplným popisem os a názvem grafu.

V současné době GUI aplikace nenabízí vše, co by tvůrce aplikace uvítal, ale vývoj v této oblasti jde stále dál, takže posléze i řízené ovládání přepínačů v další verzi by mohlo být vyřešeno stejně, jako tomu bylo v případě nyní nabízené podpory pro vyhledávání souboru, což verze Release 11 (Matlab 5.3) ještě nenabízela.

Práce je zpracována za podpory Ministerstva školství, mládeže a tělovýchovy v rámci specifického výzkumu.

Literatura

- [1] Tůma, J, Wagnerová, R. Farana, R. Landryová, L. Základy automatizace. 1. vyd. Ostrava, VŠB-TU Ostrava, 280s. ISBN 978-80-248-1523-7.
- [2] Inc. The MathWorks. Matlab help, <http://www.mathworks.com/access/helpdesk/help/techdoc/matlab.html>. Internet, 1984–2008.

Jolana Škutová

Katedra automatizační techniky a řízení, Fakulta strojní, Vysoká škola Báňská - Technická univerzita Ostrava, 17. listopadu 15, 708 33 Ostrava-Poruba, tel. (+420) 59 732 4192, jolana.skutova@vsb.cz