

Studentská tvůrčí a odborná činnost
STOČ 2017

**PROGRAMOVÝ MODUL PRO LABORATORNÍ MODEL
VENTILÁTORU**

Petr WZATEK

Vysoká škola báňská – Technická univerzita Ostrava
Fakulta strojní
Katedra automatizační techniky a řízení

20. dubna 2017
FAI UTB ve Zlíně

Klíčová slova: MATLAB, sběr dat, přechodová charakteristika, identifikace, GUI aplikace

Anotace: Práce se zabývá tvorbou grafické uživatelské aplikace pro měření dat a identifikaci laboratorního modelu ventilátoru. První část práce se věnuje měření a zpracování dat laboratorního modelu s využitím programu MATLAB a knihovny funkcí Data Acquisition (DAQ) Toolbox ve tvaru přechodové charakteristiky. Dalším cílem práce bylo aplikovat několik aproximačních metod pro identifikaci laboratorního modelu ventilátoru, zhodnotit výsledky a přesnost experimentálních metod identifikace a zobrazit výsledné obrazové přenosy identifikovaného laboratorního modelu. V případě, že uživatel nemá dostupnou knihovnu funkcí DAQ, může využít grafickou aplikaci v režimu offline s tím, že budou dostupné sady souborů naměřených dat pro zkoumání identifikačních metod.

Obsah

1.	Úvod.....	4
2.	Laboratorní model ventilátoru	4
3.	Programový modul.....	5
3.1	Komunikace s hardwarem a nastavení kanálu	5
3.2	Získání dat a transformace signálu na otáčky	5
3.3	Filtrace signálu.....	7
3.4	Identifikace systému pomocí přechodové charakteristiky	8
4.	Grafická uživatelská aplikace	9
5.	Závěr	10
	Literatura	10

1. Úvod

S rozvojem informačních technologií se zvyšují nároky na efektivní a přesné měření a ovládání systému. Obor mechatroniky využívá všeobecné strojírenské, informační a elektrotechnické znalosti, aby navrhl mechatronické systémy, které jsou schopny získat co nejlepší výsledky měření a využívat informace o chování dynamiky systému pro dosažení plynulého řízení.

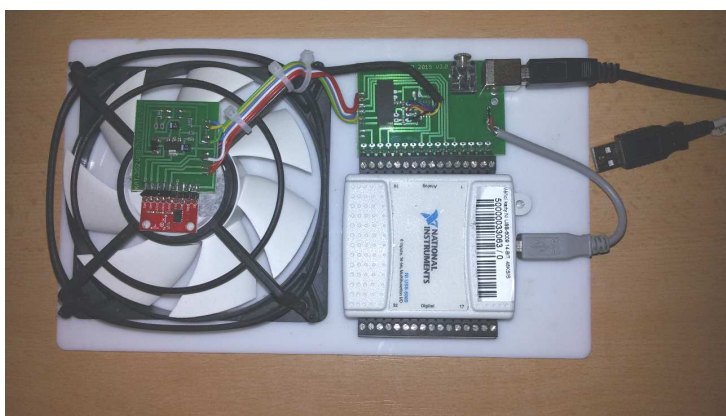
Cílem této práce je vytvořit základní programový modul pro komunikaci mezi počítačem a laboratorním modelem ventilátoru v prostředí MATLAB a nástroje Data Acquisition Toolbox pro komunikaci s měřicí kartou, která je součástí laboratorního modelu. Program bude schopen naměřit data z modelu ventilátoru, konkrétně velikost otáček za minutu, následně je upravit a určit přenos pomocí jedné z aproximačních metod identifikace systémů z přechodových charakteristik. Nakonec navrhnout a realizovat grafickou uživatelskou aplikaci pro ovládání laboratorního modelu, měření dat a jejich vyhodnocení za účelem identifikace laboratorního modelu. Program bude určený pro názornou výuku předmětů Katedry automatizační techniky a řízení.

Výsledkem této práce je tedy zajistit a prakticky testovat využití programu MATLAB pro komunikaci s měřicí kartou firmy National Instrument, funkcemi knihoven jako je Data Acquisition Toolbox pro získávání dat z reálného zařízení a identifikaci přenosu zařízení. Základy této aplikace budou určitě velkým přínosem pro další aplikace ovládání a řízení reálných systémů či laboratorních zařízení pro katedru nebo průmysl.

2. Laboratorní model ventilátoru

Řada modelů ventilátorů byla navržena a realizována pro účely měření a řízení otáček a analýzu vibrací ventilátoru s cílem využití jednotlivých modelů ve výuce předmětů na Katedře automatizační techniky a řízení. Pro tyto modely byla původně vytvořena aplikace FanControl ve vývojovém prostředí LabVIEW™.

Hlavním důvodem vytvoření nové aplikace v programu MATLAB® je možnost přímo využít naměřená data v programu pro identifikaci systému z přechodových charakteristik s využitím dostupných knihoven funkcí.



Obr.1: Laboratorní model ventilátoru

Laboratorní model ventilátoru se skládá z PC ventilátoru, měřicí karty NI USB-6009 od společnosti National Instruments™, snímače vibrací, optického snímače, integrovaného

obvodu a USB kabelu. Pro měření se využívá optický snímač pohybu nad lopatkami, který generuje napětí podle rozkladu světla. Výstup snímače je na vstupním analogovém kanálu „ai7“. Analogový kanál „ao0“ slouží pro nastavení napětí na vstupu ventilátoru.

3. Programový modul

Následuje tedy popis důležitých částí zdrojového kódu, které bylo nutné navrhnout, implementovat a testovat na reálném modelu ventilátoru.

3.1 Komunikace s hardwarem a nastavení kanálu

Měřicí karta laboratorního modelu ventilátoru se připojí k počítači pomocí USB kabelu a ke komunikaci mezi měřicí kartou a programem MATLAB se využívá knihovna funkcí Data Acquisition Toolbox. Pro nastavení komunikačního kanálu se vytvoří a definuje objekt "Session" (relace) pro nastavení vstupního a výstupního kanálu měřicí karty NI. Pro relaci se definuje typ a rychlost komunikace pomocí konkrétní vzorkovací frekvence. V případě měřicí karty NI USB-6009 byla využita nejvyšší možná frekvence vzorkování.

Při konfiguraci komunikační relace nastal problém při současné konfiguraci vstupního i výstupního kanálu pro jednu relaci. Poté nebylo možné použít příkazy pro synchronizované čtení dat z výstupu měřicí karty. Z důvodu získání dat přechodové charakteristiky v přesných synchronizovaných časových intervalech je vhodné použití daných příkazů pro získání dat. Problém byl vyřešen konfigurací dvou relací (obr. 2.), zvláště pro nastavení vstupního napětí na ventilátoru a zvláště pro relaci, která bude získávat data z výstupu ventilátoru, tj. snímání dat ze senzoru otáček. Konfigurace dvou relací má za následek zanedbatelné zpoždění měření dat, které neovlivní přesnost měření, neboť model ventilátoru je zatížen nízkou hodnotou dopravního zpoždění.

```
1      %% připojení zařízení
2 -    devices = daq.getDevices;
3 -    devices(1);
4      %% vytvoření session a nastavení kanálu pro měření
5 -    o = daq.createSession ('ni');
6 -    o.addAnalogOutputChannel ('Dev1', 'ao0', 'Voltage');
7 -    i = daq.createSession ('ni');
8 -    i.addAnalogInputChannel ('Dev1', 'ai6', 'Voltage');
9      %% parametry
10 -    i.Rate = 48000;
11 -    dobaZast=20; %%čas zastavení
12 -    dobaPust=20; %%čas měření
13 -    uZast=1; %%počáteční úroveň skoku
14 -    uPust=5; %%velikost akční veličiny
```

Obr.2: Připojení zařízení a vytvoření relace komunikace

3.2 Získání dat a transformace signálu na otáčky

Po nastavení komunikace je možné zahájit nastavení a získání dat z měřicí karty. Získání dat ve tvaru přechodové charakteristiky probíhá ve dvou fázích. První fáze je nastavení otáček ventilátoru do požadovaného počátečního stavu jeho otáček a druhá fáze pro získání odezvy modelu ventilátoru ve tvaru přechodové charakteristiky při skokové změně napětí na vstupu ventilátoru.

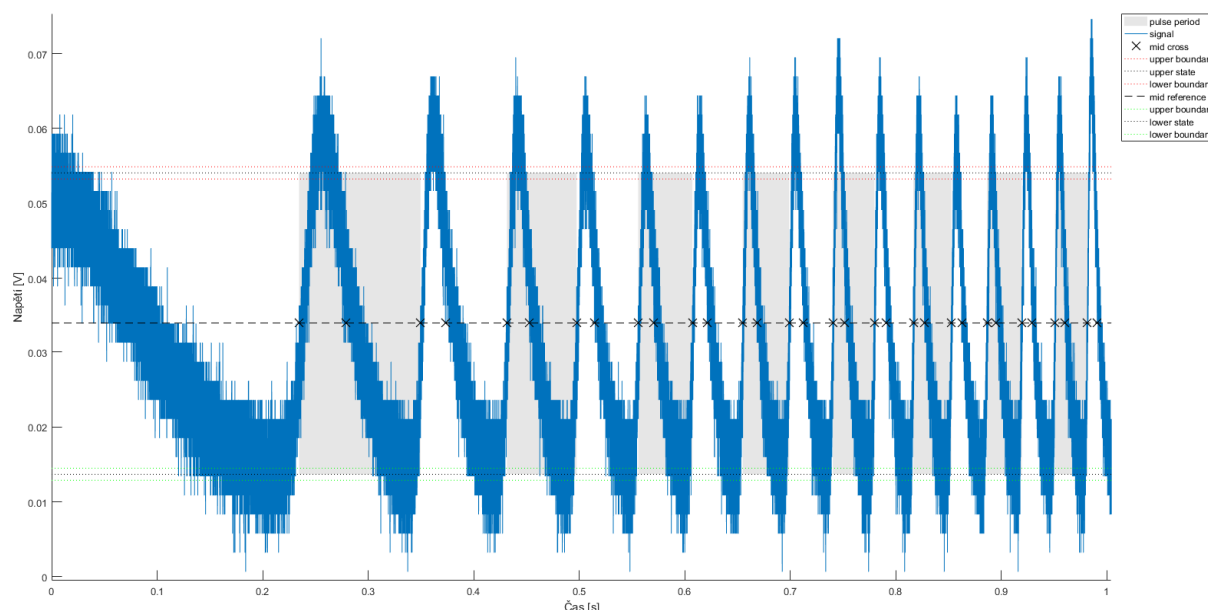
```

17 %% zastaveni ventilatoru (1.cast mereni prechodove charakteristiky)
18 - prepare(o); %připrava
19 - prepare(i);
20 - i.DurationInSeconds = dobaZast; %nastavení času operace
21 - o.outputSingleScan(uZast); %změna hodnoty
22 - [dZ,tZ]=i.startForeground; %zahájení zastavení
23 %% spusteni ventilatoru (2.cast mereni prechodove charakteristiky)
24 - i.DurationInSeconds = dobaPust; %nastavení času druhé operace
25 - o.outputSingleScan(uPust); %změna hodnoty
26 - [dP,tP]=i.startForeground; %zahájení měření
27
28 %% výpočet šířky period
29 - [ppZ,icZ,icZM,icZN]=pulseperiod(dZ, tZ), 'Polarity', 'positive', 'Tolerance', '10';
30 - [ppP,icP,icPM,icPN]=pulseperiod(dP, tP), 'Polarity', 'positive', 'Tolerance', '10';

```

Obr.3: Zahájení měření a výpočet šířky

Na obr. 4 je zobrazena část grafu naměřených hodnot napětí snímače v závislosti na čase ve tvaru diskrétně modulovaného signálu. Otáčky se stanoví pomocí funkce *pulseperiod*, která měří délku period signálu procházející středovou čarou. Podmínkou je rovněž, aby signál dosáhl nastavené horní a dolní úrovně signálu. Funkce také informuje o umístění jednotlivých period.



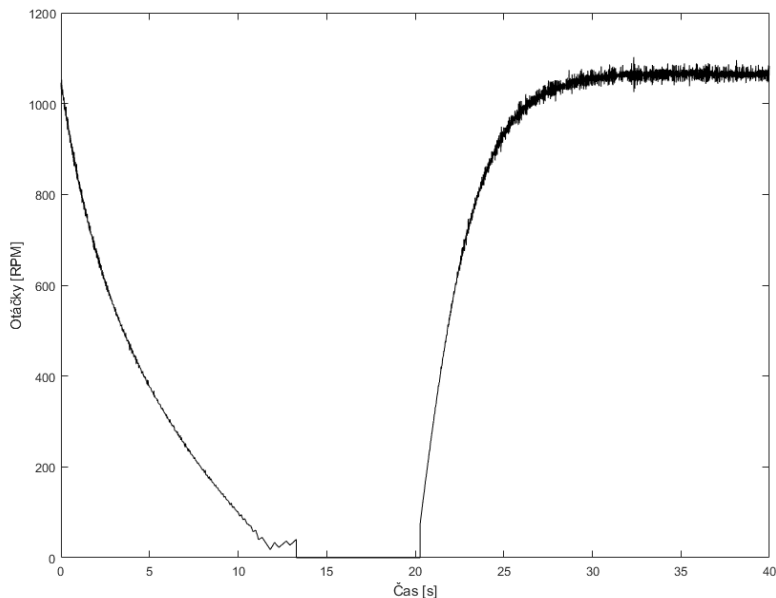
Obr.4: Část naměřených hodnot a využití měření period

Jednotlivé periody odpovídají rychlosti otáček od jedné lopatky ventilátoru ke druhé. Čím rychleji se ventilátor pohybuje, tím užší jsou periody. Z toho vyplývá, že při vyšších rychlostech je možné získat více naměřených hodnot za sekundu než na začátku. Což se dá považovat za chybu měření, protože vzorky nejsou ekvidistantní a pro přechodovou charakteristiku je požadováno co nejvíce hodnot hlavně při rozběhu otáček ventilátoru pro proces identifikace systému. Vzhledem k tomu že na ventilátoru je 7 lopatek, velikost otáček je dána vztahem

$$f = \frac{60}{T} \cdot \frac{1}{7} [RPM]. \quad (1)$$

Nízké otáčky modelu ventilátoru nejsou měřitelné z důvodu nevhodného diskrétně modulovaného signálu, jehož tvar není pro funkci *pulseperiod* čitelný. Nízké otáčky

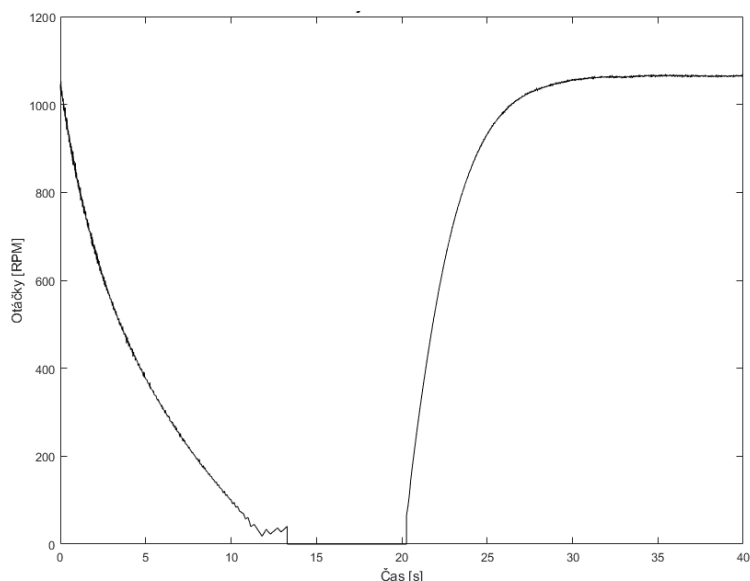
ventilátoru také nejsou rovnoměrné a považujeme je za část, která je mimo pracovní oblast použití ventilátoru. Nízké hodnoty jsou tedy ve zdrojovém kódu striktně nahrazeny nulovou hodnotou otáček. Významné hodnoty pro měření, identifikaci a jiné použití jsou nad úrovní zhruba 50 otáček za minutu.



Obr.5: Vypočtené otáčky ventilátoru z naměřených dat ve tvaru PWM signálu

3.3 Filtrace signálu

Naměřené otáčky se v ustáleném čase nabývají nerovnoměrných hodnot. Pro identifikaci je vhodné tento šum vyfiltrovat. Druhá část grafu (odezva ve tvaru přechodové charakteristiky) je filtrována pomocí příkazu *filter*, což je proměřovací pohybující se filtr. Byl aplikován filtr o velikosti 7 vzorků, protože na ventilátoru je 7 lopatek a dosahují tak konstantních hodnot. Vyfiltrované hodnoty je nutné poté posunout v čase, protože nenáleží svému původnímu umístění. Je možné využít pro filtraci i násobek 7 vzorků, například 14 vzorků, pro jemnější průběh, ale to se nepříznivě projevilo na přechodu v části grafu, kdy dochází ke změně vstupního skokového signálu.



Obr.6: Otáčky ventilátoru upravené filtrací signálu

3.4 Identifikace systému pomocí přechodové charakteristiky

Identifikace je proces zjištění matematického modelu systému. Pro identifikaci pomocí aproximace přechodových charakteristik využijí některé z experimentálních deterministických metod. Konkrétně tedy metod aproximace:

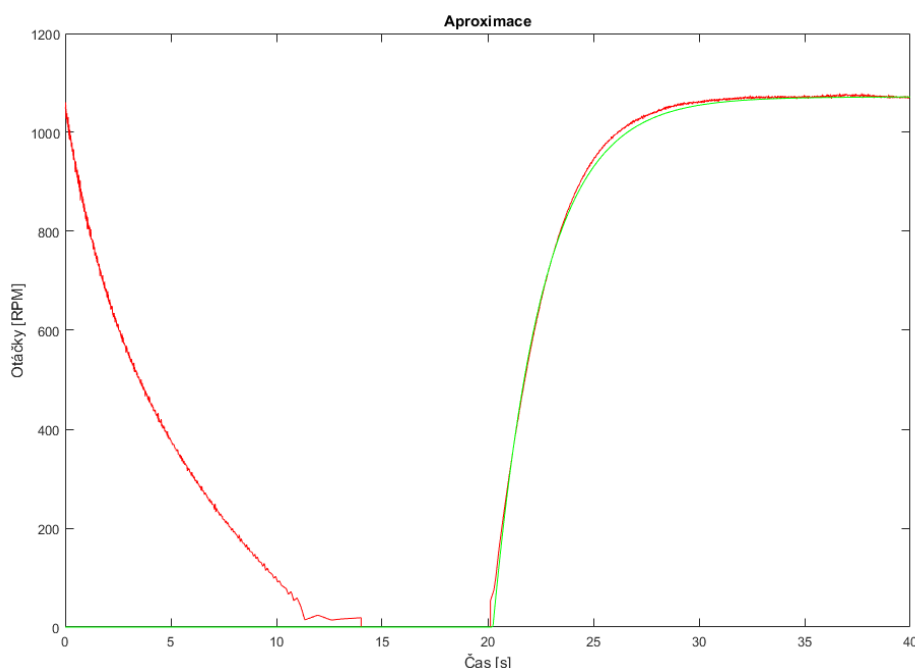
- přechodových charakteristik s aperiodickým průběhem – Strejcova metoda,
- proporcionální soustavou se setrvačností 1.řádu,
- pomocí tečny v inflexním bodě,
- pomocí sečny v inflexním bodě,
- dvoubodovou metodou – varianta 1,
- dvoubodovou metodou – varianta 2,
- třibodovou metodou.

Pro názornost návrhu aproximačních metod je uveden dále příklad aproximace pomocí dvoubodové metody (varianta 1). Každá aproximace má svůj algoritmus výpočtu, tedy zdrojový kód implementovaný v programu MATLAB. Cílem metod je tedy stanovení konkrétních neznámých parametrů předem stanoveného obecného tvaru matematického modelu. Většina metod vyžaduje určení ustálené hodnoty pro výpočet zesílení systému.

Výsledkem je obrazový přenos modelu ventilátoru, který lze porovnat s původní naměřenou charakteristikou.

```
1      %Aproximace dvoubodovou metodou - varianta č. 1
2 -    hUst=h(end); % ustálená hodnota
3 -    ku=5-1 %velikost akčního skoku
4 -    kz=hUst/ku % zesileni
5
6 -    vzorek=find(h<=0.7*hUst,1,'last'); % nalezení hodnoty odpovídající 70% * K
7 -    dh=h(vzorek+1,1)-h(vzorek,1);
8 -    dh70=0.70*hUst-h(vzorek,1);
9 -    dt=t(vzorek+1,1)-t(vzorek,1);
10 -    dt70=dh70*dt/dh;
11 -    t70=dt70+t(vzorek,1)
12 -    vzorek=find(h<=0.33*hUst,1,'last'); % nalezení hodnoty odpovídající 33% * K
13 -    dh=h(vzorek+1,1)-h(vzorek,1);
14 -    dh33=0.33*hUst-h(vzorek,1);
15 -    dt=t(vzorek+1,1)-t(vzorek,1);
16 -    dt33=dh33*dt/dh;
17 -    t33=dt33+t(vzorek,1)
18
19 -    T5=1.245*(t70-t33); %časová konstanta
20 -    TdAprox5=1.498*t33-0.498*t70; %dopravní zpoždění
21 -    tt=0:0.05:(dobaPust+dobaZast); %vzorkování
22 -    Gaprox5=tf(kz,[T5 1],'InputDelay',TdAprox5); %přenos aproximace
23 -    [hAprox5,tAprox5]=step(Gaprox5,tt);
24 -    hA5=hAprox5*ku;
25
26 -    plot(t,h,'r',tt,hA5,'g'); %zobrazení grafu
27 -    xlabel('Čas [s]');
28 -    ylabel('Otáčky [RPM]');
```

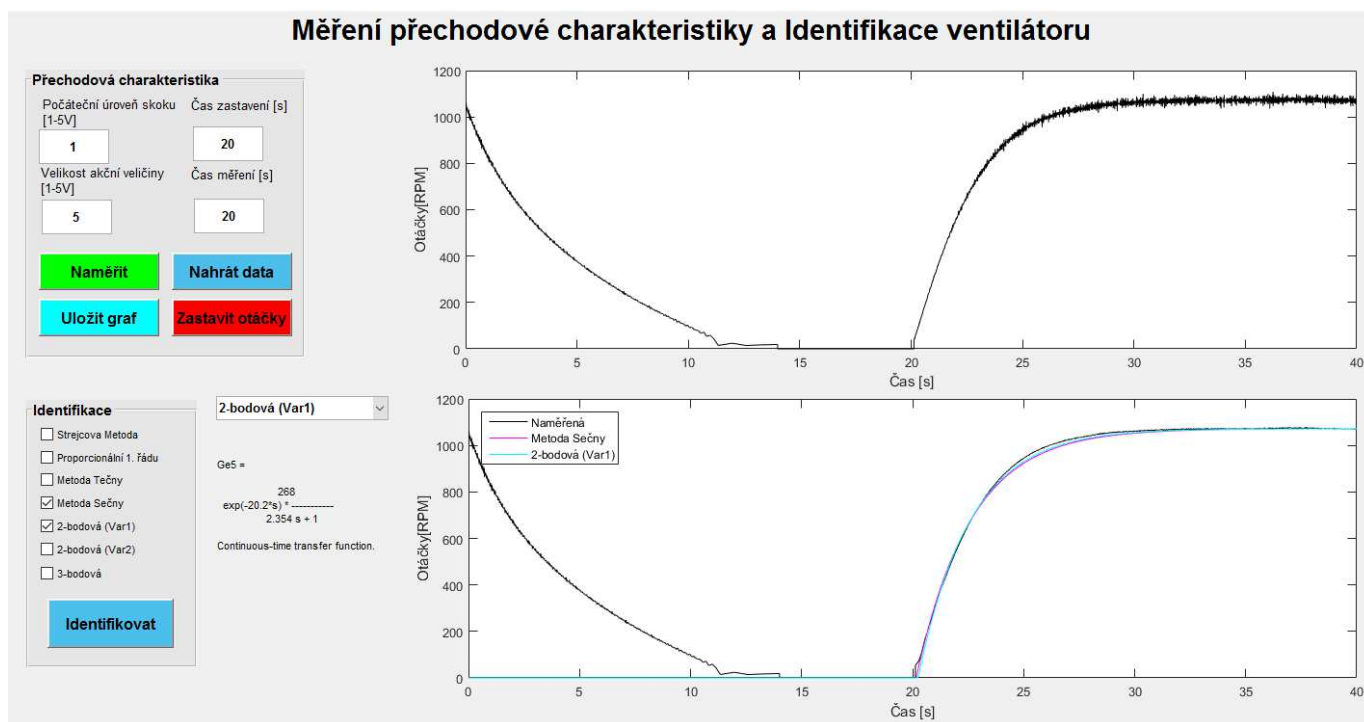
Obr. 7: Aproximace dvoubodovou metodou (var 1) – program



Obr.8: Aproximace dvoubodovou metodou (var 1) – porovnání s původní charakteristikou

4. Grafická uživatelská aplikace

Pro koncové uživatele bude k dispozici navržená grafická uživatelská aplikace, která byla vytvořena v grafickém prostředí programu MATLAB. Části zdrojových kódů, které vznikly v průběhu celé práce jsou zahrnuté v programovém souboru grafické aplikace v dílčích částech provázaných na funkce tlačítek nebo zaškrtnutých polí.



Obr.9: Grafická uživatelská aplikace pro měření dat a identifikaci modelu ventilátoru

V levé části uživatel nastavuje parametry pro měření přechodové charakteristiky. V případě, že nemá dostupné všechny funkce pro měření dat, tak může načíst data s cílem

testování aproximačních metod. V pravé části jsou zobrazovány grafické výstupy otáček modelu ventilátoru a v dolní pravé části pak průběhy matematických modelů získaných uživatelem vybraných aproximačních metod. Protože grafická uživatelská aplikace nenabízí nástroj pro zobrazení matematických vzorců a obrazové přenosy, tak je v grafické části uprostřed zobrazen obrazový přenos alespoň v podobě známé z prostředí MATLAB, tj. proměnná ve tvaru obrazového přenosu. Proto vzhledu matematických vzorců nyní není dána taková důležitost, ale později bude posléze využit nástroj, který bude implementován na úrovni prostředí MATLAB.

5. Závěr

V bakalářské práci byla realizována a popsána komunikace mezi prostředím MATLAB a laboratorním modelem ventilátoru, která může být i nadále pro tento model rozšířena do dalších úloh a potřeb výuky. Byl kladen důraz na přesnost získání dat a sestavení přechodové charakteristiky převodem pulsně šířkově modulovaného signálu ze senzoru modelu ventilátoru tak, aby průběh odpovídal skutečným otáčkám na modelu. Byla navržena grafická uživatelská aplikace pro potřeby výuky.

Metody identifikace vyžadují velkou přesnost měření. Naměřená přechodová charakteristika je nepřesná hlavně v dopravním zpoždění a v menším množství bodů na začátku náběhu otáček. To se nepříjemně projevuje v určení inflexního bodu u některých z aproximačních metod, za předpokladu že se tam vůbec nachází, a zkreslení výsledků některých metod, např. metody tečny.

Je možné říct, že omezení z pohledu přesnosti měření jsou dány hardwarově a práce byla zpracována s co největším důrazem na přesnost a věrohodnost naměřených dat dle možností, které jsou dány skutečným laboratorním modelem, jeho strukturou, snímači a danou měřicí kartou. Za předpokladu využití výkonnější měřicí karty by bylo možné získat přesnější odezvu na skok vstupního napětí. Rovněž přesnější a výkonnější snímač otáček by zajistil měření většího množství vzorků při velkých změnách rychlosti, nebo by mohl získat ekvidistantní vzorky při měření.

V každém případě lze závěrem konstatovat, že všechny získané zkušenosti a poznatky lze použít i na jiné modely a zařízení. Také vlastní identifikační část může být aplikována na jiné úlohy identifikace či účely jako samostatný celek zdrojových kódů implementujících aproximační metody pro získání matematických modelů neznámých reálných systémů.

Literatura

- [1] Data Acquisition Toolbox User's Guide. © 1994-2016 The MathWorks, Inc.. [online]. [cit. 2016-11-19]. Dostupné z: https://uk.mathworks.com/help/pdf_doc/daq/daqug.pdf
- [2] Signal processing Toolbox User's Guide. © 1988-2016 The MathWorks, Inc.. [online]. [cit. 2016-12-27]. Dostupné z: https://uk.mathworks.com/help/pdf_doc/signal/signal_tb.pdf
- [3] Noskievič, P. 1999. Modelování a identifikace systémů. Ostrava: Montanex, 1999. ISBN 80-7225-030-2.
- [4] Zaplatílek, K & Doňar, B. 2004. Matlab: tvorba uživatelských aplikací. 1. vyd. Praha – BEN – technická literatura, 2004- ISBN 80-7300-133-0.
- [5] Data Acquisition Toolbox Documentation. © 1994-2017 The MathWorks, Inc.. [online]. [cit. 2017-12-22]. Dostupné z: <https://uk.mathworks.com/help/daq/>