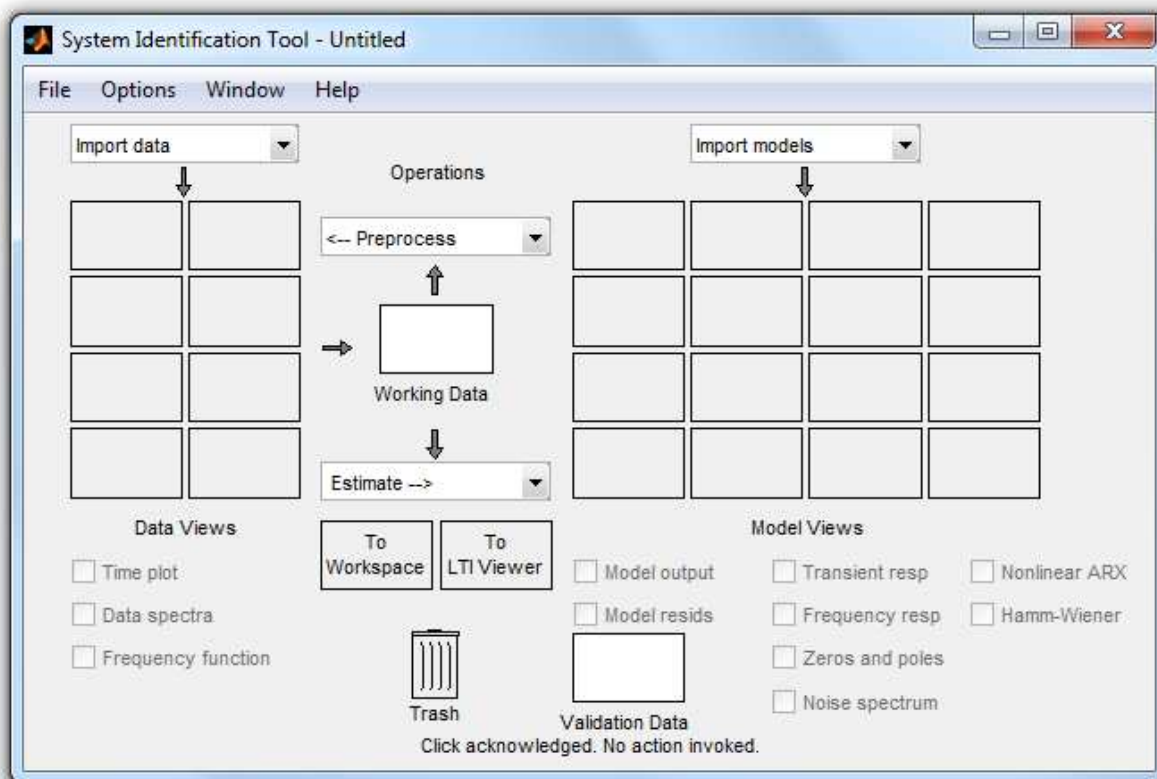


Obrázok 12: Kľúčové bloky SIT v prostredí Simulink

### 6.3 Práca v grafickom rozhraní

Veľmi zaujímavý prístup k identifikácii ponúka GUI - grafické rozhranie SIT. Spúšťa sa príkazom `ident` a ľahko sa ovláda. Umožňuje rýchlu prácu s dátami, ich úpravami, delením, normovaním. Okrem toho je tu veľa iných výhod ako grafy, rôzne modely a vysoká prehľadnosť. Taktiež je pre prácu s GUI vytvorené demo. Jedinou nevýhodou je, že nejaký čas trvá, kým sa ho používateľ naučí používať.

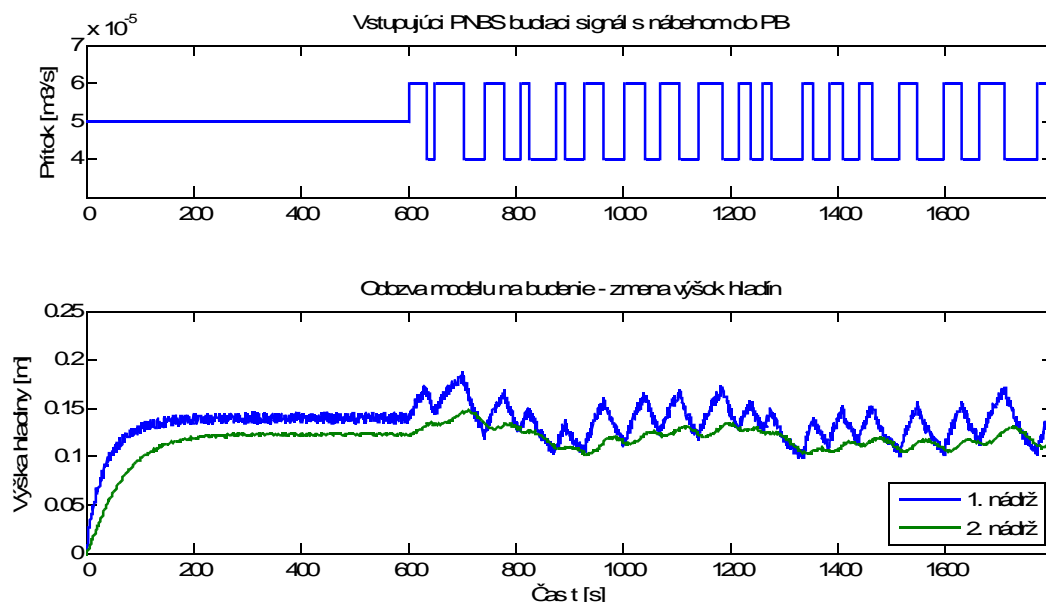


Obrázok 13: GUI pre experimentálnu identifikáciu

## PRAKTICKÁ ČASŤ

### 1 Ako máme reprezentované dáta

Predpokladáme, že sme získali nasledujúce dáta:



Obrázok 14: Dáta pre ktoré vykonávame experimentálnu identifikáciu

Dáta sú uložené v súbore *Data.mat* a sú vo workspace reprezentované nasledovne:

Premenná	Popis	Veľkosť
t	Časový vektor simulácie	<1800x1 double>
Uin	Vektor prítoku do prvej nádrže	<1800x1 double>
H1	Vektor zašumených dát hladiny prvej nádrže	<1800x1 double>
H2	Vektor zašumených dát hladiny druhej nádrže	<1800x1 double>
Tvz	Periódna vzorkovania	1

Tabuľka 2: Premenné v Matlab-e dostupné pre experimentálnu identifikáciu

Príkazy v Matlab-e:

```
load data.mat
```

Naším cieľom je vykonať experimentálnu identifikáciu tromi prístupmi SIT. Skôr ako začneme identifikovať, vytvoríme si IDDATA objekty pre tieto dáta. Tento krok nie je nutný avšak následne nám uľahčí prácu a reprezentáciu dát.

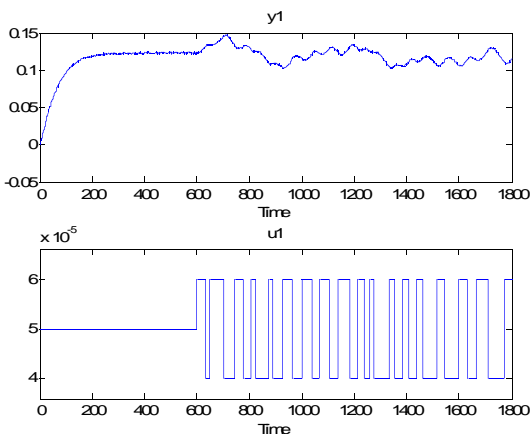
## 2 Vytvorenie IDDATA objektu

*IDDATA* objekt je unifikovaná dátová štruktúra pre reprezentáciu dát pre experimentálnu identifikáciu. Inými slovami obsahuje jak vektor budiacich dát, tak vektor odozvy. Navyše v sebe zahŕňa čas - periódu vzorkovania a mnoho iných dodatočných informácií. Príklad použitia pre

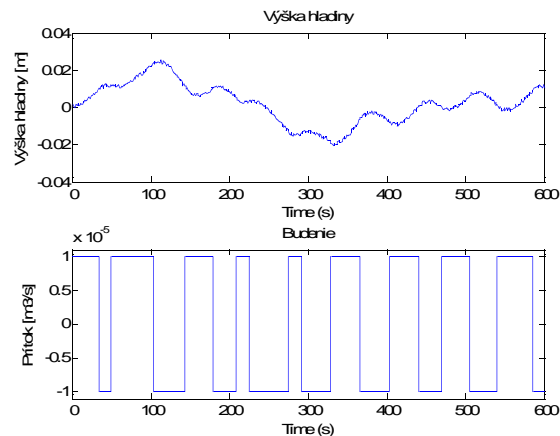
Príklady v Matlab-e:

```
ExpData = iddata(H2,Uin,Tvz);
idplot(ExpData)
```

Okrem toho, že funkcie SIT môžu používať túto dátovú štruktúru, je možné tieto dáta zobraziť v grafe, ktorý obsahuje dodatočné informácie. Pre túto úlohu je príkaz *idplot(iddata objekt)* a jeho argumentom je *IDDATA* objekt.



Obrázok 15: IDPLOT bez doplnkových informácií



Obrázok 16: IDPLOT s doplnkovými informáciami

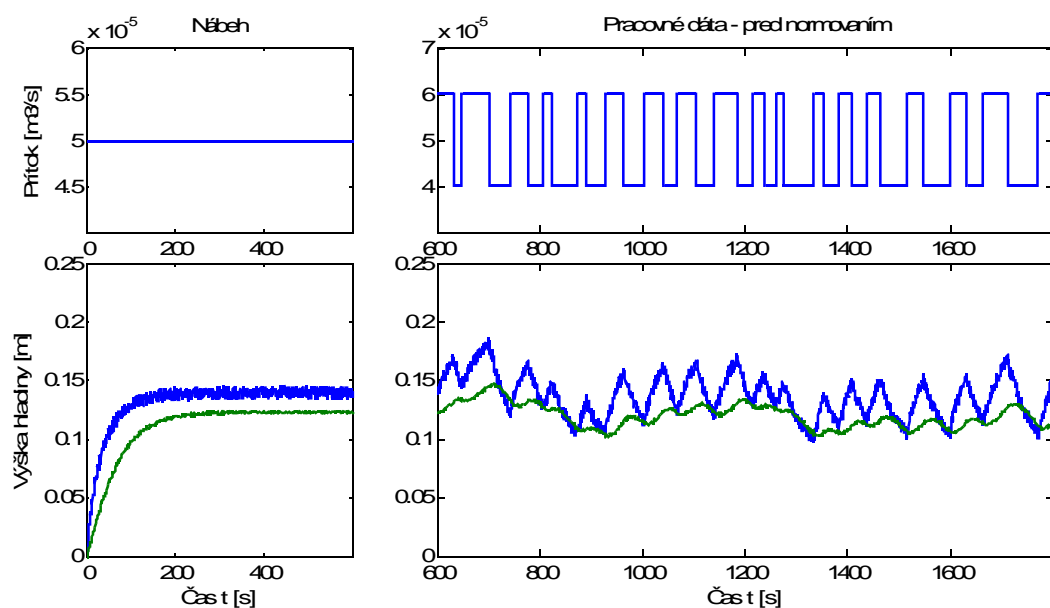
## Úlohy

1. Vytvorte *IDDATA* objekt *ExpData* pre hladinu H2 a to rozdelením dát na 600 vzorkové vektory.
2. Zobrazte *IDPLOT* pre *ExpData*.
3. Pomocou príkazu *set* nastavte parametre: meno dátového súboru, názov a jednotky vstupu, výstupu, časovú jednotku a názov experimentu. (Pre zistenie presného označenia vlastností použite príkaz *get(ExpData)*).
4. Znova zobrazte *IDPLOT*.
5. Použite príkaz *present(ExpData)*.

### 3 Práca v Matlab-e

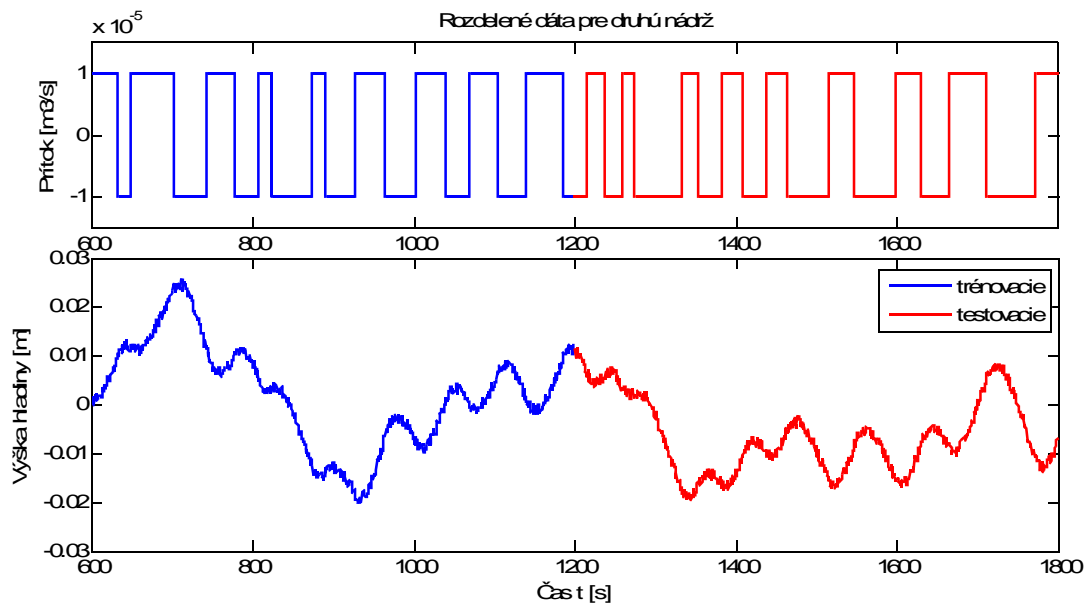
#### 3.1 Predspracovanie dát

Ako prvé potrebujeme odstrániť nábeh, pretože ho pre identifikáciu netreba. Avšak vieme, že nábeh končí 601 vzorkou vrátane, preto si zapamätáme hodnotu budenia  $U_{norm}$  a výšky hladiny  $H2_{norm}$ . Tieto hodnoty použijeme pri normovaní dát. Vektory  $U_{in}$  a  $H2$  orežeme o 601 vzoriek od začiatku. Využijeme indexovanie, pretože  $U_{in}$  a  $H2$  sú vektory. Nezabudnúť orezať aj časový vektor. Výsledkom má byť bude



Obrázok 17: Odstránenie nábehu

Pracovné dáta budeme normovať tak, že od každej hodnoty  $U_{in}$  odpočítame  $U_{ust}$ . Obdobne pre  $H2$ . Po normovaní nasleduje delenie dát. Delíme ich na tréningové - TrenData a validačné TestData. Ak sme postupovali správne, pre identifikáciu máme pripravené takto zobrazené dáta.



Obrázok 19: Správne predspracované dáta

## Úlohy

1. Odstráňte nábeh a porovnajzte výsledok príkazu `mean(Uin)` s `Qust`, zdôvodníte výsledok.
2. Vytvorte `TrenData` a `TestData` aj ako `IDDATA` objekty.
3. Zobrazte ich vo spoločnom grafe.

### 3.2 Určenie modelov ARX a ARMAX

Keďže predpokladáme model 2. rádu v diskretnom prenose v tvare:

$$F_M(z^{-1}) = \frac{\Delta H_2(z^{-1})}{\Delta U_{bud}(z^{-1})} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \quad (4)$$

Na základe tejto vedomosti vieme zvoliť parametre funkcií ARX a ARMAX. Funkcie požadujú ako prvý parameter dáta a druhý je vektor rádov. Ako dáta použijeme dvojicu [výstup vstup], konkrétne [H2tren Utren]. Alternatívne je možné použiť `IDDATA` objekt `TrenData`.

Rády pre ARX volíme [na nb nk], kde na je rád menovateľa, nb je rád čitateľa a nk je rád oneskorenia. Volíme [2 2 1]. Pre armax je [na nb nc nk], nc je rád filtra. Volíme [2 2 2 1]. Funkcie nám vracajú `theta(idpoly)` model. My hlavne potrebujeme iba vektory čitateľa *B* a menovateľa *A*. Vytvoríme diskretné prenosi `sysdarx` a `sysdarm`.

## Príkazy v Matlab-e:

```

% ARX

th1 = arx([H2tren Utren],[2 2 1]); % Určenie Theta modelu cez ARX
sysdarx = tf(th1.B,th1.A,Tvz,... % Vytvorenie diskretného prenosu
    'Variable','z^-1'); % pre Tvz = 1

disp('Aproximácia systému ARX modelom:'); sysdarx

% ARMAX

th2 = armax([H2tren Utren],[2 2 2 1]); % Určenie Theta modelu cez ARMAX
sysdarm = tf(th2.B,th2.A,Tvz,... % Vytvorenie diskretného prenosu
    'Variable','z^-1'); % pre Tvz = 1

disp('Aproximácia systému ARMAX modelom:'); sysdarm

```

Pre ARX aproximáciu dostávame nestabilný systém:

$$F_{ARX}(z^{-1}) = \frac{\Delta H_2(z^{-1})}{\Delta U_{bud}(z^{-1})} = \frac{-21.67 z^{-1} + 43.7 z^{-2}}{1 - 0.6591 z^{-1} - 0.3409 z^{-2}} \quad (5)$$

Pre ARMAX aproximáciu dostávame stabilný systém:

$$F_{ARMAX}(z^{-1}) = \frac{\Delta H_2(z^{-1})}{\Delta U_{bud}(z^{-1})} = \frac{1.229 z^{-1} + 1.635 z^{-2}}{1 - 1.952 z^{-1} + 0.9521 z^{-2}} \quad (6)$$

## Úlohy

1. Vypočítajte ARX a ARMAX aproximáciu.
2. Vykonajte jednotkový skok pre ARMAX model.
3. Preved'te ARMAX model do časovej oblasti príkazom d2c.

### 3.3 Validácia ARMAX modelu

Cieľom validácie ARMAX je porovnávať odozvu aproximovaného modelu voči pôvodne meranému modelu a musíme si odsimulovať pôsobenie rovnakého budenia. Pre tento účel je pripravená funkcia IDSIM, odvodená od funkcie SIM. Argumentami sú budiace napätie a *theta* model.

## Príkazy v Matlab-e:

```

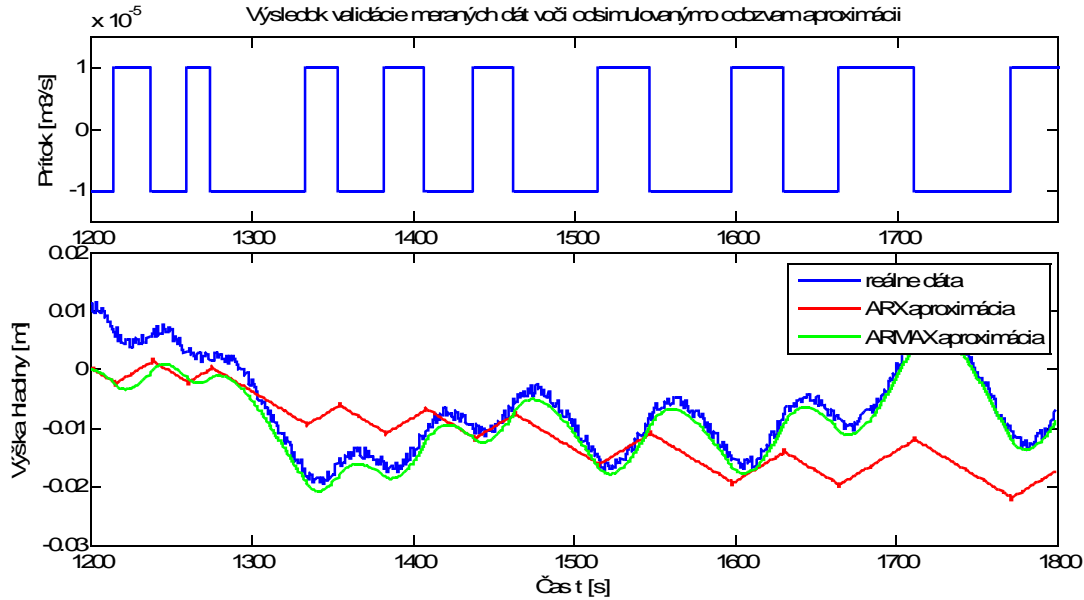
H2armax = idsim(U_test,th_Armax);
stairs(,H2-H2armax,'linewidth',2);

```

Alebo pre IDMODEL:

Príklady v Matlab-e:

```
H2armax = idsim(TestData.u,th_Armax);
stairs(t,H2-H2armax,'linewidth',2);
```



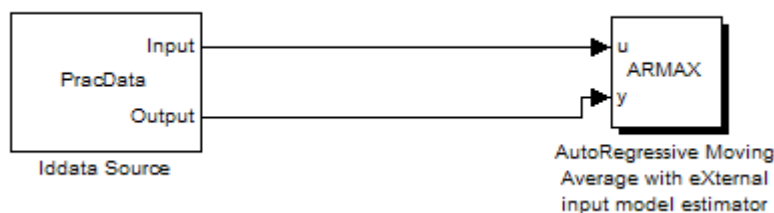
Obrázok 20: Výsledná validácia modelov ARX a ARMAX

### Úlohy

1. Zobrazte priebeh  $H2val$  a  $H2armax$  v jednom grafe.
2. Zobrazte priebeh odchýlky.
3. Skúste to isté urobiť pre ARX model.
4. Vyhodnotte výsledky identifikácie

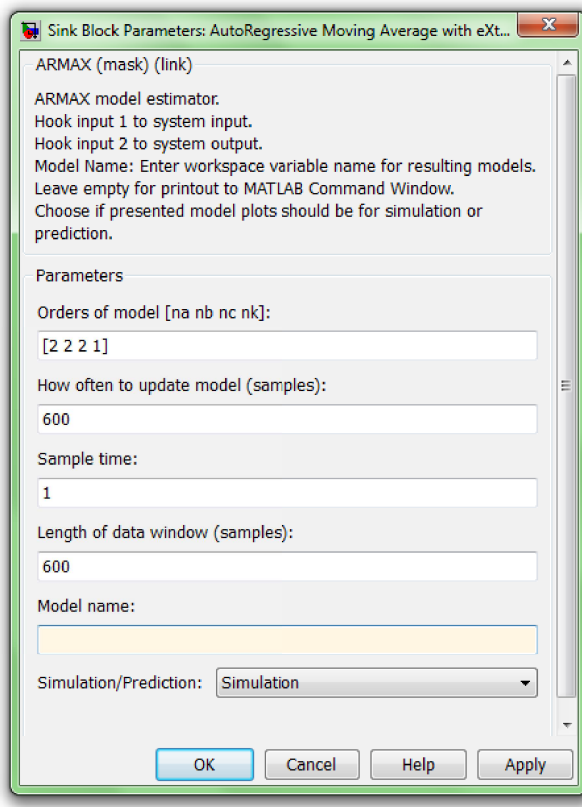
## 4 Práca v prostredí Simulink

Predpokladáme, že máme vytvorený *IDDATA* objekt *PracData*, kde sú normované pracovné dáta v rozsahu 1200 vzoriek. V Simulink-u si vytvoríme schému:



Obrázok 21: Schéma pre identifikáciu v Simulink-u

Nastavíme *solver* ako *discrete*, periódu vzorkovania *Tvz* a *Tstart* a *Tfin*. Nezabudnúť na *fixed step*. Otvoríme masku ARMAX a nastavujeme:

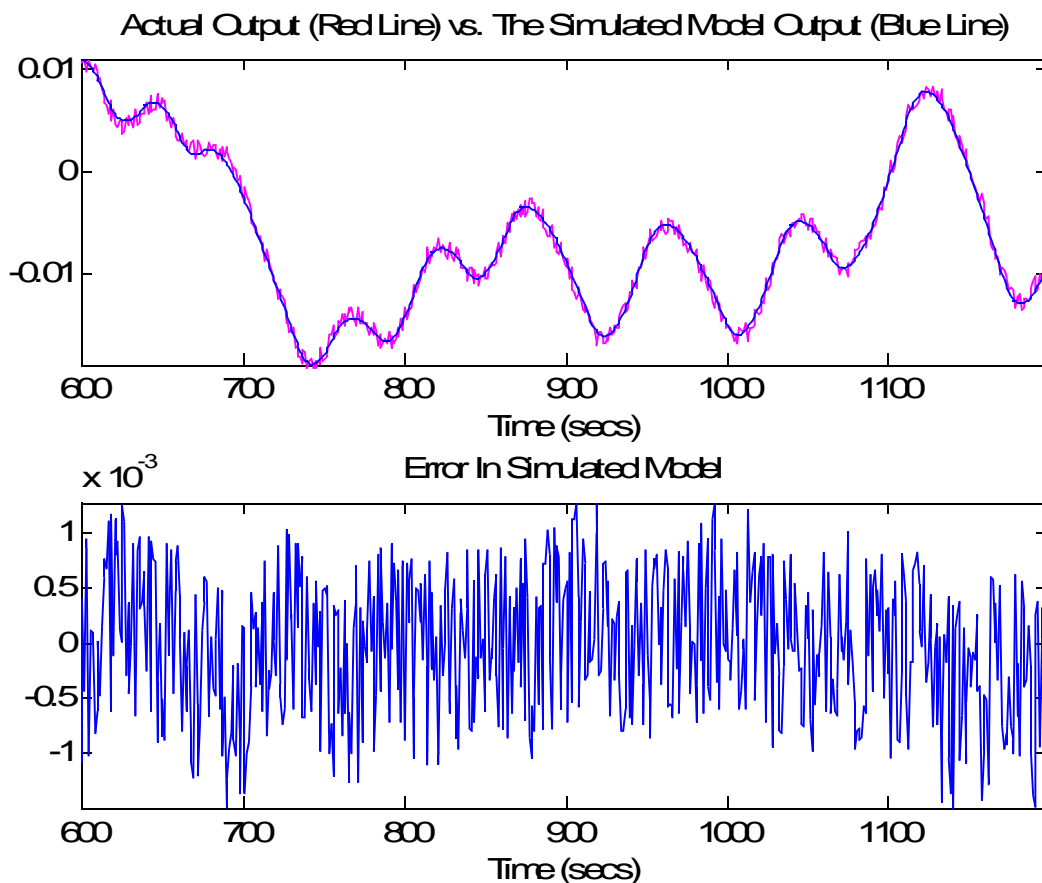


Obrázok 22: Možnosti bloku ARMAX

V Orders sú zvolené rády, model aktualizujeme iba raz - po 600 vzorkách, Vzorkujeme po sekundách a zobrazujeme posledných 1200 vzoriek. Takto využijeme celý súbor dát pre určenie modelu. Ak Model name necháme voľné, po spustení sa nám do príkazového riadku vypíše výsledok, respektíve priebežné výsledky ak budeme častejšie aktualizovať. Súčasne sa nám otvorí graf kde hore je v grafe odozva laboratórneho modelu spolu s aproximovaným. Dole je priebeh odchýlky.

Simulink počíta ARMAX model inak - aktualizuje ho na základe súboru dát - v našom prípade najprv zo 600 hodnôt tréningových dát vypočíta aproximáciu a testovacie dáta, s ktorými aproximáciu porovnáva použije zároveň aj na spresnenie samotného výsledku.





Obrázok 23: Výstup bloku ARMAX pre posledných 600 vzoriek

### Výstup v Matlab-e:

Transfer function:

num/den =

$$\frac{1.0111 z + 1.7815}{z^2 - 1.9518 z + 0.95246}$$

Noise model:

num/den =

$$\frac{z^2 - 1.8844 z + 0.88451}{z^2 - 1.9518 z + 0.95246}$$

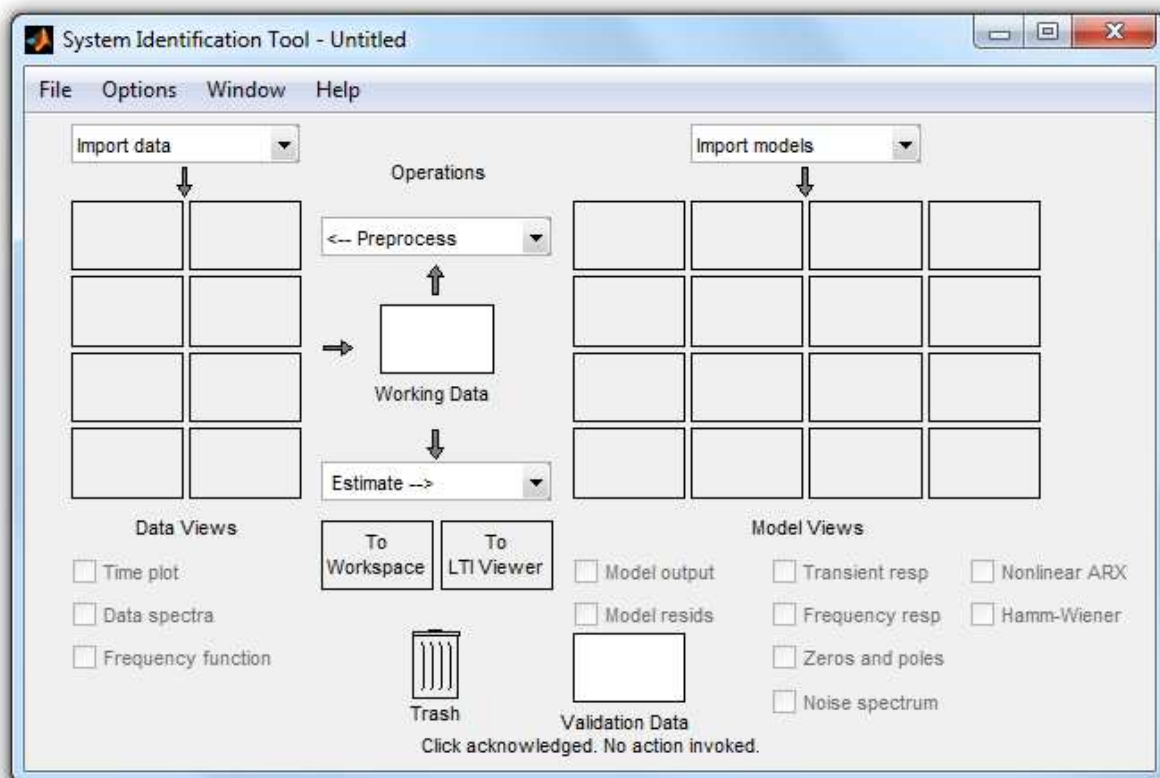
Sampling time: 1

## Úlohy

1. Pomenujte model a zobrazte ho v Matlab-e ako diskretný prenos.
2. Urobte identifikáciu pre ARX

## 5 Práca s grafickým rozhraním GUI

Máme namerané dáta pred predspracovaním a to vo vektoroch a IDDATA objekte *ExpData*. Príkazom *ident* otvoríme grafické rozhranie GUI.

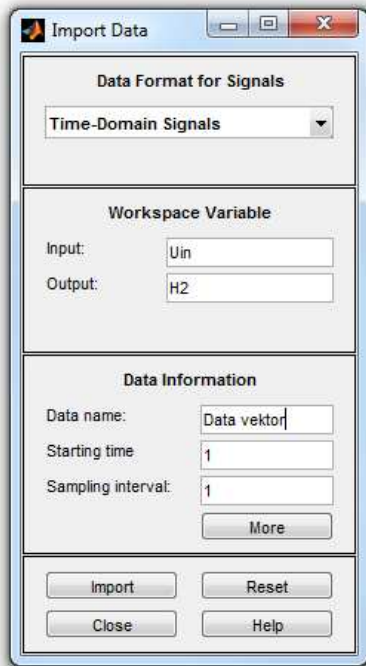


Obrázok 24: GUI bez dát

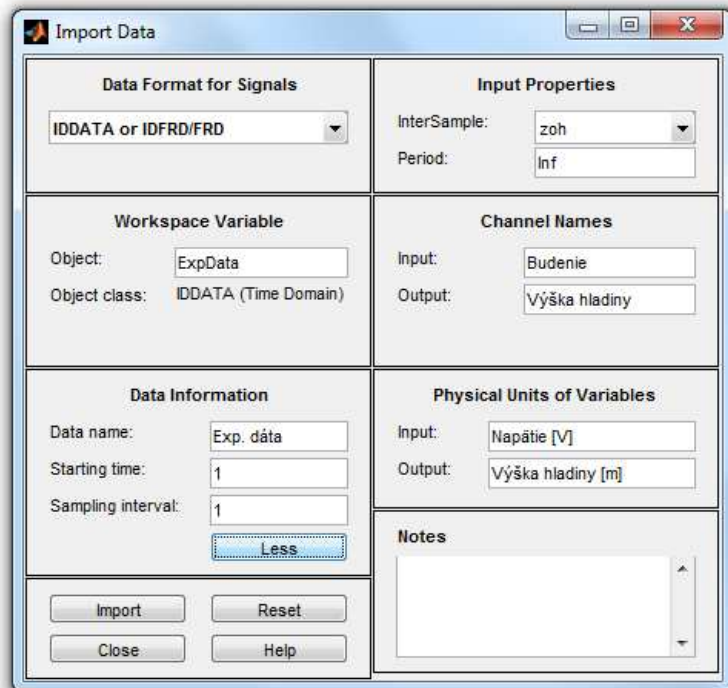
### 5.1 Import dát do GUI

Cez položku Import data/Time domain data... otvoríme okno kde môžeme vybrať zdroj dát. Prednastavený je Time domain signals - teda jednotlivé vektory. Môžeme zadať Input ako Uin a Output ako H2. Data name bude Data Vektor.

Ak prepne na voľbu IDDATA or IDFRD/FRD, môžeme importovať dáta ako ExpData, pričom sa importujú aj podrobnosti o dátovom súbore. Data name sa preberie z ExpData. Prístup cez tlačidlo More. Import sa potvrdzuje tlačidlom Import. Po dokončení zavrieme tlačidlom Close. Naše dáta by sa mali zobrazit' v hlavnom okne vľavo.

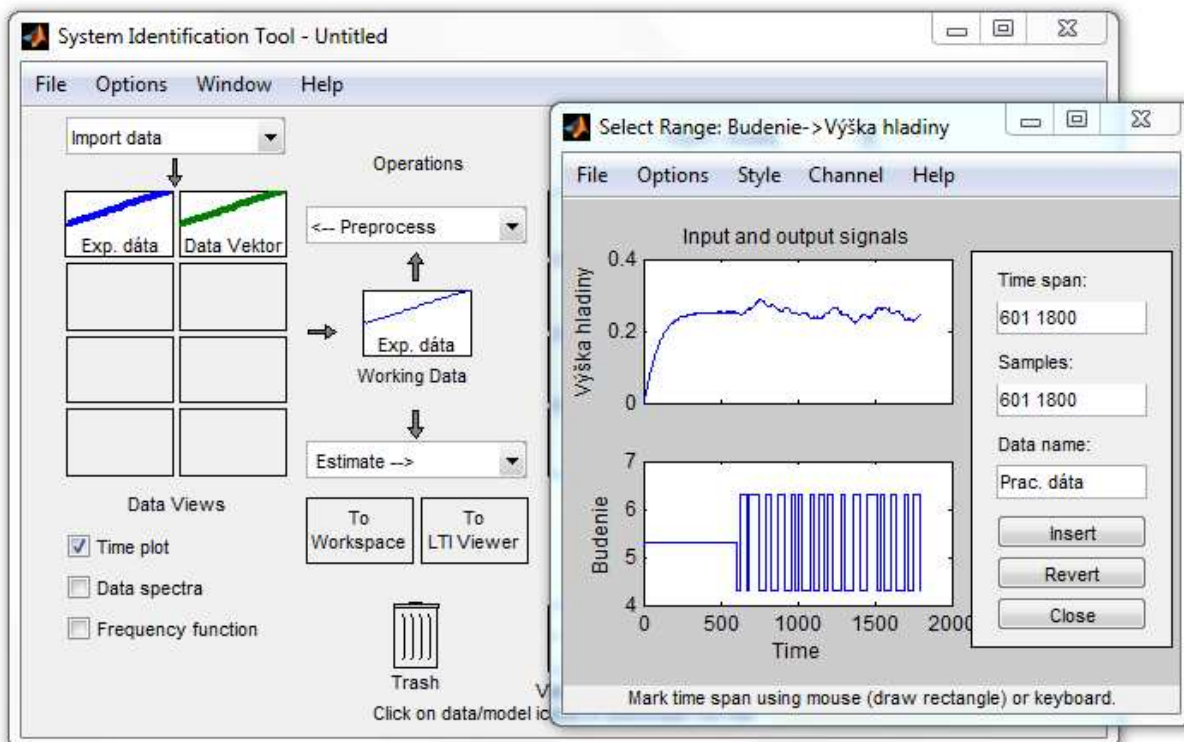


Obrázok 25: Import vektory



Obrázok 26: Import IDDATA

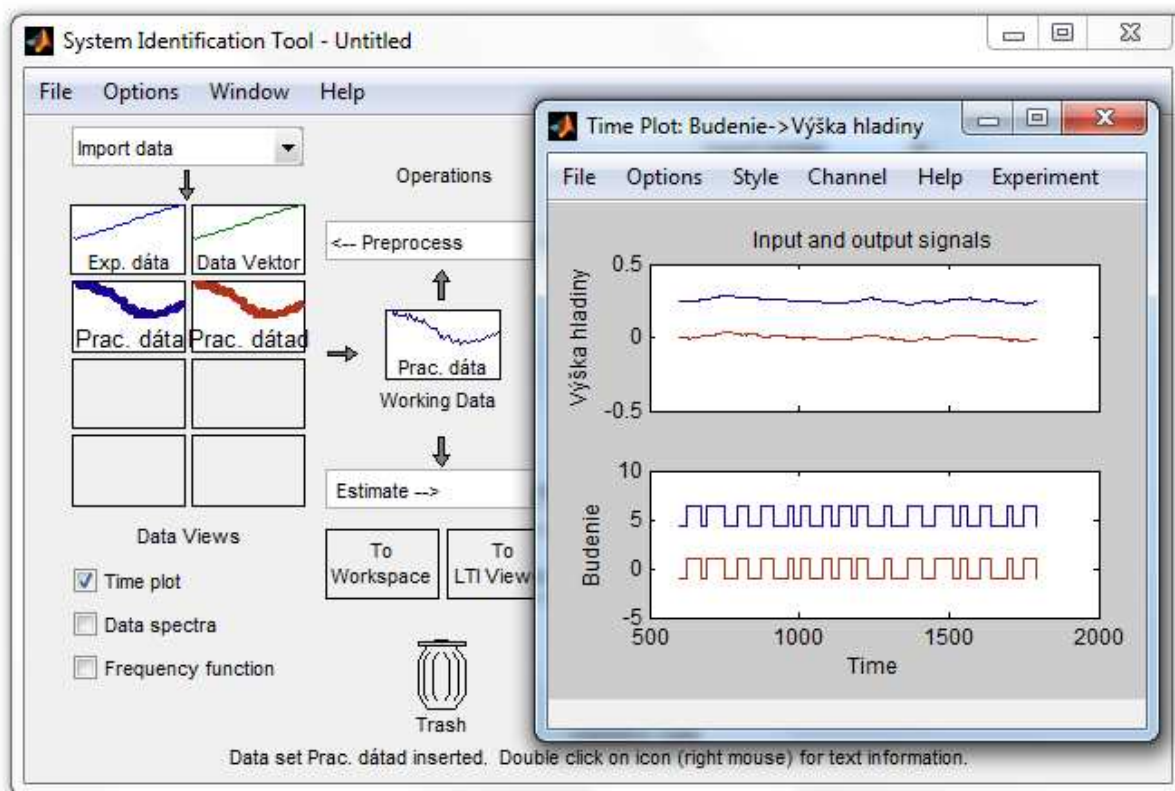
Ako *Working data* mám zvolené Exp. Data. Cez voľbu *Preprocess/Select range...* odstránim nábeh do pracovného bodu. Nový súbor dát nazvem Prac. Data. Potvrdí tlačidlom *Insert*.



Obrázok 27: Orezávanie nábehu - pracovné dáta

## 5.2 Normovanie dát

Ďalší krok je normovanie dát. Ako *Working data* si teraz volím Prac. Dáta. Cez voľbu *Preprocess/Remove means* odstránim konštantné budenie motora Unorm a výšku hladiny v ustálenom stave H2norm. Ich hodnoty sú automaticky vyrátané cez funkciu mean. Vznikne my nový dátový súbor Prac. Datad. Môžem si zobrazit' označené dáta v časovom grafe zaškrtnutím voľby *Time plot*. Ďalej už pracujem s normovanými dátami.



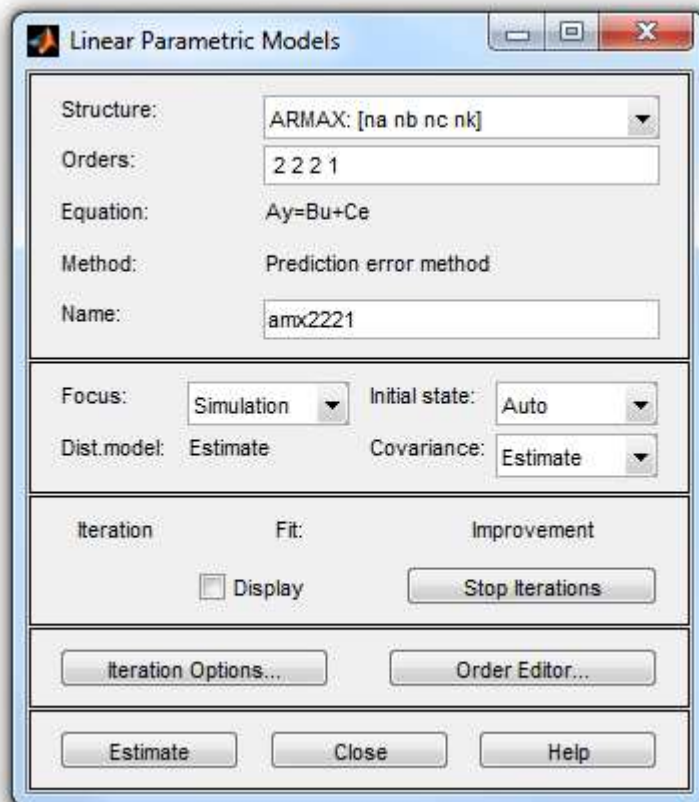
Obrázok 28: Normované dáta a graf porovnania

## 5.3 Delenie dát

Teraz nasleduje delenie tohto súboru dát na TrenData a TestData po 600 vzorkách. Pokiaľ ich máme v Matlab-e, môžeme ich importovať. TrenData budú ďalej používané ako *Working data* a TestData budú použité ako *Validation data*. Overenie, či sú dáta rozdelené dobre je cez *Time plot*.

## 5.4 Určenie ARMAX modelu

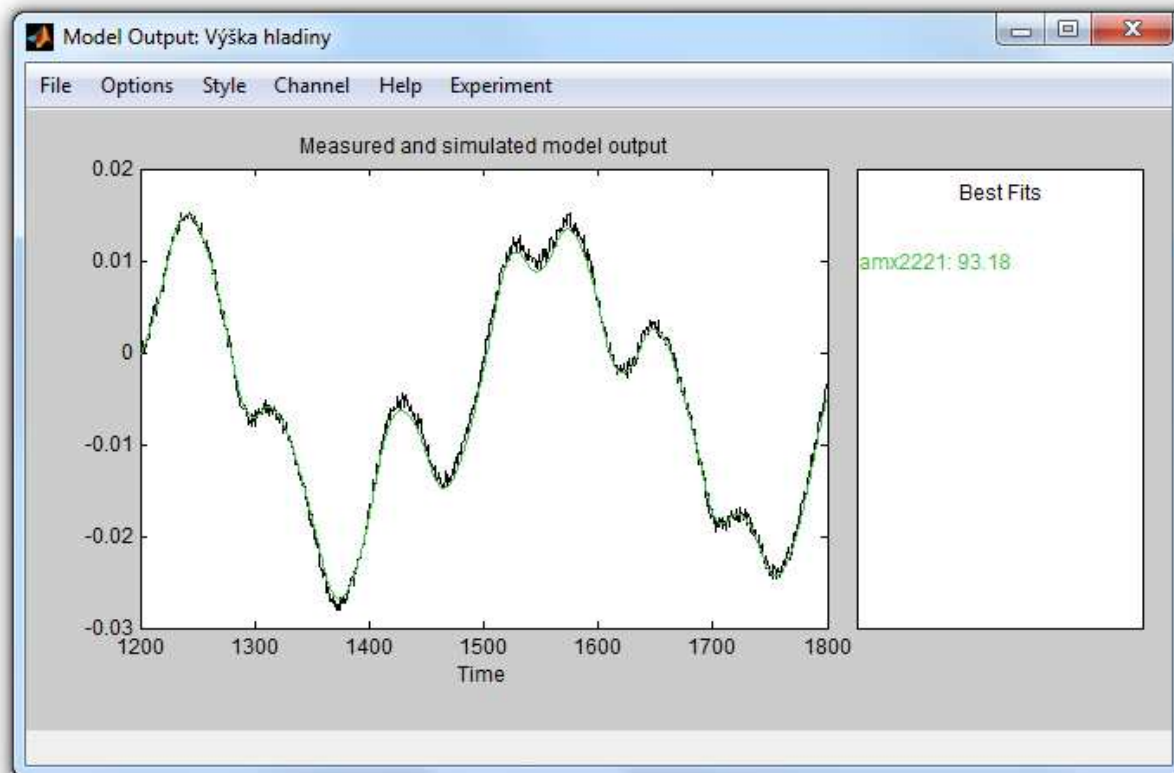
Cez voľbu Estimate/Linear parametric models... si nastavíme ARMAX model s rádmi [2 2 2 1], zameraný na Simuláciu. Vytvoríme ho tlačidlom Estimate a bude sa volať amx2221.



Obrázok 29: Parametre ARMAX modelu

## 5.5 Validácia modelu

Vytvorený model validujeme cez voľbu Model Output. Výsledok validácie je zobrazený v grafe:



Obrázok 30: Model Output graf

## Úlohy

1. Vyskúšajte s *ident* demo
2. Vyskúšajte identifikovať na polovičnom intervale
3. Vytvorte model *ARX* a skúste zvýšiť rády