Solution 1:

The first step would be to plot the data:

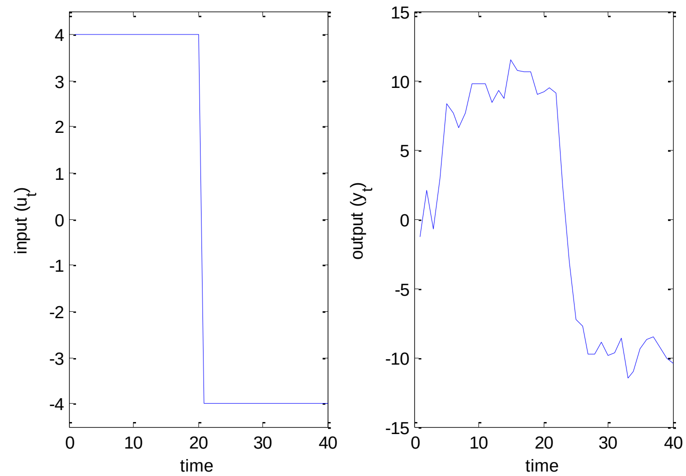

Figure 1: Step response

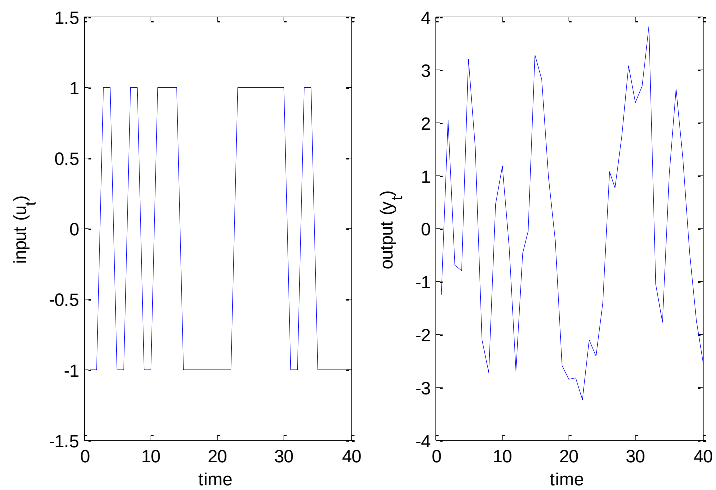

Figure 2: PRBS response

It can be noted from the plot that the data is quite noisy. Before we fit models through the data, it is important to convert the raw input-output data into deviation form. There are various ways to convert data into deviation form, shown below are the two popular choices.

Option 1

Ya_d=ya-ya(1)

Ua_d=ua-ua(1)

Option 2

Ya_d=ya-mean(ya)

Ua_d=ua-mean(ua)

In this assignment, we will use Option 1; however, students are encouraged to try Option 2 to see how the results compare against Option 1.

Again plotting the two datasets in their deviation form (using Option 1):

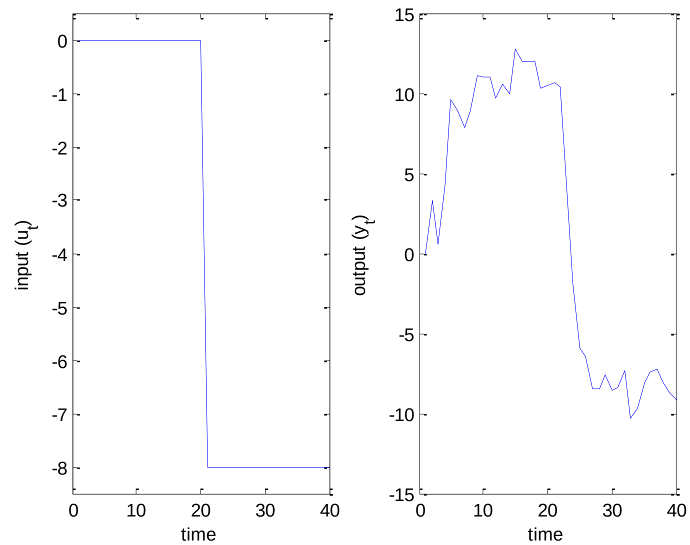

Figure 3: Step response in deviation form

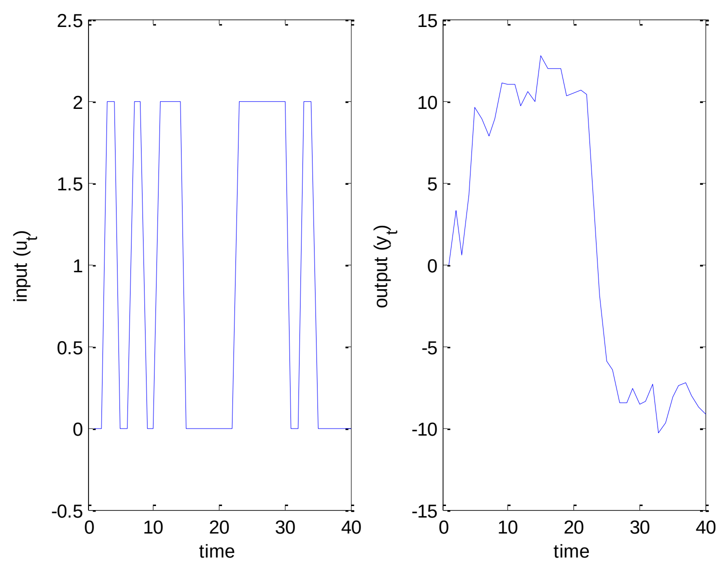

Figure 4: PRBS response in deviation form

The model that we are trying to fit is the ARX model of the form:

$$y_t = \frac{(b_0 + b_1 z^{-1}) z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} u_t + \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}} \varepsilon_t$$

In difference eqn form, we can rewrite the above eqn as:

$$y_t (1 + a_1 z^{-1} + a_2 z^{-2}) = (b_0 + b_1 z^{-1}) z^{-2} u_t + \varepsilon_t$$

Taking inverse Z-transform:

$$y_t + a_1 y_{t-1} + a_2 y_{t-2} = b_0 u_{t-2} + b_1 u_{t-3} + \varepsilon_t$$

$$y_t = -a_1 y_{t-1} - a_2 y_{t-2} + b_0 u_{t-2} + b_1 u_{t-3} + \varepsilon_t$$

Rewriting this in the matrix form:

$$y_t = \begin{bmatrix} -y_{t-1} & -y_{t-2} & u_{t-2} & u_{t-3} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ b_0 \\ b_1 \end{bmatrix}$$

* The good thing about ARX models is that it is linear in parameters and therefore, the techniques of Linear Regression can be applied to estimate them.

We know, that for a linear equation of the form $Y = X\beta$, the value of the parameters are estimated as follows:

$$\hat{\beta} = (X^T X)^{-1} X^T Y.$$

for our problem, the X and Y matrices are given as follows:

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ y_N \end{bmatrix}; \quad X = \begin{bmatrix} -y_0 & -y_1 & u_{-1} & u_{-2} \\ -y_1 & -y_0 & u_0 & u_{-1} \\ \vdots & -y_1 & u_1 & u_0 \\ \vdots & \vdots & & \vdots \\ -y_{N-1} & -y_{N-2} & u_{N-2} & u_{N-3} \end{bmatrix}$$

$$X = \begin{bmatrix} -y_0 & 0 & 0 & 0 \\ -y_1 & -y_0 & u_0 & 0 \\ \vdots & y_1 & u_1 & u_0 \\ & \vdots & & \vdots \\ -y_{N-1} & -y_{N-2} & u_{N-2} & u_{N-3} \end{bmatrix}$$

a)

$$\hat{\beta} = \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \\ \hat{b}_0 \\ \hat{b}_1 \end{bmatrix} = \begin{bmatrix} -1.1895 \\ 0.2257 \\ 0.7963 \\ -0.7028 \end{bmatrix}. \quad ; \hat{\sigma}_\epsilon^2 = \frac{1}{N-p}(Y - X\hat{\beta})^T(Y - X\hat{\beta})$$

$$= 3.6438.$$

$$\text{cov}(\hat{\beta}) = (X^TX)^{+}\hat{\sigma}_\epsilon^2 = \begin{bmatrix} 0.0227 & -0.0210 & -0.0014 & 0.0058 \\ * & 0.0211 & -0.0009 & -0.0020 \\ * & * & 0.0599 & -0.0617 \\ * & * & * & 0.0687 \end{bmatrix}$$

* indicates symmetric terms.

b) Adopting a similar approach as in (a), we get the following results using PRBS data.

$$\hat{\beta} = \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \\ \hat{b}_0 \\ \hat{b}_1 \end{bmatrix} = \begin{bmatrix} -0.5202 \\ 0.0907 \\ 1.2052 \\ -0.2073 \end{bmatrix}, \quad \hat{\sigma}_\epsilon^2 = \frac{1}{N-p}(Y - X\hat{\beta})^T(Y - X\hat{\beta})$$

$$= 1.5780$$

$$\mathrm{Cov}(\hat{\beta}) = (X^T X)^{-1} \hat{\sigma}_\varepsilon^2$$

$$= \begin{bmatrix} 0.0299 & -0.0106 & -0.0083 & 0.0407 \\ * & 0.0125 & -0.0014 & -0.0039 \\ * & * & 0.0486 & -0.0484 \\ * & * & * & 0.1120 \end{bmatrix}$$

* indicates symmetric enteries.

c) True system/process

$$Y_t = \frac{(1-0.25z^{-1})z^{-2}}{1-0.9z^{-1}+0.2z^{-2}} u_t + \frac{1}{1-0.9z^{-1}+0.2z^{-2}} \varepsilon_t$$

Model identified with step data:

$$Y_t = \frac{(0.7963-0.7028z^{-1})z^{-2}}{1-1.1895z^{-1}+0.2257z^{-2}} u_t + \frac{1}{1-1.1895z^{-1}+0.2257z^{-2}} \varepsilon_t$$

model identified with PRBS data:

$$Y_t = \frac{(1.2052-0.2073z^{-1})z^{-2}}{1-0.5202z^{-1}+0.0907z^{-2}} u_t + \frac{1}{1-0.5202z^{-1}+0.0907z^{-2}} \varepsilon_t$$

Comparing the two models against the True system, there doesn't appear to be much of a statistical difference between the two models. In terms of parameter estimates and parameter covariance, the model identified with 'steptest' data appears to be closer to the actual process in atleast 2 parameters.

Overall, if we include the variance of the prediction error, the model identified with the PRBS data appears to be superior.

d). few reccomendations on choosing an appropriate input.

a) PRBS does better than Step, hence the next input to be picked should be PRBS.

b) Using the step data, the time constant for the process appears to be ~ 2 samples (from graphical method). The base switching time for the PRBS should be ≥ 2 samples (current design has a switching time of 1 sample).

c) Run longer experiments.

e) True system:

System poles are: $1 - 0.9z^{-1} + 0.2z^{-2} = 0$

$z = 0.5; 0.4$

$\because |0.5| < 1$ and $|0.4| < 1$

$\Rightarrow$ Stable !.

Model with step data.

System poles are: $1 - 1.1895z^{-1} + 0.2251z^{-2} = 0$

$z = 0.9534; 0.2361$.

Using the same arguement as previously.

$\Rightarrow$ stable !

Model with PRBS data:

System poles are: $1 - 0.5202z^{-1} + 0.0907z^{-2} = 0$

$z = (0.2601 \pm 0.1518i)$

$\Rightarrow$ Stable with decaying oscillations! (not good!).

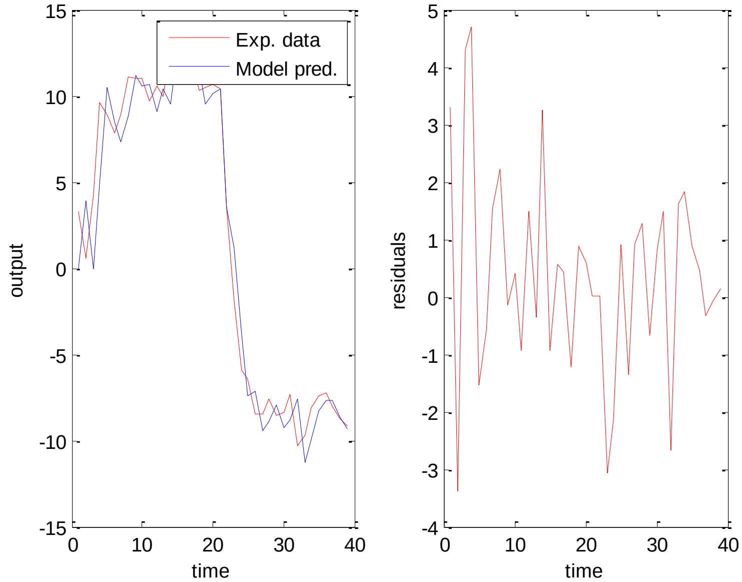

Figure 5: Model predictions and residuals based on step test data

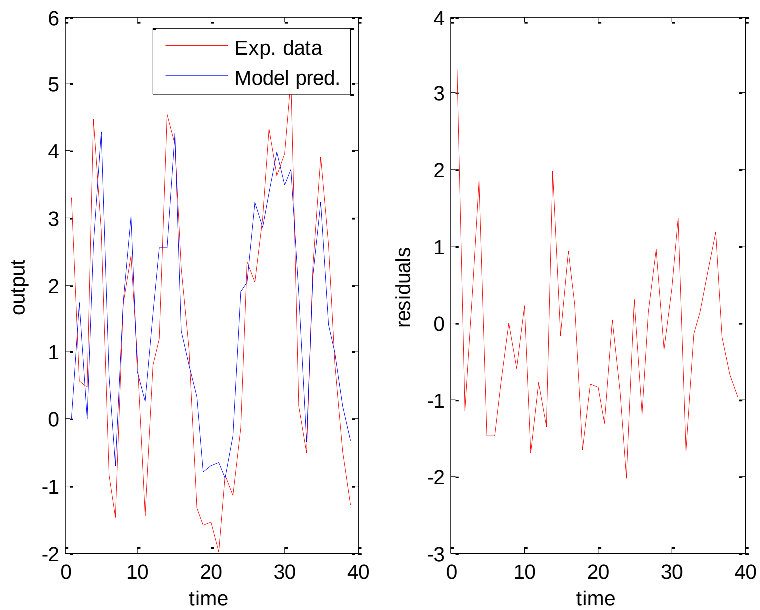

Figure 6: Model predictions and residuals based on PRBS test data

<u>An alternate approach</u>

Instead of using <u>*'linear regression'*</u> approach to compute the parameters of the ARX model, inbuilt MATLAB functions based on <u>*'PEM'*</u> (Prediction Error Method) algorithm can also be used. MATLAB provides a comprehensive platform for model identification of a variety of model forms, namely: ARX (Auto Regressive model); ARMAX (Auto Regressive Moving Average model); OE (output-Error model);

and BJ (Box-Jenkins model). Given below is a pseudo code to compute the parameters of the ARX model using MATLAB.

1a and 1b) we will try to fit an ARX model through the step and PRBS test data. To do this, we use the MATALB command ARMAX (MATLAB uses the same command for both ARMAX and ARX models)

For the given ARX model form, we have

Sample delay: 1

Process delay (dead time): 1

# of poles: 2

# of zeros: 1

Using the following command in MATLAB :

>>dat1=iddata(Ya_d,Ua_d,1);

>>sys1=armax(dat1, [2, 2, 0, 2])

>>dat2=iddata(Yb_d,Ub_d,1);

>>sys2=armax(dat2, [2, 2, 0, 2])

Please refer to the MATLAB documentation on ARMAX for more details on how to define the function arguments.

Results:

Parameters	Estimates using step data	Estimates using PRBS data
b0	0.8008 (+-0.2456)	1.232 (+-0.1994)
b1	-0.722 (+-0.2632)	-0.338 (+-0.3207)
a1	-1.264 (+-0.1566)	-0.616 (+-0.1754)
a2	0.2940 (+-0.1505)	0.1246 (+-0.1045)

The bracketed term is the standard deviation for the estimate obtained using the following command:

>>present(sys1)

>>present(sys2)

You can also use commands like 'compare' to compute the residuals.

Students are encouraged to compare the results obtained here with the linear regression. Why do we get different estimates with the two methods?

Solution 2:

a) The output data is **non-stationary.**
Reason: Assuming that the process stated at some steady state, a careful look in the interval between t=0 and 6000 samples suggests a downward drifting trend. This is indicative of an integrating term in the noise model. The same trend is seen in the interval between t=6000 to 10000 samples, where it process or output appears to respond to a step change in the inputs. The continuing upward trend in the system response around the end of the experiment makes the output data non-stationary.

b) Figure 2 shows an estimate of the impulse response based on MATALB command *'cra'*. There appears to be a lag of 2 samples in the response. Clearly, this indicates that the process dead time is 1 and sample delay is 1. (Note that in discrete time system there is always a sample lag of 1). Also, the trajectory of the impulse response suggests that the underlying process is typically a second order system.
Figure 3 shows a graph of estimated step weights, the inverse response is indicative of a presence of a zero in the underlying process.

Conclusion about the process model:

- Poles: 2
- Dead time: 1
- Zeros: 1

c) Figure 3 and Figure 4 shows the ACF and the PACF response, respectively, based on once differenced data. ACF trajectory gradually dies off in the confidence belt whereas the PACF takes around 2 lags before dipping down into the confidence belt. Clearly, this is a signature of a second order AR model. Also, since the results do not show any signature of the integrating part. Clearly, this suggests that a single time data differencing has taken care of all the underlying integrating terms in the noise model.

Conclusion about the noise model:

- Order of the AR part: 2
- Order of the integrating part: 1
- Order of the MA part: 0

d) The general model for the underlying system is of the following form:

$$y_t = \frac{(b_0 + b_1 z^{-1})z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} u_t + \frac{1}{(1 + p_1 z^{-1} + p_2 z^{-2})(1 - z^{-1})} e_t$$