Assignment 5
Subspace Identification

Subjects: Subspace identification: Singular value decomposition, similarity transform, order selection

Important notes

- Reports handed in after the deadline will not be marked, resulting in a zero.
- Working in teams of (not more than) two students is strongly encouraged.

How to get help

- Always check the *Assignment Guide* on Blackboard for support.
- The command functions *HELP* or *DOC* <*FUNCTIONNAME*> can be used for instant support to MATLAB functions.
- Assistance hours (see schedule).
- Help each other using the *Discussion Board* on Blackboard.

Report requirements

- Reports should be written in English.
- Each student hands in a report using blackboard. Always provide the name of the partner at the report.
- Always provide explanations in your answers and be precise and concise (providing irrelevant information can negatively influence your mark). Formulate your answer in such a way that you show you understood the question.
- Use figures to illustrate your findings. If possible use the same figure for the results of multiple sub-questions; this allows for easy comparison. When plotting more than one curve in a figure, use legends. Do not forget a caption and the units on the axes. Choose proper axes scales. Unclear Figures and Tables will be excluded from judgment.
- Do not submit MATLAB coding, unless it is clearly stated that you should.
- Put your name (and that of your colleague if applicable), page number, and the assignment number on each page of your report.

Goal

The goal of this assignment is to understand the basic principles of subspace identification.

Literature

Chapter 9 of the book 'Filtering and system identification' of Verhaegen and Verdult (can be found through netlibrary) gives the underlying theory of subspace identification.

Accessories: *SSASS1.M* *SSASS2.M* *SSASS3.M* *SSASS4.M* *SSASS5.M* *GETSSHP.P* *GETLPV.P* *MVAF.M* *VAF.M* *TVDATSET.MAT* *SIMLPV.P*

Question 1: Identification of autonomous systems

In this exercise a second order mass-spring-damper system is excited with an initial condition and the system matrices are estimated back using a standard subspace method.

a) Run part A & B of SSAss1.M. In figure 2 the data matrix '*YsN*', which is a Hankel matrix of the output data, is plotted. Put figure 2 in your report containing the trajectories of 3 different initial conditions and with the correct axis labels. What do you notice when looking at the different trajectories in both figures? *Hint: Also take a look at the plot of the different output signals versus time.*

b) Run part C of SSAss1.M . How do the lines which now appear in figure 2 relate to the matrices '*U*' and '*S*' of the singular value decomposition? Do this for 3 different initial conditions and explain the differences and similarities.

c) Now make a plot of at least two different systems in a new figure 2 (Change the parameters in line 10-11) . Explain your observations. What happens with the trajectories in figure 2 when you change the damping ('*B*') parameter only?

d) Make an estimate of the order of the system by looking at the plot of the singular values in figure(3). Fill in the system order in part D (variable 'n') and running it will extract the system matrices out of the SVD. Compare the eigenvalues (MATLAB command: EIG) of the original system '*A0*' with that of the identified system '*Aid*'. What does this tell you about the dynamics of the original and identified system?

e) In figure(1) the output of the original and identifies system are plotted together. Why is there a difference between the outputs when using the same initial value vector? Find a solution and implement it. *Hint: Make use of the data equation and the SVD.*

f) Run the script with added output noise (set variance to 1e-2, line 18), and comment on what happens in each of the 3 plots compared to the noiseless situation. (p.e. data space, singular values, quality of estimation).

g) Results obtained with a dataset subjected to noise can be improved by changing the size of the data-matrix '*YsN*'. What difference can you see in the magnitudes of the singular values compared to f?

h) Now change the system ('*sys0*' in part A) to a (stable!) 4rd order system. Add a considerable amount of output noise and estimate the system matrices out of the simulated data. Show that you made a good estimate of the original system.
Hint: when changing the system, check the time-response to see how it behaves.

Question 2: Identification with general inputs

a) In SSAss2.m a system is excited with an input *uk(1)*=1, *uk(2:end)*=0. Now there is a bias on the resulting output in figure(3). Use the data equation (lecture sheets) to explain how this can be fixed by setting the start index of the data-matrices ('*ii*' in part B) from 0 to 1, instead of estimating initial conditions as you have done before.

b) If you set the start-index of the data-matrix ('*ii*' in part B) higher than 1, again you end up with an incorrect estimate (Zoom in if you don't believe it). Which of the system matrices (A to D) is/are estimated incorrectly? Make an estimate for the B-matrix which will work for *ii>1*. *Hint: Find the formula in the lecture slides which is used to estimate the B-matrix and find out why it does not work for ii>1*

c) Look at how the D-matrix is estimated in part D of SSAss2.m . Why is this estimate sensitive to output noise?

d) In SSAss3.m the system is excited with an input *uk*=white noise. What happens with the data-space figure(2) because of the noise input?

e) To find the correct system matrices we need to do two things:

   1) Remove the influence of the input on the output data ('*YsN*') matrix. One way to do this is by multiplying '*YsN*' with '*PsN*' (the orthogonal projection onto the column space of '*UsN*') giving '*YsNP*'. Uncomment the appropriate lines in part C of SSAss3.m to do so. Plot '*YsN*' and '*YsNP*' in one figure.

   2) Estimate the B and D system matrices using the least squares routine given in part E of SSAss3.m and plot the results;

f) Use the data-equation to explain why we need to conduct step 1 in e).

g) This same routine yields unbiased results for output noise. Try by adding some noise on the output (p.e. variance 1e-3) and increasing the '*s*' and '*N*' parameters. *Note: The routine presented here is very slow, modern subspace routines are a lot faster: be patient*

To illustrate the power of subspace identification in the next question you will use a modern subspace algorithm. Speed and performance is hugely improved by using smart matrix factorizations and projections.

h) Create a MIMO state space system with at least 3 inputs and 3 outputs using SSAss4.m. Add noise on both the inputs and outputs and identify the system using GETSSHP.P. Give a bode-plot of the original and identified system and a short fragment of the in – and output time series.

3. **Time varying systems.**

i) Open and run SSAss5.m. This script loads TVdatset.mat and all signals are plotted in Figure(1). Let's say this data describes recorded torques (uk) and rotations (yk) on some joint. Three EMG signals (ek) are also recorded. Question a to c use the same algorithm as 2h. First look at the singular values of figure(3). What order would be suitable to identify this system?

j) Now look at the results in figure(2). The bottom plot gives a moving variance accounted for (VAF) value and gives an impression of the system estimates over time. What do you notice?

k) Now try to see if maybe adding the EMG signals improves the estimates by simply replacing 'uk' by '[uk ek]' in line 28&30. Does this help?

l) Undo the changes made in c). Maybe the EMG signals do not act as an extra input, but do manipulate the system dynamics. It is believed that a low-pass filtered version of one of the EEG signals is responsible for changing some gains within the system. In this case an LPV structure might find a good estimation. Try each of the EMG signals as a scheduling function, or just give a good guess based on the signals in figure(1). Do this by uncommenting part B and enter the correct ek-signal in line 51.

m) By replacing 'ek(:,x)' with 'ek(:,:)' in line 51 all three EEG signals are used as a scheduling function (might take some time to execute…). Now look at the entries of 'Al', which is the LPV A-matrix (or try 'mesh(abs(Al))') . Can you deduct from this which EEG signal was the correct one and why?

n) Undo the changes made in m) and enter the right EEG-signal as a scheduling. Now run the script together with part C. How did the EEG signal affect the stiffness of the system?