Bonus assignment KEN2430 Mathematical Modeling 2022/2023

March 7, 2023

Instructions

Please hand in your solution as **a single .m file**. Do not use a livescript when you hand in. You are free to work with livescripts before handing in, but we will be running a plagiarism scan and will need .m files. The assignments should be uploaded as .m and not be compressed to facilitate automatic plagiarism checks. Your m-file can contain multiple functions to keep things organized. You have to upload your solution **before 11:00h (Maastricht time) on 20 March, 2023 through Canvas**. Please submit in time and account for the time you need to submit. If Canvas closes the submission it closes and the Canvas time counts. This is an individual graded assignment worth **0.5 bonus points**. What you submit should be your work and **your work alone**, and by submitting it you testify that this is indeed the case. Doing it alone also means that you did not use AI to generate your submitted work. :-) Your work should adhere to the **template supplied below**, which therefore obviously can be used.

Challenge

We ask you to write a Matlab function that assists you on practicing exercises on computing the discrete-time response $y[k]$ corresponding to a given proper rational function $Y(z)$. Your function should start from the template given below and do the following:

1. Generate a random example of $Y(z)$ and nicely display it. The function $Y(z)$ should be proper rational. The template has optional arguments in the function header, which should be taken into account (whenever provided by the user) to allow for a decent amount of variation in the problems generated.

2. Display the solution steps, adhering to the solution strategy discussed in this course:

   (a) Divide by z and display the rational function $Y(z)/z$.

   (b) Compute the poles and use them to display the appropriate form of the partial fraction expansion for $Y(z)/z$. I.e., display an expression containing all the elementary terms you aim to use to rewrite $Y(z)/z$, still having unspecified coefficients.

   (c) Compute and display the coefficients for each of the elementary partial fractions. [You are *not* allowed to use the `residuez` function.]

   (d) Multiply by z and display the resulting partial fraction expansion for $Y(z)$.

   (e) Show the result of the inverse z-transformation: present the solution $y[k]$.

What you should produce is an .m file with the following header:

```
%% practiceDTpfe
%  function to assist on practicing discrete-time partial fraction expansion
%
%  inputs:
%     order: the desired order of the rational function Y(z) to be generated.
%          This can be a scalar, but can also be a vector of length 2,
%          which gives the minimum and maximum order and this is randomly
%          sampled in this range according to a uniform distribution.
%          Optional argument, default [2,4].
%     repeatedPoles: if 0: repeated poles might occur by chance, if 1:
%          repeated poles are enforced to happen.
%          Optional argument, default: 0.
%     complexPoles: if 0: there are no complex conjugate pairs of poles, if 1:
%          complex conjugate pairs of poles are enforced to happen.
%          Optional argument, default: 0.
%     num: Fix the numerator polynomial, rendering the first three
%           arguments irrelevant. The 5th argument must be provided as well.
%     den: Fix the denominator polynomial, rendering the first three
%           arguments irrelevant. The 4th argument must be provided as well.
%
%  outputs:
%     PFEinfo: An mx3 matrix, where $m$ denotes the number of
%          poles of the rational function Y(z)/z. (This equals the number of
%          elementary terms in the PFE of Y(z)/z.) Each row should contain:
%          [the pole location p, its multiplicity j, its coefficient A]
%
%  When running the function, various info and details are displayed:
%    - the proper rational function Y(z),
%    - the form of the PFE of Y(z)/z showing its elementary terms,
%    - the result of the PFE for Y(z)/z, as well as for Y(z),
%    - the corresponding time sequence y[k].
%  In the result, poles at z=0 give rise to (possibly delayed versions of) the
%  impulse sequence delta[k].
%
function [PFEinfo] = practiceDTpfe(order,repeatedPoles,complexPoles,num,den)

if(nargin==5)
    % if the numerator and denominator are provided we don't have to
    % generate them. As we do not want a non-causal system, we check on this:
    if(length(num)>length(den)), error('The transfer function must be proper'); end
elseif(nargin==4)
    error('If you provide a numerator, you should also provide a denominator');
else
    % we will have to generate a numerator and denominator. We first will
    % handle the optional arguments by specifying default values.
    % (We are just providing an example start here - feel free to modify ):
    switch nargin
        case 2
            complexPoles=0;
    end
    % generate num and den randomly.
end
```

So what do we expect as output `PFEinfo`?

Example: suppose $Y(z) = \dfrac{5z^4 - z^3 + \frac{1}{3}z^2 - \frac{2}{5}z + 3}{z^4 + \frac{1}{2}z^3 - \frac{1}{4}z^2 - \frac{1}{8}z}$, then we can compute its poles (and their multiplicities) to factorize the denominator and obtain:

$$\frac{Y(z)}{z} = \frac{5z^4 - z^3 + \frac{1}{3}z^2 - \frac{2}{5}z + 3}{z^2(z - \frac{1}{2})(z + \frac{1}{2})^2}.$$

Therefore, the appropriate form of the partial fraction expansion of $Y(z)/z$ is:

$$\frac{Y(z)}{z} = \frac{B_{1,1}}{z} + \frac{B_{1,2}}{z^2} + \frac{A_1}{z - \frac{1}{2}} + \frac{B_{2,1}}{z + \frac{1}{2}} + \frac{B_{2,2}}{(z + \frac{1}{2})^2}.$$

Once you computed the values for the coefficients $B_{1,1}$, $B_{1,2}$, A_1, $B_{2,1}$, $B_{2,2}$ you can generate the output variable `PFEinfo` as follows:

$$\text{PFEinfo} = \begin{pmatrix} 0 & 1 & B_{1,1} \\ 0 & 2 & B_{1,2} \\ \frac{1}{2} & 1 & A_1 \\ -\frac{1}{2} & 1 & B_{2,1} \\ -\frac{1}{2} & 2 & B_{2,2} \end{pmatrix}$$

in which the actual values for the coefficients should now be displayed.

When running the function, the **final output displayed** for $y[k]$ should leverage the information contained in PFEinfo to show two types of terms: **impulses** $\delta[k]$ (possibly delayed) for poles located at $z = 0$, and **general powers** $\begin{pmatrix} k \\ j-1 \end{pmatrix} b^{k-(j-1)}$ for poles located at b. See Table 4.4 in the lecture notes for full details. Note that $\begin{pmatrix} k \\ j-1 \end{pmatrix}$ is a binomial coefficient, which for small values of j may be worked out to make explicit that it defines a polynomial in k of degree $j - 1$. Note also that the pole b may be complex (rather than real), in which case the coefficients in the PFE may also be complex. In that case, complex conjugate counterparts should also feature in the decomposition as well as in the solution.

To give an example of what we would consider **fancy formatting** to display certain output:

```
The partial fraction expansion found is:
           -2.33333z      3z
H(z)= 1 + ---------- +  -----
             z + 3       z + 2
```

Keep in mind that, because the tool is intended to help you practice exercising, you may want to only generate coefficients, poles, zeros with nice (integer or rational) values.

As an optional add on: can you make sure to only display real expressions for the final solution?

Have fun!