Short guide to Control Systems Toolbox

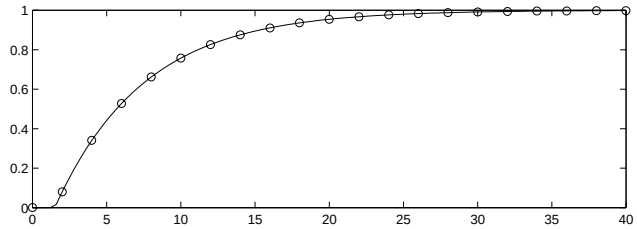
This guide is an introduction on how to use Control Systems Toolbox for control analysis and design, especially of computer controlled systems.

The basic data structure is the LTI (linear time-invariant) model. There is a number of ways to create, manipulate and analyze models. Some operations are best done on the LTI system, and others directly on the matrices and polynomials of the model. First, some examples on basic system manipulation:

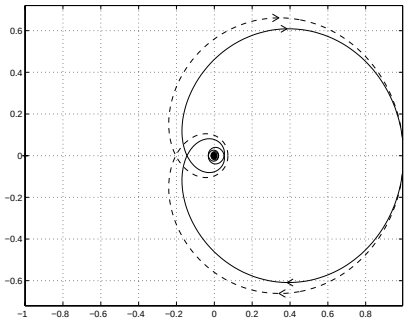
```
>> % Continuous transfer function G1(s)=e^(-3s)/(6s+1) :
>> G1 = tf(1,[6 1],'Td',1.5)
      Transfer function:
         1
      -------
      6 s + 1

      Input delay: 1.5
>> % ZOH sampling of G1 using h=2:
>> H1 = c2d(G1,2)
      Transfer function:
      0.07996 z + 0.2035
      ------------------
        z^2 - 0.7165 z

      Sampling time: 2
>> % Extract zeros, poles and gain from H1 as vectors:
>> [z,p,k] = zpkdata(H1,'v')
z =
   -2.5453
p =
         0
    0.7165
k =
    0.0800
>> % Calculate step responses:
>> [yc,tc] = step(G1,40); [yd,td] = step(H1,40);
>> plot(tc,yc,'-',td,yd,'o')
```




```
>> % Nyquist plots of G1 and H1 (positive and negative frequencies):
>> nyquist(G1,H1);
```



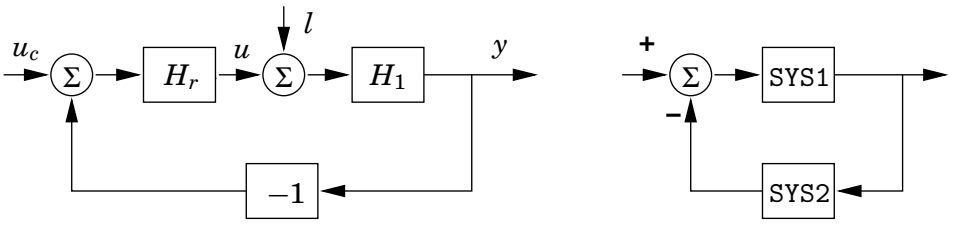
State feedback example

Problem 5a from the exam in August 1998, available on the web.

```
>> % Discrete-time state space model, set for example h=1;
>> Phi = [1.7 -0.72; 1 0]; Gamma = [1 0]'; C = [1 -0.7]; D=0; h=1;
>> sys1 = ss(Phi,Gamma,C,D,h);
>> % Check reachability:
>> Wc = ctrb(sys1)
Wc =
    1.0000    1.7000
         0    1.0000
>> rank(Wc)
ans =
     2
>> % Solve the discrete time characteristic polynomial:
>> desired_poles = roots([1 -1.2 0.5]);
>> % Place the poles:
>> L = place(Phi,Gamma,desired_poles)
L =
    0.5000   -0.2200
>> % Use Lc to set static gain = 1:
>> Lc = 1 / (C/(eye(2)-(Phi-Gamma*L))*Gamma)
Lc =
    1.0000
```

Connecting systems

LTI systems can be interconnected in a number of ways. For example, you may add and multiply systems (or constants) to achieve parallel and series connections, respectively. Assume that H_1 above should be controlled by a PI controller with $K = 1$, $Ti = 4$ and $h = 2$, according to the standard block diagram to the left:



There is a function `feedback` for constructing feedback systems. The call `feedback(SYS1,SYS2)` results in a system according the block diagram to the right. Note the sign conventions. To find transfer functions from the set point u_c and the load disturbance l to control signal u and ouput y, you must identify what `SYS1` and `SYS2` should be for each case. From u_c to y it is easy to see that `SYS1` is H_1H_r and `SYS2` is 1.

```
>> % Discrete-time PI controller with K=1, Ti=4, h=2:
>> K=1; Ti=4; h=2;
>> Hr = K*(1+tf(h/Ti,[1 -1],h))
      Transfer function:
      z - 0.5
      -------
       z - 1

      Sampling time: 2
>> Hyuc = feedback(H1*Hr,1);  % from above
>> % Try to verify the following transfer functions as well:
```

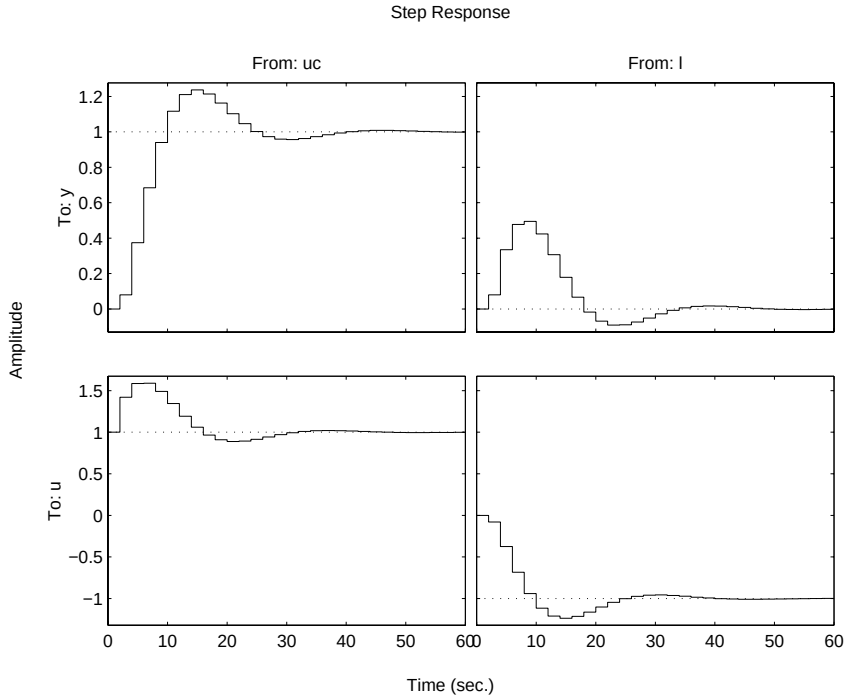
```
>> Huuc = feedback(Hr,H1);
>> Hyl = feedback(H1,Hr);
>> Hul = feedback(-Hr*H1,-1);
>> % You may put the transfer functions together into a matrix of
>> % transfer functions, with inputs uc and l, outputs y and u:
>> %
>> %                         +--------+
>> %               uc ------>|        |----> y
>> %                         | CLSYS  |
>> %                l ------>|        |----> u
>> %                         +--------+
>> CLSYS = [Hyuc Hyl; Huuc Hul];
>> % It is possible to assign names to the signals:
>> set(CLSYS,'InputName',{'uc','l'},'OutputName',{'y','u'})
>> CLSYS
      Transfer function from input "uc" to output...
             0.07996 z^2 + 0.1635 z - 0.1018
       y:  -----------------------------------
           z^3 - 1.637 z^2 + 0.8801 z - 0.1018

              z^3 - 1.217 z^2 + 0.3583 z
       u:  -----------------------------------
           z^3 - 1.637 z^2 + 0.8801 z - 0.1018

      Transfer function from input "l" to output...
             0.07996 z^2 + 0.1236 z - 0.2035
       y:  -----------------------------------
           z^3 - 1.637 z^2 + 0.8801 z - 0.1018

            -0.07996 z^2 - 0.1635 z + 0.1018
       u:  -----------------------------------
           z^3 - 1.637 z^2 + 0.8801 z - 0.1018

      Sampling time: 2
>> % Show step responses from set point and load disturbance to output and control signal:
>> step(CLSYS)
```



In order to mix continuous-time and discrete-time systems, you must simulate in Simulink. The plots above do NOT show the output of the continuous system between the sampling points.

Some useful functions from Control Systems Toolbox

Do `help` <*function*> to find possible input and output arguments.

```
Creation and conversion of continuous or discrete time LTI models.
  ss          - Create/convert to a state-space model.
  tf          - Create/convert to a transfer function model.
  zpk         - Create/convert to a zero/pole/gain model.
  ltiprops    - Detailed help for available LTI properties.
  ssdata etc. - Extract data from a LTI model.
  set         - Set/modify properties of LTI models.
  get         - Access values of LTI model properties.

Sampling of systems.
  c2d         - Continuous to discrete conversion.
  d2c         - Discrete to continuous conversion.

Model dynamics.
  pole, eig   - System poles.
  pzmap       - Pole-zero map.
  covar       - Covariance of response to white noise.

State-space models.
  ss2ss       - State coordinate transformation.
  canon       - State-space canonical forms.
  ctrb, obsv  - Controllability and observability matrices.

Time response.
  step        - Step response.
  impulse     - Impulse response.
  initial     - Response of state-space system with given initial state.
  lsim        - Response to arbitrary inputs.
  ltiview     - Response analysis GUI.
  gensig      - Generate input signal for LSIM.
  stepfun     - Generate unit-step input.

Frequency response.
  bode        - Bode plot of the frequency response.
  nyquist     - Nyquist plot.
  ltiview     - Response analysis GUI.

System interconnections.
  + and -     - Add and subtract systems (parallel connection).
  *           - Multiplication of systems (series connection).
  / and \     - Division of systems (right and left, respectively).
  inv         - Inverse of a system.
  [ ]         - Horizontal/vertical concatenation of systems.
  feedback    - Feedback connection of two systems.

Classical design tools.
  rlocus      - Root locus.
  rlocfind    - Interactive root locus gain determination.
  rltool      - Root locus design GUI
  place       - Pole placement (state feedback or estimator).
  estim       - Form estimator given estimator gain.
  reg         - Form regulator given state-feedback and estimator gains.

LQG design tools. Notation differs from CCS.
  lqr,dlqr    - Linear-quadratic (LQ) state-feedback regulator.
  lqry        - LQ regulator with output weighting.
  lqrd        - Discrete LQ regulator for continuous plant.
  kalman      - Kalman estimator.
  kalmd       - Discrete Kalman estimator for continuous plant.
  lqgreg      - Form LQG regulator given LQ gain and Kalman estimator.

Matrix equation solvers.
  dlyap       - Solve discrete Lyapunov equations.
  dare        - Solve discrete algebraic Riccati equations.
```