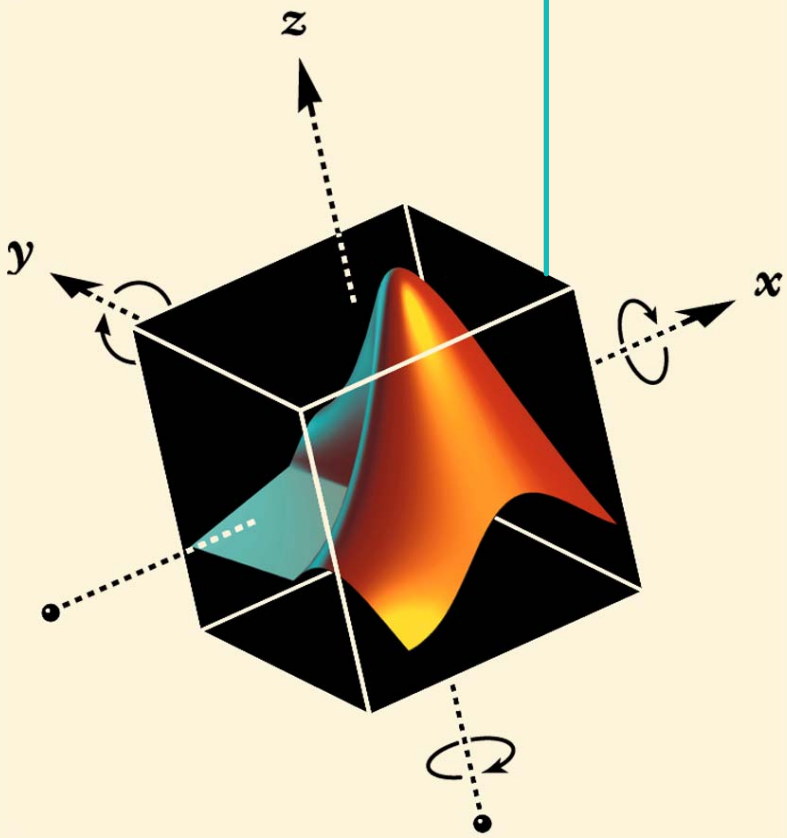
Control Design and Analysis in MATLAB

>> 1997

>> matlab conference



Karen Gondoly and Pascal Gahinet

kgondoly@mathworks.com

pascal@mathworks.com

The MathWorks Inc.

24 Prime Park Way

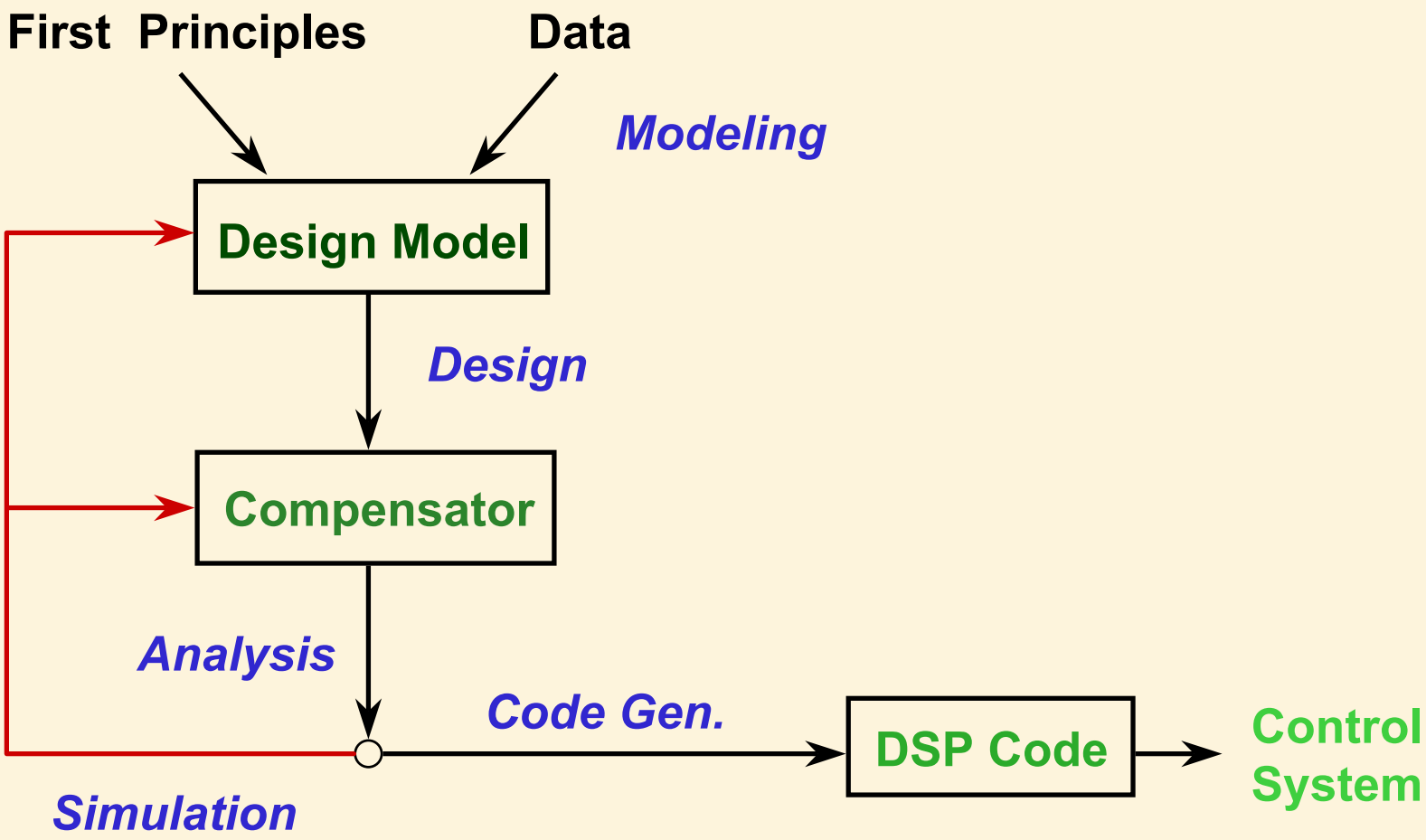
Natick, MA 01760-1500, USA

Outline

- **The Control Design Process**
- **Control System Toolbox 4.1**
  - *LTI objects*
  - *New GUIs*
  - *Simulink-based Tools*
- **Advanced Control Toolboxes**
- **Future Directions**

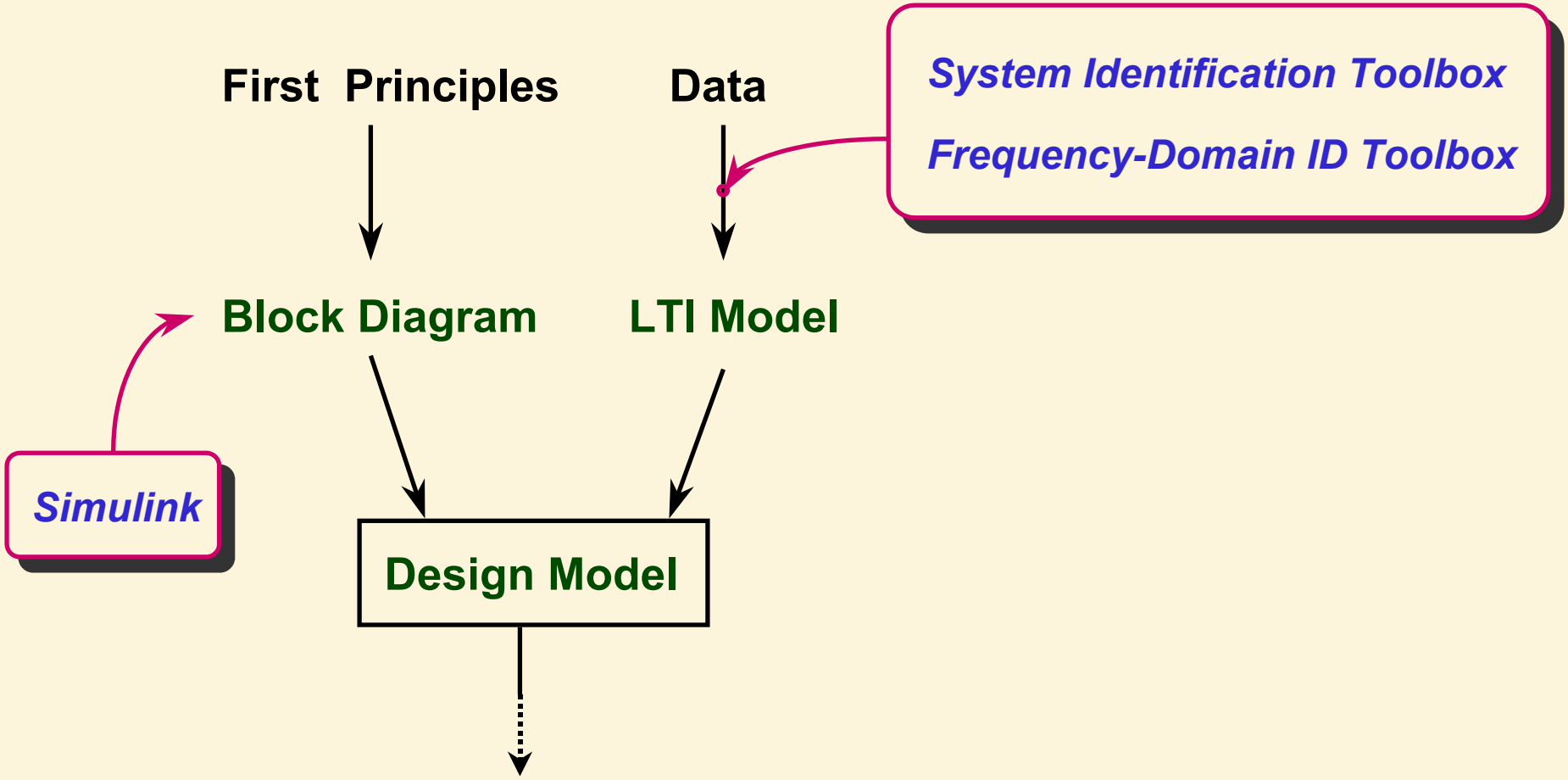
Copyright © 1984 - 1997 by The MathWorks, Inc.

The Control Design Process



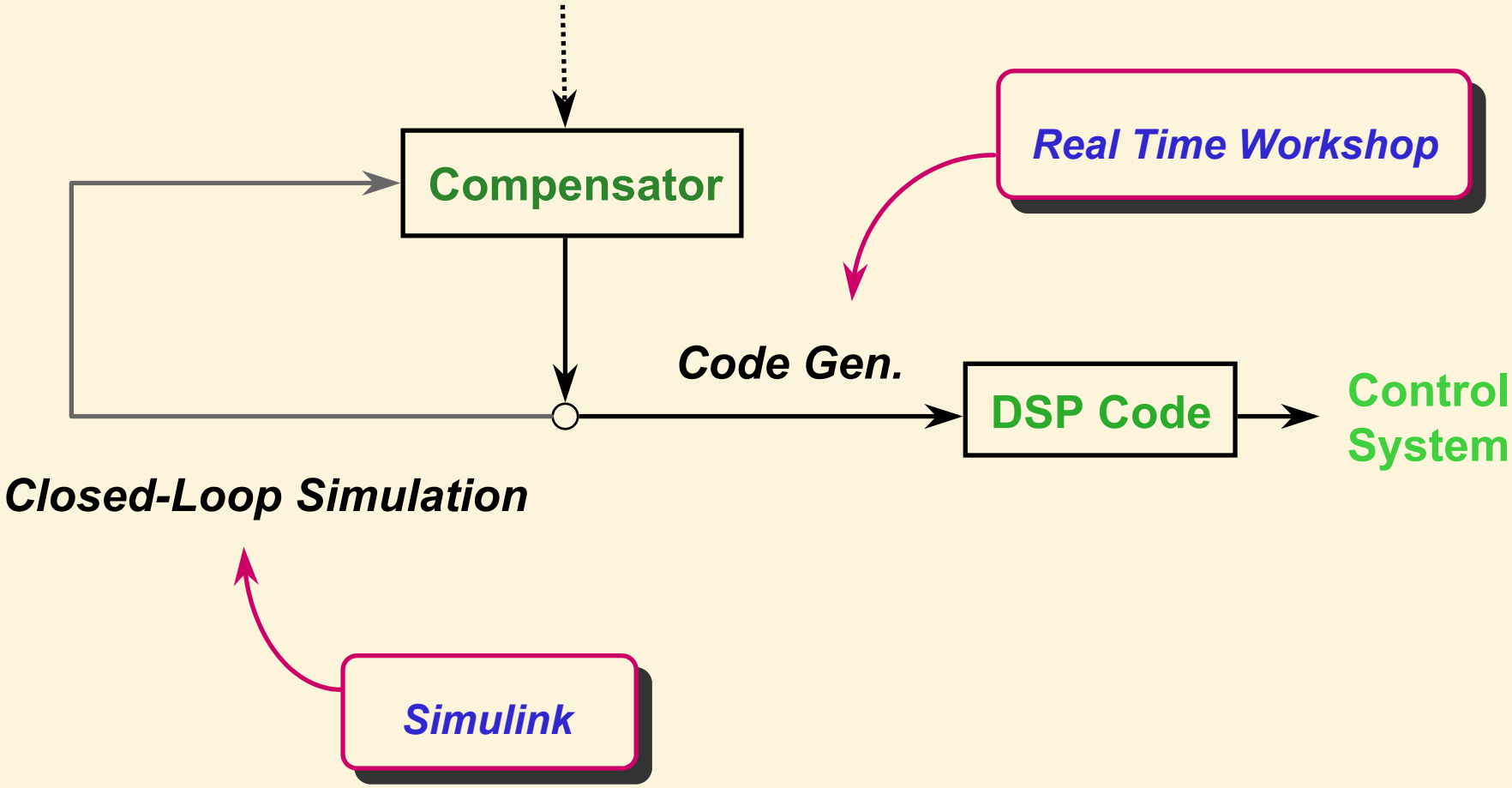
Copyright © 1984 - 1997 by The MathWorks, Inc.

Modeling Tools



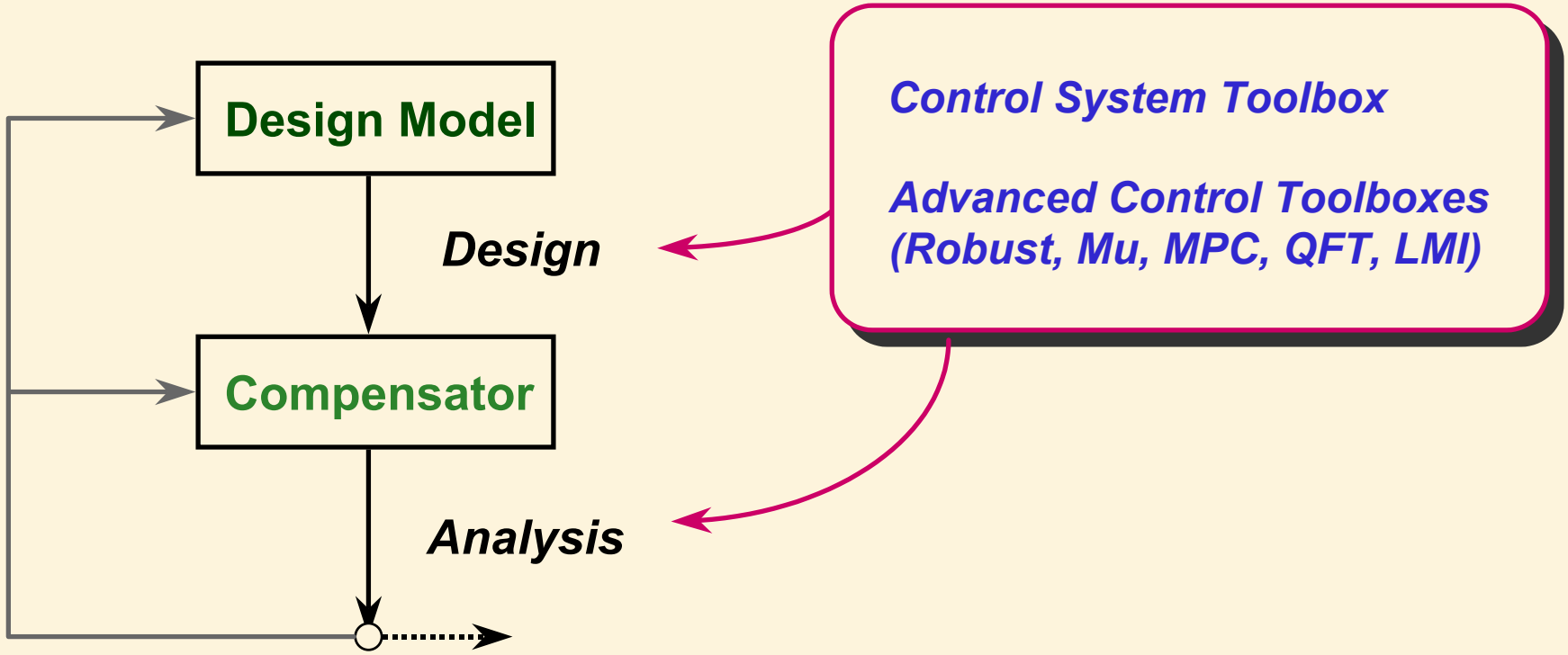
Copyright © 1984 - 1997 by The MathWorks, Inc.

Simulation and Implementation



Copyright © 1984 - 1997 by The MathWorks, Inc.

Design and Analysis Tools



Copyright © 1984 - 1997 by The MathWorks, Inc.

Outline

- The Control Design Process
- Control System Toolbox 4.1
- Advanced Control Toolboxes
- Future Directions

Copyright © 1984 - 1997 by The MathWorks, Inc.

Control System Toolbox 4.1

- Core Features
- LTI Objects
- Root Locus Design GUI NEW!
- LTI Viewer

Copyright © 1984 - 1997 by The MathWorks, Inc.

Core Features

- **Tools to manipulate LTI models**
- **Classical analysis and design:**
  - *Bode, Nyquist, Nichols diagrams*
  - *Step and impulse response*
  - *Gain/phase margins*
  - *Root locus design*
- **Modern state-space techniques:**
  - *Pole placement*
  - *LQG regulation*

Copyright © 1984 - 1997 by The MathWorks, Inc.

LTI Objects

(LTI = Linear Time Invariant)

- **Customized data structures for LTI models**
- **Three flavors:**

```
sys = tf(num,den)      % transfer function
sys = zpk(z,p,k)       % zero-pole-gain
sys = ss(a,b,c,d)      % state space
```

LTI object `sys` ***encapsulates the model data***

Copyright © 1984 - 1997 by The MathWorks, Inc.

SISO Example:

$$H(z) = \frac{2}{(z-0.2)(z^2 - z + 1.25)}$$

```
>> Ts = 0.1   % sample time

>> zpk([],[0.2 0.5-i 0.5+i],2,Ts)

Zero/pole/gain:
          2
---------------------
(z-0.2) (z^2 - z + 1.25)

Sampling time: 0.1
```

Copyright © 1984 - 1997 by The MathWorks, Inc.

MIMO Example:

$$H(s) = \begin{bmatrix} 0 & \frac{1}{s} \\ \frac{s+1}{s^2+s+1} & 2 \end{bmatrix}$$

```
>> h = [0                          tf(1,[1 0]) ;
        tf([1 1],[1 1 1])                    2 ]
```

Full support for MIMO models!

Copyright © 1984 - 1997 by The MathWorks, Inc.

LTI Objects: Conversions

```
>> sys=tf(1,[1 -3 2])
Transfer function:
      1
-------------
s^2 - 3 s + 2
```

ss

```
>> ss(sys)
a =
              x1          x2
   x1    3.00000    -1.00000
   x2    2.00000           0

b =
              u1
   x1    0.50000
   x2          0

c =
              x1          x2
   y1          0     1.00000

d =
              u1
   y1          0
```

zpk

```
>> zpk(sys)
Zero/pole/gain:
     1
-----------
(s-2) (s-1)
```

Copyright © 1984 - 1997 by The MathWorks, Inc.

LTI Objects: Properties

Access/modify properties using get and set ...

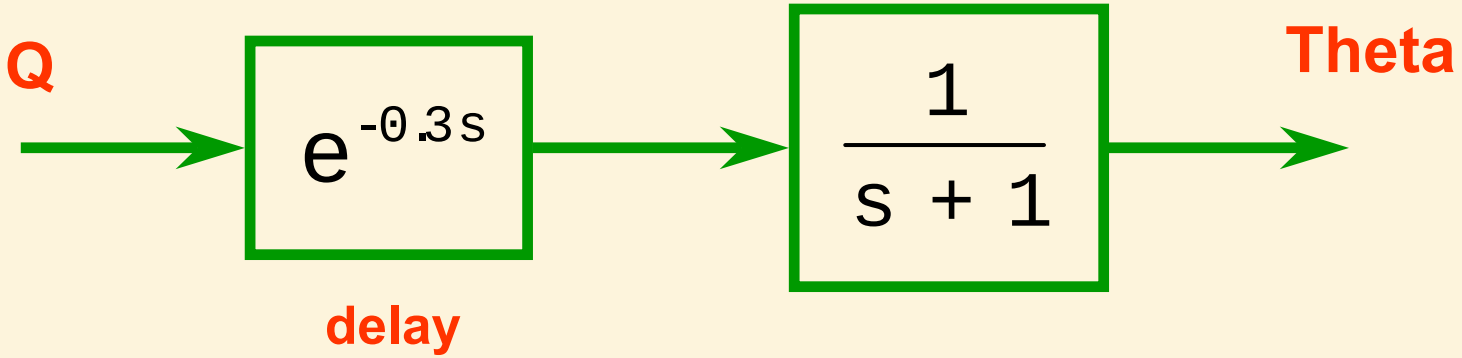
Ts
Td
InputName
OutputName
Notes
Userdata

Or use structure-like syntax sys.num

a, b, c, d, e StateName	z, p, k Variable	num, den Variable
State space	Zero-pole-gain	Transfer function

Copyright © 1984 - 1997 by The MathWorks, Inc.

Example:




```
>> sys = tf(1,[1  1],'Td',0.3)

>> set(sys,'InputName','Q',...
           'OutputName','Theta')

>> set(sys,'Notes','A simple heater model')
```

Copyright © 1984 - 1997 by The MathWorks, Inc.

Check result:

```
>> get(sys)
        num = {[0 1]}
        den = {[1 1]}
        Variable = 's'
        Ts = 0
        Td = 0.3
        InputName = {'Q'}
        OutputName = {'Theta'}
        Notes = {'A simple heater model'}
        UserData = [ ]
```

Copyright © 1984 - 1997 by The MathWorks, Inc.

LTI Objects: Benefits

- **Manipulate model as single MATLAB variable**
- **Overloaded arithmetic:**

```
sys1+sys2,  sys1*sys2,  [sys1;sys2],…
```

- **Fewer commands**
- **Simpler syntax**
- **In short, more power with less effort!**

Copyright © 1984 - 1997 by The MathWorks, Inc.

Example: Bode response

Before

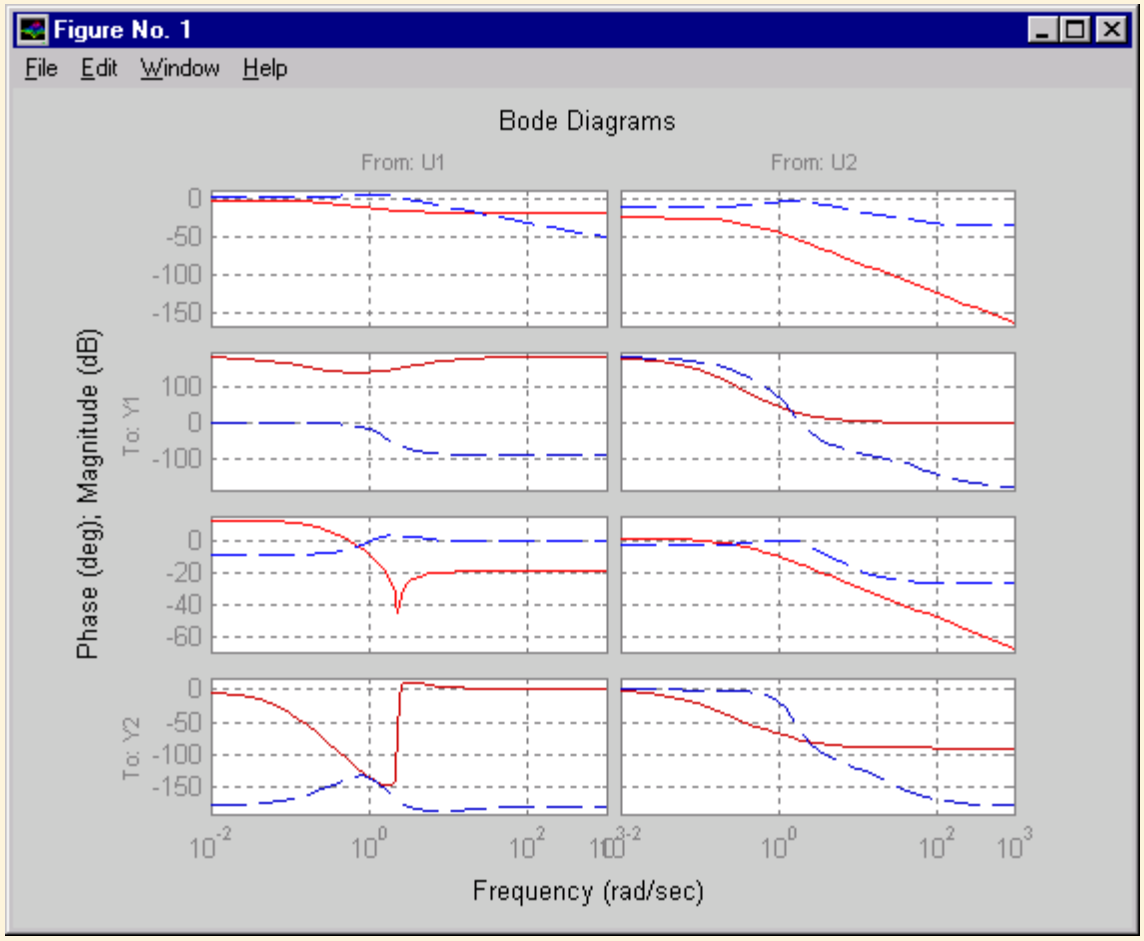
```
bode(num,den)
bode(a,b,c,d,iu)
dbode(num,den,Ts)
dbode(a,b,c,d,Ts,iu)
```

Now

```
bode(sys)
```

Copyright © 1984 - 1997 by The MathWorks, Inc.

```
>> bode(sys1,'r',sys2,'b--')
```



Copyright © 1984 - 1997 by The MathWorks, Inc.

New GUIs:
Root Locus Design GUI

Allows you to interactively:

- *Specify and modify compensator gain and pole/zero locations*
- *View changes in closed-loop system responses*
- *Zoom into sections of the root locus*
- *Designate boundaries and constraints on the root locus (e.g. maximum damping ratio)*

Copyright © 1984 - 1997 by The MathWorks, Inc.

New GUIs: LTI Viewer

Integrated tool for model analysis:

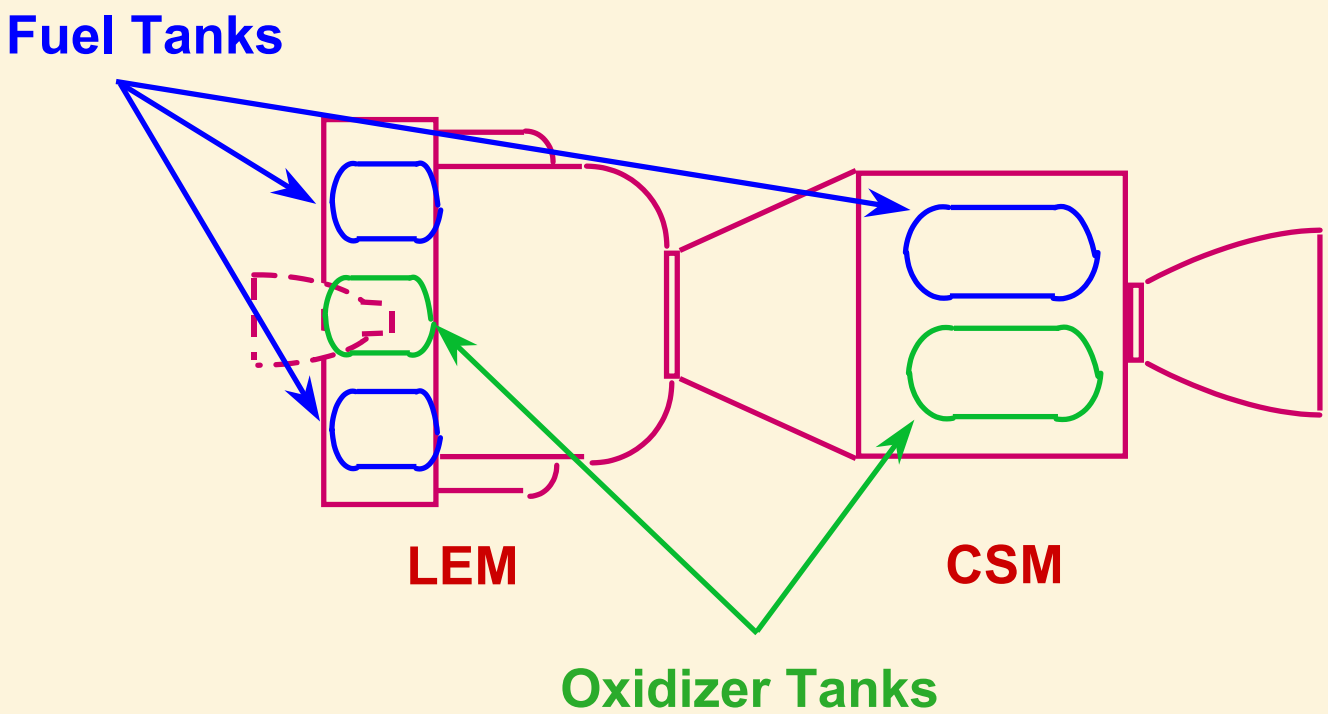
- *All response types in single GUI*
- *Simultaneous analysis of multiple systems*
- *Customizable MIMO plots*
- *Zooming for finer analysis*
- *Response characteristics (e.g. settling time)*

NEW! SIMULINK-based Viewer

Copyright © 1984 - 1997 by The MathWorks, Inc.

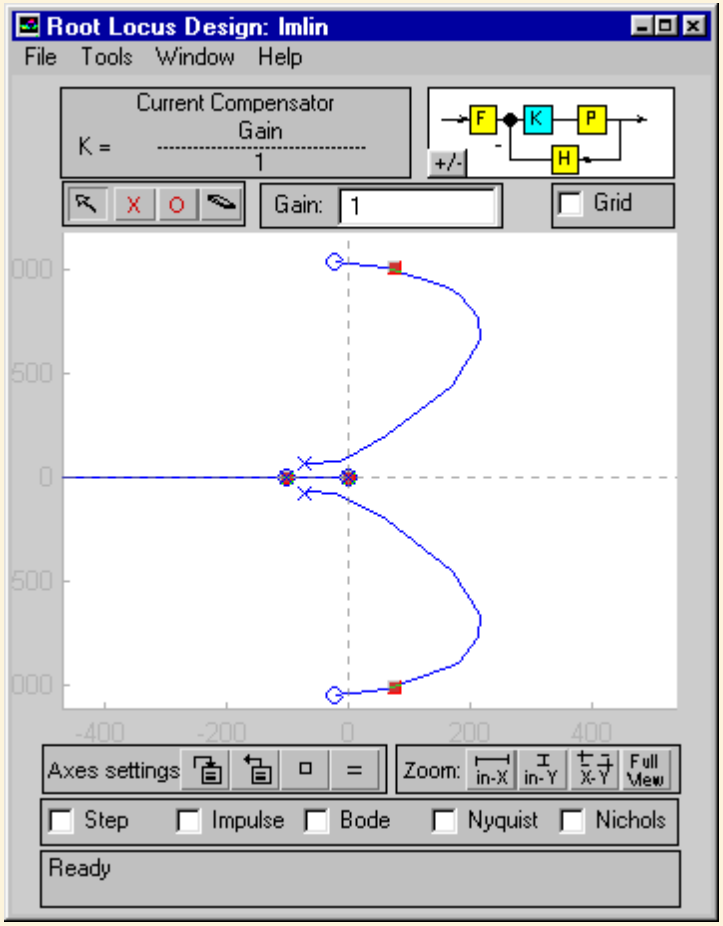
LEM Demo

Purpose: Stabilize fuel sloshing modes



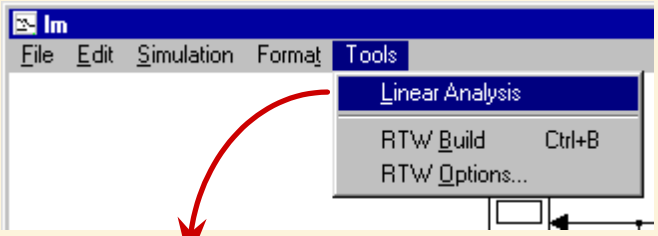
Copyright © 1984 - 1997 by The MathWorks, Inc.

Root Locus GUI



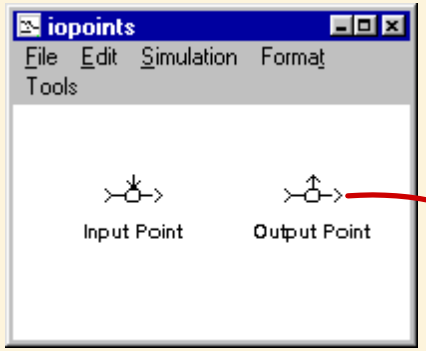
Copyright © 1984 - 1997 by The MathWorks, Inc.

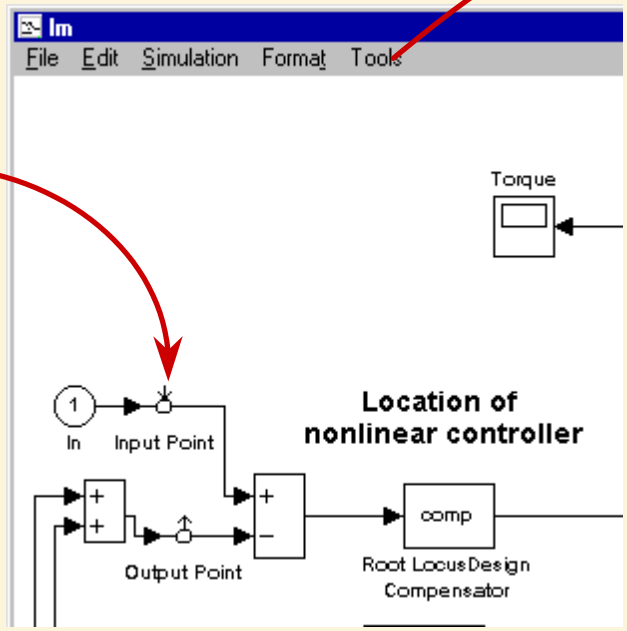
Simulink LTI Viewer

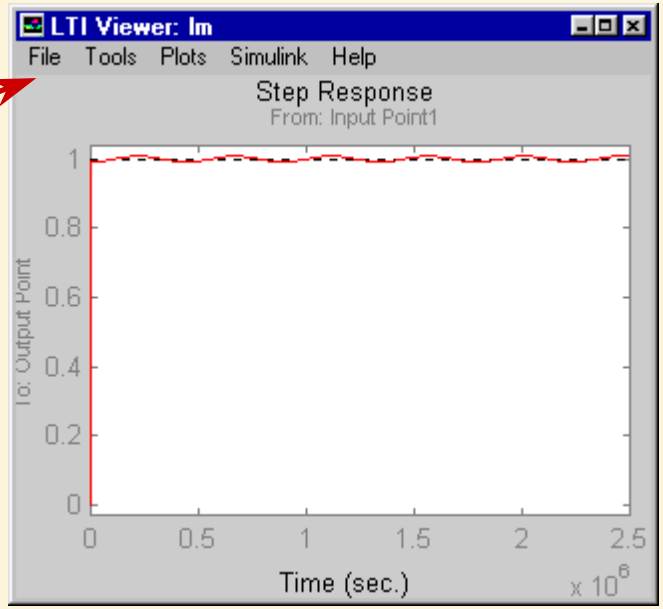


Drag-and-drop Input/Outputs blocks

LTI Viewer linked to Simulink diagram







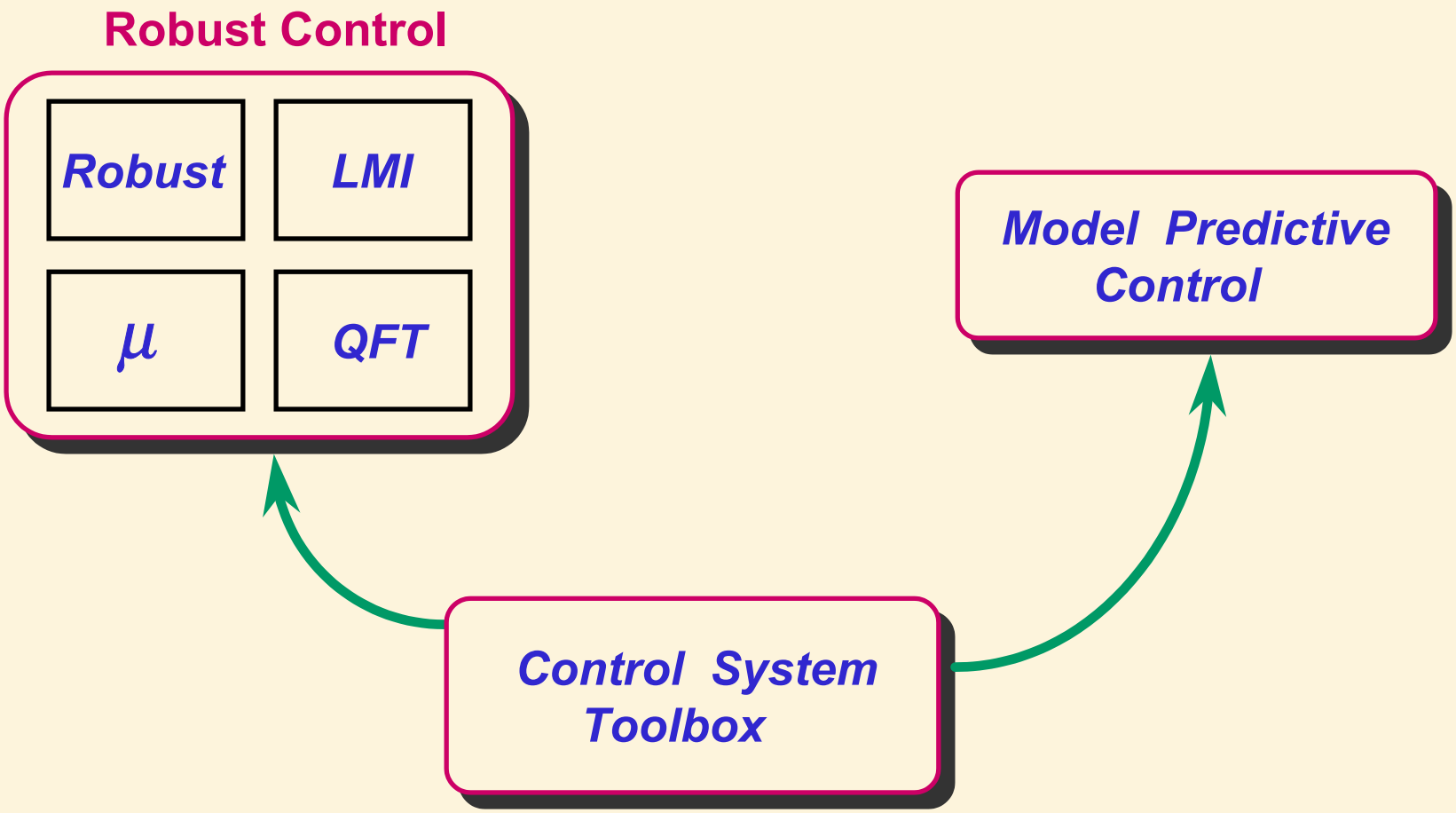
Copyright © 1984 - 1997 by The MathWorks, Inc.

Outline

- The Control Design Process
- Control System Toolbox 4.1
- Advanced Control Toolboxes
- Future Directions

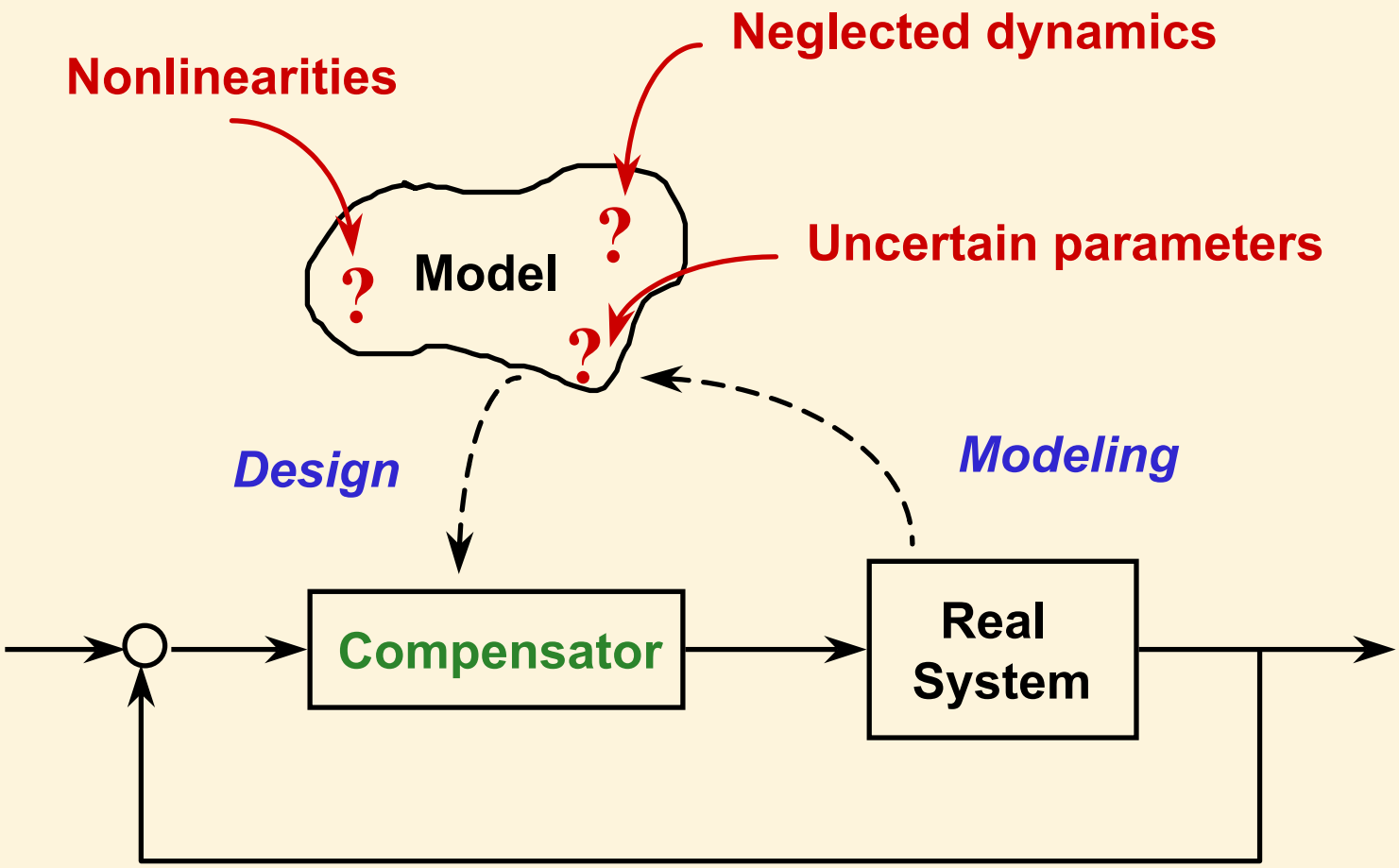
Copyright © 1984 - 1997 by The MathWorks, Inc.

Advanced Control Toolboxes



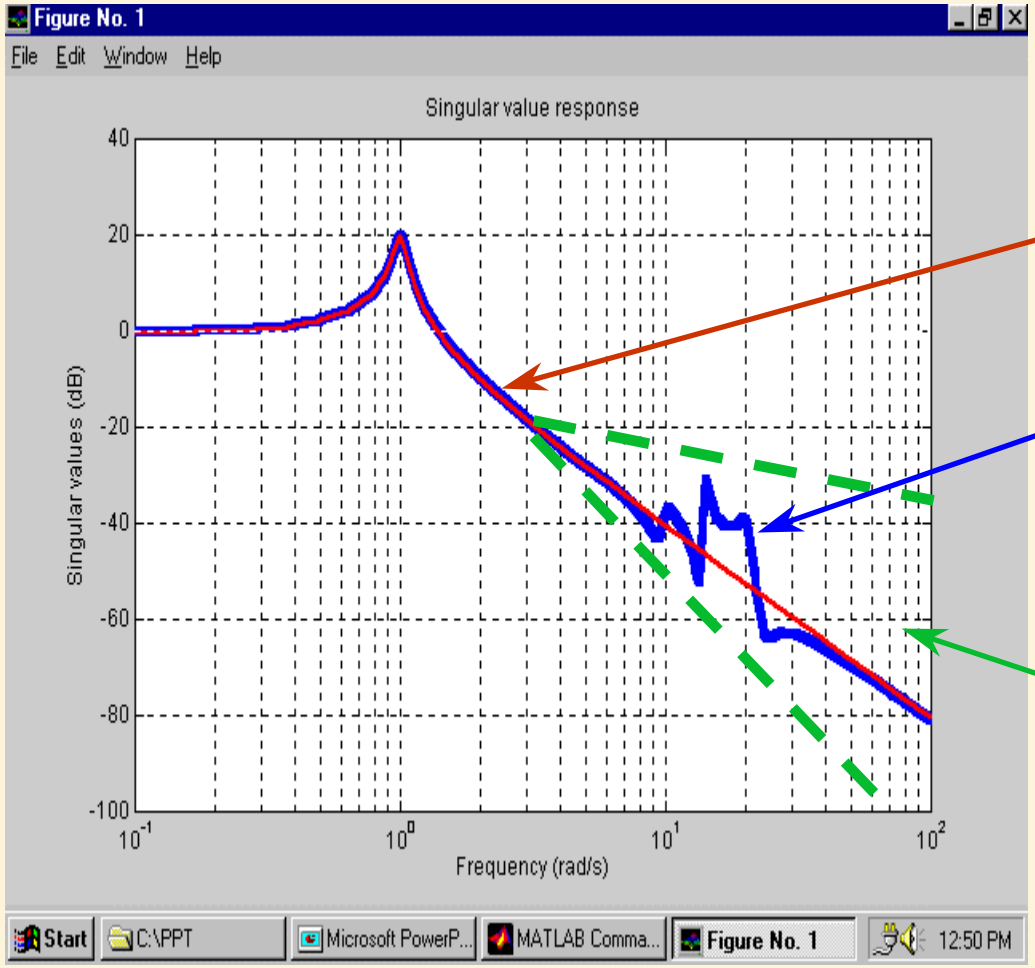
Copyright © 1984 - 1997 by The MathWorks, Inc.

Robust Control: Why?



Copyright © 1984 - 1997 by The MathWorks, Inc.

Example



nominal model (2nd order)

true model

model uncertainty

Copyright © 1984 - 1997 by The MathWorks, Inc.

Robust Control: Goals

- **Take into account model uncertainty**
- **Incorporate knowledge about the unknown**
- **Quantify and optimize trade-off between performance and robustness**

using:

- *Modern analysis techniques*
- *Multivariable design tools*

Copyright © 1984 - 1997 by The MathWorks, Inc.

Related Toolboxes

- **Robust Control Toolbox**
  - *SISO and MIMO Loop Shaping*
  - *H-infinity synthesis*
  - *Model order reduction*

- **μ–Analysis and Synthesis Toolbox**
  - *μ-Analysis (modern gain/phase margins)*
  - *μ-Synthesis GUI*
  - *Robustness to dynamic + parameter uncertainty*

Copyright © 1984 - 1997 by The MathWorks, Inc.

Related Toolboxes

- **LMI Control Toolbox** (***LMI = Linear Matrix Inequality***)
  - *Multi-objective control*
  - *Gain scheduling*
  - *High-performance LMI solvers*

- **QFT Toolbox** (***QFT = Quantitative Feedback Theory***)
  - *Classical "one-loop-at-a-time" methodology*
  - *Interactive loop shaping in Nichols plane*
  - *Low-order compensators*

Copyright © 1984 - 1997 by The MathWorks, Inc.

Sample Application: Herbst Maneuver for F-18 Hornet

- F18 Hornet with "thrust vectoring" engine nozzles
- Post-stall maneuvering
- High-alpha flight
- Uncertain dynamics

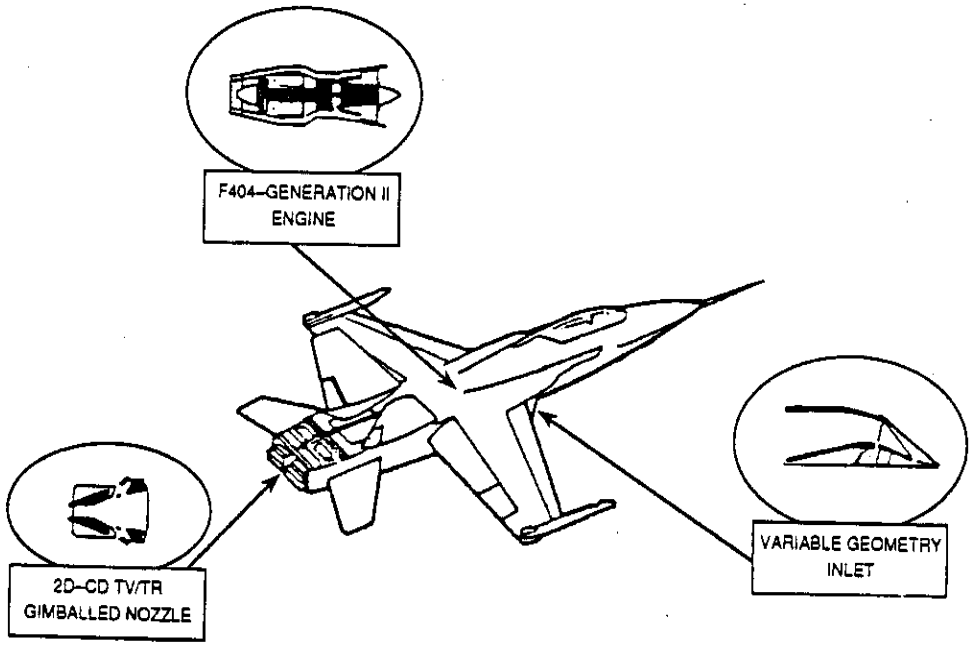


Figure 1: Supermaneuverable fighter.

Herbst Maneuver

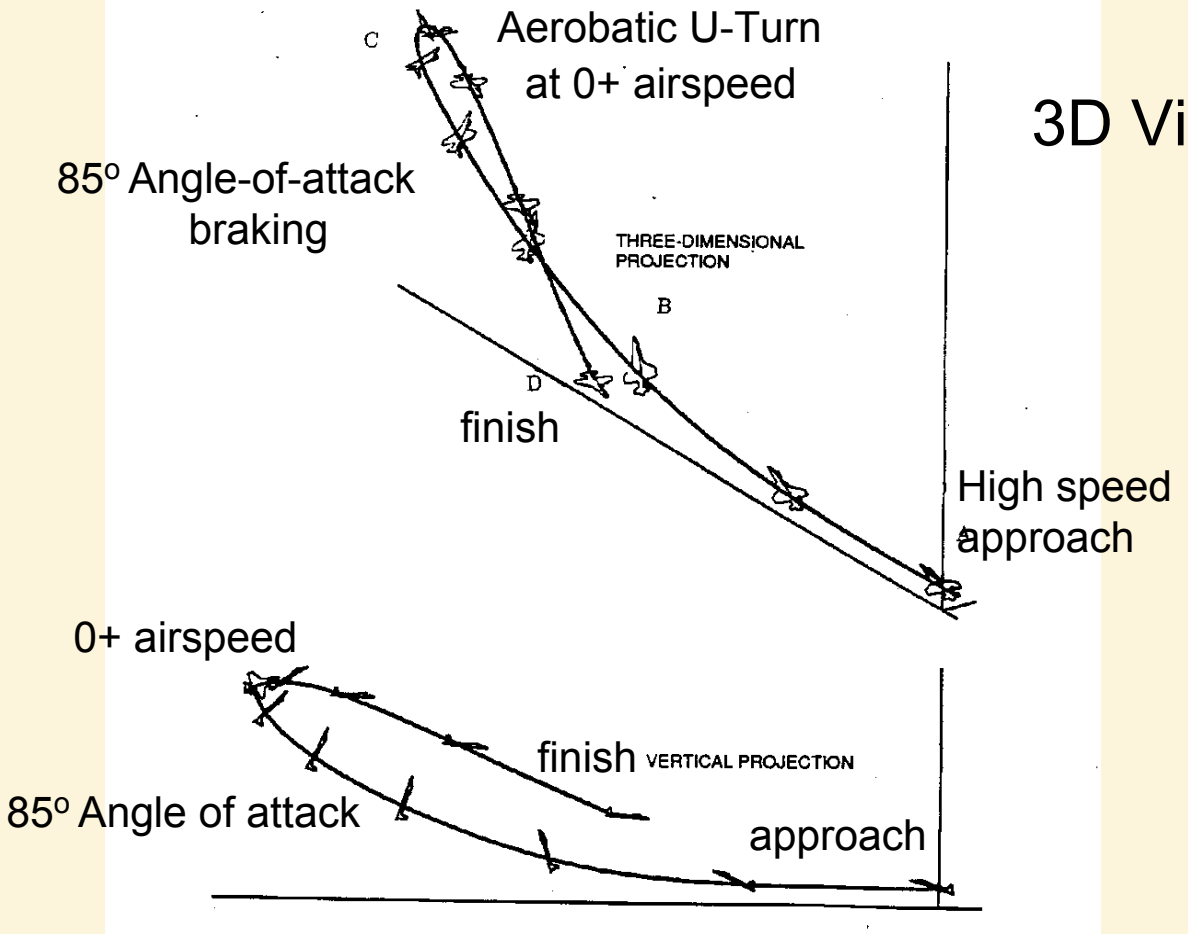


Figure 2: Herbst-Type maneuver.

7 by The MathWorks, Inc.

Herbst Maneuver: Robust Design

- **Design with RCT:**
  - *H^∞ Loop-Shaping design*
  - *MIMO controller*
- **H^∞ controller flies F18 Hornet through the entire "Herbst Maneuver"**
- **Robust high-precision control despite extreme nonlinearity & uncertain dynamics**

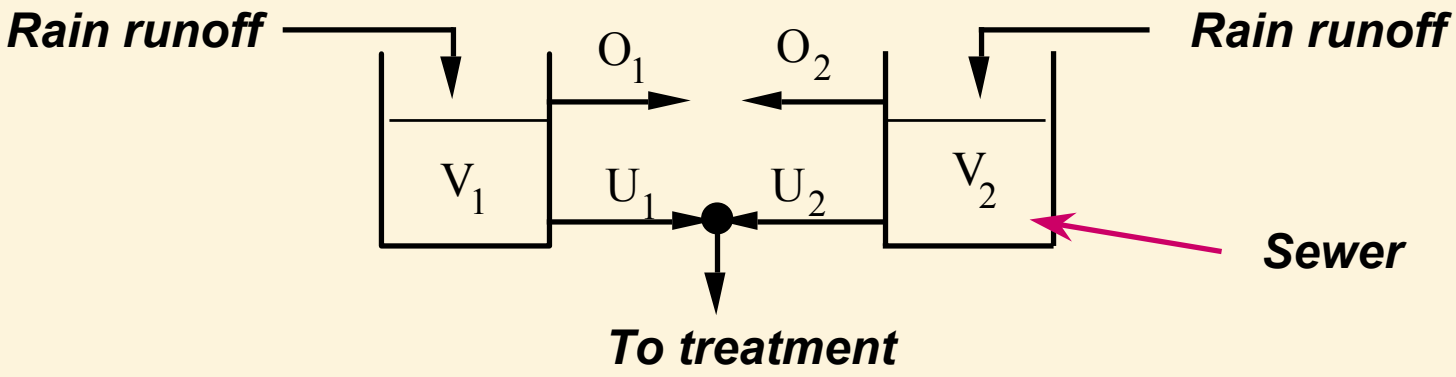
Copyright © 1984 - 1997 by The MathWorks, Inc.

Model Predictive Control Toolbox

- Ideal for Process Control
- On-line optimization of control law
- Can handle saturations and delays

Copyright © 1984 - 1997 by The MathWorks, Inc.

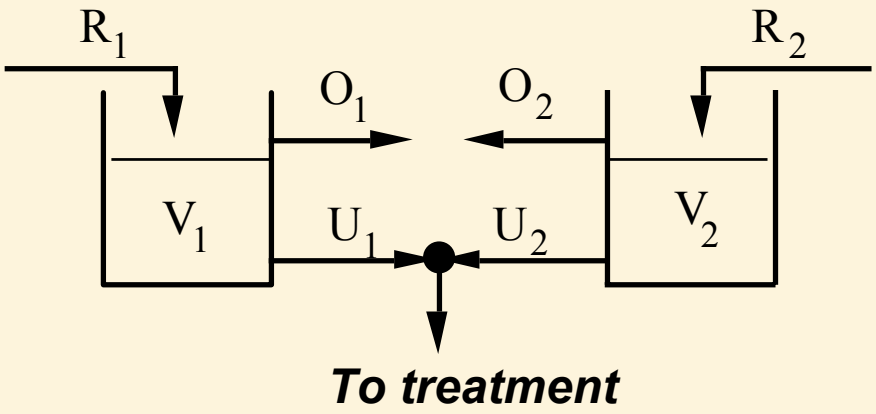
Sample Application: Minimization of Sewer Overflows



U = Controlled flow to treatment plant

O = Overflow of raw sewage to environment

Copyright © 1984 - 1997 by The MathWorks, Inc.



MPC design specs:

- For predicted runoff R, adjust U to minimize cumulative overflows O
- Hard bounds on U, V, and total flow to treatment plant

(Installed in Seattle in 1993, adjusts flows at 23 locations)

Copyright © 1984 - 1997 by The MathWorks, Inc.

Outline

- The Control Design Process
- Control System Toolbox 4.1
- Advanced Control Toolboxes
- Future Directions

Copyright © 1984 - 1997 by The MathWorks, Inc.

Future Enhancements

- Greater integration with Simulink
- More GUIs for analysis and design
- Tighter integration among Control Toolboxes

Copyright © 1984 - 1997 by The MathWorks, Inc.