ActuatorSelection_Test3_2

October 27, 2021

1 Benefit of Multiplicative (MPL) Models over Nominal (Nom) Models of Systems

Testing actuator selection and feedback of Nominal and MPL models on simulations of True system

Py Packages

```
[1]: import numpy as np
     from copy import deepcopy as dc
     # %matplotlib widget

     from functionfile_system_definition import sys_from_file, system_display_matrix
     from functionfile_system_mplcost import simulation_nom_vs_mpl,␣
      ↪plot_simulation_nom_vs_mpl_1, plot_simulation_nom_vs_mpl_2,␣
      ↪actuator_comparison
```

1.1 Code

```
[2]: test_set = 'System Model 4'
     S_True = sys_from_file(test_set + ' True')
     S_MPL = sys_from_file(test_set + ' MPL')
     S_Nom = sys_from_file(test_set + ' Nominal')
```

```
System read from file @ system_model/System Model 4 True.pickle

System read from file @ system_model/System Model 4 MPL.pickle

System read from file @ system_model/System Model 4 Nominal.pickle
```

```
[3]: ret_sim = simulation_nom_vs_mpl(S_Nom, S_MPL, S_True)
```
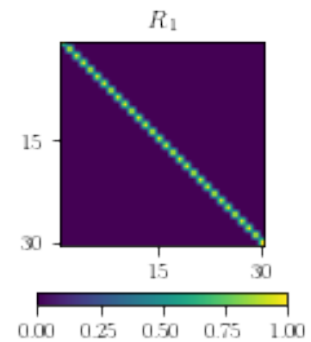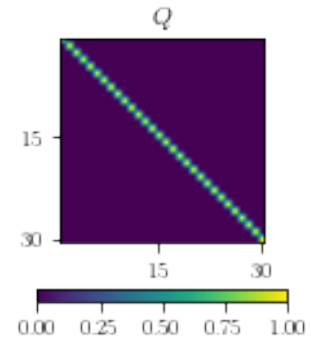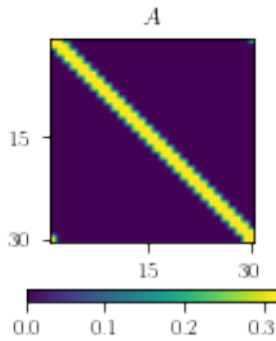
1.2 Output

System Models

True System

```
[4]:
```

```
print('max(abs(eigvals(A)))= %.4f' % (np.max(np.abs(np.linalg.
 ↪eigvals(S_True['A'])))))
system_display_matrix(S_True)
```

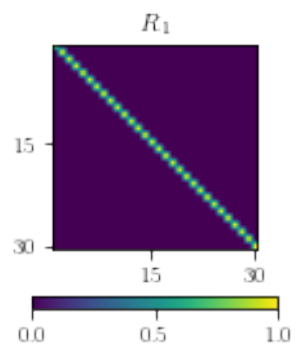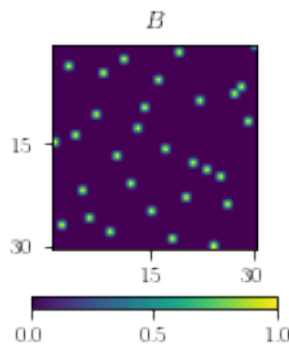max(abs(eigvals(A)))= 0.9500
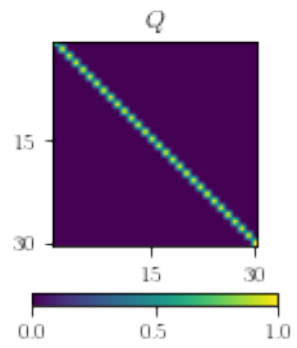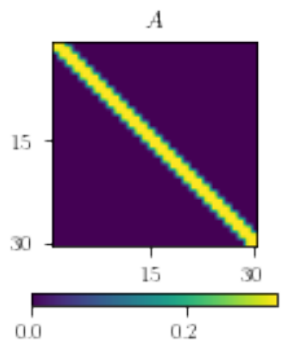

System Model 4 True

Nominal Model

```
[5]: print('max(abs(eigvals(A)))= %.4f' % (np.max(np.abs(np.linalg.
      ↪eigvals(ret_sim['system_nom']['A'])))))
     system_display_matrix(ret_sim['system_nom'])
```

max(abs(eigvals(A)))= 0.9500

System Model 4 Nominal

A

Q

B

R_1

Multiplicative Noise Model

```
[6]: print('max(abs(eigvals(A)))= %.4f' % (np.max(np.abs(np.linalg.
     ↪eigvals(ret_sim['system_mpl']['A']))))))
     system_display_matrix(ret_sim['system_mpl'])
```
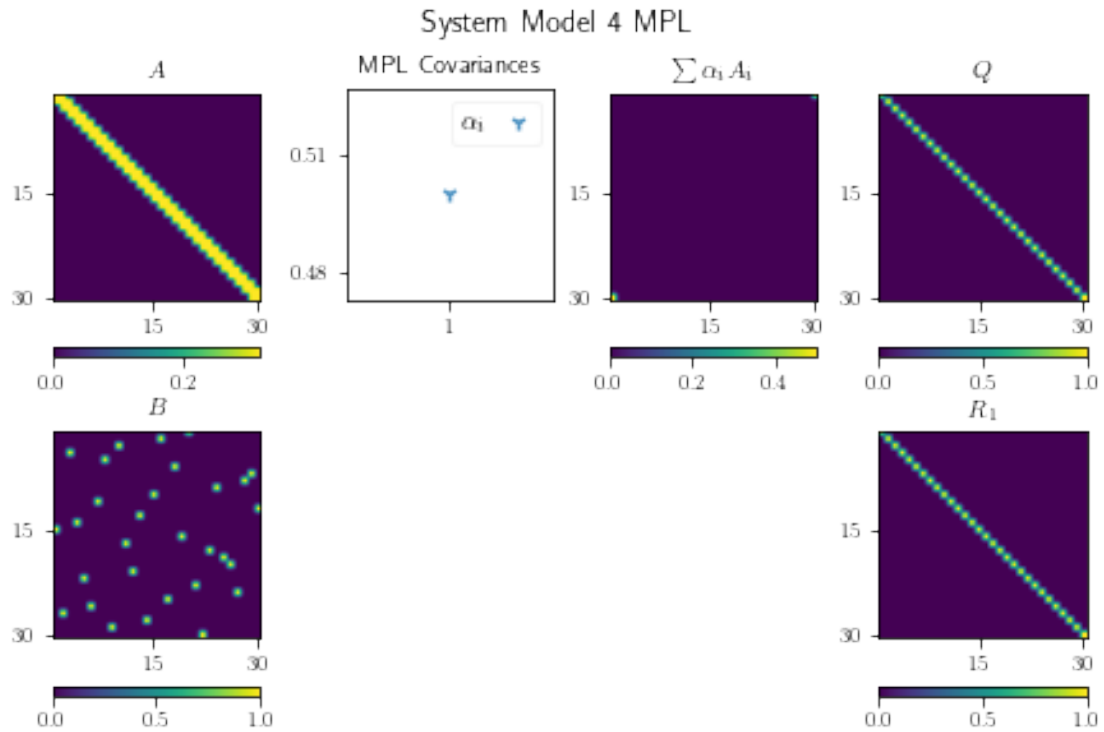
```
max(abs(eigvals(A)))= 0.9500
```

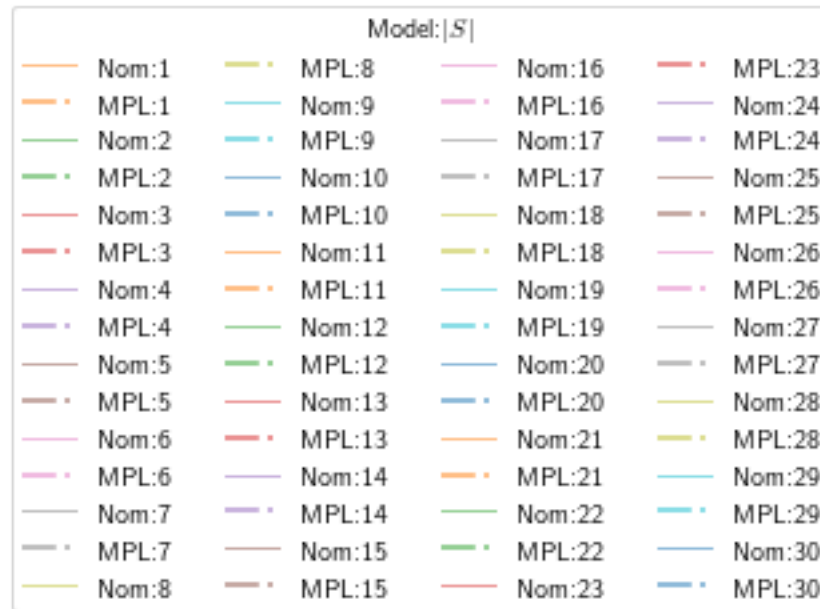**Simulation - Trajectory, Control Input and Costs**

```
[7]: plot_simulation_nom_vs_mpl_1(ret_sim)
```

C:\Users\kxg161630\Anaconda3\envs\work\lib\site-
packages\IPython\core\pylabtools.py:137: UserWarning: constrained_layout not
applied because axes sizes collapsed to zero. Try making figure larger or axes
decorations smaller.
  fig.canvas.print_figure(bytes_io, **kw)
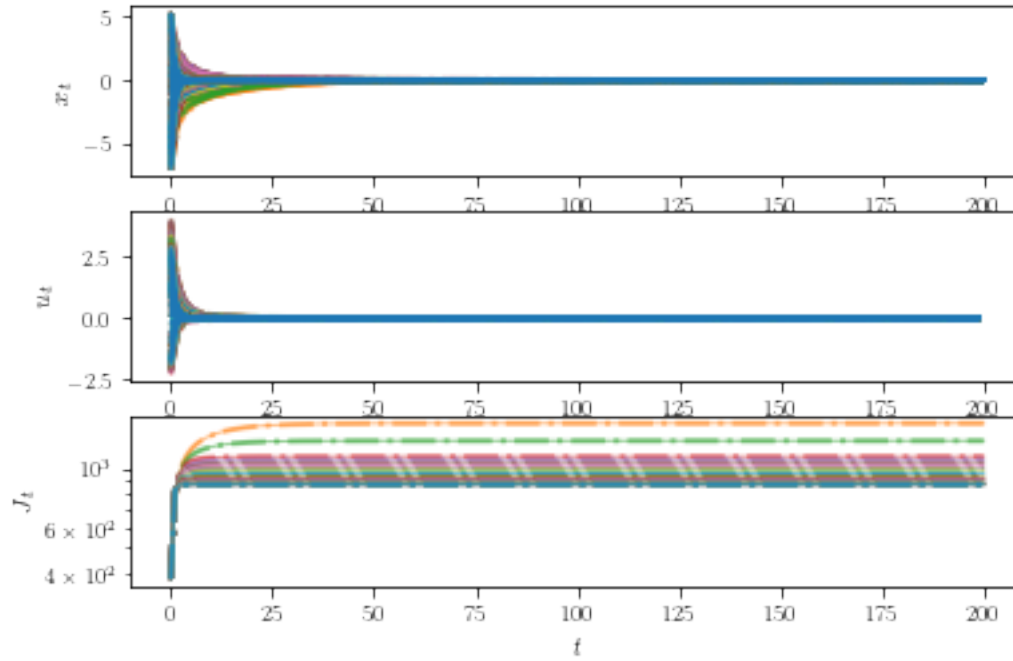
Actuator set comparison

```
[8]: actuator_comparison(ret_sim['system_mpl'], ret_sim['system_nom']);
```

Control sets are different

Actuator Set (B) comparison

Simulated costs

```
[9]: print('True simulation cost with <___> feedback (4decimal approx)')
     for key in ret_sim['T_Nom']['costs']:
         print ("|S|: %s | Nom: %.4f | MPL: %.4f | Diff (Nom-MPL) %.4f (%.4f %% of␣
     ↪Nom)" % (key, ret_sim['T_Nom']['costs'][key][-1],␣
     ↪ret_sim['T_MPL']['costs'][key][-1],␣
     ↪ret_sim['T_Nom']['costs'][key][-1]-ret_sim['T_MPL']['costs'][key][-1],␣
     ↪(ret_sim['T_Nom']['costs'][key][-1]-ret_sim['T_MPL']['costs'][key][-1])*100/
     ↪ret_sim['T_Nom']['costs'][key][-1]))
```
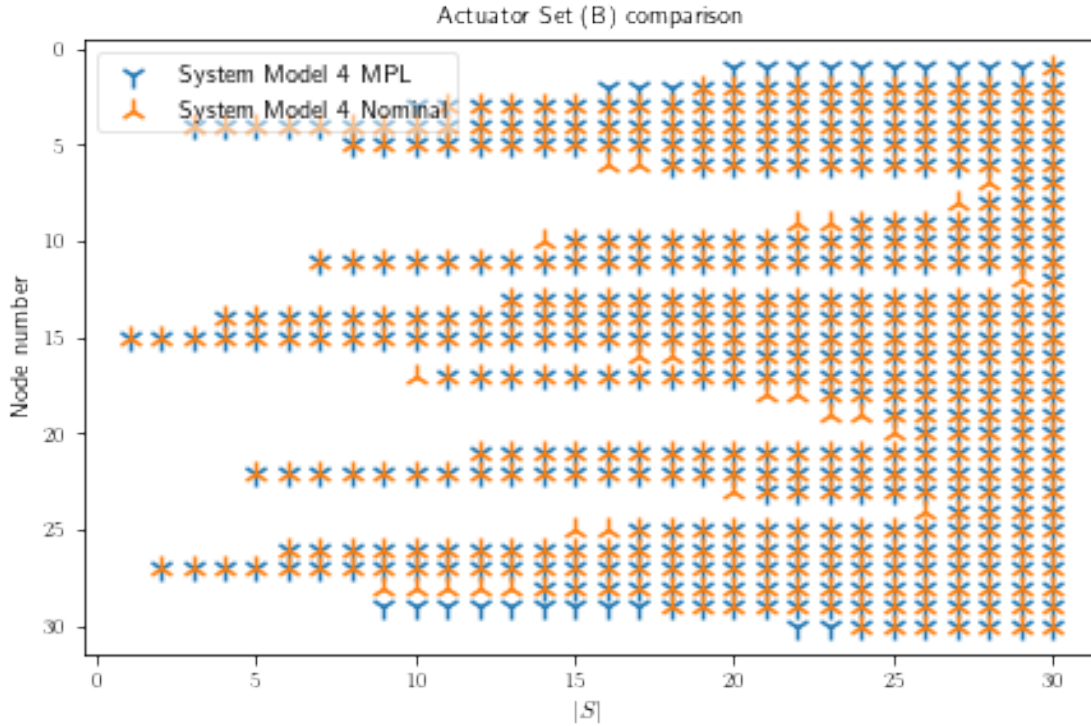
```
True simulation cost with <___> feedback (4decimal approx)
|S|: 1 | Nom: 1492.7694 | MPL: 1492.7694 | Diff (Nom-MPL) -0.0000 (-0.0000 % of
Nom)
|S|: 2 | Nom: 1280.8156 | MPL: 1280.5066 | Diff (Nom-MPL) 0.3089 (0.0241 % of
Nom)
|S|: 3 | Nom: 1114.1547 | MPL: 1113.9239 | Diff (Nom-MPL) 0.2308 (0.0207 % of
Nom)
|S|: 4 | Nom: 1083.0782 | MPL: 1082.8451 | Diff (Nom-MPL) 0.2331 (0.0215 % of
Nom)
|S|: 5 | Nom: 1052.9252 | MPL: 1052.7126 | Diff (Nom-MPL) 0.2126 (0.0202 % of
Nom)
|S|: 6 | Nom: 1028.6757 | MPL: 1028.4202 | Diff (Nom-MPL) 0.2555 (0.0248 % of
Nom)
```

```
|S|: 7 | Nom: 1005.4750 | MPL: 1005.2309 | Diff (Nom-MPL) 0.2440 (0.0243 % of
Nom)
|S|: 8 | Nom: 987.1752 | MPL: 986.8858 | Diff (Nom-MPL) 0.2894 (0.0293 % of Nom)
|S|: 9 | Nom: 965.1240 | MPL: 960.2865 | Diff (Nom-MPL) 4.8375 (0.5012 % of Nom)
|S|: 10 | Nom: 952.3236 | MPL: 943.5963 | Diff (Nom-MPL) 8.7273 (0.9164 % of
Nom)
|S|: 11 | Nom: 933.1954 | MPL: 930.7929 | Diff (Nom-MPL) 2.4025 (0.2574 % of
Nom)
|S|: 12 | Nom: 923.9573 | MPL: 921.5569 | Diff (Nom-MPL) 2.4004 (0.2598 % of
Nom)
|S|: 13 | Nom: 916.1615 | MPL: 913.7610 | Diff (Nom-MPL) 2.4004 (0.2620 % of
Nom)
|S|: 14 | Nom: 910.0392 | MPL: 907.1621 | Diff (Nom-MPL) 2.8772 (0.3162 % of
Nom)
|S|: 15 | Nom: 904.6858 | MPL: 901.0400 | Diff (Nom-MPL) 3.6458 (0.4030 % of
Nom)
|S|: 16 | Nom: 899.7913 | MPL: 894.9418 | Diff (Nom-MPL) 4.8495 (0.5390 % of
Nom)
|S|: 17 | Nom: 895.0838 | MPL: 889.5428 | Diff (Nom-MPL) 5.5410 (0.6190 % of
Nom)
|S|: 18 | Nom: 886.2789 | MPL: 884.6043 | Diff (Nom-MPL) 1.6746 (0.1889 % of
Nom)
|S|: 19 | Nom: 880.0418 | MPL: 879.8968 | Diff (Nom-MPL) 0.1450 (0.0165 % of
Nom)
|S|: 20 | Nom: 877.8384 | MPL: 878.8924 | Diff (Nom-MPL) -1.0540 (-0.1201 % of
Nom)
|S|: 21 | Nom: 875.9573 | MPL: 876.6888 | Diff (Nom-MPL) -0.7315 (-0.0835 % of
Nom)
|S|: 22 | Nom: 874.3454 | MPL: 875.9345 | Diff (Nom-MPL) -1.5891 (-0.1817 % of
Nom)
|S|: 23 | Nom: 873.5669 | MPL: 874.0533 | Diff (Nom-MPL) -0.4864 (-0.0557 % of
Nom)
|S|: 24 | Nom: 872.3160 | MPL: 872.4411 | Diff (Nom-MPL) -0.1250 (-0.0143 % of
Nom)
|S|: 25 | Nom: 871.9599 | MPL: 871.6626 | Diff (Nom-MPL) 0.2974 (0.0341 % of
Nom)
|S|: 26 | Nom: 871.8029 | MPL: 871.3065 | Diff (Nom-MPL) 0.4964 (0.0569 % of
Nom)
|S|: 27 | Nom: 871.7061 | MPL: 871.1495 | Diff (Nom-MPL) 0.5566 (0.0639 % of
Nom)
|S|: 28 | Nom: 871.6184 | MPL: 871.0526 | Diff (Nom-MPL) 0.5658 (0.0649 % of
Nom)
|S|: 29 | Nom: 871.5365 | MPL: 870.9650 | Diff (Nom-MPL) 0.5715 (0.0656 % of
Nom)
|S|: 30 | Nom: 870.8460 | MPL: 870.8832 | Diff (Nom-MPL) -0.0372 (-0.0043 % of
Nom)
```

```
[10]: plot_simulation_nom_vs_mpl_2(ret_sim)
```
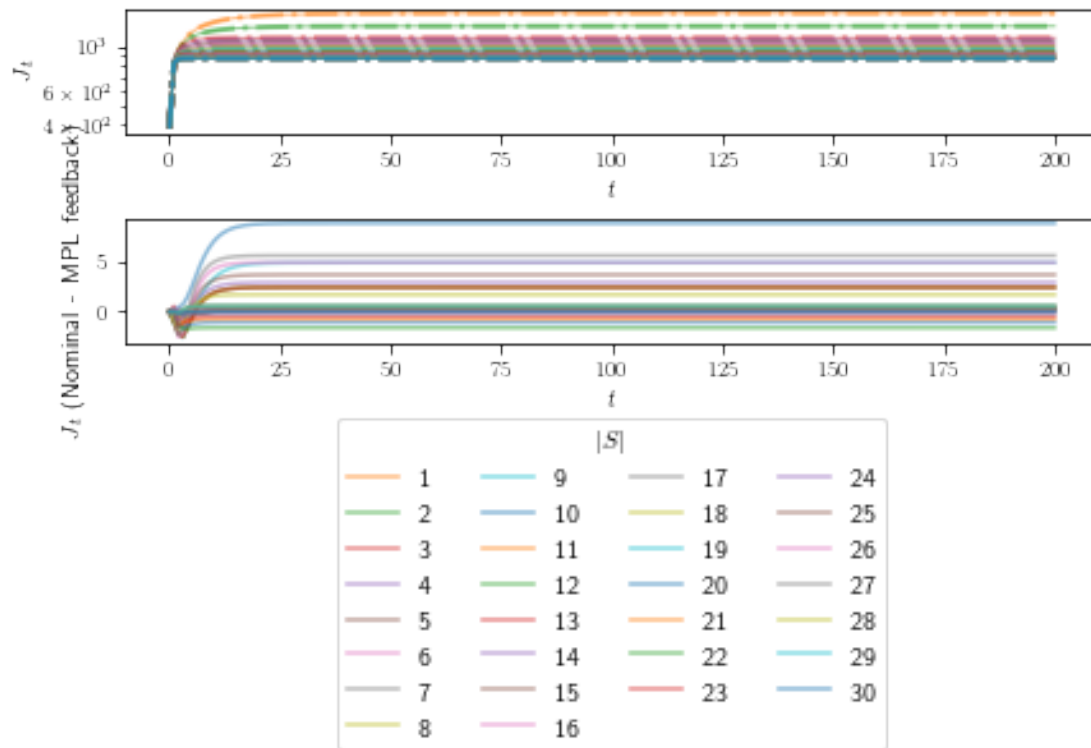

1.3 Run Complete

```
[11]: print('Run Complete')
```

```
Run Complete
```