PROJECT REPORT

Nonlinear Systems Evaluation Using Fast Orthogonal Search

Non-Linear Systems: Analysis and Identification

(ELEC 841)

Submitted to: Prof. M. J. Korenberg

Submitted by

Vidhan Dholakia

(20151326)

(18vbd@queensu.ca)

CONTENT:

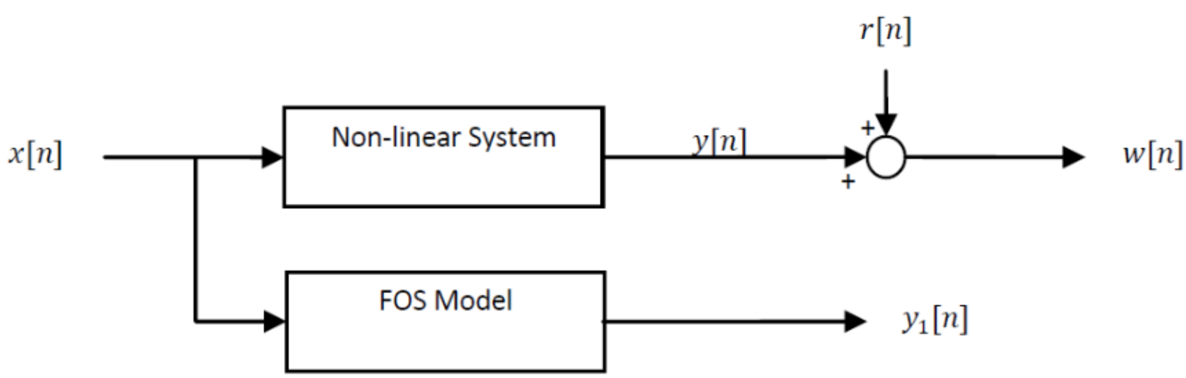
OVERVIEW:

The primary goal is to analyse the performance of three non-linear differential equations across a range of 3000 data points and to use Fast Orthogonal Search (FOS) to achieve an approximation for each of the three calculations with the most accurate model. In this project, we used the Fast-Orthogonal Search Algorithm to assess 3 sets of non-linear second-order systems. We identify the top-performing model for all of the systems under various circumstances. We subject the systems to different levels of noise, i.e. 0%, 25%, 50%, and 100%. We have used MATLAB to test the systems.

INTRODUCTION:

Fast Orthogonal Search (FOS) is an algorithm designed and developed by Queen's University Prof. M.J Korenberg for the detection and examination of discrete system models. As shown below, the generic nonlinear system used for this project is:

y(n) = F[y(n-1) ,……, y(n-k),x(n),……,x(n-L)]



(Reference : https://www.researchgate.net/publication/20419817_Applications_of_Fast_Orthogonal_Search_Time-Series_Analysis_and_Resolution_of_Signals_in_Noise)

Here, input to the system is x(n) and output is y(n). The above equation is a discrete Volterra Series of time-limited memory whose kernels should be assessed at the point where F is a function of only input values.

In addition, in cases where F is a function of both input and output values, the above equation is a non-linear differential equation whose coefficients are then analysed. In this project, we make a comparison of the top-performing model's Mean Square Error to the Ideal MSE for calculating the functioning of the overall project.

Analysis & Comparisons of the SYSTEMS:

SYSTEM 1

y(n)= 1.2 + 0.124*y(n-1) + 0.2*x(n-2)*x(n-3) - 0.26*x(n-1)*y(n-2) ...- 1.03*x(n) - 0.03*y(n-2) *y(n-1);

RESULTS:

Best Model 6 (when Noise is 0%)

y[n]=1.4243+(-1.0006)[n]+(-0.19769)y[n-1]x[n-2]+(0.18297)x[n-1]x[n-2]+(-0.044839)y[n-1]y[n-3]+(-0.047351)x[n-2]x[n-5]+(-0.03849)y[n-5]x[n-2]+(-0.1802)x[n-2]+(-0.010691)x[n-4]x[n-5]+(-0.011683)x[n-2]x[n-4]+(-0.01014)y[n-4]x[n-2]+(-0.013506)y[n-2]x[n-4]+(-0.018821)y[n- 2]y[n- 4]+(0.058696)x[n- 3]x[n- 4]+(-0.0004073)y[n- 5]x[n- 6]+(-0.0027642)y[n-1]x[n-3]+(-0.028066)x[n-4]x[n-4]+(-0.0068583)y[n-2]x[n-7]+(-0.032285)y[n-4]x[n-4]+(-0.0001558)y[n-4]x[n-6]+(0.038381)x[n-4]+(-0.00096461)y[n-5]x[n-3]+(0.18402)y[n-2]+(-0.024992)y[n-3]x[n-4]+(0.037575)y[n-5]+(0.017881)y[n-4]+(-0.036532)y[n-2]x[n-5]+(0.016149)y[n-3]x[n-6]+(0.014084)x[n-3]x[n-6]+(-0.016731)x[n- 6]+(-0.0070717)y[n- 3]x[n-5]+(-0.02857)y[n-2]y[n-5]+(0.061224)x[n-5]+(0.0020791)y[n-5]x[n-4]+(-0.0057002)x[n-4]x[n-6]+(-0.00054043)y[n-5]y[n-5]+(-0.0038926)x[n-3]x[n-5]+(-0.0088514)x[n-2]x[n-7]+(0.0076673)x[n-7]+(0.016129)y[n-3]y[n-4]+(-0.1366)y[n-1]+(0.01619)y[n-4]x[n-3]+(-0.0079603)x[n-5]x[n-5]+(0.0029058)y[n-5]x[n-7]+(0.0025272)x[n-5]x[n-7]+(0.0018151)x[n-7]x[n-7]+(0.0044315)y[n-1]x[n-5]+(0.0040865)y[n-1]y[n-5]+(-0.0075302)x[n-1]x[n-4]+(-0.0055301)y[n-4]x[n-1]+(0.038929)x[n-2]x[n-3]+(0.00076735)y[n-1]x[n-6]+(0.062397)y[n-3]+(0.0044512)y[n-1]y[n-2]+(-0.12977)x[n-1]+(-0.0053108)x[n-5]x[n-6]+(-0.014224)y[n-2]x[n-3]+(-0.0055941)x[n-6]x[n-7]+(-0.011212)y[n-5]x[n-5]+(0.013852)y[n-3]y[n-5]+(0.0094517)y[n-2]y[n-3]+(0.009057)y[n-3]x[n-2]+(0.010025)y[n-4]x[n-5]+(0.0013845)y[n-3]x[n-7]+(0.0014985)x[n-6]x[n-6]+(-0.0039751)y[n-4]y[n-4]+(0.0027694)y[n-4]y[n-5]+(0.014118)y[n-3]y[n-3]+(-0.025988)y[n- 3]x[n- 3]+(-0.012207)x[n- 3]x[n- 3]+(0.04833)x[n-3]+(0.001068)x[n- 2]x[n- 6]

Best Model 5 (when Noise is 25%)

y[n]=1.1121+(-0.95238)x[n]+(-0.22822)y[n-1]x[n-2]+(0.12911)x[n-1]x[n-2]+(-0.064947)x[n- 1]x[n-3] +(0.066385)x[n- 3]+(0.025072)y[n- 2]x[n-2]

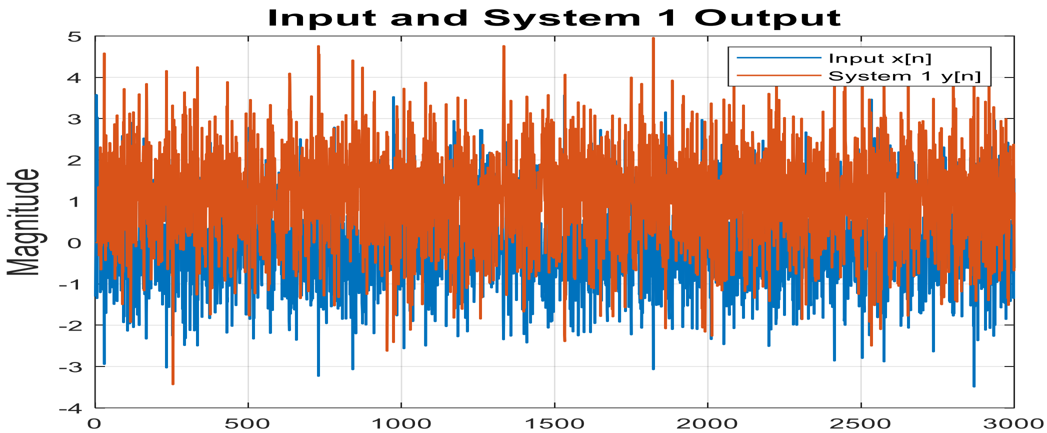
Best Model 4 (when Noise is 50%)

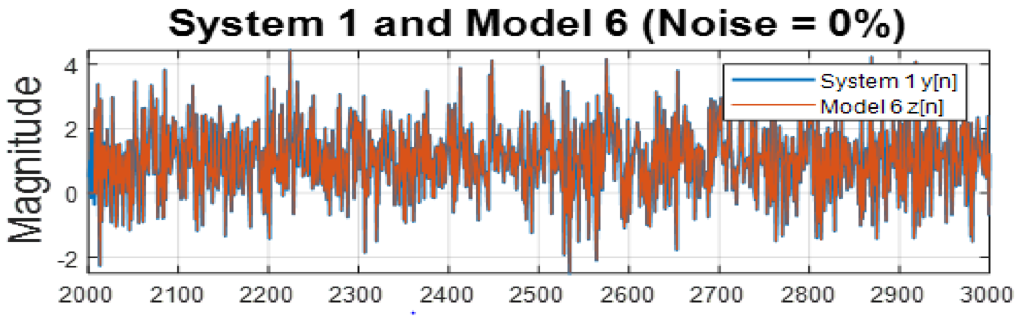
y[n]=1.0754+(-0.9784)x[n]+(-0.12418)y[n-1]x[n-2]+(0.18844)x[n-1]x[n-2]+(-0.10354)x[n-2] +(0.04367)y[n- 3]x[n- 1]+(0.072241)x[n]x[n- 2]+(-0.039672)x[n- 2]x[n- 2]

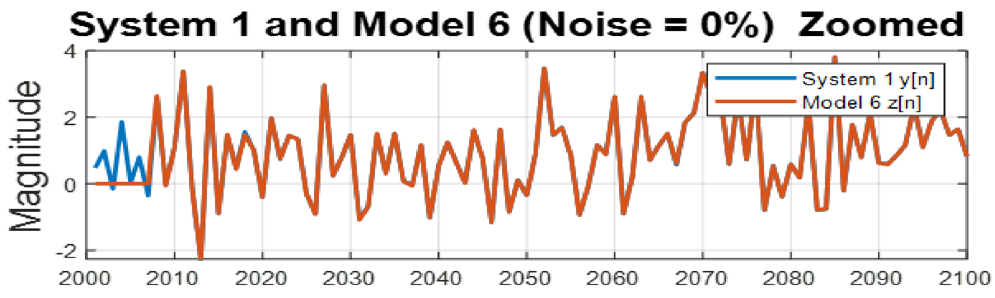
Best Model 2 (when Noise is 100%)

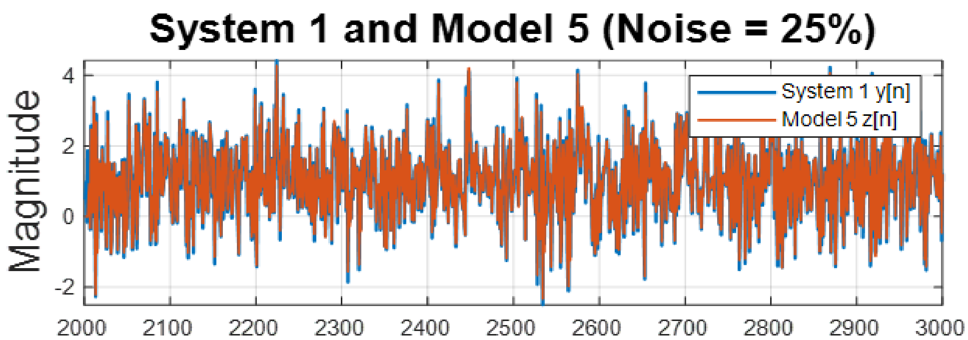
y[n]=1.1709+(-0.95141)x[n] +(-0.089388)y[n- 1]x[n- 2]+(0.27629)x[n- 1]x[n- 2]+(-0.1335)x[n- 2] +(-0.049713)y[n- 4] +(-0.050095)x[n- 2]x[n- 2]

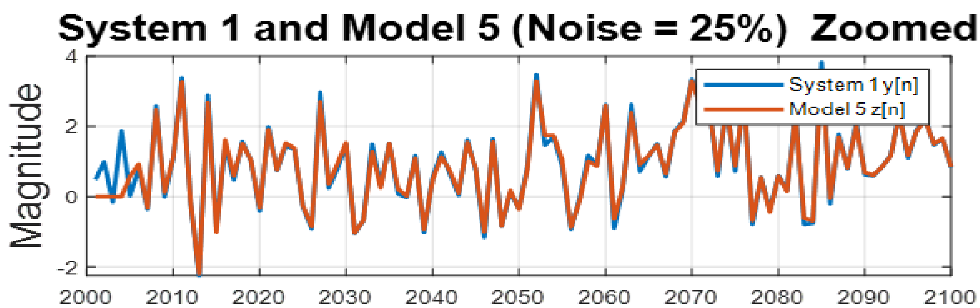
Input & Output

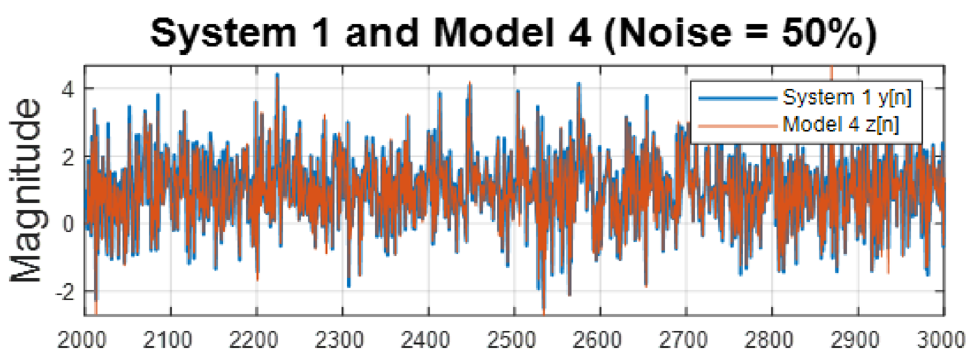


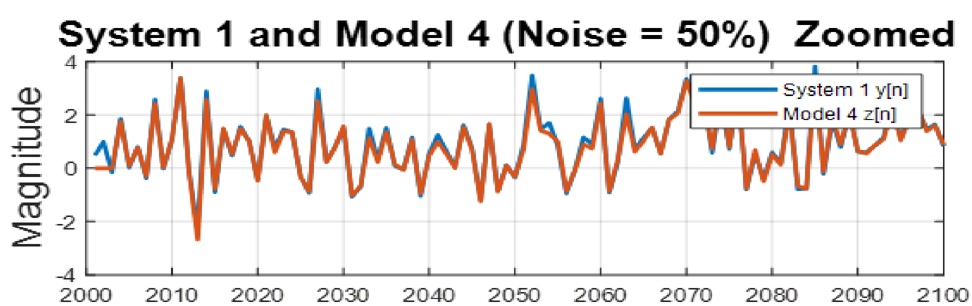


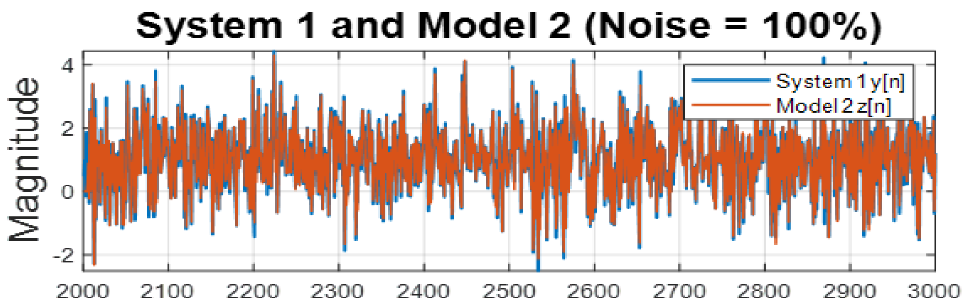


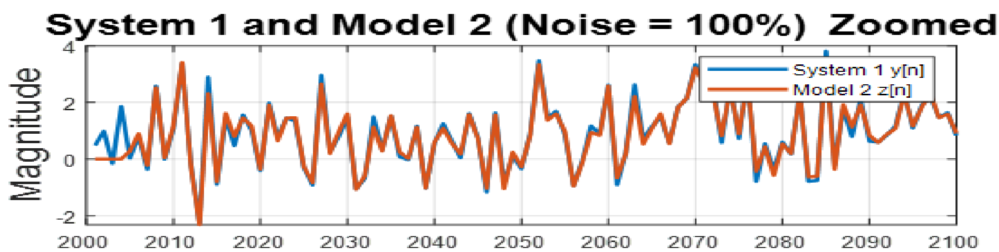

System 1 and Model 5 (Noise = 25%)
Magnitude
System 1 y[n]
Model 5 z[n]
4
2
0
-2
2000
2100
2200
2300
2400
2500
2600
2700
2800
2900
3000


System 1 and Model 5 (Noise = 25%) Zoomed
Magnitude
System 1 y[n]
Model 5 z[n]
4
2
0
-2
2000
2010
2020
2030
2040
2050
2060
2070
2080
2090
2100


System 1 and Model 4 (Noise = 50%)
Magnitude
System 1 y[n]
Model 4 z[n]
4
2
0
-2
2000
2100
2200
2300
2400
2500
2600
2700
2800
2900
3000


System 1 and Model 4 (Noise = 50%) Zoomed
Magnitude
System 1 y[n]
Model 4 z[n]
4
2
0
-2
-4
2000
2010
2020
2030
2040
2050
2060
2070
2080
2090
2100





System 1	Model	Model 1	Model 2	Model3	Model 4	Model 5	Model 6
	K	2	1	2	5	4	8
	L	1	4	1	3	2	3
	Order	2	2	2	2	2	2
Noiseless	Ideal % MSE	0	0	0	0	0	0
	Absolute MSE over Segment 1	0.0783475	0.0137175	0.0783475	0.0015758	0.0017013	1.91E-05
	Percentage MSE over Segment 1	13.17753	2.3077376	13.17753	0.2651302	0.2862192	0.0032206
	Absolute MSE over Segment 2	0.0829685	0.0124261	0.0829685	0.0001347	0.0006763	9.60E-05
	Percentage MSE over Segment 2	12.665577	1.8953289	12.665577	0.0205333	0.103151	0.014597
	Absolute MSE over Segment 3	0	0	0	0	0	0.0005072
	Percentage MSE over Segment 3	0	0	0	0	0	0.0748896
Noise P=25%	Ideal % MSE	20	20	20	20	20	20
	Absolute MSE over Segment 1	0.2430046	0.16595	0.2432091	0.1589566	0.1739589	0.1689893
	Percentage MSE over Segment 1	32.048128	22.522372	32.666363	20.380061	22.052877	22.103193
	Absolute MSE over Segment 2	0.2429215	0.1791157	0.2389039	0.1635376	0.1658437	0.1713185
	Percentage MSE over Segment 2	30.023656	22.522262	30.097161	20.348153	20.172148	21.151026
	Absolute MSE over Segment 3	0	0	0	0.1701467	0	0
	Percentage MSE over Segment 3	0	0	0	20.534546	0	0

Noise P=50%	Ideal % MSE	33.333333	33.333333	33.333333	33.333333	33.333333	33.333333
	Absolute MSE over Segment 1	0.4113164	0.3366392	0.3885216	0.3089431	0.3333934	0.3204254
	Percentage MSE over Segment 1	42.503883	36.508464	41.671738	34.734289	36.187001	34.662083
	Absolute MSE over Segment 2	0.4586157	0.329963	0.4088381	0.3189927	0.330926	0.3554397
	Percentage MSE over Segment 2	44.869097	33.392017	43.38752	32.451166	33.935851	37.179829
	Absolute MSE over Segment 3	0	0	0	0.3370938	0	0
	Percentage MSE over Segment 3	0	0	0	34.017315	0	0
Noise P=100%	Ideal % MSE	50	50	50	50	50	50
	Absolute MSE over Segment 1	0.713191	0.6960616	0.7368265	0.7143913	0.6534855	0.7011039
	Percentage MSE over Segment 1	56.354474	52.867553	57.036521	55.500429	49.774433	53.313235
	Absolute MSE over Segment 2	0.7775184	0.6442713	0.730105	0.6565405	0.6617343	0.6689816
	Percentage MSE over Segment 2	55.29764	51.036318	59.07619	53.528239	52.355873	51.244127
	Absolute MSE over Segment 3	0	0.6491978	0	0	0	0
	Percentage MSE over Segment 3	0	52.51448	0	0	0	0

SYSTEM 2

y(n)= -0.03 + 0.13*x(n-1) - 0.234*x(n) - 1.022*x(n-2)*x(n) - 1.13*y(n-2)... - 0.043*y(n-3) + 2.3*y(n-1)*x(n-2);

RESULTS:

Best Model 3 (when Noise is 0%)

y[n]=-0.019626+(-0.20318)x[n] +(-0.081927)x[n]x[n- 2]+(0.29992)y[n- 1]x[n-2]+(0.040008)x[n- 2]x[n- 2]+(-0.25376)y[n- 2]y[n- 4]+(0.013167)x[n- 1] +(-0.042361)y[n- 3]

Best Model 5 (when Noise is 25%)

y[n]=-0.026147+(-0.20323)x[n]+(-0.078808)x[n]x[n- 2]+(0.045445)x[n- 2]x[n- 2]+(0.13937)y[n- 1]x[n- 2]+(-0.029676)x[n- 1]x[n- 2]+(-0.015065)x[n- 2]x[n- 4]

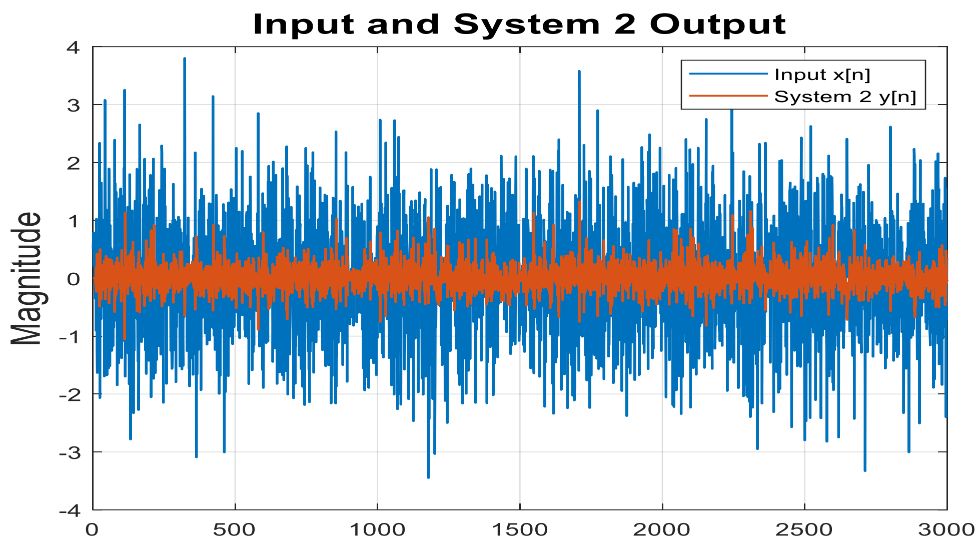
Best Model 1 (when Noise is 50%)

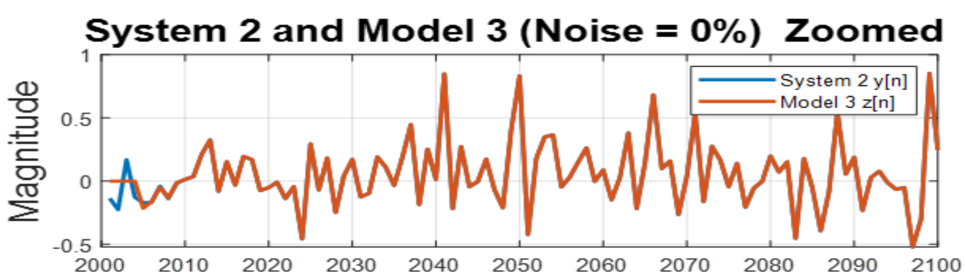
y[n]=-0.025115+(-0.20392)x[n]+(-0.079375)x[n]x[n- 2]+(0.13494)y[n- 1]x[n- 2]+(0.042306)x[n- 2]x[n- 2]+(-0.032126)x[n- 1]x[n- 2]+(0.019577)x[n- 1]

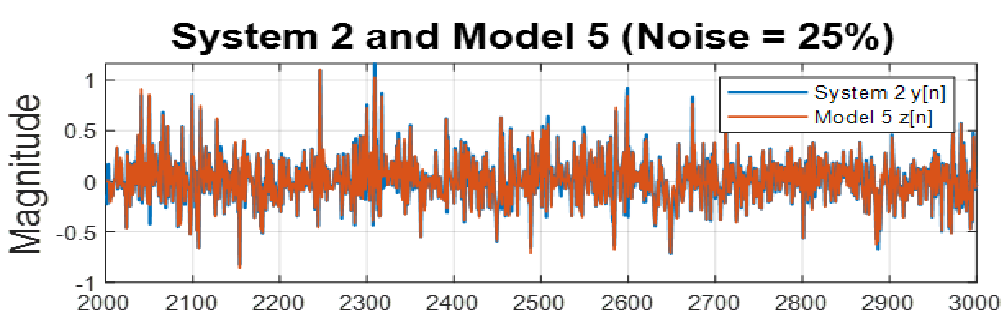
Best Model 6 (when Noise is 100%)

y[n]=-0.021631+(-0.19895)x[n]+(-0.090434)x[n]x[n-2]+(0.046997)x[n-2]x[n-2]+(0.082539)y[n- 1]x[n- 2]+(-0.05302)y[n- 1] +(0.057606)y[n- 2]x[n- 4]+(0.018746)x[n]x[n- 8]+(-0.024678)x[n- 1]x[n- 2]+(-0.017892)x[n- 5]+(0.050408)y[n- 1]x[n- 4]

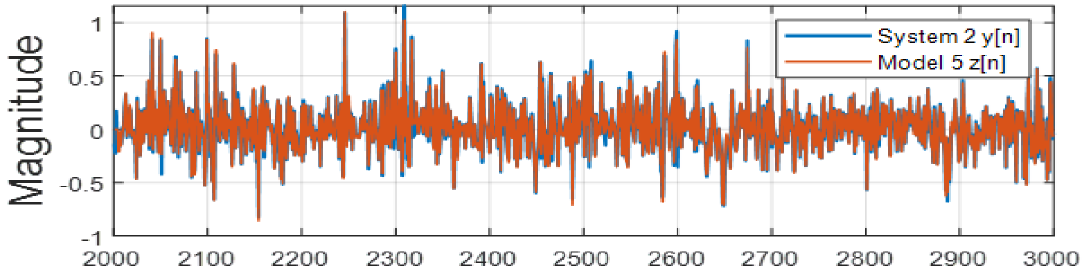
I/P & O/P


Input and System 2 Output
Magnitude
Input x[n]
System 2 y[n]
4
3
2
1
0
-1
-2
-3
-4
0
500
1000
1500
2000
2500
3000

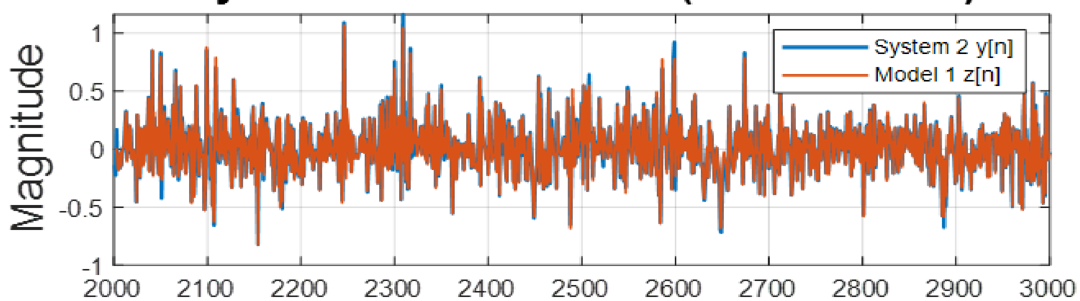

System 2 and Model 3 (Noise = 0%) Zoomed
Magnitude
System 2 y[n]
Model 3 z[n]
1
0.5
0
-0.5
2000
2010
2020
2030
2040
2050
2060
2070
2080
2090
2100


System 2 and Model 5 (Noise = 25%)
Magnitude
System 2 y[n]
Model 5 z[n]
1
0.5
0
-0.5
-1
2000
2100
2200
2300
2400
2500
2600
2700
2800
2900
3000

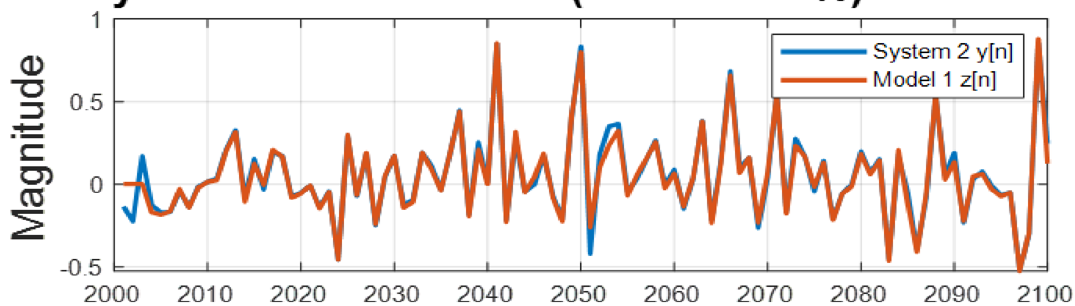
System 2 and Model 5 (Noise = 25%)



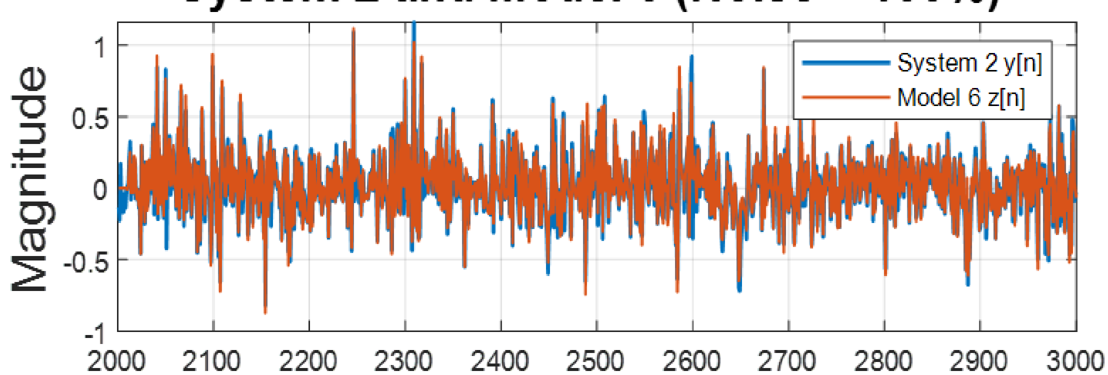
System 2 and Model 1 (Noise = 50%)

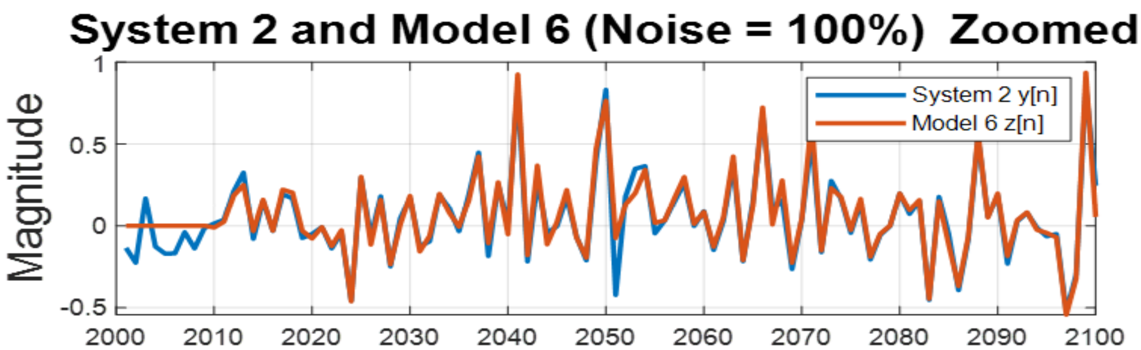


System 2 and Model 1 (Noise = 50%) Zoomed



System 2 and Model 6 (Noise = 100%)





System 2	Model	Model 1	Model 2	Model3	Model 4	Model 5	Model 6
	K	2	6	4	5	1	2
	L	3	1	2	3	4	9
	Order	2	2	2	2	2	2
Noiseless	Ideal % MSE	0	0	0	0	0	0
	Absolute MSE over Segment 1	0.0004422	0.0039333	0.0002035	0.0001883	0.0003623	5.33E-05
	Percentage MSE over Segment 1	0.7832479	6.9463444	0.3600349	0.3329297	0.641188	0.0939337
	Absolute MSE over Segment 2	0.0002236	0.0039621	2.80E-06	3.56E-06	0.0001954	8.20E-05
	Percentage MSE over Segment 2	0.4205746	7.4417897	0.0052648	0.0066939	0.3670479	0.1537538
	Absolute MSE over Segment 3	0	0	2.57E-06	0	0	0
	Percentage MSE over Segment 3	0	0	0.0041389	0	0	0
Noise P=25%	Ideal % MSE	20	20	20	20	20	20
	Absolute MSE over Segment 1	0.0151518	0.0187009	0.0143447	0.0154772	0.0151453	0.014721
	Percentage MSE over Segment 1	20.607645	27.642098	20.016686	21.568929	21.38054	20.696446
	Absolute MSE over Segment 2	0.016491	0.021193	0.0154888	0.0142476	0.0146913	0.0158964
	Percentage MSE over Segment 2	25.160887	31.006734	23.307592	21.366509	21.578472	22.486141
	Absolute MSE over Segment 3	0	0	0	0.0137635	0	0
	Percentage MSE over Segment 3	0	0	0	17.642236	0	0
Noise P=50%	Ideal % MSE	33.333333	33.333333	33.333333	33.333333	33.333333	33.333333
	Absolute MSE over Segment 1	0.02815	0.0333529	0.0285921	0.0283774	0.0266101	0.027518
	Percentage MSE over Segment 1	32.341118	41.657266	32.96142	34.061543	30.42109	33.27687
	Absolute MSE over Segment 2	0.027949	0.0348747	0.0278486	0.0300174	0.0278338	0.0318109
	Percentage MSE over Segment 2	35.068831	42.43039	33.577572	34.841593	34.580042	38.14107
	Absolute MSE over Segment 3	0	0	0	0	0.0287472	0
	Percentage MSE over Segment 3	0	0	0	0	32.404112	0
Noise P=100%	Ideal % MSE	50	50	50	50	50	50
	Absolute MSE over Segment 1	0.0573694	0.0629078	0.0620172	0.0552932	0.0561809	0.0623545
	Percentage MSE over Segment 1	51.950615	56.958966	51.251158	52.69041	49.642866	52.457601
	Absolute MSE over Segment 2	0.0578605	0.0664279	0.0602397	0.066415	0.0584274	0.0577022
	Percentage MSE over Segment 2	52.790895	59.388768	54.350325	57.519499	51.356275	52.177798
	Absolute MSE over Segment 3	0	0	0	0	0	0.062298
	Percentage MSE over Segment 3	0	0	0	0	0	50.391229

SYSTEM 3

y(n)= 0.46 + 0.152*y(n-3) - 2.162*x(n-1) - 1.02*x(n-2)*x(n)*y(n-1) - 0.233*y(n-1);

RESULTS:

Best Model 6 (when Noise is 0%)

y[n]=0.64265+(-0.72007)x[n-1]+(0.35302)y[n-3]+(-0.12237)y[n-1]+(0.027375)y[n-4]x[n]+(-0.016631)x[n] +(-0.010049)y[n-7]x[n]+(0.0030129)y[n- 5]x[n]

Best Model 5 (when Noise is 25%)

y[n]=0.59979+(-0.74165)x[n- 1]+(0.29396)y[n- 3] +(0.10636)x[n- 2]+(-0.024001)y[n- 4]y[n- 4]+(-0.023061)y[n- 1]x[n- 2]

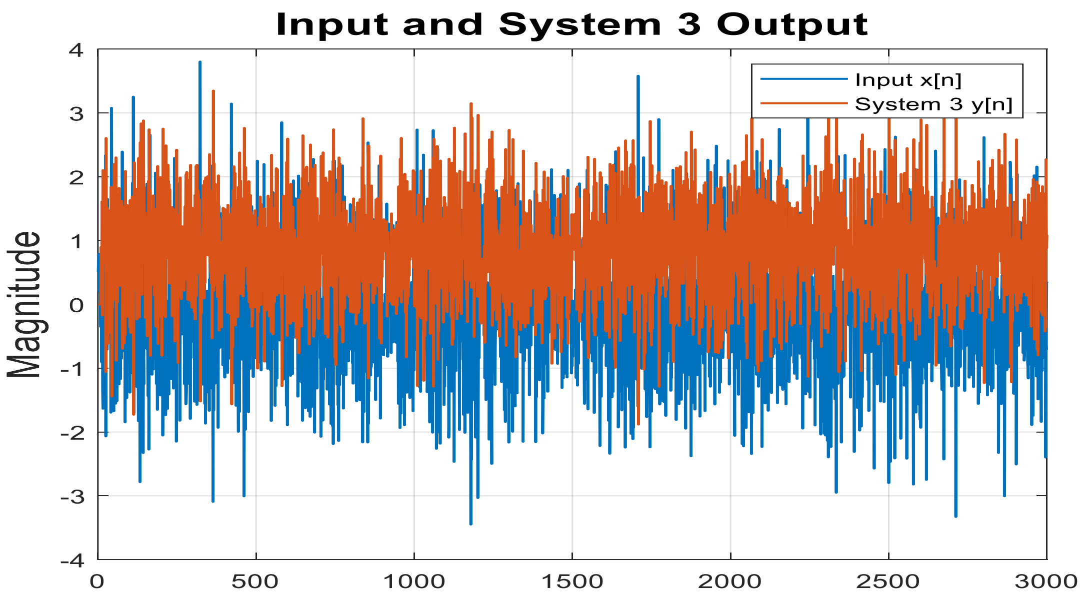
Best Model 2 (when Noise is 50%)

y[n]=0.88361+(-0.73637)x[n- 1] +(-0.23077)x[n- 4] +(0.071994)x[n- 2]+(-0.071076)y[n- 1]

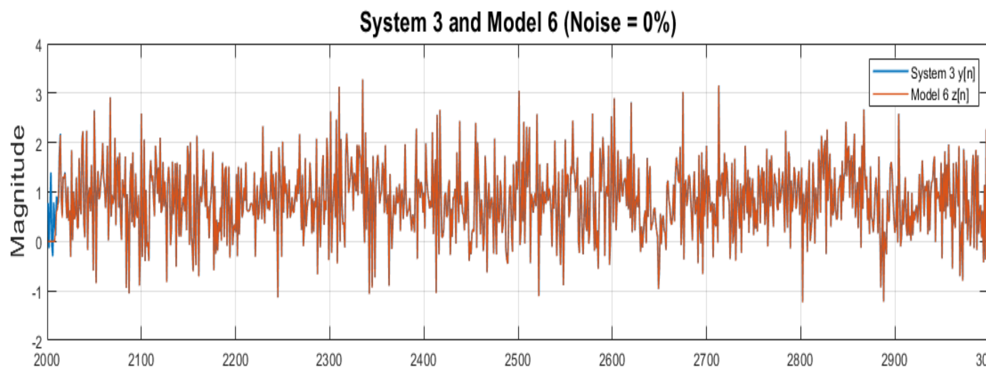
Best Model 6 (when Noise is 100%)

y[n]=0.69793+(-0.64277)x[n-1] +(0.14062)y[n-3]+(0.13399)x[n-2]+(-0.088802)y[n-1]x[n- 1]+(-0.075228)y[n-7]x[n-2]+(0.057866)y[n-6]+(0.039608)y[n- 3]y[n-5]+(-0.043177)y[n-8]x[n- 3]+(-0.061499)y[n-1]

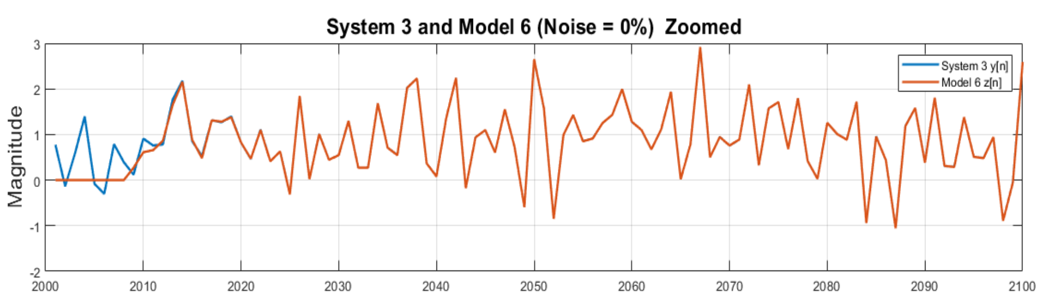
I/P & O/P



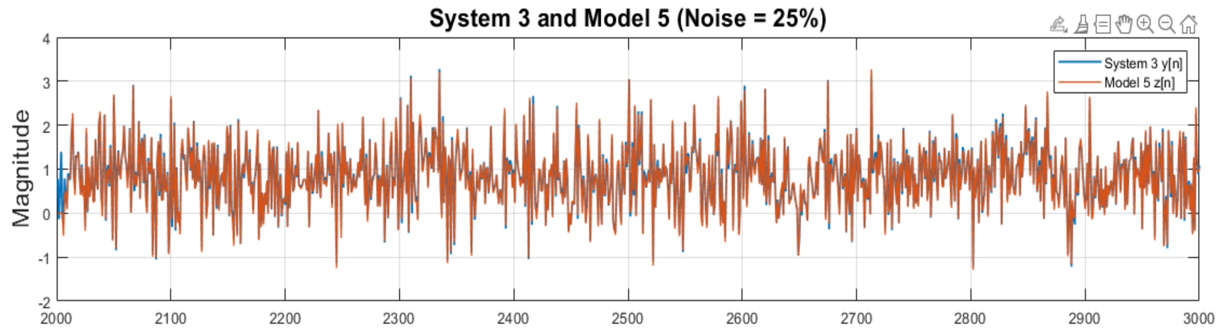
Input and System 3 Output
Magnitude
Input x[n]
System 3 y[n]

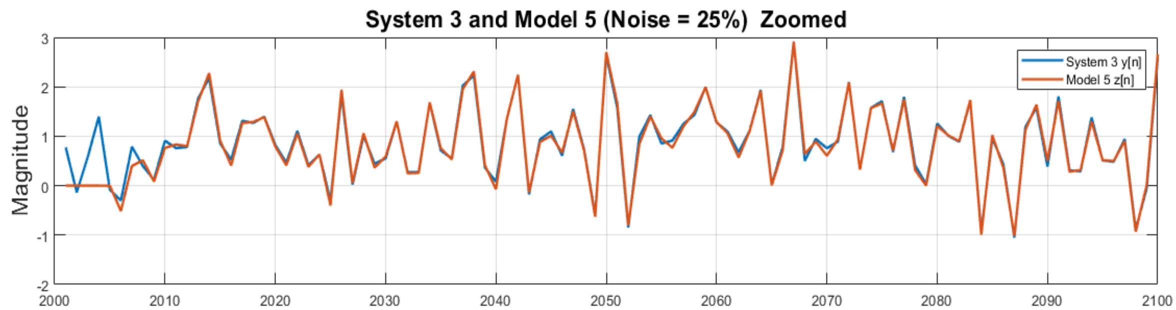


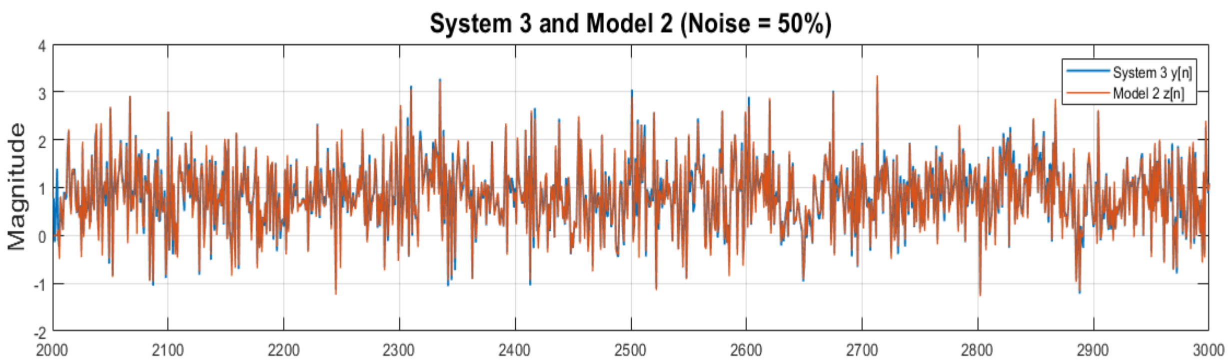
System 3 and Model 6 (Noise = 0%)
Magnitude
System 3 y[n]
Model 6 z[n]

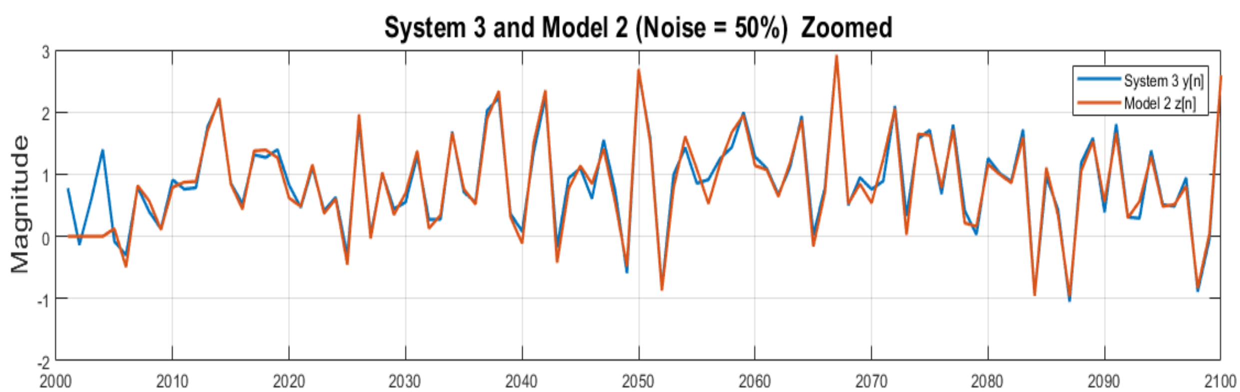


System 3 and Model 6 (Noise = 0%) Zoomed
Magnitude
System 3 y[n]
Model 6 z[n]

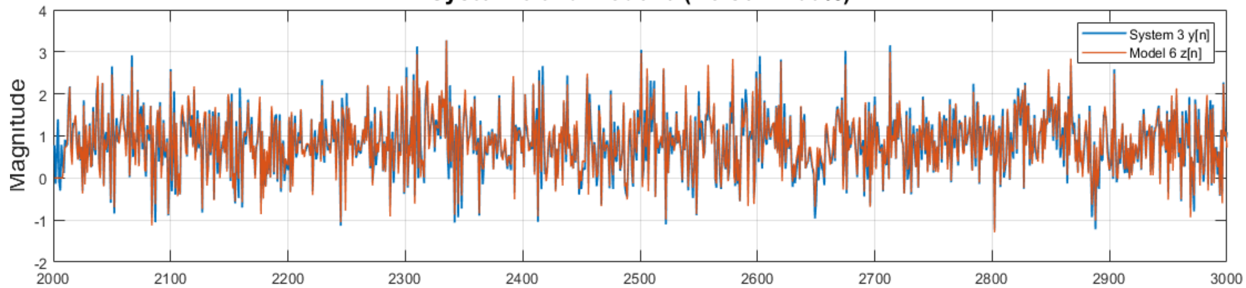

System 3 and Model 5 (Noise = 25%)
System 3 y[n]
Model 5 z[n]
Magnitude


System 3 and Model 5 (Noise = 25%) Zoomed
System 3 y[n]
Model 5 z[n]
Magnitude

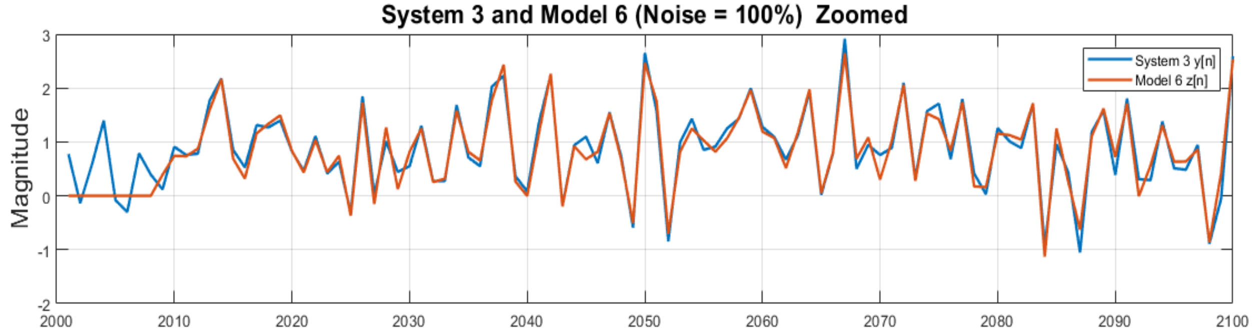

System 3 and Model 2 (Noise = 50%)
System 3 y[n]
Model 2 z[n]
Magnitude


System 3 and Model 2 (Noise = 50%) Zoomed
System 3 y[n]
Model 2 z[n]
Magnitude

System 3 and Model 6 (Noise = 100%)



System 3 and Model 6 (Noise = 100%) Zoomed



System 3	Model	Model 1	Model 2	Model3	Model 4	Model 5	Model 6
	K	2	1	2	5	4	8
	L	1	4	1	3	2	3
	Order	2	2	2	2	2	2
Noiseless	Ideal % MSE	0	0	0	0	0	0
	Absolute MSE over Segment 1	0.0726656	0.0119264	0.0726656	0.0005746	0.0011798	5.21E-04
	Percentage MSE over Segment 1	12.928608	2.1233027	12.928608	0.1023378	0.2100445	0.0928739
	Absolute MSE over Segment 2	0.0724258	0.0101747	0.0724258	0.0007025	0.0006888	1.26E-04
	Percentage MSE over Segment 2	12.654643	1.7762808	12.654643	0.1225271	0.1202546	0.0219922
	Absolute MSE over Segment 3	0	0	0	0	0	0.0001526
	Percentage MSE over Segment 3	0	0	0	0	0	0.0283348
Noise P=25%	Ideal % MSE	20	20	20	20	20	20
	Absolute MSE over Segment 1	0.2100678	0.1549566	0.2011401	0.1408435	0.1461398	0.1372498
	Percentage MSE over Segment 1	30.246273	21.668214	29.436241	20.650927	19.851668	19.545763
	Absolute MSE over Segment 2	0.2149479	0.1599387	0.2111984	0.1561841	0.1406333	0.1572976
	Percentage MSE over Segment 2	30.838342	21.88239	29.902068	21.548046	19.80605	21.568352
	Absolute MSE over Segment 3	0	0	0	0	0.1423781	0
	Percentage MSE over Segment 3	0	0	0	0	21.428054	0
Noise P=50%	Ideal % MSE	33.333333	33.333333	33.333333	33.333333	33.333333	33.333333
	Absolute MSE over Segment 1	0.3453788	0.3062632	0.3445067	0.325612	0.3121013	0.2794405
	Percentage MSE over Segment 1	42.831823	34.905599	44.54182	36.572635	35.182616	32.903036
	Absolute MSE over Segment 2	0.3396947	0.2758818	0.3446941	0.2792043	0.2988818	0.2967516
	Percentage MSE over Segment 2	42.129159	33.556735	39.059924	33.476692	35.092014	34.47477
	Absolute MSE over Segment 3	0	0.2909267	0	0	0	0
	Percentage MSE over Segment 3	0	34.853624	0	0	0	0
Noise P=100%	Ideal % MSE	50	50	50	50	50	50
	Absolute MSE over Segment 1	0.6048714	0.5406462	0.6048857	0.5982303	0.6211956	0.6014254
	Percentage MSE over Segment 1	59.938206	49.32034	55.144606	52.424735	52.063254	51.601422
	Absolute MSE over Segment 2	0.659136	0.5865348	0.6665342	0.5881986	0.5716447	0.5490081
	Percentage MSE over Segment 2	58.397433	48.389914	59.314207	53.131377	49.403226	51.260184
	Absolute MSE over Segment 3	0	0	0	0	0	0.5483043
	Percentage MSE over Segment 3	0	0	0	0	0	54.184635

Conclusion

For each equation, we note that the absolute value of the M.S.E. of the best model also improves as the value of P increases, computing the noisy values of the unmodelled output over the second data point segment.

We could observe that the Mean Square Error values of the best models obtained for the three systems are at all times close to the ideal MSE values. So, this establishes the successful implementation of the Fast-Orthogonal Search algorithm across various noise scenarios on all of these systems.

References

[1] Korenberg, Michael J. "Identifying nonlinear difference equation and functional expansion representations: the fast-orthogonal algorithm." Annals of biomedical engineering 16.1 (1988): 123-142.

[2] Korenberg, Michael J., and Larry D. Paarmann. "Orthogonal approaches to time-series analysis and system identification." Signal Processing Magazine, IEEE 8.3 (1991): 29-43.

[3] Korenberg, Michael J. "A robust orthogonal algorithm for system identification and time-series analysis." Biological cybernetics 60.4 (1989): 267-276.

[4] https://link.springer.com/article/10.1007/BF02368043

[5]https://www.researchgate.net/publication/20419817_Applications_of_Fast_Orthogonal_Search_Time-Series_Analysis_and_Resolution_of_Signals_in_Noise

Appendix:

MATLAB Code:

clc;

close all;

clear all;

n_points = 3000; **(% Number of points)**

System = 3; **(%1,2, or 3 Choose which system to run)**

h=randn (1, number_of_points);

(% Selecting the values of K and L)
l = [1, 2, 2, 3, 4, 7; % L for System 1
    3, 1, 2, 3 ,4 ,9; % L for System 2
    1, 4, 1, 3, 2, 3]; % L for System 3

k = [1, 4, 2, 3, 2, 5; % K for System 1
    2, 6, 4, 5, 1, 2; % K for System 2
    2, 1, 2, 5, 4, 8]; % K for System 3

l = l(System,:); **(%Assign L with the correct row)**

k = k(System,:); **(%Assign K with the correct row)**

Order=[2,2,2,2,2,2];

percent=[0, 25, 50, 100]; (**%Percent of noise)**

(%%% System 1)

y(n)= 1.2 + 0.124*y(n-1) + 0.2*x(n-2)*x(n-3) - 0.26*x(n-1)*y(n-2) ...- 1.03*x(n) - 0.03*y(n-2) *y(n-1);

mse_1st_percent = zeros(length(k), length(percent));

figure('Color',[1 1 1],'OuterPosition',[395 226 576 513]);

figure('Color',[1 1 1],'OuterPosition',[395 226 576 513]);

(% Create axes)

axes1 = axes;

hold(axes1,'on');

(% Create multiple lines using matrix input to plot)

plot1 = plot(YMatrix1,'LineWidth',1);

set(plot1(1),'DisplayName','Input x[n]');

set(plot1(2),'DisplayName','System 3 y[n]');

(% Create ylabel)

ylabel('Magnitude','FontSize',16);

(% Create title)

title('INPUT & OUTPUT','FontSize',16);

box(axes1,'on');

grid(axes1,'on');

% Create legend

legend(axes1,'show');

(% Create subplot)

subplot1 = subplot(2,1,1);

hold(subplot1,'on');

(% Create multiple lines using matrix input to plot)

plot1 = plot(X1,YMatrix1,'LineWidth',1.75);

set(plot1(1),'DisplayName','System 2 y[n]');

set(plot1(2),'DisplayName','Model 5 z[n]');

(% Create ylabel)

ylabel('Magnitude');

%%% System 2

y(n)= -0.03 + 0.13*x(n-1) - 0.234*x(n) - 1.022*x(n-2)*x(n) - 1.13*y(n-2)... - 0.043*y(n-3) + 2.3*y(n-1)*x(n-2);

(% Create figure)

figure('Color',[1 1 1],'OuterPosition',[395 226 576 513]);

(% Create subplot)

subplot1 = subplot(2,1,1);

hold(subplot1,'on');

(% Create multiple lines using matrix input to plot)

plot1 = plot(X1,YMatrix1);

set(plot1(1),'DisplayName','System 2 y[n]','LineWidth',1.35);

set(plot1(2),'DisplayName','Model 6 z[n]','LineWidth',1);

(% Create ylabel)

ylabel('Magnitude');

(% Create title)

title('System 2 and Model 6 (Noise = 100%)');

box(subplot1,'on');

grid(subplot1,'on');

(% Create legend)

legend(subplot1,'show');

(% Create title)

title('System 2 and Model 5 (Noise = 25%) Zoomed');

box(subplot1,'on');

grid(subplot1,'on');

(% Create legend)

legend(subplot1,'show');

%%% System 3

y(n)= 0.46 + 0.152*y(n-3) - 2.162*x(n-1) - 1.02*x(n-2)*x(n)*y(n-1) - 0.233*y(n-1);

mse_1st_magnitude = zeros(length(k),length(percent));

mse_2nd_magnitude = zeros(length(k),length(percent));

mse_3rd_percent = zeros(length(k),length(percent));

for i=1:length(percent)

```
for model=1:length(k)
disp(['Model ' num2str(model) ' where Noise = ' num2str(percent(i)) '%']);
tic
a(model)=FOS(x,y,percent(i),k(model),l(model),Order(model),...
number_of_points);
toc
mse_1st_magnitude(model,i)=a(model).mse_1st_magnitude;
mse_1st_percent(model,i)=a(model).mse_1st_percent;
mse_2nd_magnitude(model,i)=a(model).mse_2nd_magnitude;
mse_2nd_percent(model,i)=a(model).mse_2nd_percent;
end
mse_2nd_percent = zeros(length(k), length(percent));
Print_Model(a(p(i)).Selected_candidates,a(p(i)).a,a(p(i)).K,...
a(p(i)).L,percent(i),a(p(i)).Order,p(i));
v3(i,1:1000) = Model_Output(a(p(i)).a,a(p(i)).Selected_candidates,...
x(2001:3000),a(p(i)).K,a(p(i)).L,a(p(i)).N,a(p(i)).Order);
mse_3rd_magnitude(p(i),i)=a(p(i)).mse_3rd_magnitude;
mse_3rd_percent(p(i),i)=a(p(i)).mse_3rd_percent;
```

(% Create figure)

```
figure('Color',[1 1 1],'OuterPosition',[395 226 576 513]);
%CREATEFIGURE(X1, YMatrix1)
% X1: vector of x data
% YMATRIX1: matrix of y data
```

(% Create figure)

```
figure('Color',[1 1 1],'OuterPosition',[395 226 576 513]);
```

(% Create subplot)

```
subplot1 = subplot(2,1,1);
hold(subplot1,'on');
```

(% Create multiple lines using matrix input to plot)

plot1 = plot (X1, YMatrix1);

set(plot1(1),'DisplayName','System 2 y[n]','LineWidth',1.35);

set(plot1(2),'DisplayName','Model 6 z[n]','LineWidth',1);

(% Create ylabel)

ylabel('Magnitude');

(% Create title)

title('System 2 and Model 6 (Noise = 100%)');

box(subplot1,'on');

grid(subplot1,'on');

(% Create legend)

legend(subplot1,'show');

(% Create axes)

axes1 = axes;

hold(axes1,'on');

(% Create multiple lines using matrix input to plot)

plot1 = plot(YMatrix1,'LineWidth',1);

set(plot1(1),'DisplayName','Input x[n]');

set(plot1(2),'DisplayName','System 2 y[n]');

% Create ylabel

ylabel('Magnitude','FontSize',16);

(% Create title)

title('INPUT & OUTPUT','FontSize',16);

box(axes1,'on');

grid(axes1,'on');

(% Create legend)

legend(axes1,'show');

subplot(2,1,2) %Error Plotting

error = v3(i,1:1000)-y(1,2001:3000);

plot(2001:3000, error, 'linewidth',1.5, 'DisplayName','Error')

ylabel('Magnitude', 'FontSize',16)

title('ERROR', 'FontSize', 16)

legend('show')

grid on

% End of Figure 1-------------------------------------

function createfigure(X1, YMatrix1)

%CREATEFIGURE(X1, YMatrix1)

% X1: vector of x data

% YMATRIX1: matrix of y data

(% Create figure)

figure('Color',[1 1 1],'OuterPosition',[395 226 576 513]);

(% Create subplot)

subplot1 = subplot (2,1,1);

hold(subplot1,'on');

(% Create multiple lines using matrix input to plot)

plot1 = plot(X1,YMatrix1);

set(plot1(1),'DisplayName','System 2 y[n]','LineWidth',1.35);

set(plot1(2),'DisplayName','Model 6 z[n]','LineWidth',1);

(% Create ylabel)

ylabel('Magnitude');

(% Create title)

title('System 2 and Model 6 (Noise = 50%)');

box(subplot1,'on');

grid(subplot1,'on');

(% Create legend)

legend(subplot1,'show');

```
subplot (2,1,2) %Error Plotting Zoom
errorzoom = v3(i,001:100)-y (1,2001:2100);
plot (2001:2100, errorzoom, 'linewidth',2, 'DisplayName', 'Error')
ylabel('Magnitude', 'FontSize',16)
title ('ERROR Zoomed', 'FontSize', 16)
legend('show')
grid on
----------------------------------------
end
```

FOS Code

```
s = Add_Noise(y, Noise_percentage, n_points);
for e = 2:(M+1) Q=zeros(M,1); c=zeros(M,2);
for j= 2:(M+1)
if ip(j,1)>0 P(e,1:2) =ip(j,1:2);
if ip(j,1) ==2
Cand(1,1:1000) =X_Terms(ip(j,2),1:1000);
end;
G (1) = sum(s(N+1:1000))/(1000-N);
Selected_candidates(1) = Candidates(1);
A = 0;
M_S_E = sum(s(N+1:1000). ^2)/(1000-N)-G (1) ^2*E (1);
F (1, position) = g(next_candidate);
D (1, position) = e(next_candidate,position);
Selected_candidates(position) = next_candidate;
A (position,1: position-1) = Alpha (next_candidate,1: position-1);
M= size(a,1);
```

```
P_cand=P_candidates3rd(k,l,x,y_noisy);

P_cand=P_candidates3rd(k,l,x,y_noisy);

end;

Best_Model_Display_1=plot(2001:3000,y_noise_less(1,2001:3000),'',2001:3000,Model_Output
_3RD_SEGMENT(1,2001:3000));set(Best_Model_Display_1(1),'Color','red','linewidth',2.5);
set(Best_Model_Display_1(2),'Color','blue','linewidth',1.0);

w=y;

e = rand(1,3000);

per = 0.60;

y = y + sqrt(per*var(y)/var(e)) * (e-0.5);

figure;plot(1:N1,x,1:N1,y);

ac = 0;

Pal = 0;

yn2 = sum(y(N0+1:N).^2)/(N-N0);

diff = yn2 - g(1)^2 * D(1);

%%%P(1,e) - are the g(e) terms

%%%P(2,e) - are the D(e,e) terms

while(e1 < M)

P(1:O+2,e1) = -1;

P(1,e1) = g(pos);

P(2,e1) = D(pos,e1);

P(3:2+O,e1) = Pp(pos,1:O);

Pal(e1,1:e1-1) = alpha(pos,1:e1-1);

e1 = e1 + 1;

D1(1:M,e1) = 0;

alpha = 0;

for e = 2 : M

j = Pp(e,:);

if(sum(ismember(ac,e)) == 0)
```

```
SE = g(pos)^2 * D(pos,e1);
if(SE < (4/(N-N0))*diff || Qmax <=0 || diff < 0)
break;
end
diff = diff - (g(pos)^2) * D(pos,m1);
v1 = Model_Output(a,Selected_candidates,x(1,1:1000),K,L,N,Order);
mse_1st_magnitude = (sum((v1(N+1:1000)-s(N+1:1000)).^2))/(1000-N);
mse_1st_percent = (100*sum((v1(N+1:1000)-s(N+1:1000)).^2))/...
((1000-N)*var(s(N+1:1000)));
v2 = Model_Output(a,Selected_candidates,x(1,1001:2000),K,L,N,Order);
mse_2nd_magnitude = (sum((v2(N+1:1000)-s(N+1001:2000)).^2))/(1000-N);
mse_2nd_percent = (100*sum((v2(N+1:1000)-s(N+1001:2000)).^2))/...
((1000-N)*var(s(N+1001:2000)));
v3 = Model_Output(a,Selected_candidates,x(1,2001:3000),K,L,N,Order);
mse_3rd_magnitude = (sum((v3(N+1:1000)-s(N+2001:3000)).^2))/(1000-N);
mse_3rd_percent=(100*sum((v3(N+1:1000)-s(N+2001:3000)).^2))/...((1000-
N)*var((N+2001:3000)));
Best_Model_Display_1=plot(2001:3000,y_noise_less(1,2001:3000),'--
',2001:3000,Model_Output_3RD_SEGMENT(1,2001:3000));set(Best_Model_Display_1(1),'Col
or','red','linewidth',2.5); set(Best_Model_Display_1(2),'Color','blue','linewidth',1.0);figure;
Best_Model_Display_2=plot(2900:3000,y_noise_less(1,2900:3000),'--
',2900:3000,Model_Output_3RD_SEGMENT(1,2900:3000));set(Best_Model_Display_2(1),'Col
or','red','linewidth',2.5); set(Best_Model_Display_2(2),'Color','blue','linewidth',1.0);
```