

Appendix A

Simulation Models

This appendix explains how to obtain simulation models for a cantilever beam with different loading conditions. The models are referred to as Model 1 to Model 4 with the loading condition shown in Fig. 2.2. Using modal analysis, it can be found that the transfer function from the applied force $u(s)$ to the transverse deflection of the beam $y(s, x)$ is

$$\frac{Y(s, x)}{U(s)} = \sum_{m=1}^{\infty} \frac{y_m(x)y_m(x_1)}{s^2 + 2\zeta_m\omega_m s + \omega_m^2} \quad (\text{A.1})$$

where $y_m(x)$ is the eigenfunction of the m^{th} mode at position x , $y_m(x_1)$ is the eigenfunction of the m^{th} mode at position x_1 , ζ_m is the damping ratio of the m^{th} mode and ω_m is the eigenvalue or the natural frequency of the m^{th} mode. Substituting $s = j\omega = j0 = 0$ to (A.1), one obtains the m^{th} mode frequency response at dc (dc gain) as

$$\frac{Y(s, x)}{U(s)} = \frac{y_m(x)y_m(x_1)}{\omega_m^2}. \quad (\text{A.2})$$

Equation (A.1) and (A.2) show that in general every transfer function is a superposition of single degree of freedom systems, with each system having its dc gain determined by the proper eigenfunction entry product divided by the square of the eigenvalue, and with resonant frequency defined by the proper eigenvalue.

The dc gain can be used as a mode selection criterion to sort modes of vibration so that only the most important modes are kept in the models.

To obtain the simulation models in the form of (A.1), the models are built using the ANSYS program. Since the values of eigenfunctions ($y(x)$) and eigenvalues (ω) are obtained as return parameters from modal analysis in ANSYS, the only parameter that needs to be found is ζ . The value of ζ can be found by measuring the deflection at the resonance frequency. At resonance, the peak amplitude of each mode is given by substituting $s = j\omega_i$, $s^2 = -\omega_m^2$ into (A.1)

$$\begin{aligned} \frac{Y_m(x)}{u} &= \frac{y_m(x)y_m(x_1)}{s^2 + 2\zeta_m\omega_i s + \omega_m^2} \\ &= \frac{y_m(x)y_m(x_1)}{-\omega_m^2 + 2\zeta_m\omega_m^2 j + \omega_m^2} \\ &= \frac{y_m(x)y_m(x_1)}{2\zeta_m\omega_m^2 j}. \end{aligned} \quad (\text{A.3})$$

From (A.3) it can be seen that ζ_m can be calculated if the maximum deflection $y_i(x)$ at each resonance frequency for a certain force amplitude u is known. To find $y_i(x)$ and u , harmonic analysis using ANSYS is conducted. The harmonic analysis is conducted by sweeping the system with a certain u along the frequency band of interest and measuring the deflection at certain points x along the beam. The output of the harmonic analysis using ANSYS is a frequency response function (FRF). From the FRF the maximum deflection $y(x)$ at the resonance frequency for each of the modes can be measured, then the value of ζ can be found from (A.3).

The modal analysis in ANSYS for the first 25 modes of vibration for all models investigated in this research shows that the dc gain for the first three modes are the most dominant and the dc gain of the modes higher than the tenth-mode are very small ($< 10^{-2}$) compared to the first three modes. Therefore, the simulation modes for all the models will be based on the first ten modes of

vibration. The natural frequencies, eigenfunctions and damping ratio, ζ , for the first ten modes for Model 1 to Model 4 are tabulated in Table A.1 to Table A.4. The eigenfunctions are obtained for position x_1 in the middle of the piezoceramic patch (3.5 cm from the fixed end).

Natural frequency (Hz)	eigenfunction $y(x)$	ζ
5.9750	0.0125574	4.97E-02
45.236	-0.0772385	9.02E-04
133.50	-0.264567	4.42E-04
319.90	-0.594745	2.29E-05
542.84	0.697097	5.35E-05
733.80	-1.02103	5.99E-06
1113.5	-1.54031	3.58E-05
1576.1	-1.60791	4.59E-05
1844.0	1.41085	3.25E-05
2350.6	-2.12969	3.26E-05

Table A.1: The first ten natural frequencies, eigenfunctions and damping ratios of Model 1.

Natural frequency (Hz)	eigenfunction $y(x)$	ζ
6.8345	0.0140025	9.31E-03
53.871	-0.108671	8.41E-03
162.86	0.304482	8.31E-04
331.95	-0.565164	6.92E-04
560.58	0.858715	8.73E-06
846.88	-1.15324	2.89E-04
1187.5	1.41837	5.36E-04
1576.2	-1.62605	2.66E-04
2003.3	1.75746	1.52E-03
2460.5	-1.82730	7.27E-04

Table A.2: The first ten natural frequencies, eigenfunctions and damping ratios of Model 2.

Since the eigenfunctions obtained from modal analysis using ANSYS are normalized with respect to the mass of the beam without loads, the eigenfunctions for Model 1 to Model 3 must be multiplied by a correction factor before they can be substituted into (A.1). The correction factor can be defined as the ratio of

Natural frequency (Hz)	eigenvalue $y(x)$	ζ
7.7432	0.0170449	6.17E-02
59.243	-0.105044	8.58E-04
145.34	0.268462	3.69E-03
326.86	-0.604026	6.59E-05
557.56	0.719626	4.47E-05
739.16	-0.997956	3.00E-05
1113.6	-1.53892	2.61E-05
1580.7	-1.64910	4.61E-05
1871.3	1.41052	4.08E-05
2370.7	-2.14833	3.73E-05

Table A.3: The first ten natural frequencies, eigenfunctions and damping ratios of Model 3.

Natural frequency (Hz)	eigenvalue $y(x)$	ζ
10.020	0.0217182	1.85E-03
62.790	-0.124932	2.69E-04
175.80	0.321682	2.36E-03
344.47	-0.574135	1.44E-04
569.36	0.857314	1.88E-05
850.39	-1.14589	-1.74E-05
1187.5	1.41674	1.02E-05
1580.7	-1.64939	-2.10E-05
2029.8	1.82671	-8.62E-06
2534.8	-1.93543	-8.05E-06

Table A.4: The first ten natural frequencies, eigenfunctions and damping ratios of Model 4.

the mass of the beam plus the mass of the load to the mass of the beam without a load. Based on the mass of the beam (m) and mass of the load (M_1 and M_2), the correction factors for Model 1, Model 2 and Model 3 are 1.8087, 1.3281 and 1.4806, respectively. By inserting the values of corrected eigenfunctions, damping ratios and natural frequencies for each of the models, the transfer functions of Model 1 to Model 4 for the first 10 modes are obtained. Note that because the sensor and actuator are collocated then $y_m(x) = y_m(x_1)$. The transfer functions are then implemented in Simulink and formed in a twentieth-order system.

$$\frac{Y(s)}{U(s)} = \frac{b_{19}s^{19} + b_{18}s^{18} + \dots + b_1s + b_0}{s^{20} + a_{19}s^{19} + a_{18}s^{18} + \dots + a_1s + a_0}. \quad (\text{A.4})$$

The coefficients of the transfer function and the position of the poles of Model 1 to Model 4 are tabulated in Table A.5 - A.8. All the tables show that the poles of all the models are located in the left side of the s-plane, therefore all the models are stable.

i	a_i	b_i	pole positions
0	1.125e067	8.814e060	-8.7382e+001 +1.4769e+004i
1	4.36e063	2.821e057	-8.7382e+001 -1.4769e+004i
2	8.143e063	5.272e057	-6.8550e+001 +1.1586e+004i
3	6.337e059	3.195e053	-6.8550e+001 -1.1586e+004i
4	1.138e059	5.443e052	-5.8591e+001 +9.9028e+003i
5	4.016e054	1.528e048	-5.8591e+001 -9.9028e+003i
6	1.876e053	6.761e046	-4.1394e+001 +6.9962e+003i
7	3.746e048	1.076e042	-4.1394e+001 -6.9962e+003i
8	6.542e046	1.78e040	-2.7279e+001 +4.6105e+003i
9	8.74e041	1.879e035	-2.7279e+001 -4.6105e+003i
10	7.872e039	1.625e033	-2.0180e+001 +3.4107e+003i
11	7.328e034	1.15e028	-2.0180e+001 -3.4107e+003i
12	4.028e032	6.071e025	-1.1892e+001 +2.0100e+003i
13	2.579e027	2.779e020	-1.1892e+001 -2.0100e+003i
14	9.447e024	9.767e017	-4.9628e+000 +8.3879e+002i
15	4.024e019	2.671e012	-4.9628e+000 -8.3879e+002i
16	1.052e017	6.734e009	-1.6816e+000 +2.8422e+002i
17	2.723e011	8481	-1.6816e+000 -2.8422e+002i
18	5.373e008	16.26	-2.2212e-001 +3.7541e+001i
19	644.3	-2.274e-013	-2.2212e-001 -3.7541e+001i

Table A.5: The transfer function coefficients and pole positions of Model 1.

i	a_i	b_i	pole positions
0	6.989e067	5.506e061	-3.6238e+001 +1.5460e+004i
1	9.355e063	6.126e057	-3.6238e+001 -1.5460e+004i
2	3.861e064	2.537e058	-2.9505e+001 +1.2587e+004i
3	1.02e060	5.192e053	-2.9505e+001 -1.2587e+004i
4	3.824e059	1.832e053	-2.3214e+001 +9.9035e+003i
5	4.739e054	1.8e048	-2.3214e+001 -9.9035e+003i
6	4.59e053	1.642e047	-1.7489e+001 +7.4613e+003i
7	3.443e048	9.787e041	-1.7489e+001 -7.4613e+003i
8	1.375e047	3.706e040	-1.2473e+001 +5.3211e+003i
9	6.877e041	1.447e035	-1.2473e+001 -5.3211e+003i
10	1.443e040	2.888e033	-8.2562e+000 +3.5222e+003i
11	4.998e034	7.556e027	-8.2562e+000 -3.5222e+003i
12	6.382e032	9.201e025	-4.8890e+000 +2.0857e+003i
13	1.533e027	1.576e020	-4.8890e+000 -2.0857e+003i
14	1.314e025	1.293e018	-2.3986e+000 +1.0233e+003i
15	2.114e019	1.325e012	-2.3986e+000 -1.0233e+003i
16	1.305e017	7.851e009	-7.9341e-001 +3.3848e+002i
17	1.276e011	3688	-7.9341e-001 -3.3848e+002i
18	5.975e008	16.63	-1.0066e-001 +4.2942e+001i
19	270.7	4.547e-013	-1.0066e-001 -4.2942e+001i

Table A.6: The transfer function coefficients and pole positions of Model 2.

i	a_i	b_i	pole positions
0	4.523e067	3.498e061	-4.8345e+001 +1.4895e+004i
1	7.552e063	4.73e057	-4.8345e+001 -1.4895e+004i
2	1.951e064	1.217e058	-3.8161e+001 +1.1758e+004i
3	7.012e059	3.387e053	-3.8161e+001 -1.1758e+004i
4	1.692e059	7.593e052	-3.2235e+001 +9.9318e+003i
5	3.145e054	1.131e048	-3.2235e+001 -9.9318e+003i
6	2.32e053	7.896e046	-2.2709e+001 +6.9969e+003i
7	2.528e048	6.943e041	-2.2709e+001 -6.9969e+003i
8	7.549e046	1.984e040	-1.5073e+001 +4.6443e+003i
9	5.517e041	1.153e035	-1.5073e+001 -4.6443e+003i
10	8.705e039	1.758e033	-1.1370e+001 +3.5032e+003i
11	4.439e034	6.844e027	-1.1370e+001 -3.5032e+003i
12	4.33e032	6.447e025	-6.6655e+000 +2.0537e+003i
13	1.52e027	1.62e020	-6.6655e+000 -2.0537e+003i
14	9.956e024	1.022e018	-2.9639e+000 +9.1319e+002i
15	2.321e019	1.53e012	-2.9639e+000 -9.1319e+002i
16	1.089e017	6.942e009	-1.2081e+000 +3.7223e+002i
17	1.54e011	4777	-1.2081e+000 -3.7223e+002i
18	5.468e008	16.5	-1.5790e-001 +4.8652e+001i
19	357.8	-4.206e-012	-1.5790e-001 -4.8652e+001i

Table A.7: The transfer function coefficients and pole positions of Model 3.

i	a_i	b_i	pole positions
0	2.919e068	2.248e062	-1.5927e+001 +1.5927e+004i
1	1.204e064	7.642e057	-1.5927e+001 -1.5927e+004i
2	7.586e064	4.907e058	-1.2754e+001 +1.2754e+004i
3	7.804e059	3.534e053	-1.2754e+001 -1.2754e+004i
4	5.627e059	2.167e053	-9.9318e+000 +9.9318e+003i
5	2.871e054	8.744e047	-9.9318e+000 -9.9318e+003i
6	5.86e053	1.659e047	-7.4613e+000 +7.4613e+003i
7	1.85e048	4.16e041	-7.4613e+000 -7.4613e+003i
8	1.629e047	3.464e040	-5.3432e+000 +5.3432e+003i
9	3.459e041	5.74e034	-5.3432e+000 -5.3432e+003i
10	1.641e040	2.586e033	-3.5774e+000 +3.5774e+003i
11	2.418e034	2.876e027	-3.5774e+000 -3.5774e+003i
12	7.074e032	8.014e025	-2.1644e+000 +2.1644e+003i
13	7.218e026	5.829e019	-2.1644e+000 -2.1644e+003i
14	1.427e025	1.102e018	-1.1046e+000 +1.1046e+003i
15	9.718e018	4.791e011	-1.1046e+000 -1.1046e+003i
16	1.387e017	6.57e009	-3.9452e-001 +3.9452e+002i
17	5.714e010	1305	-3.9452e-001 -3.9452e+002i
18	6.18e008	13.63	-6.2958e-002 +6.2957e+001i
19	117.4	-2.842e-013	-6.2958e-002 -6.2957e+001i

Table A.8: The transfer function coefficients and pole positions of Model 4.

Appendix B

Passivity

The following definitions of a positive real or passive system are found in the literature [99, 56]

Definition B.1 A rational function $H(s)$ of the complex variables $s = \sigma + j\omega$ is positive real if

1. $H(s)$ is real for real s .
2. $\operatorname{Re}[H(s)] \geq 0$ for all $\operatorname{Re}[s] > 0$.

For discrete time systems the passivity or positive realness of a system is defined in the following definitions:

Definition B.2 [70] An $m \times m$ discrete matrix $H(z)$ of real rational functions is positive real if and only if:

1. All the elements $H(z)$ are analytic in $|z| > 1$ (i.e., they do not have poles in $|z| > 1$).
2. The eventual poles of any element of $H(z)$ on the unit circle $|z| = 1$ are simple and the associated residue matrix is a positive semidefinite Hermitian.

3. The matrix $H(e^{j\omega}) + H^T(e^{-j\omega})$ is a positive semidefinite Hermitian for all real values of ω which are not poles of any element of $H(e^{j\omega})$.

In the case of a scalar transfer function $H(z)$ condition (3) is replaced by :

$$\operatorname{Re}H(z) \geq 0; \quad \forall |z| = 1. \quad (\text{B.1})$$

Definition B.3 [70] An $m \times m$ discrete matrix $H(z)$ of real rational functions is strictly positive real if and only if:

1. All the elements $H(z)$ are analytic in $|z| \geq 1$.
2. The matrix $H(e^{j\omega}) + H^T(e^{-j\omega})$ is a positive definite Hermitian for all real ω .

Appendix C

SimulinkTM Models

This appendix presents the Simulink models that are used for simulation and experimental implementations. The models are referred to as Model 1 to Model 4 with the loading condition shown in Fig. 2.2. The parameters of the models are shown in Table 2.2, and the natural frequencies of the models for simulation (obtained from analytical results) and experimental implementations are shown in Table 2.5.

A set of C S-functions is implemented for the simulation and experimental implementations. The functions are stored as files with *.c* extension. The files are listed in Table C.1

C.1 Simulations

In the simulations all the models are formed as transfer functions where the denominators and numerators for all the models are stored in the *mdl_TF.mat* file. A bank of Band-Pass Filters (BPFs) with resonant controller structure for all the models is stored in the *rescon.mdl* file. Six Simulink model (with *.mdl* extension) are built for the simulations. The models and the explanation about the models are shown in Table C.2. The fixed-parameter models used in the filter bank in all the simulation models are based on Model 1, Model 3 and Model 4.

File	Note
<i>rlsid2</i>	RLS estimator with constant forgetting factor.
<i>fc11</i> <i>fc12</i> <i>fc13</i>	All the algorithms are identical. These algorithms are used to convert the coefficients from estimation results to natural frequencies.
<i>adacon</i>	Resonant controller algorithm. To make the controller adjustable, natural frequency is used as an input to this file.
<i>super1</i> <i>super2</i> <i>super3</i>	All the algorithms are identical. Supervisor algorithms that are used in the M ⁴ ARC method to determine whether the estimator is still in the transient phase or close to the steady state condition, and to supply the adjustable controller with appropriate parameters.
<i>superx1</i> <i>superx2</i> <i>superx3</i>	All the algorithms are identical. Algorithms that are used to observe the switching behaviour in the M ⁴ ARC method.
<i>maxi</i>	Supervisor algorithm that is used in the M ⁴ RC method to determine which model is the closest model to the current loading condition and to supply the adjustable controller with the appropriate parameters.
<i>supsor</i>	Existing supervisor algorithm (MMSE) that is used in the M ⁴ RC method to determine which model is the closest model to the current loading condition and to supply the adjustable controller with the appropriate parameters.
<i>saklar</i>	Algorithm that is used to observe the switching behaviour in the M ⁴ RC method with the MMSE supervisor scheme.

Table C.1: C S-functions that are used for simulation and experimental implementations.

Model	Can be used for
<i>adapt_sim</i>	Simulate the adaptive resonant controller compared to the fixed-parameter resonant controller. Test the natural frequency estimator.
<i>m4rc_sim</i>	Simulate the M ⁴ RC using Butterworth BPF in the filter bank compared to the fixed-parameter resonant controller or to the adaptive resonant controller.
<i>rc_m4rc_sim</i>	Simulate the M ⁴ RC using BPF with resonant controller structure in the filter bank compared to the fixed-parameter resonant controller.
<i>narendra_sim</i>	Simulate the M ⁴ RC using the existing MMSE supervisor scheme.
<i>m4arc_sim</i>	Simulate the M ⁴ ARC using Butterworth BPF in the filter bank compared to adaptive resonant controller.
<i>rc_m4arc_sim</i>	Simulate the M ⁴ ARC using BPF with resonant controller structure in the filter bank compared to the adaptive resonant controller.

Table C.2. Simulink model for simulations.

adapt_sim

The schematic diagram of the *adapt_sim* Simulink model is shown Fig C.1. In the figure, the blocks **in_md11** to **in_md14** are the disturbances that are formed as a combination of three sinusoidal signals with frequencies are the same as the first three natural frequencies of Model 1 to Model 4, respectively. The block **plant** is the plant under control that can be changed to Model 1, Model 2, Model 3, or Model 4. The blocks **1st mode natural frequency estimator**, **2nd mode natural frequency estimator** and **3rd mode natural frequency estimator** are three parallel second-order estimators. The detail of the blocks are shown in Fig. C.2 (a) to (c). The **BPF** and **BPF1** blocks in all the natural frequency estimator blocks are band-pass filter that are used to prefiltering the input to the **RLS Estimator** block. The sampling time for the 1st, 2nd and 3rd estimator are 0.02 sec, 0.002 sec, and 0.001 sec, respectively. The **controller** block is an adjustable controller that is built from three parallel resonant controllers. The detail of the **controller** block is shown in Fig. C.3.

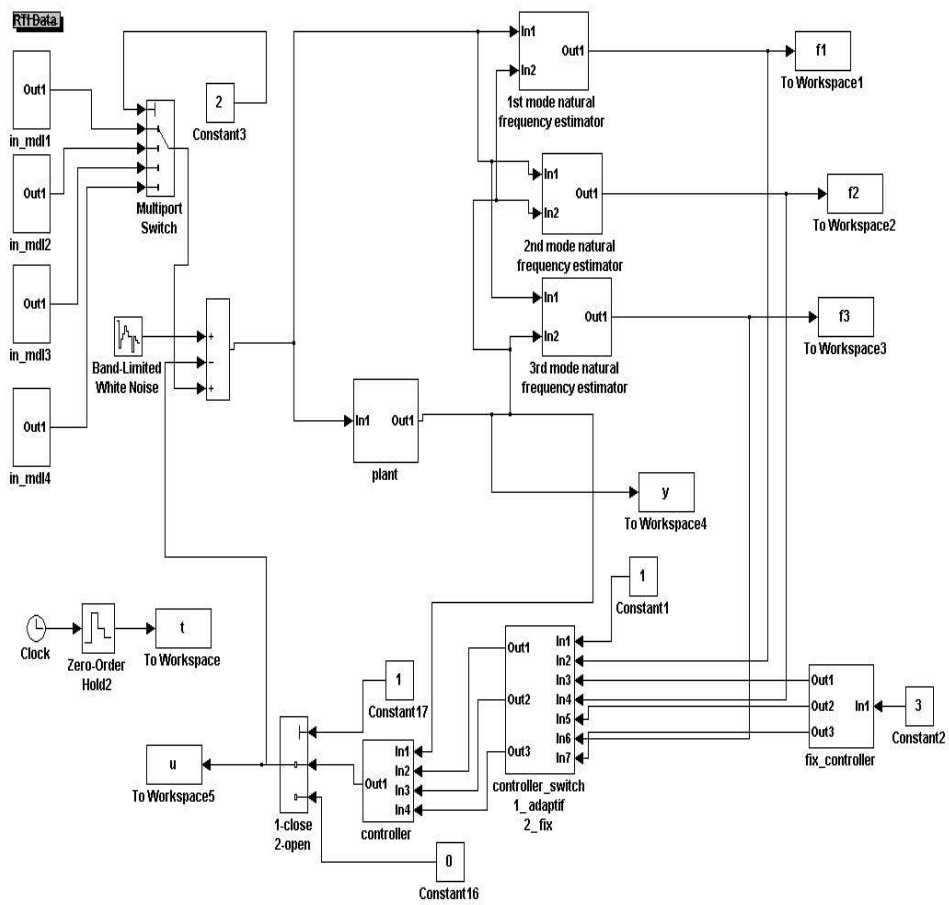


Figure C.1. Schematic diagram of *adap_sim* model.

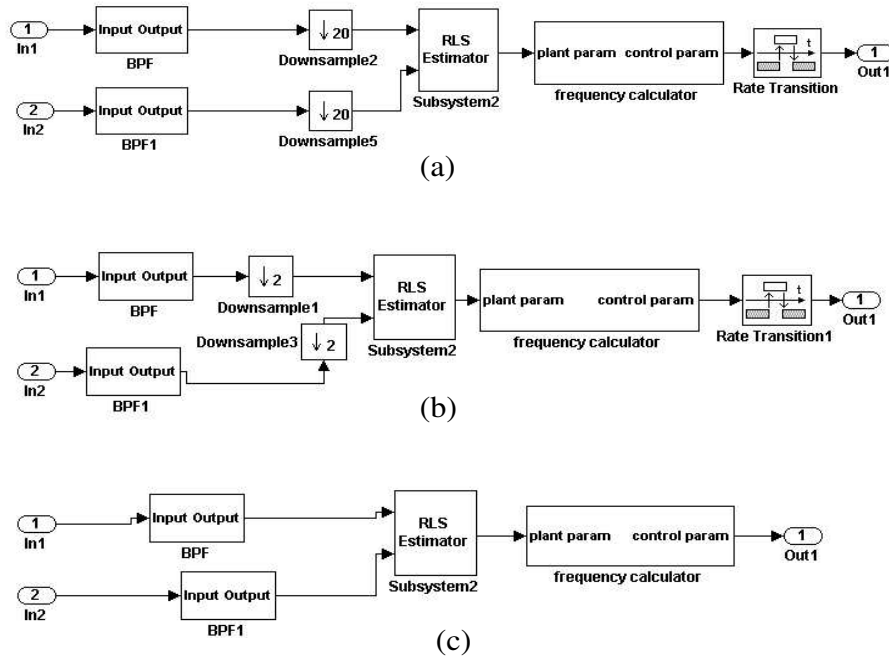


Figure C.2: Detail schematic diagram of (a) 1st mode natural frequency estimator block (b) 2nd mode natural frequency estimator block (c) 3rd mode natural frequency estimator block.

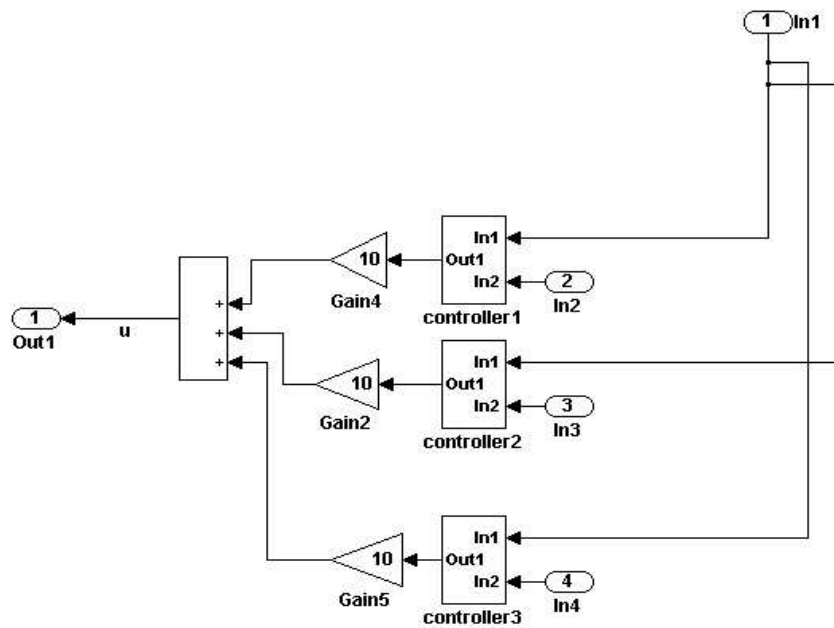


Figure C.3. Detail schematic diagram of the controller block.

To start a simulation, set solver type under the simulation parameters menu as variable-step (ode45). Set the constant value in the **Constant3** block according to which model is want to be controlled. Set the numerator and denominator in the **plant** block accordingly (e.g. set the numerator as num.mdl1 and the denominator as den.mdl1 to make Model 1 as the plant under control). The model and the disturbance can be changed while the simulation is running by using the `set_param` command in the Matlab editor. To observe the output and the control signal, the variable `y` and `u` can be plotted by using the `plot(t,y)` or `plot(t,u)` command in the Matlab editor. To observe the natural frequency of the estimation results, plot the variable `f1` to `f3` using the `plot` command from the Matlab editor.

m4rc_sim

The schematic diagram of the *m4rc_sim* model is shown Fig. C.4. From the figure, it can be seen that the *m4rc_sim* model is similar to *adapt_sim* model

with addition of the **1st mode filter bank and switching systems** to **3rd mode filter bank and switching systems** blocks. These blocks are bank filter systems and switching systems for M⁴RC method as shown in Fig. C.5. The BPFs in the filter bank system for each mode are designed as sharp band-pass filters with frequency cut-off the same as natural frequencies of Model 1, Model 3 and Model 4 for that particular mode.

A similar procedure to *adapt_sim* simulation is used to start a simulation. The **controller_switch** block is used to select whether the M⁴RC method or adaptive method want to be used. The variable s, s1, and s2 can be used to observe the switching behaviour of the M⁴RC method.

rc_m4rc_sim

The schematic diagram of the *rc_m4rc_sim* model is similar to the schematic diagram of the *m4rc_sim* model. The difference is that for the *rc_m4rc_sim* model the band-pass filters in the **1st mode filter bank** to **3rd mode filter bank** blocks are formed from the band-pass filter with resonant controller structure.

narendra_sim

The schematic diagram of the *narendra_sim* model is shown Fig. C.6. In this model, the **plant** block consist of Model 1 to Model 4, therefore, to change the model under control while simulation is running can be achieved by changing the constant value in the **Constant3** block.

m4arc_sim

The schematic diagram of the *m4arc_sim* model is shown Fig. C.7. From the figure it can be seen that the *m4arc_sim* model is similar to *m4rc_sim* model as shown in Fig. C.4. The difference is that in the *m4arc_sim* model, the output of the natural frequency estimator is fed into the filter bank and switching system. The filter bank and switching system for *m4arc_sim* model is shown Fig. C.8.

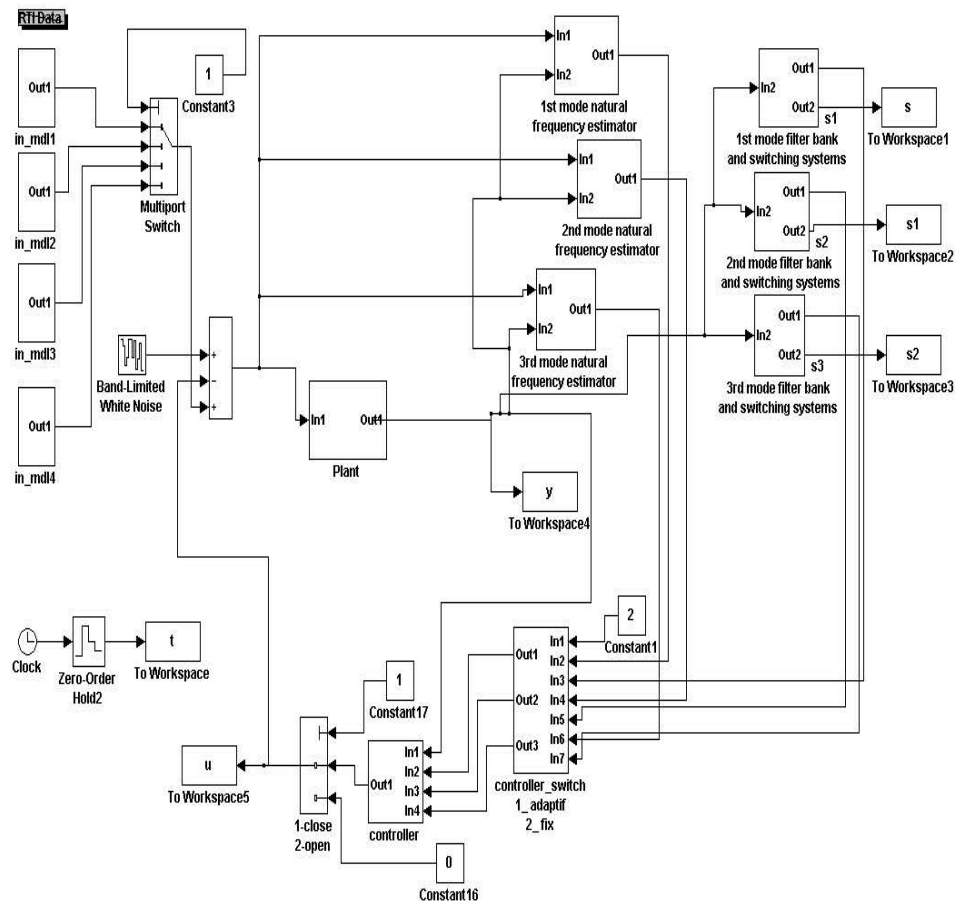


Figure C.4. Schematic diagram of *m4rc_sim* model.

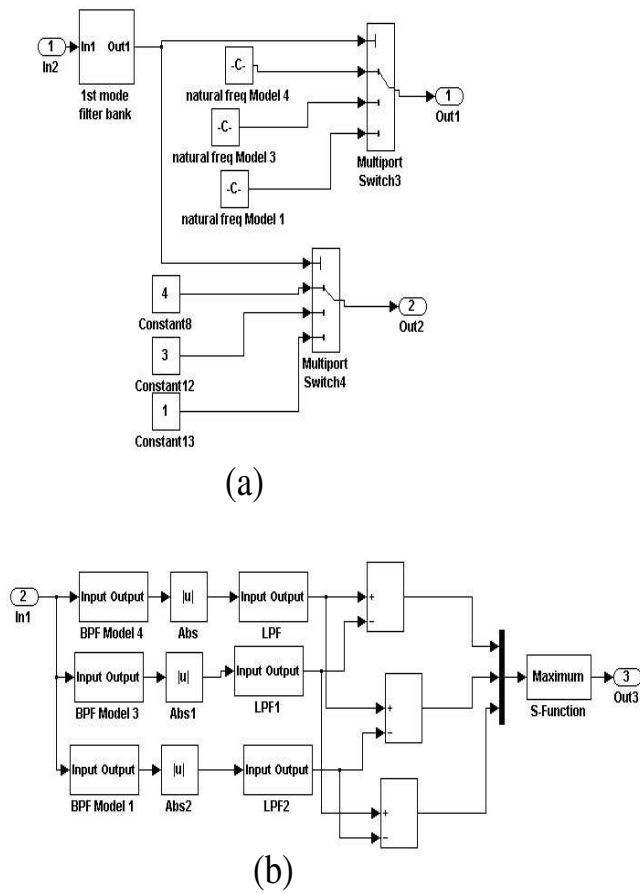


Figure C.5: Schematic diagrams of (a) Filter bank and switching system block (b) Detail of bank filter system block.

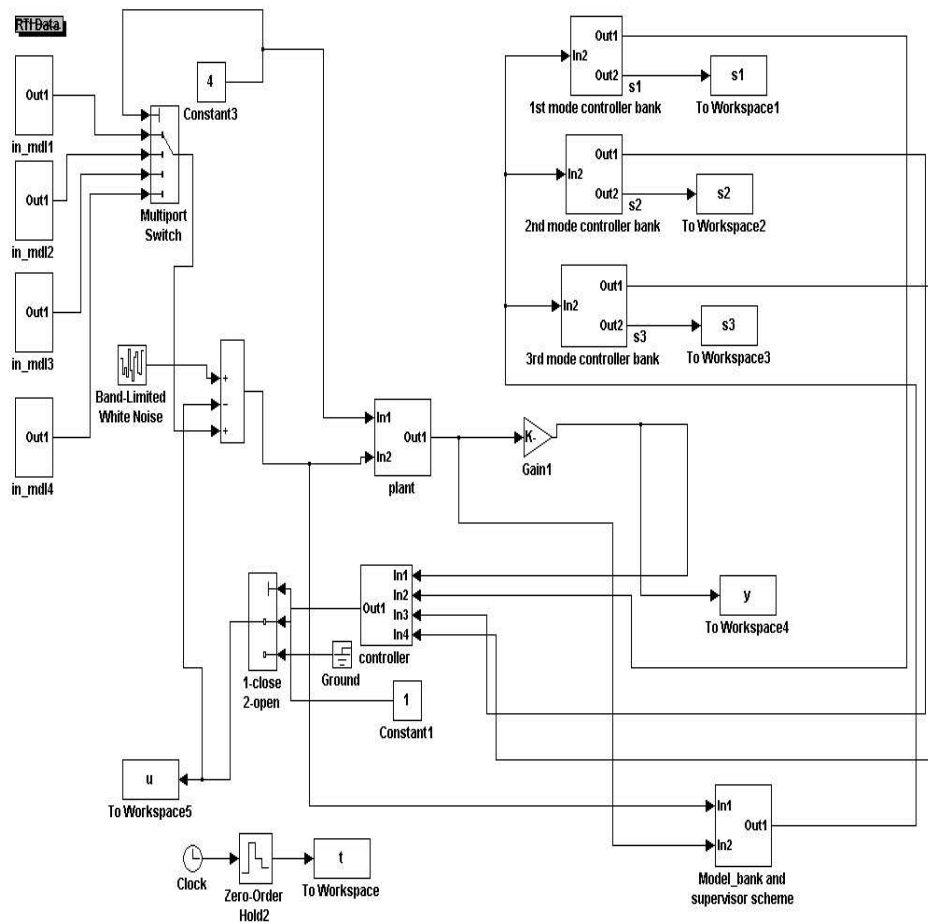


Figure C.6. Schematic diagram of *narendra_sim* model.

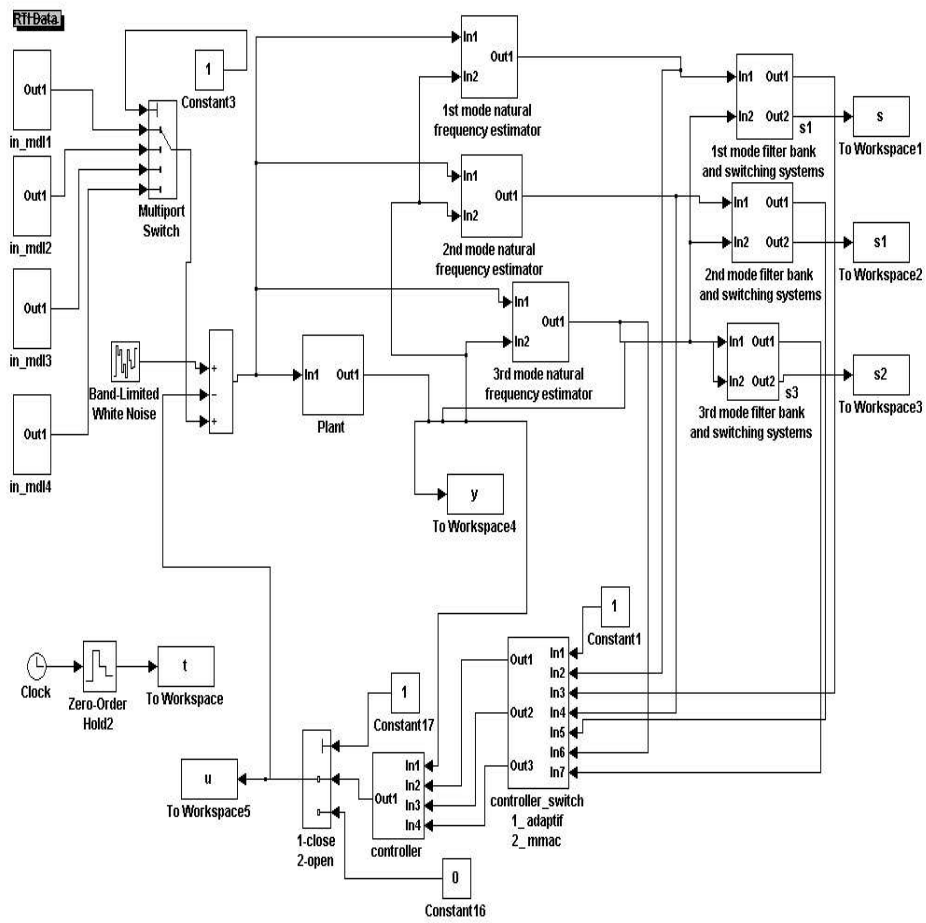


Figure C.7. Schematic diagram of *m4arc_sim* model.

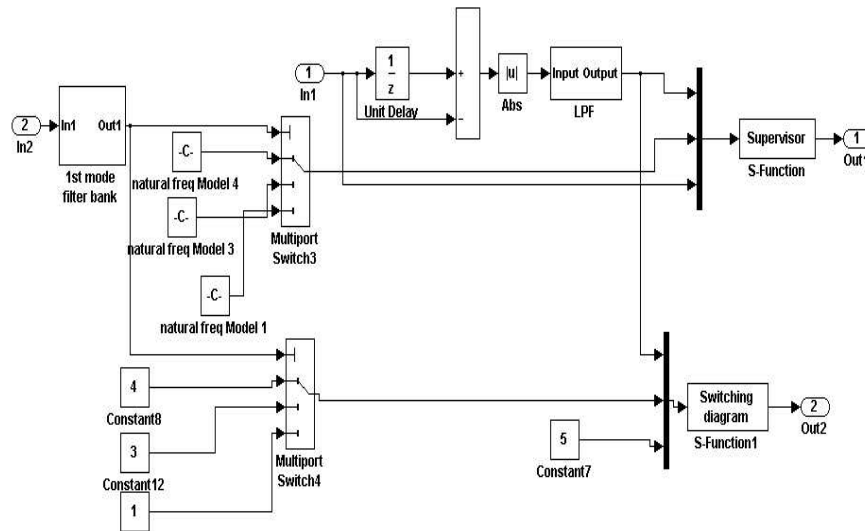


Figure C.8: Detail of filter bank and switching systems block of *m4arc_sim* model.

rc_m4arc_sim

The schematic diagram of the *rc_m4arc_sim* model is similar to the schematic diagram of the *m4arc_sim* model. The difference is that for the *rc_m4arc_sim* model the band-pass filters in the **1st mode filter bank** to **3rd mode filter bank** blocks are formed from the band-pass filter with resonant controller structure.

C.2 Experiments

The schematic of the experimental set-up is shown in Fig. 3.34. For the experiments, the controllers are implemented in the C-S-function in the Real-Time Workshop Simulink, and embedded into dSpace board to obtain a real-time implementation. All the files that need for compilation process are stored in the file with .dll extension. A bank of Band-Pass Filters (BPFs) with resonant controller structure for all the models is stored in the *rescon_exp.mdl* file. Five Simulink

model files are implemented for the experiments. The files and the explanation about the files are shown in Table C.3.

Model	Can be used for
<i>adapt_exp</i>	Simulate the adaptive resonant controller compared to fixed-parameter resonant controller. Test the natural frequency estimator.
<i>m4rc_exp</i>	Simulate the M ⁴ RC using Butterworth BPF in the filter bank.
<i>rc_m4rc_exp</i>	Simulate the M ⁴ RC using BPF with resonant controller structure in the filter bank.
<i>m4arc_exp</i>	Simulate the M ⁴ ARC using Butterworth BPF in the filter bank compared to fixed-parameter resonant controller, and compared to adaptive resonant controller.
<i>rc_m4arc_exp</i>	Simulate the M ⁴ ARC using BPF with resonant controller structure in the filter bank compared to fixed-parameter resonant controller, and compared to adaptive resonant controller.

Table C.3. Simulink model for experimental implementation.

The schematic of all the Simulink models for the experiments are similar to the corresponding model for the simulations. The difference is that in the experimental models, the plant block in the simulation models is replaced by a pair of ADC-DAC block. For example, the schematic of *adapt_exp* model is shown in Fig. C.9. From the figure it can be seen that the control signal and disturbance signal are fed into channel 1 DAC of the dSpace board (**DS1103DAC_C1** block), and the output from piezosensor is fed into channel 1 ADC of the dSpace board (**DS1103MUX_ADC_CON1** block).

In the experiments, the loading condition is changed by changing the constant value in the **Constant12** block. With the cable from loads M_1 and M_2 are connected to the switch box as in Fig. 3.34, Model 1, Model 2, and Model 4 is chosen by setting the constant value as 1, 2 and 3, respectively. To choose Model 3, first swap the cable from loads M_1 and M_2 so that M_1 connected to point B in the switch box and M_2 connected to point A. Then set the constant value in

the **Constant12** block to be 2.

For all the Simulink models for the experimental implementation set the solver type in the simulation parameters menu to fixed-step, and set the fixed step size to 1/1000.

Execution of an experiment and acquisition of an experimental data can be obtained by the following steps:

1. Open Matlab program.
2. Open a Simulink model file that want to be executed.
3. Open dSpace ControlDesk program.
4. With the Simulink as the active screen, press Ctrl B to start the compilation process. For success compilation Matlab editor will display :
Successful compilation of Real-time Workshop build procedure for model: file name. *** Finish RTI build procedure.
5. To start the experiment, choose Simulation in the Simulink window menu and select connect to target.
6. To be able to observe the variables and collect the experimental data, choose sdf file on the bottom of the screen on ControlDesk program.
7. On the file menu choose New and select new layout.
8. On data acquisition choose plotter for data observation and collection.
9. Drag the variable that want to be observed (collected) from the labels to the y-axis of the plotter.
10. Repeat steps 7 to 9 to observe (collect) more variables (one plotter for one variable).

11. Change the length to the desired observation time (in seconds).
12. Press animation button to start the observation (collection).

Note: Current Matlab directory must be set to be the same as the directory where the Simulink model resides.