Optimization in Active Vibration Control: Virtual Experimentation Using COMSOL Multiphysics - MATLAB Integration

Khairul A. M. Nor 1, Asan G. A. Muthalif 2, Azni N. Wahid 3
Smart Structures, Systems & Control Research Laboratory, Department of Mechatronics Engineering, International Islamic University Malaysia
Jalan Gombak, 53100, Kuala Lumpur, Malaysia
1 khairulaffendynmdnor@yahoo.com, 2 asan@iium.edu.my, 3 azni.nabela.wahid@gmail.com

Abstract—This paper demonstrates optimization of collocated sensor-actuator location and the controller gains of active vibration control system. Ant colony optimization algorithm is employed for this purpose. Instead of using equation-based modeling for the system, the plant is a finite element model developed in COMSOL Multiphysics software, which later interfaced with the MATLAB-coded optimization algorithm using the Liveliink for MATLAB feature. The benchmark model is a simply supported thin plate excited and attenuated by two piezoelectric patches. The optimization is based on the average energy reduction across a frequency range between 11 Hz to 50 Hz, which covers the first three modes. It is found that the maximum attenuation achieved is 68.31% using optimal values of sensor-actuator location and controller gains.

Keywords-COMSOL Multiphysics – MATLAB integration; active vibration control; Ant Colony Optimization

I. INTRODUCTION

Recently, many researchers look upon developing light-weight yet stronger and more flexible structures using smart materials such as piezoelectric material, shape memory alloy (SMA), magneto-rheological fluid (MRF) etc. Piezoelectric material has the capability to produce an electric field when deformed, and conversely deformed with the application of an electric field. By surface-bonding them to a structure such as plate and beam, a superior structure is produced, which is capable to give actuating force accordingly with the help of feedback control, thus making it the suitable choice as vibration suppressor. This strategy is called active vibration control (AVC).

Apart from the successes reported in optimization using common optimization techniques such as linear quadratic regulator (LQR) [1], H2 norms [2-3] and H (infinity) norm [4], the use of biologically-inspired optimization algorithm in AVC system has gained significant attention due to its reliability. One of them is Ant Colony Optimization (ACO) [5-7]. ACO algorithm is inspired by the behavior of real ants in which the ants uses pheromones as a communication medium in finding food source [8].

One of the powerful features in COMSOL Multiphysics software; a commercially available finite element package, is the ability to modify the parameters of its finite element model from within the MATLAB’s scripting environment via Liveliink for MATLAB function. This integration is realizable either by using MATLAB as scripting interface to set up and solve the COMSOL Multiphysics models, or by calling MATLAB functions when modeling within the COMSOL Multiphysics desktop. As a result, it is deemed possible to employ MATLAB-coded controller algorithm to virtually control vibration in a finite element model developed in COMSOL Multiphysics.

This paper presents optimization of sensor-actuator location and PID controller gains in AVC system. The benchmark model is a simply-supported rectangular plate attached with two piezoelectric patches (as vibration exciter and suppressor) developed in finite element software namely COMSOL Multiphysics. Optimization is conducted by interfacing the ACO algorithm written in MATLAB to the said benchmark model. This paper is arranged as follows. Section II presents the equation of motion (EOM) of the system. Integration between COMSOL Multiphysics and MATLAB software is discussed in detail in Section III. Section IV focuses on optimization and comprises of explanation on the ACO algorithm and objective function, as well as discussions on optimizing the location of the collocated sensor-actuator and the controller gains. The selected optimal values are based on the highest reduction of frequency-averaged energy.

II. MATHEMATICAL MODELING

The constitutive equation of piezoelectric material as actuator is [9]:

\[ \sigma = c^e \varepsilon - eE \] (1)

where \( \sigma \) is the stress vector, \( c^e \) is the matrix of elastic coefficients under constant electric field, \( \varepsilon \) is the strain vector, \( e \) is the dielectric permittivity matrix and \( E \) is the electric field vector. Considering the piezoelectric patch as a linear isotropic material, therefore (1) can be expanded to:
\[
\begin{bmatrix}
\sigma_{xx} \\
\sigma_{yy} \\
\sigma_{xy}
\end{bmatrix} = 
\begin{bmatrix}
\frac{E_p}{1-v_{pz}^2} & \frac{v_{pz}E_p}{1-v_{pz}^2} & 0 \\
\frac{v_{pz}E_p}{1-v_{pz}^2} & \frac{E_p}{1-v_{pz}^2} & 0 \\
0 & 0 & \frac{E_p}{2(1+v_{pz})}
\end{bmatrix} \begin{bmatrix}
f_{xx} \\
f_{yy} \\
f_{xy}
\end{bmatrix}
\]

where \(d_{31}\) and \(d_{32}\) are the piezoelectric strain constants (\(d_{32} = d_{31}\), because of isotropic), \(E\) and \(E_{pz}\) are the Young’s modulus for plate and piezoelectric patch, respectively, \(v\) and \(v_{pz}\) are the Poisson’s ratio for plate and piezoelectric patch, respectively, \(\epsilon(t)\) is voltage supplied to the piezoelectric actuator and \(t_p\) is the piezoelectric patch thickness.

In the case of a simply supported plate at all edges, the deflection of the plate during vibration is represented as the summation of modes in double series [10]:

\[
w(x,y) = \sum_{m} \sum_{n} W_{mn} \sin\frac{m\pi x}{a} \sin\frac{n\pi y}{b}
\]

where \(W_{mn}\) is the magnitude and subscripts \(m\) and \(n\) refer to half-wave number in the \(x\) and \(y\) directions, respectively. Lagrange’s method is employed to derive the equation of motion of a simply supported plate with attached piezoelectric patch. Thus, the equation of motion of plate with 2 patches is written as:

\[
(-\omega^2 M + K_C)w_{mn} = f_e + f_c
\]

where \(\omega\) is angular frequency in rad/s, \(M\) is mass matrix, \(K_C = (j+1)K\), \(K\) is stiffness matrix, \(\eta\) is modal loss factor, \(j\) is imaginary unit, \(w_{mn}\) is the modal amplitude vector, \(f_e\) and \(f_c\) are modal force vector generated by exciter and controller patches, respectively.

### III. Active Vibration Control Using COMSOL Multiphysics – MATLAB Integration

#### A. Building the benchmark model using COMSOL Multiphysics

A simply-supported rectangular thin plate attached with two piezoelectric patches is modeled in COMSOL Multiphysics and equipped with piezoelectric devices physics interface. Details of the said model are given in Table I. The exciter patch is powered by 100 V and located at \((0.15\, \text{m}, 0.15\, \text{m})\) while the controller patch’s location is determined using optimization technique described in Section 4. The model structure is meshed using triangular mesh with 3428 number of elements. Natural frequencies for the said structure are calculated using Eigenfrequency study and tabulated in Table II.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Frequency (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>16.561</td>
</tr>
<tr>
<td>2</td>
<td>36.924</td>
</tr>
<tr>
<td>3</td>
<td>46.162</td>
</tr>
<tr>
<td>4</td>
<td>66.697</td>
</tr>
<tr>
<td>5</td>
<td>71.662</td>
</tr>
</tbody>
</table>

The plate energy (denoted by \(EP\)) is calculated by integrating the energy density for the whole system, i.e. plate and piezoelectric patches; over the volume using the following equation.

\[
EP = \int \frac{1}{2} \rho(\dot{u}\dot{u} + \dot{\psi}\dot{\psi} + \dot{w}\dot{w})dV
\]

where \(\rho\) is the density, \(V\) is the volume, \(\dot{u}\), \(\dot{\psi}\) and \(\dot{w}\) are the velocity in \(x\), \(y\) and \(z\) directions, respectively. Thus, the frequency-averaged energy for a certain frequency range can be calculated using

\[
\overline{EP} = \frac{\sum_{m=1}^{M} EP(f)}{f_i-f_f}
\]

where \(\overline{EP}\) is the frequency-averaged plate energy, and \(f_i\) and \(f_f\) are the initial and final frequencies, respectively.

#### B. Interfacing of MATLAB-coded controller with COMSOL Multiphysics model

The benchmark model described in section 3.1 is loaded into MATLAB scripting environment using Livelink for MATLAB function. To actively control vibration of the benchmark model, a PID controller with collocated sensor-actuator configuration is developed in MATLAB script and used to manipulate the voltage of the controller patch in the model. Velocity at the sensor-actuator location for uncontrolled system is taken as the input of the controller, which is used to generate the voltage of the controller patch (denoted by \(V_o\)) as follows:

\[
V_o = k_p \cdot (0 - \dot{w}_{nc}) + k_i \cdot \int (0 - \dot{w}_{nc})dt + k_d \cdot \frac{d(0 - \dot{w}_{nc})}{dt}
\]
where $k_p$, $k_i$ and $k_d$ are the PID controller gains, and $w_{nc}$ is the velocity of uncontrolled system measured from sensor-actuator location. This interfacing is illustrated in Fig. 1.

![Integration of COMSOL Multiphysics and MATLAB.](image)

**IV. OPTIMIZATION**

A. Ant Colony Optimization algorithm

Simple-ACO (SACO), a variation of ACO, is employed in this paper. Global cooperation among ants in a colony can produce promising paths towards the food source by using pheromones as a communication medium. An ant deposits pheromone trails on the ground while looking for the food. The next ants will tend to follow the stronger trails and also reinforce the trail with their own pheromone, which consequently making the path more favorable as compared to others. Meanwhile, the weaker trails progressively decreased by evaporation.

The probability equation in SACO can be written as:

$$P_{ik}(t) = \begin{cases} \frac{[\tau_{ik}(t)]^\alpha}{\sum_{j \in T_k} [\tau_{ij}(t)]^\alpha}, & \text{if } j \in T_k \\ 0, & \text{otherwise} \end{cases}$$

(8)

where $\tau$ is the pheromone trail, $t$ is time, $\alpha$ is the constant that defines the relative importance of the pheromone values and $T_k$ is the path effectuated by the ant $k$ at a given time. The value of tour taken by each ant is:

$$\Delta \tau_{ijk}(t) = \begin{cases} Q/L_k, & \text{if } (i,j) \text{walked by ant } k \\ 0, & \text{otherwise} \end{cases}$$

(9)

where $Q$ is a constant and $L_k$ is the cost of tour by ant $k$ or the objective function. In this paper, $L_k$ is the inverse of energy reduction. The pheromone trails are updated using:

$$\tau_{ij}(t) = \sigma \tau_{ij}(t-1) + \sum_{k=1}^{N_A} \Delta \tau_{ijk}(t)$$

(10)

where $\sigma$ is the pheromone decay $0 < \sigma < 1$ to introduce the forgetfulness of the bad choices and $N_A$ is the number of ants. The ACO parameters used throughout the paper are listed in Table III.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of ants</td>
<td>5</td>
</tr>
<tr>
<td>Maximum generation</td>
<td>10</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>1</td>
</tr>
<tr>
<td>Pheromone decay, $\sigma$</td>
<td>0.95</td>
</tr>
</tbody>
</table>

B. Objective Function

The objective function in this paper is to maximize the energy reduction (denoted by $ER$), which can be defined as

$$\text{maximize } ER = \frac{EP_{nc} - EP_c}{EP_{nc}}$$

(11)

where $EP_c$ and $EP_{nc}$ are the average plate energy with and without control, respectively. The frequency range of interest is between 11 Hz to 50 Hz, which covers the first three modes.

All optimization are computed on a 2.20 GHz Intel® Core™ i7 processor with 8 GB of random access memory. Since optimization of five variables ($x, y, k_p, k_i$, and $k_d$) is computationally expensive, the process is conducted in two stages as follows:

1) Two variables, $x$- and $y$-coordinates of the collocated sensor-actuator $(x_{sa}, y_{sa})$, are optimized while applying simple velocity feedback control with an initial gain of 4850, which is approximately equal to 50 times of the infinite plate driving point impedance.

2) The remaining three variables, $k_p$, $k_i$, and $k_d$ of the PID controller are determined while fixing location of sensor-actuator with the values found in the first step.

C. Optimal sensor-actuator location

There is a very high chance that the optimization algorithm will end up choosing location near the exciter patch i.e. location of the disturbance; as the optimal sensor-actuator location. This is because the best way to reduce vibration is by directly locking the disturbance force [11]. However, this is not practical because in reality it is difficult to know the origin of disturbance. Having said that, it is made a condition that the controller patch must be placed 0.05 m apart along $x$ and $y$ axes from the exciter patch. The variables to be optimized i.e. $(x_{sa}, y_{sa})$ are coded by 10000 nodes making 10000 x 10000 possible combinations.

The energy plots obtained using different sensor-actuator location and the corresponding energy reduction with respect to the mode are shown in Fig. 2(a) and Fig. 2(b), respectively. Based on these figures, locations of sensor-actuator are varied if one wants to reduce vibration at a particular mode. For mode number 1, 2 and 3, the best sensor-actuator locations are at (0.2097 m, 0.2760 m), (0.3504 m, 0.4370 m) and (0.3504 m, 0.3901 m), respectively. However, considering broadband vibration attenuation within the frequency range of interest (11 Hz to 50 Hz), the optimal location to place the controller patch is at (0.3519 m, 0.4132 m) with frequency-averaged energy reduction of 41.51%. Fig. 3 shows the convergence plot for sensor-actuator location optimization.
Figure 2. (a) Energy plot and (b) energy reduction with respect to mode, using different sensor-actuator locations.

<table>
<thead>
<tr>
<th>Sensor-actuator location</th>
<th>Energy reduction (%)</th>
<th>1st mode</th>
<th>2nd mode</th>
<th>3rd mode</th>
<th>Frequency-averaged</th>
</tr>
</thead>
<tbody>
<tr>
<td>(xsa,ysa) = (0.0800, 0.0800)</td>
<td>5.94</td>
<td>21.40</td>
<td>15.05</td>
<td>10.51</td>
<td></td>
</tr>
<tr>
<td>(xsa,ysa) = (0.2097, 0.2760)</td>
<td>56.12</td>
<td>9.83</td>
<td>25.71</td>
<td>13.49</td>
<td></td>
</tr>
<tr>
<td>(xsa,ysa) = (0.3048, 0.2297)</td>
<td>46.17</td>
<td>51.98</td>
<td>40.94</td>
<td>28.40</td>
<td></td>
</tr>
<tr>
<td>(xsa,ysa) = (0.3504, 0.3921)</td>
<td>33.46</td>
<td>79.53</td>
<td>71.00</td>
<td>40.12</td>
<td></td>
</tr>
<tr>
<td>(xsa,ysa) = (0.3504, 0.4370)</td>
<td>26.19</td>
<td>77.58</td>
<td>56.28</td>
<td>40.92</td>
<td></td>
</tr>
<tr>
<td>(xsa,ysa) = (0.3519, 0.4132)</td>
<td>29.90</td>
<td>70.04</td>
<td>65.43</td>
<td>41.51</td>
<td></td>
</tr>
</tbody>
</table>

### D. Optimal PID controller gains

Optimization of PID gains is conducted while considering the optimal sensor-actuator, which is (0.3519 m, 0.4132 m). The range of values for $k_p$, $k_i$, and $k_d$ obtained are 1000 to 20000, 1000 to 2000 and 0.1 to 10, respectively. Each parameter is coded by 10000 nodes making 10000 x 10000 x 10000 possible combinations.

Figs. 4(a) and 4(b) illustrate the energy plots obtained using different PID gains while fixing the controller patch at its optimal location and the bar graph of the corresponding amount of energy attenuation in percentage for each mode, respectively. Overall, the result presented in Table V shows improvement in energy reduction as compared to attenuation using velocity feedback controller in the previous section. This is proven by the increasing of the maximum frequency-averaged energy reduction to 68.31 % by tuning the $k_p$, $k_i$ and $k_d$ values to 12211, 11511 and 4.6158, respectively. For this controller optimization, the convergence profile of the objective function is depicted in Fig. 5.

<table>
<thead>
<tr>
<th>PID gains</th>
<th>Energy reduction (%)</th>
<th>1st mode</th>
<th>2nd mode</th>
<th>3rd mode</th>
<th>Frequency-averaged</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k_p=4850$, $k_i=1000.0$, $k_d=0.1000$</td>
<td>29.91</td>
<td>70.00</td>
<td>65.40</td>
<td>41.56</td>
<td></td>
</tr>
<tr>
<td>$k_p=9948$, $k_i=4243.6$, $k_d=4.3594$</td>
<td>54.65</td>
<td>97.07</td>
<td>95.52</td>
<td>65.35</td>
<td></td>
</tr>
<tr>
<td>$k_p=13448$, $k_i=17819$, $k_d=3.4198$</td>
<td>67.81</td>
<td>88.97</td>
<td>88.74</td>
<td>68.07</td>
<td></td>
</tr>
<tr>
<td>$k_p=12211$, $k_i=10309$, $k_d=4.7634$</td>
<td>63.58</td>
<td>94.64</td>
<td>93.80</td>
<td>68.30</td>
<td></td>
</tr>
<tr>
<td>$k_p=12211$, $k_i=11511$, $k_d=4.6158$</td>
<td>63.55</td>
<td>94.62</td>
<td>93.81</td>
<td>68.31</td>
<td></td>
</tr>
</tbody>
</table>

![Figure 3](image_url)  convergence plot for optimization of sensor-actuator location.
V. CONCLUSION

In this paper, optimization of collocated sensor-actuator location and PID controller gains using COMSOL Multiphysics – MATLAB integration is successfully performed. The time taken to complete each optimization is about the same, i.e. 8.25 hours. It is expected that, better results can be obtained if more ‘ants’ and number of iterations in ACO algorithm are used, however the computational cost will inevitably be high.

REFERENCES