DIRECT CLOSED-LOOP ACTIVE VIBRATION CONTROL SYSTEM PROTOTYPING IN ANSYS

Gergely Takács[1*] and Boris Rohaľ-Ilkiv[1,2]

[1]Institute of Automation, Measurement and Applied Informatics
Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava
Nám. slobody 17, Bratislava 812 31, Slovakia
Email: gergely.takacs@stuba.sk

[2] Email: boris.rohal-ilkiv@stuba.sk

ABSTRACT

Designing an active vibration control (AVC) system involves numerous important engineering decisions in the planning phase. Finite element modeling (FEM) software is traditionally used to configure the components of smart systems and test their dynamic properties. Although modern FEM software allows the integration of the electro-mechanical parts into the AVC system, these do not routinely implement tools to evaluate control systems in conjunction with the model. As the dynamic effect of the applied control law is the most important aspect AVC systems, direct prototyping of controllers inside FEM software can be an essential tool for engineering practitioners.

This paper proposes a direct closed-loop prototyping method for controllers in AVC systems. A clamped-free beam with piezoelectric actuation is modeled in ANSYS, and verified using an experimental measurement — first in open-loop, without control. A linear-quadratic (LQ) controller with a Kalman-filter is then implemented to this FEM model using the APDL language. The simulated closed-loop transient response is contrasted to the experimental results, confirming the validity of the approach by matching simulated and measured responses. It is concluded that APDL can be effectively used to integrate controllers into ANSYS models of active vibration control systems.

1 INTRODUCTION

Although the concept of finite element analysis (FEA) and the finite element method (FEM) has been established as early as the time of World War II [1], its proliferation to everyday engineering analysis and product development was made only possible by the development of mainstream computing hardware. Tools for vibration analysis are now a part of most off-the-shelf FEM packages, allowing engineers to investigate the vibration response of concept products using modal, harmonic and transient tests. One particular area in which readily available FEA software lags behind current trends is, the integration of the tools of the trade of the control engineering specialist. Though this is also true for most fields where FEA is applied, vibrating mechanical systems are being converted to complex *smart* structures. These smart structures integrate sensors, actuators, computing hardware and control logic to create systems enabled with active vibration control (AVC) [2, 3].

Elements allowing the simulation of coupled physical effects, such as the electro-mechanical interaction of piezoceramics, are routinely implemented in most modern FEM packages. However, their use is limited to passive open-loop tests, without assessing the effect of the control system itself. The traditional prototyping approach involves dividing the design process into two stages: creating an open-loop FEA model and performing a simple system identification procedure, then constructing a closed-loop control system using a simplified mathematical model of the structure [4]. Unlike the highly integrated physical system, fine-tuning of the structural and control design becomes a separate process.

Integration of the control system inside the FEA model not only simplifies the design process by eliminating a need of a separate design stage, but also removes the need to use a severely truncated version of the full model. Although the design and verification of controllers using process models yields almost instantaneous results, the level of precision may not be sufficient for all prototyping tasks. In case the control system is implemented inside the FEA software, one may assess the effect of the controller on a model that represents the real system much better than the process model. Moreover, given the controller is tuned or designed within the FEA procedure, it is possible to directly refine the controlled structure alongside with the controller. Unfortunately, such a one-stage and direct structural/control design process is not a standardly available option in mainstream FEM software.

Most researchers adopted this detached design flow: create a numerical model of the structure and possibly optimize actuator locations in open-loop, then based on a process model design a controller separately. Recently however, numerous articles appeared in which the control prototyping of smart systems has been integrated into the FEM simulation. Dong et al. used the linear quadratic Gaussian algorithm in a closed-loop numerical simulation and then compared the results to experimental measurements with good results [5]. A PID controller has been used in ANSYS by Xiaojin et al., verifying actuator placement on a plate with vibration control [6]. In a more elaborate article, Malgaca has implemented direct velocity control (DVC) to suppress vibrations of a composite cantilever beam. These experimental results were subsequently matched to the simulations performed using DVC applied to an ANSYS simulation loop [7]. The authors reported a good match between the FEA and experimental results, while noting the promising potential of integrating control systems into FEA software.

In a previous work, a PID controller was implemented in ANSYS APDL, designed to govern the maximal allowable temperature on a pipe while subjected to in-service maintenance welding [8]. In this paper, we will propose a method to implement a linear quadratic controller (LQ) inside the ANSYS FEM software package. This LQ controller will control the input voltage on the piezoceramic actuators embedded in a smart structure with active vibration control. The validity of the approach will be confirmed by contrasting experimental measurement results with the simulations performed by the integrated FEM / control system.

2   THE VIBRATION CONTROL SYSTEM AND ITS FEM MODELING

2.1   Experimental hardware

The experimental hardware featured throughout this paper is the classic example of a clamped cantilever beam. The dimensions of the beam are $550 \times 40 \times 3$ mm and is made of commercially pure aluminium. The beam is fixed to a clamping system, resting on a heavy base. This cantilever beam is equipped with several piezoelectric transducers, of which only the two closest to the clamped end are used as actuators[1]. The transducers were manufactured by Midé Technology and are bearing the product designation QP16n. The type of piezoceramic used is PZT5A and the size of the actuator (without its thin film casing is) $45.9 \times 20.7 \times 0.25$ mm. The two actuators are co-located on the opposite sides of the beam, and receive the same signal with a reversed polarity.

---

[1]The active transducers are bonded 14 mm from the clamped edge, inactive at 90 and 320 mm.

2.2 Finite element modeling

2.2.1 Theoretical foundations

Let us begin the discussion by recapitulating the foundations for creating an equation of motion for systems with piezoelectric actuation. One may express the equation of motion from a purely mechanical viewpoint for a multiple DOF (or discretized) system by using [3]:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{B_s}\dot{\mathbf{q}} + \mathbf{K_s}\mathbf{q} = \mathbf{f_e} \tag{1}$$

where matrix $\mathbf{M}$ is the mass matrix, $\mathbf{B_s}$ is the structural damping matrix, $\mathbf{K_s}$ is the structural stiffness matrix and $\mathbf{f_e}$ is an external force supplied to the system, for example resulting from an actuator. Vector $\mathbf{q}$ contains the displacement coordinates for the individual DOF. Of course, for a mechanical system including piezoelectric actuators or sensors, one has to account for the coupled electrical and mechanical effects. The piezoelectric constitutive equations in the stress-charge form are given by [2, 9]:

$$\sigma = \mathbf{c}_E\mathbf{S} - \mathbf{e_p}\mathbf{E_e} \tag{2}$$
$$\mathbf{D_e} = \mathbf{e_p}^T\mathbf{S} + \epsilon_S\mathbf{E_e} \tag{3}$$

where $\mathbf{S}$ is strain, σ is stress, $\mathbf{D_e}$ is the electric displacement, $\mathbf{E_e}$ is the electric field strength, $\mathbf{e_p}$ are the piezoelectric coupling coefficients in the stress-charge form, $\mathbf{c}_E$ contains stiffness coefficients under constant electric field and ϵ_S is the electric permittivity matrix under constant stain. Subscripts $_E$ indicate zero or constant electric field and $_\sigma$ zero or constant stress field, while superscript ^{T} denotes matrix transposition [10]. This can be compacted in a matrix form to yield:

$$\begin{bmatrix} \sigma \\ \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{c}_E & \mathbf{e_p} \\ \mathbf{e_p}^T & -\epsilon_S \end{bmatrix} \begin{bmatrix} \mathbf{S} \\ -\mathbf{E} \end{bmatrix} \tag{4}$$

Using the variational principle and applying element discretization, we may arrive at

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\mathbf{p}} \end{bmatrix} + \begin{bmatrix} \mathbf{B_s} & \mathbf{0} \\ \mathbf{0} & \mathbf{B_p} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{p}} \end{bmatrix} + \begin{bmatrix} \mathbf{K_s} & \mathbf{K_z} \\ \mathbf{K_z}^T & \mathbf{K_p} \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_e \\ \mathbf{L_e} \end{bmatrix} \tag{5}$$

where $\mathbf{B_p}$ is responsible for electric damping (dielectric loss), $\mathbf{K_p}$ is the electric stiffness (dielectric permittivity coefficient matrix) and $\mathbf{K_z}$ the piezoelectric coupling matrix [11]. Vector variables $\mathbf{q}$ and $\mathbf{p}$ express structural and electrical degrees of freedom.

2.2.2 Practical realization

The experimental hardware was modeled in the ANSYS FEA software package. The beam and the piezoelectric materials have been modeled by the SOLID5 element, which has 8 nodes. Only the displacement degrees of freedom (DOF) were used for the beam, while the numerical representation of the piezoceramic transducers employed the electrical DOF as well.

The solid volumetric model of the beam and the transducers was created from block shapes, using the *BLC4* command. After creating the main blocks, the beam was sectioned along the piezoelectric transducers using rotated work planes. Material properties have been assigned using data provided by the manufacturer of the piezoceramics and commonly available engineering tables [12]. The volumetric model has been meshed by a 8 mm maximal global element size. Because the beam was sectioned to sub-blocks, the nodes in the piezoceramics line up perfectly with the nodes of the beam located underneath. This way adjacent nodes may be joined by the *NUMMRG* command, which is aimed at merging nodes located within a tolerance range. The merged nodes will behave as one in simulation, meaning that if the piezoceramic material deforms, so will the

(a) FEM of the cantilever beam showing elements   (b) Detail of the model and boundary conditions
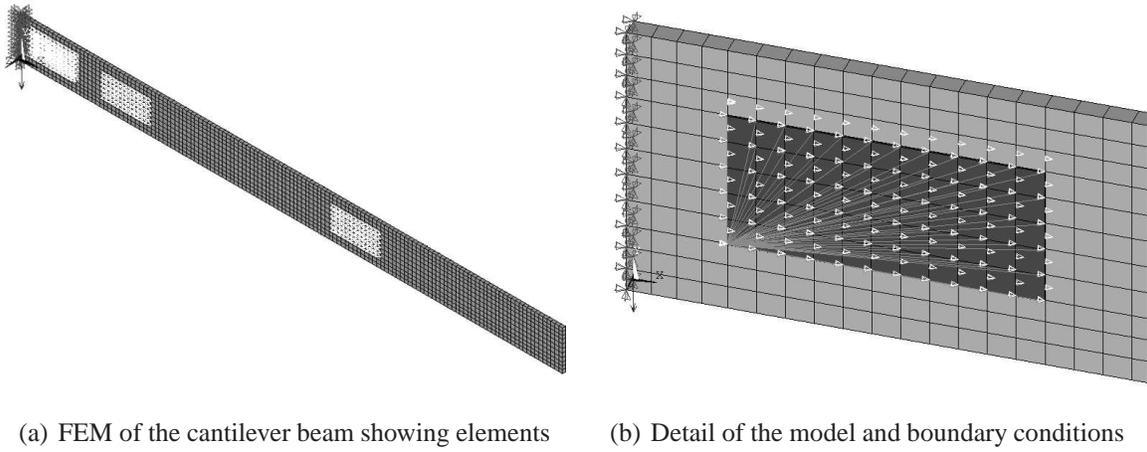
Figure 1: The finite element model of the clamped cantilever beam equipped with piezoceramic patches is shown in (a), while a detailed view of the actuator and its coupled DOF is given in (b)

beam underneath. This assumes a perfect bond between the beam and the actuators, discarding the minor effects the thin layer of resin might have on the behavior of the structure.

At the fixed end of the beam, zero displacement DOF is specified for the x, y and z directions, thereby simulating the effect of the clamp. Gravitational pull is taken into account by specifying an acceleration vector in the z direction. Finally, the electrodes on the piezoceramics are modeled by defining the appropriate boundary conditions. The upper layer of nodes creates the top (outwards) electrode on each transducer, where a coupling is created by the CP command and the appropriate voltage potential is assigned. The electrodes that will be engaged in the control system have this value constantly changing, in fact, this is the where the input from the controller will be utilized later. The unused transducers are short-circuited, which is represented by a zero voltage potential on each node in top layer. Electrodes on the bottom (bonded) side are defined in a similar fashion. Only this time, a fixed zero voltage potential is defined on the bottom layers of all transducers, representing the ground terminal of the actuators.

The FEA model of the cantilever beam is illustrated in Figure 1. The whole beam is shown in Figure 1(a), where the distribution and relative placement of actuators is featured. The symbols denoting the different boundary conditions are depicted in a close-up detail in Figure 1(b), where the darker triangle symbols on the left denote the zero displacement DOF (clamping) and the lighter triangle symbols on the right and their connection denotes the definition of the coupled electric DOF (electric terminals).

Rayleigh damping was used in the finite element analysis presented here, assuming that the damping matrix is constructed as the weighted sum of mass and stiffness matrices:

$$\mathbf{B_s} = \alpha\mathbf{M} + \beta\mathbf{K_s} \tag{6}$$

where α and β are damping constants. These may be calculated from the modal damping ratios or alternatively, chosen to suit the given application. All simulations featured in this paper assume $\alpha = 0$ and $\beta = 0.00015$ damping constants. The choice for the seemingly small β coefficient is confirmed by the correctness of both the open- and the closed-loop verification experiments presented in the upcoming sections.

3 THE CLOSED-LOOP CONTROL SYSTEM AND ITS FEM MODELING

3.1 Linear quadratic control

3.1.1 Theoretical foundations

A linear quadratic (LQ) controller attempts to minimize the overall control energy, indicated by a cost function given in a quadratic form. When multiplied by the state, the resulting feedback matrix yields an input which will keep the cost function at its minimum. Let us consider a discrete, linear time-invariant state-space system defined by [3, 4, 13]:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \tag{7}$$
$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k \tag{8}$$

where $\mathbf{x}_k$ is the state at time (k), $\mathbf{y}_k$ is the output at time (k), $\mathbf{A}$ is the state transition matrix, $\mathbf{B}$ the input matrix, $\mathbf{C}$ the output matrix and $\mathbf{D} = 0$ is the direct input feed-through matrix.

The LQR controller is the feedback gain $\mathbf{K}$ that minimizes the infinite horizon cost function, which uses a weighted square of the current state and current input to express cost [13, 14]:

$$J = \sum_{k=0}^{\infty} \left(\mathbf{x}_k^T \mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^T \mathbf{R}\mathbf{u}_k \right) \tag{9}$$

where $\mathbf{Q}$ is a positive semi-definite state weighting matrix, $\mathbf{R}$ is a positive definite input weighting matrix and $\mathbf{P}_f$ is a terminal weighting matrix. The output voltage at the piezoelectric actuators is defined as the multiple of the LQ gain and the currently measured or observed system state:

$$\mathbf{u}_k = -\mathbf{K}\mathbf{x}_k \tag{10}$$

Minimizing the cost function defined by Equation (9) results the LQ feedback gain [14, 15]:

$$\mathbf{K} = (\mathbf{R} + \mathbf{B}^T \mathbf{P}\mathbf{B})^{-1}\mathbf{B}^T \mathbf{P}\mathbf{A} \tag{11}$$

where $\mathbf{P}$ is the solution of the discrete time algebraic Ricatti equation (DARE) defined by [14, 15]:

$$\mathbf{P} = \mathbf{Q} + \mathbf{A}^T \left(\mathbf{P} - \mathbf{P}\mathbf{B} \left(\mathbf{R} + \mathbf{B}^T \mathbf{P}\mathbf{B} \right)^{-1} \mathbf{B}^T \mathbf{P} \right) \mathbf{A} \tag{12}$$

3.1.2 Practical realization

In order to compute the LQ gain introduced by Equations (9)-(12), a simplified state-space process model has to be defined according to Equation (7). This process model may be acquired based on a first principles model, FEA results, reduced and truncated FEA matrices or experimental identification results. This work utilizes the results of an experimental identification procedure, in which the piezoelectric actuators used a chirp signal to excite the beam in the bandwidth of interest. The resulting input and output data has been identified using the Matlab System Identification Toolbox, using subspace identification methods [16]. The data set was estimated using N4SID algorithm [16] and resulted in a loss function of 0.015 and a final prediction error (FPE) of 0.014. The identification procedure yielded the following second order process model:

$$\mathbf{A} = \begin{bmatrix} 0.867 & 1.112 \\ -0.214 & 0.870 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 9.336\text{E}{-2} \\ 5.301\text{E}{-2} \end{bmatrix} \quad \mathbf{C} = [-0.553 \quad 0.705] \tag{13}$$

Computation of the controller gain can be automated in Matlab, given one has already defined the mathematical model of the system and the tunable penalty matrices $\mathbf{Q}$ and $\mathbf{R}$. Matrix

P can be computed using the command *dare* based on Equation (12) and then substituted to (11) to get the LQ gain. Alternatively, these two steps can be performed in one stage by using the *dlqr* command. The control gain calculated for the smart beam featured in this system is:

$$\mathbf{K} = \begin{bmatrix} 12.97 & 125.50 \end{bmatrix} \tag{14}$$

In addition to the controller gain, an observer must be created to estimate system states $\mathbf{x}_k$, based on output measurements $\mathbf{y}_k$. The discrete Kalman estimator can be computed by using the *kalman* command in Matlabt that—amongst others—results in the Kalman innovation gain **M**, which in this case was computed as:

$$\mathbf{M} = \begin{bmatrix} -4.337 \\ 0.354 \end{bmatrix} \tag{15}$$

Discarding process and measurement noise results in a simple observer equation, given by [3]:

$$\tilde{\mathbf{x}}_k = \mathbf{A}\tilde{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{M}(\mathbf{y} - \mathbf{C}\tilde{\mathbf{x}}_{k-1}) \tag{16}$$

where $\tilde{\mathbf{x}}_k$ denotes the current state estimate, while $\tilde{\mathbf{x}}_{k-1}$ denotes the state estimate in the previous iteration and matrices **A**,**B**,**C** represent the state-space model of the system.

3.2 Closed-loop FEM in ANSYS

We begin our discussion on implementing a simple LQ controller into ANSYS by assuming that the solid model has been created, meshed and boundary conditions have been defined according to the discussion presented previously in Section 2.2.2.

The controller implementation in the ANSYS parametric design language (APDL) involves utilizing the commands for elementary matrix operations and basic program flow controls. Given that a process model of the the discrete linear state-space system defined by **A**,**B**,**C** is available, moreover the LQ gain **K** has been computed along with the Kalman filter innovation gain **M**, these variables must be stored in a matrix form. Matrices are sized using the **DIM* command, while their elements can be defined using the **SET* command in APDL. Along with the required controller and observer parameters, certain temporary matrix variables must be defined.

Another necessary initialization task is defining the type of the analysis, which in this case is a transient analysis. This definition is identical to the customary steps used in ANSYS for transient tests [11]:

```
/SOLU
ANTYPE,4
TRNOPT,FULL,,DAMP
LUMPM,0
DELTIM,Tres
TINTP,,0.25,0.5,0.5
TIME,Tstep
```

At the starting point of the simulation, we define a boundary condition emulating an initial force effect that in turn causes the beam to statically deflect to the desired initial deflection position. This very first step in the transient solution is solved, and the initial force effect is removed from the test, the nodal deflection is retrieved and converted into mm:
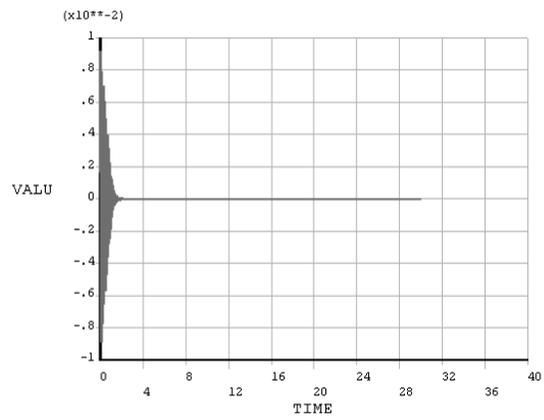
```
F,350,FZ,frce
KBC,1
SOLVE
FDELE,350,ALL
*GET,y,NODE,351,U,Z
y=y*1000
```
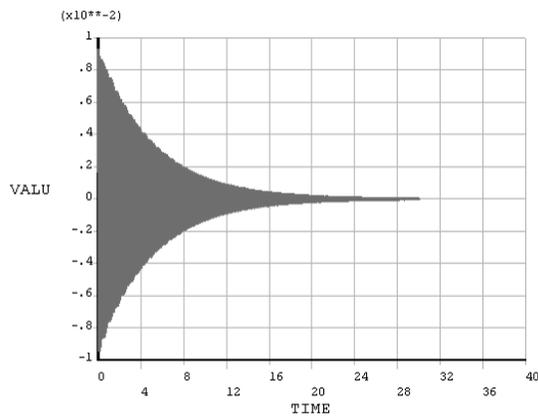
(a) Open-loop response (no control)

(b) Closed-loop response (with control)

Figure 2: The figure shows the results of the open-loop analysis without control in (a), while (b) denotes the closed-loop response without control. The default ANSYS output uses *TIME* to denote time in s and *VALU* as the deflection in m at the beam tip

In order to compare the results with the experimental ones, the steps are written to a comma separated value (CSV) file. The file is created by the command *CFOPEN*[2] while lines are added using the *VWRITE* command augmented by formatting instructions:

```
*CFOPEN,results,csv
*VWRITE,tstep,y,u,0,0
\%0.0e,\%0.0f,\%0.0f,\%0.0f,\%0.0f,
```

The main controller algorithm can be called by a programming cycle, realized by the *DO–*ENDDO* commands. This cycle uses the time variable as its index variable, while the step size is the discrete sampling time. At each cycle, the transient simulation time is set, the nested controller macro is used

```
*DO,t,2*Tstep,Tstop,Tstep
        TIME,t
        *USE,controller.mac
        *VWRITE,t,y,u,x1,x2
        \%0.0e,\%0.0f,\%0.0f,\%0.0f,\%0.0f
        SOLVE
*ENDDO
```

The sensor is emulated by reading out the nodal position at the end of the beam:

```
*GET,y,NODE,351,U,Z
y=y*1000
```

The state observer is implemented according to Equation (16) by using the elementary matrix manipulation commands in APDL, for example *MOPER* for multiplication, *VFACT* for scaling, *VFUN* for copying and *VOPER* for adding:

```
*MOPER,t1,A,MULT,x
*VFACT,u
*VFUN,t2,COPY,B
*MOPER,t3,C,MULT,x
t3(1,1)=y-t3(1,1)
t3B=t3(1,1)
```

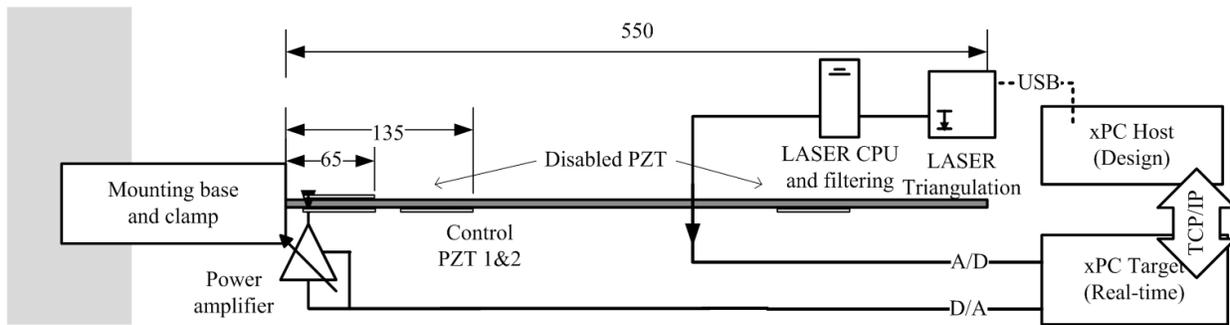[2]The file needs to be closed by a *CFCLOSE* at the end of the simulation.

Figure 3. Simplified experimental connection scheme of the smart cantilever beam

```
*VFACT,t3B
*VFUN,t4,COPY,M
*VOPER,t4,t2,ADD,t4
*VOPER,x,t1,ADD,t4
```

The implementation of the LQ controller itself is very simple and it is based on Equation (10) involving only a matrix multiplication to compute the input $\mathbf{u}_k$, based on the estimated state $\tilde{\mathbf{x}}_k$:

```
*MOPER,ut,K,MULT,x
u=-ut(1,1)
```

The saturation used as a safety measure is solved by a simple *IF–*ELSEIF condition:

```
*IF,u,LT,uL,THEN
 u=uL
*ELSEIF,u,GT,uH
 u=uH
*ENDIF
```

Finally, both of the actuators are set to receive the desired voltage by invoking *d,p1,VOLT,u,0* and *d,p2,VOLT,u,0* . The results of the open-loop simulation in ANSYS are shown in Figure 2(a), while the closed-loop simulation is featured in Figure 2(b).

4 EXPERIMENT

4.1 Experimental setup

The experiments utilized the device described in Section 2.1, with the additional piezoelectric transducers short-circuited in order to prevent electro-mechanical interaction with the structure. Only the two collocated transducers closest to the clamped end were engaged. Deflections at the beam tip were measured using a Keyence LK-G82 industrial laser triangulation system, which—using its proprietary LK-G3001V central processing unit—supplies an analog voltage signal scaled linearly to the the true deflection measured in mm. Power to the pair of piezoelectric actuators is supplied via the Midé Technology EL-1225 operational amplifier capable of driving high-capacitance loads. The amplifier is set to a constant $20\times$ gain and uses an analog voltage input. The analog voltage output from the distance measurement system and the analog input to the actuators is connected to a National Instruments BNC-2111 connector board, which in turn supplies signals to and from a National Instruments PCI-6030E laboratory measurement card. This measurement card is located in a *Target* computer running the Matlab xPC Target rapid control prototyping platform, which in turn is controlled using a *Host* computer running Matlab Software. The Target and Host computers are communicating via the TCP/IP protocol. The simplified experimental connection scheme is featured in Figure 3.
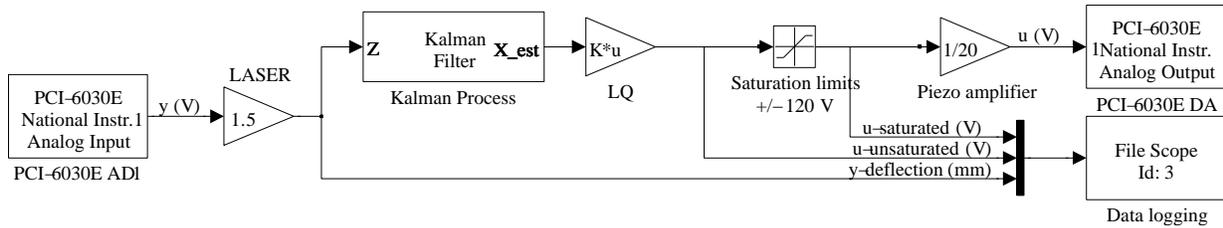
Figure 4. Block scheme of the linear quadratic control system used in the experiments

The linear quadratic control logic was implemented as a Matlab Simulink block scheme, then it was subsequently translated automatically into the C language. Figure 4 shows the resulting block scheme that was used in the experimental test. The scheme reads its input from the A/D port of the measurement card and then compensates for scaling. This signal is fed to a Kalman filter, creating estimates of the state, which is then multiplied by the fixed LQ gain to create inputs. The raw inputs are saturated, in order to prevent de-polarization of the piezoceramic layer. The signal is then compensated for the gain of the operational amplifiers and set to the D/A port of the measurement card. Variables are logged into the memory of the xPC Target system and then saved to its hard drive for later retrieval.

4.2   Experimental procedure

To perform the open- and closed-loop experiments, the tip of the beam has been deformed to $+10$ mm then released without any additional outside force interaction. Due to this initial condition, the beam starts to vibrate and in the open-loop scenario eventually settles as a result of its internal damping. Because the system is very lightly damped, the settling time of the open-loop experiment was well in the excess of 30 s. The closed-loop tests involved engaging the control system and repeating the same initial condition. The active vibration control system shortened the settling time by more than an order of magnitude, effectively rendering the blade motionless after 2-3 s.

5   RESULTS

5.1   Open-loop response

Figure 5 shows the result of the open-loop experiment, with the control system turned off. The solid black line represents the experimental results, the gray dash-dot corresponds to the ANSYS simulation results, the x axis denotes time, while the deflection is shown on the y axis in mm. The full 30 s response is featured in Figure 5(a), while a closer detail of the same measurement is provided in Figure 5(b). The open-loop sum of squares error (SSE) computed from the experimental measurement is 9 240 (-). As it is clearly visible from the figure, the response of the open-loop model matches the measured response very well. There are practically no differences between the responses, the modeling assumptions have been right. The open-loop finite element model is correct, and suitable to contrast closed-loop responses.

5.2   Closed-loop response

The simulation results are compared to the experimental measurements on Figure 6, this time with the control system engaged. The top figure contrasts the measured deflection outputs in mm (Figure 6(a)), while the bottom illustrates the control input voltage (Figure 6(b)).

The first thing to notice is the amount of damping provided by the control system. The open-loop response settles well in the excess of 30 s, while the closed-loop vibrations return to equilibrium much faster. By using active vibration control, a more than 10 times improvement can
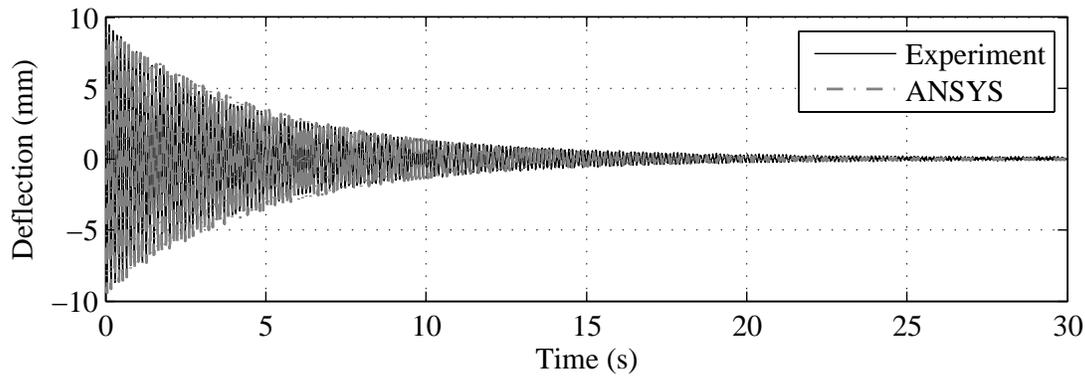
(a) Open-loop output responses (deflection)
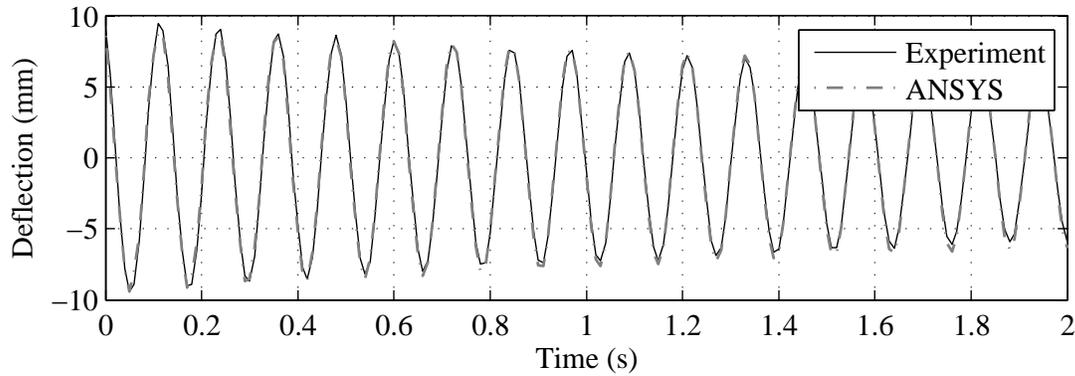
(b) Detail of the open-loop responses (deflection)

Figure 5: The figure shows the response of the cantilever beam to an initial deflection without control. The experimental measurement of the settling of the tip position is compared to the ANSYS results in (a), while (b) illustrates the first 3 seconds detail of the same response.

be observed in the total settling time of the beam tip vibrations. The damping improvement is also clear by the examining the SSE: the open-loop system produced 9240 (-), while in close-loop this fell to 1724 (-) for the ANSYS simulation, and 2105 (-) for the experiment.

Comparing the outputs $\mathbf{u}_k$ in Figure 6(a) reveals a very good match of the simulated and measured course of deflections. Assuming to have a correct open-loop finite element model, a simulation with built-in control logic emulates the behavior of the real system accurately. The input voltages supplied to the piezoelectric actuators depicted in Figure 6(b) demonstrate an excellent match as well. The only difference is the evolution of inputs, when the beam tip approaches its mechanical equilibrium. The real model is subject to slight and unintentional outside disturbances: for example heat convection in the room, mechanical noise due to people present in the laboratory, road traffic etc [10]. This difference cannot be detected by looking at the outputs, since the real controller compensates these errors, but due to this compensating action, the inputs must be different.

6 CONCLUSIONS

A method of implementing a linear quadratic controller in the ANSYS APDL macro language was introduced here. This controller was used to simulate the transient response of a smart structure under active vibration control. Based on the comparison of simulated and experimentally measured results presented in this work, we may state that the proposed simulation procedure resulted in a very close match between FEA and reality. Such a one-stage simulation method may be beneficial in the prototyping of complex shaped structures, where a simple—possibly linear or linearized—
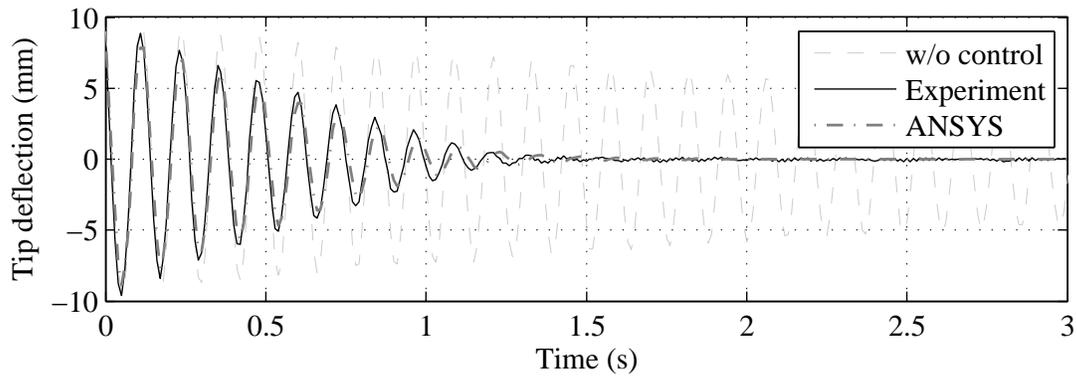
(a) Closed-loop outputs (deflection)
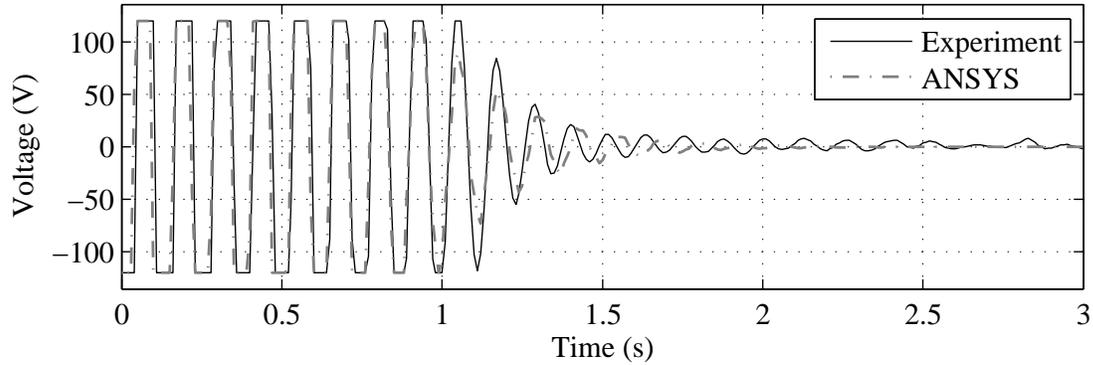
(b) Closed-loop inputs (input voltage)

Figure 6: The closed-loop simulated and measured deflection is contrasted to the free response in (a), while the corresponding inputs are shown below in (b)

process model cannot fully represent the real plant. Moreover, this integrated FEM and control prototyping process may yield better results than any of these methods alone.

The increased precision, however, comes with a price. A process model used in Matlab to represent the controlled plant allows the fast reconfiguration of the controller and almost instantaneous simulation results. The integrated FEM and control design requires to run full transient tests, which can be especially lengthy for complex systems. Therefore, our general recommendation regarding the integrated design of control systems in FEA software is the following:

- Create the open-loop FEA model with integrated actuators

- Extract a process model from FEA (results from open-loop simulations or model reduction)

- Use a process model to design, tune and verify your controller *outside FEA* software (eg. in Matlab)

- Use the controller candidate to get a more precise closed-loop plant response *inside FEA* analysis, if necessary fine-tune either the controller, the structure or both.

ACKNOWLEDGEMENTS

REFERENCES

[1] Peter Widas. Introduction to finite element analysis. Website, August 1997. Available: `http://www.sv.vt.edu/classes/MSE2094_NoteBook/97ClassProj/num/widas/history.html`.

[2] C. R. Fuller, S. J. Elliott, and P. A. Nelson. *Active Control of Vibration*. Academic Press, San Francisco, CA, United States of America, 1. edition, 1996.

[3] Daniel J. Inman. *Vibration with control*. John Wiley & Sons, Chichester, England, October 2006.

[4] M. R. Hatch. *Vibration Simulation Using MATLAB and ANSYS*. Chapman and Hall / CRC, Boca Raton, Florida, United States of America, first edition, September 2000.

[5] Xing-Jian Dong, Guang Meng, and Juan-Chun Peng. Vibration control of piezoelectric smart structures based on system identification technique: Numerical simulation and experimental study. *Journal of Sound and Vibration*, 297(3–5):680–693, 2006.

[6] Zhu Xiaojin, Zhao Miao, Gao Zhiyuan, and Chen Zhiyan. Analysis of active vibration control for piezoelectric intelligent structures by ANSYS and MATLAB. In *2010 International Conference on Computer Application and System Modeling (ICCASM)*, volume 4, pages V4–184 –V4–188, October 2010.

[7] Levent Malgaca. Integration of active vibration control methods with finite element models of smart laminated composite structures. *Composite Structures*, 92(7):1651 – 1663, 2010.

[8] Gergely Takács. *Temperature Controlled Welding Simulation: Finite Element Analysis of a PID Controlled Temperature Constrained Arc Welding Process*. VDM Verlag, Saarbrücken, Germany, 1. edition, 2010.

[9] V. Piefort. *Finite Element Modelling of Piezoelectric Structures*. PhD thesis, Université Libre de Bruxelles, 2001.

[10] Gergely Takács and Boris Rohaľ-Ilkiv. *Predictive Vibration Control: Efficient constrained MPC vibration control for lightly damped mechanical systems*. Springer, London, United Kingdom, 1. edition, 2012.

[11] Ansys Inc. *Release 12.1 Documentation for ANSYS*. Ansys Inc. / SAS IP Inc., 2009.

[12] MIDÉ Technology Corporation. *QuickPack Actuator Catalog*. MIDÉ Technology Corporation, Medford, MA, USA, 1. edition, 2007.

[13] Joseph L. Hellerstein, Yixin Diao, Sujay Parekh, and Dawn M. Tilbury. *Feedback Control of Computing Systems*. John Wiley & Sons Inc. / IEEE Press, Hoboken, NJ, United States of America, 2004.

[14] Robert L. Williams II and Douglas A. Lawrence. *Linear State-Space Control Systems*. John Wiley & Sons, Hoboken, New Jersey, United States of America, 2007.

[15] Panos J. Antsaklis and Anthony N. Michel. *Linear Systems*. Birkhäuser, Boston, United States of America, 2. edition, 2005. Originally published by McGraw-Hill, Englewood Cliffs, NJ, USA; 1997.

[16] L. Ljung. *System Identification: Theory for the User*. PTR Prentice Hall, Upper Saddle River, NJ., 2. edition, 1999.