Artificial Intelligence for Active Vibration Control Optimization on Smart Structures

Maryne Febvre, Jonathan Rodriguez, Simon Chesne, Manuel Collet

Artificial Intelligence for Active vibration control optimization on smart structures.

Maryne Febvre
INSA Lyon, CNRS, LaMCoS, UMR5259, 69621 Villeurbanne, France
CNRS, Ecole Centrale de Lyon, ENTPE, LTDS, UMR5513, 69134 Ecully, France
Jonathan Rodriguez
INSA Lyon, CNRS, LaMCoS, UMR5259, 69621 Villeurbanne, France
Simon Chesne
INSA Lyon, CNRS, LaMCoS, UMR5259, 69621 Villeurbanne, France
Manuel Collet
CNRS, Ecole Centrale de Lyon, ENTPE, LTDS, UMR5513, 69134 Ecully, France
DOI: https://doi.org/10.1115/SMASIS2023-110216
Conference: Proceedings of the ASME 2023 Conference on Smart Materials, Adaptive Structures and Intelligent Systems
SMASIS2023
September 2023
Austin, TX, USA
SMASIS2023-110216

Abstract

New meta-materials appear with the usage of piezoelectric transducers' networks. Within the number of controlling strategies for vibration mitigation, this study uses the classical derivative control law as a basis. As a preliminary work in optimization with AI, an automatic algorithm using Reinforcement Learning (RL) approached with TRPO (Trust Region Policy Optimization) tunes a controller on an experimental cantilever beam. The control law is a simple derivative feedback between two colocated piezoelectric transducers close to the beam-clamped end. The RL algorithm trains offline on an estimated model of the experimental setup. The study compares control methods between Reinforcement Learning results and a classical published approach.

Keywords : Vibration Control, Reinforcement Learning, Active Control, Cantilever Beam, Meta-material.

1 Introduction

Materials define properties and functions. Nowadays, they can be more functional by interacting with electromechanical systems. A new category of materials appears as "smart" materials. Piezoelectric materials are part of these new materials. As multi-physics materials, piezoelectric elements allow transferring of mechanical energy into electrical energy with a reciprocity effect [Tie69] and be used as sensors or as actuators. Structures and smart materials combination lead to smart structures with tunable smart transducers.

Indeed, many strategies have been developed to control smart beams and show their efficiency as LQR/LQG [SFH+05], $\|H\|_\infty$ optimization, or Fuzzy logic with sliding mode control [LSO09]. Researchers implement control strategies on numerical models, develop them experimentally with one set of co-localized piezoelectric transducers [VR07], or use it on multi-patch with distributed control [SW17]. Control strategies using Neural Networks with Neuro-controllers can suppress vibration with piezoelectric transducers [DT95, JH04, MCK15]. Numerical simulations and experimentations prove the methods' efficiency. The method's limit is the requirement of a computation unit during the training and during the controller usage.

Besides, this study aims at using neural network tools on a "smart" structure to optimize an Active Damping strategy based on a derivative feedback loop displacement and a piezoelectric patch [CWD03]. Thus, the computation unit is needed only during the training part. A Reinforcement Learning method named TRPO (Trust Region Policy Optimization) [SLA+15] is used to train neural networks. A cantilever beam with two colocated piezoelectric patches is also chosen to test this method as a preliminary experimental setup with the aim of vibration minimization of the free end.

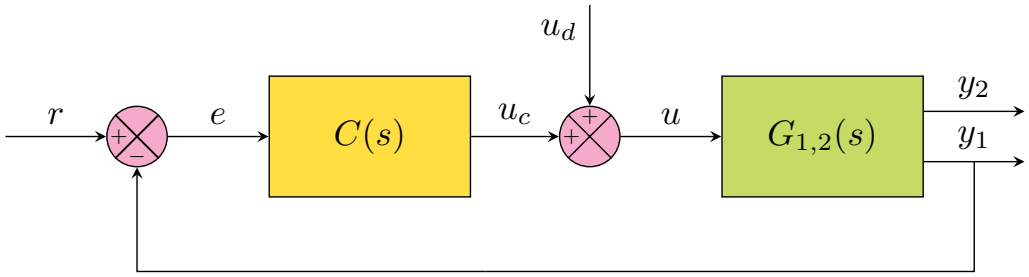


Figure 1: System architecture representation with a feedback control law Eq. (1)

2 Problem formulation

A combination of Laplace transfer functions can describe every smart structure within the linear domain. Figure 1 shows a block diagram architecture representation of a system where r is the target signal, e is the error, $C(s)$ is the control law, u_d is the perturbation signal, u_c is the correction signal, u is the input signal, $G_{1,2}(s)$ is the system, and $y_{1,2}$ are the output signals. The reference signal r is null here. One canal u introduces both the input perturbation signal and the control signal to the system.
The study aims at minimizing the vibration displacement at the beam-free end y_2. As shown in Fig. 1, a derivative feedback control law is chosen as:

$$C(s) = Ds \tag{1}$$

With s the Laplace variable of the transfer functions, $D \in R$ is the derivative value.

An optimization process tunes the feedback gain D based on the H2 norm minimization of the controlled system [DGKF89] within the frequency domain $\Omega = [\omega_1, ..., \omega_2]$ in (rad/sec) :

$$\|G_{c2}\|_2 = \sqrt{\frac{1}{2\pi}\int_{\omega_1}^{\omega_2} G_{c2}(j\omega)^2 d\omega} \quad and \quad G_{c2} = \frac{G_2}{1 + G_1 C} \tag{2}$$

The result obtained with this tuning method is shown in Tab. 2 and used as a reference to compare with the new method implemented here. Thus in this study, a Reinforcement Learning algorithm tunes the controller's parameter to minimize the free-beam end vibrations.

3 Algorithm Development

Within the Artificial Intelligence area, many algorithms exist. The Machine Learning domain differentiates three main categories: Non-Supervised Learning, Supervised Learning, and Reinforcement Learning (RL) [SB92]. Unlike the two other methods, this study chooses RL methods as it does not need any database.
RL algorithm in Fig. 2 learns how to achieve a goal by an iterative process in a limited environment. An agent leads the algorithm to associate $N_S \in N$ observations of states $S_t \in R^{N_S}$ from the environment with $N_A \in N$ actions $A_t \in R^{N_A}$ to choose at an Episode $t \in N$. The reward function $R_t \in R$ estimates the efficiency of the selected action on the environment. The agent uses the reward value to adjust his policy of action. The reward increases as the action selected gets close to the goal.

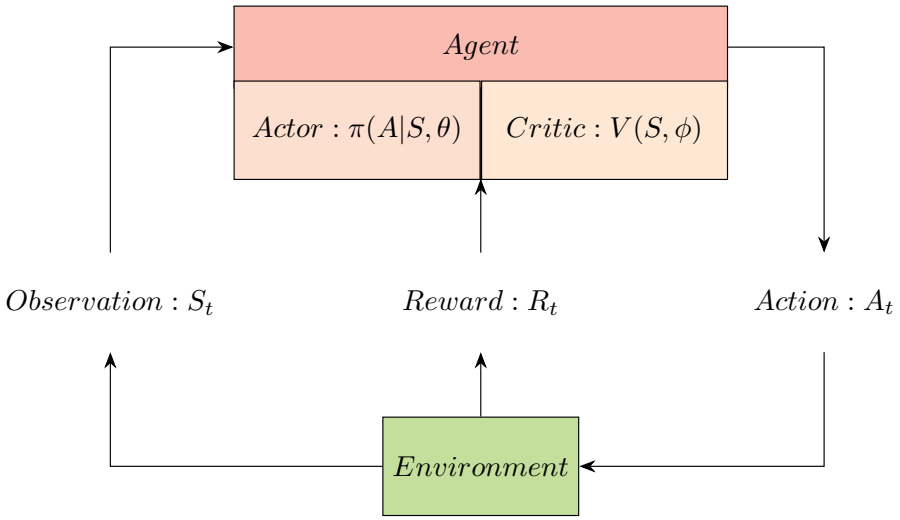


Figure 2: RL algorithm: schematic representation

Two neural network (NN) functions define the agent: the Critic and the Agent.

The Critic $V(S,\phi)$ evaluates the expected reward $R_V(S_t) \in R$ according to the observation state $S_t \in R^{2\times 1}$. ϕ defines bias $B^{(j)} \in R^{i\times 1}$ and weights $W^{(j)} \in R^{i\times i-1}$ of each layers $j \in N$ and its corresponding number of neurons $i \in N$ as in Fig. 3 with:

$$\begin{Bmatrix} j \\ i \end{Bmatrix} = \left\{ \begin{Bmatrix} 0 \\ 2 \end{Bmatrix}, \begin{Bmatrix} 1 \\ N_{Hu} \end{Bmatrix}, \begin{Bmatrix} 2 \\ N_{Hu} \end{Bmatrix}, \begin{Bmatrix} 3 \\ 1 \end{Bmatrix} \right\} \tag{3}$$

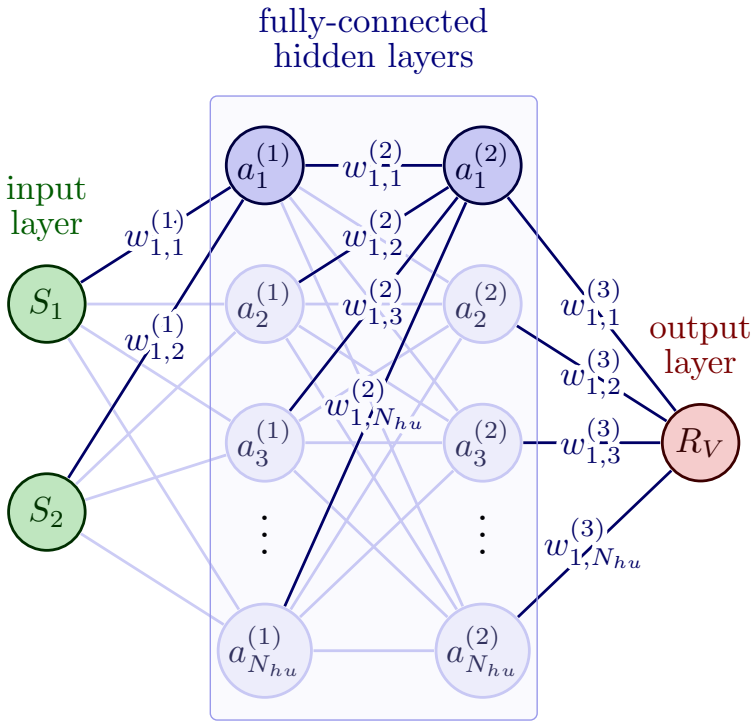


Figure 3: Neural Network definition: Critic function.

Equations (4),(5), and (7) define the hidden layers neurons computations with $\mathbf{a^{(j)}} \in R^{i\times 1}$,$\mathbf{W^{(j)}} \in R^{i\times i-1}$,$\mathbf{a^{(j-1)}} \in R^{i-1\times 1}$,$\mathbf{b^{(j)}} \in R^{i\times 1}$,$j$ the layers, i the number of neurons and $a^{(0)} = [S_1 \quad S_2]^T$.

$$\begin{bmatrix} a_1^{(j)} \\ a_2^{(j)} \\ \vdots \\ a_i^{(j)} \end{bmatrix} = f\left(\begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,i-1} \\ w_{2,1} & w_{2,2} & \dots & w_{2,i-1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{i,1} & w_{i,2} & \dots & w_{i,i-1} \end{bmatrix} \begin{bmatrix} a_1^{(j-1)} \\ a_2^{(j-1)} \\ \vdots \\ a_{i-1}^{(j-1)} \end{bmatrix} + \begin{bmatrix} b_1^{(j)} \\ b_2^{(j)} \\ \vdots \\ b_i^{(j)} \end{bmatrix} \right) \tag{4}$$

$$a^{(j)} = f(\mathbf{W}^{(j)} a^{(j-1)} + \mathbf{B}^{(j)}) \tag{5}$$

The critic output R_V is explicit in Eq. (6) and (7)

$$R_V = f(\mathbf{W}^{(3)} a^{(2)} + \mathbf{B}^{(3)}) \tag{6}$$

$$f(x) = \begin{cases} x & \text{x} \geq 0 \\ 0 & \text{x} < 0 \end{cases}, \quad \mathbf{x} \in R \tag{7}$$

The Actor $\pi(A|S,\theta)$ makes the decisions. It chooses the best action A_t, according to the observation state $S_t \in R^{2\times 1}$. θ defines bias $B^{(j)} \in R^{i\times 1}$ and weights $W^{(j)} \in R^{i\times i-1}$ of each layers $j \in N$ and its corresponding number of neurons $i \in N$ as in Fig. 4 with:

$$\begin{Bmatrix} j \\ i \end{Bmatrix} = \left\{ \begin{Bmatrix} 0 \\ 2 \end{Bmatrix}, \begin{Bmatrix} 1 \\ N_{Hu} \end{Bmatrix}, \begin{Bmatrix} 2 \\ N_{Hu} \end{Bmatrix}, \begin{Bmatrix} 3 \\ 2 \end{Bmatrix} \right\} \tag{8}$$

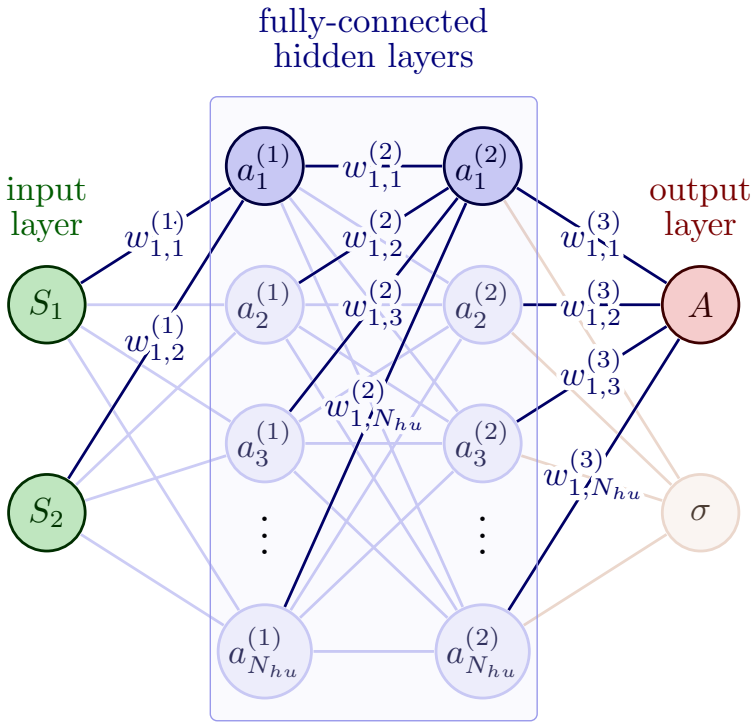


Figure 4: Neural Network definition: Actor function.

The Actor's NN output A and σ are expressed in Eq. (9) and the hidden layers with Eq. 4 and 5.

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = f(\mathbf{W}^{(3)} a^{(2)} + \mathbf{B}^{(3)}), \quad \mathbf{x_{1,2}} \in R$$
$$A = tanh(x_1)$$
$$\sigma = ln(1 + e^{x_2}) \tag{9}$$

As a first RL approach, this study uses the TRPO algorithm developed by Schulman [SLA+15]. This method guarantees improvement in policies with neural network. The critic parameters ϕ and the Actor parameters θ are updated with a gradient minimization of their loss function based on an advantage function $E_t \in R$ in Eq. 10 with the discount factor $\delta_f \in R$ and $\lambda \in R$ the Generalized Advantage Estimator (GAE). It follows the steps defined in Fig. 5

$$E_t = \lambda \delta_f (R_t - V(S_t, \phi)) \tag{10}$$

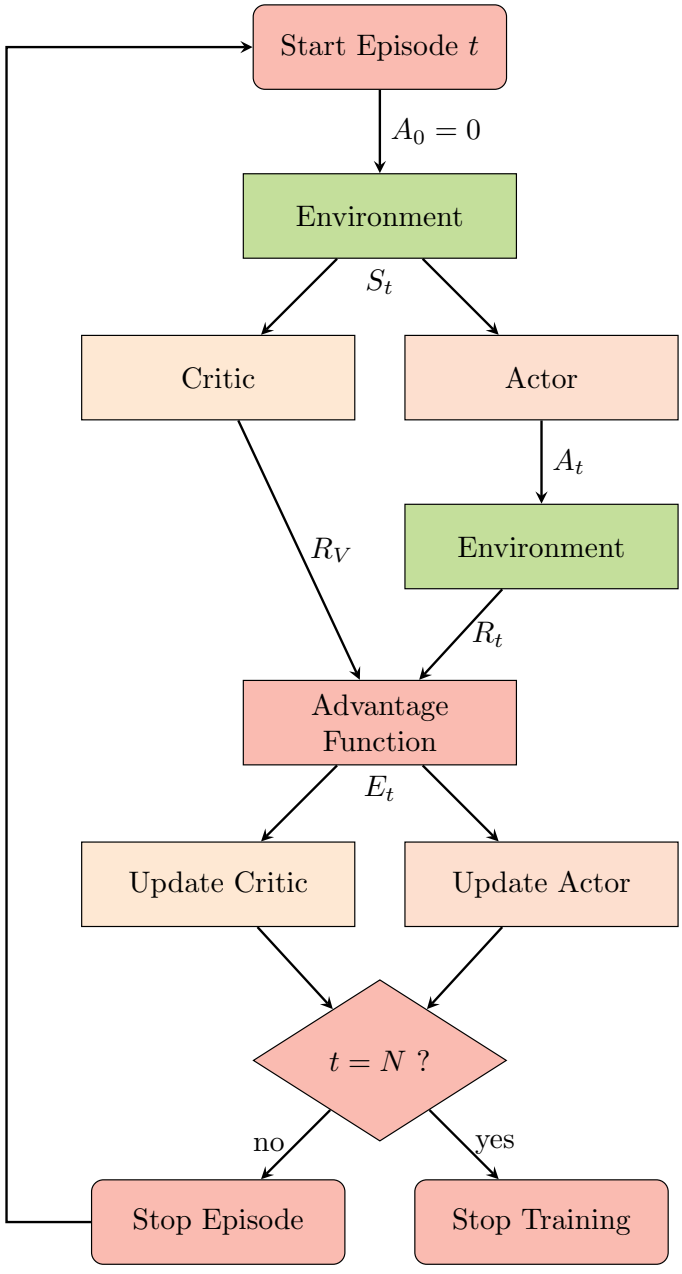


Figure 5: TRPO Training Algorithm

At the end of the training, when the number of episodes equals N, the action chosen by the actor leads to a maximum reward from the environment. The critic output value equals this reward. In our case, the maximal reward value achieves when the derivative control law value leads to the maximal minimization of vibrations at the free-beam end.

4 Experimental Implementation

The RL method is implemented on a cantilever beam as a simple structure. The beam has length L_b, width l_b, and thickness h_b. Two collocated piezoelectric elements are attached to the beam at a distance e_p from the supported end. Piezoelectric elements' dimensions are length L_p, width l_p, and thickness h_p. The experimental values are available in Appendix A. Figures 6 and 7 show the experimental setup.

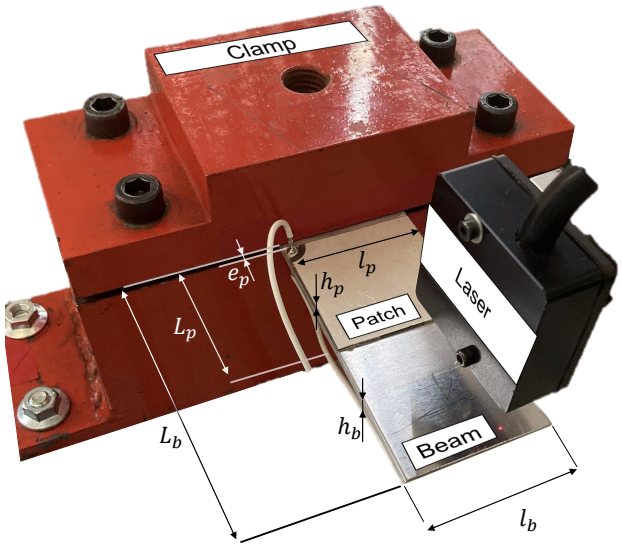


Figure 6: Experimental setup: picture

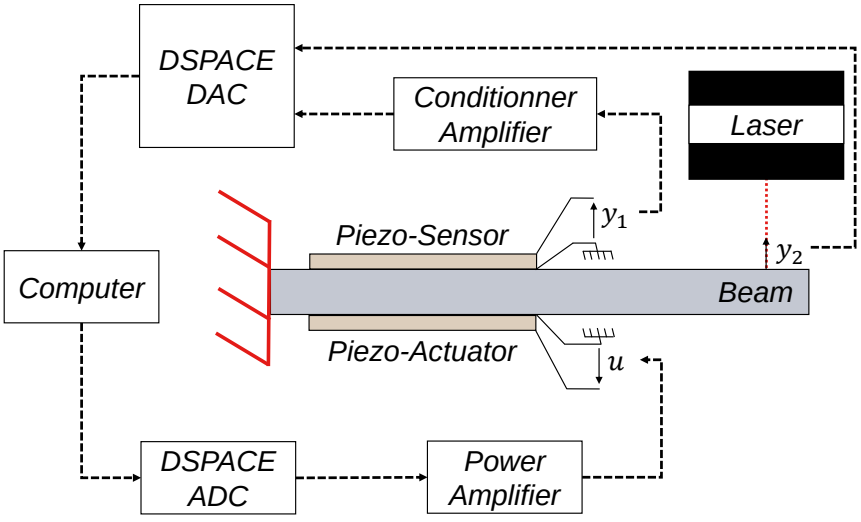


Figure 7: **Experimental setup: scheme**

A band-limited white noise (0-1000Hz) is the input perturbation d in one piezoelectric element. This frequency range brings out the 2 first modes of the beam.

The second piezoelectric patch is used as a sensor with an output y_1. A laser localized at the beam-free end measures the relative displacement with an output sensor voltage y_2. According to Fig. 1, the system representation is composed of 2 transfers functions between the input and the outputs in the Laplace domain:

$$Y_1(s) = G_1(s)U_d(s) \quad and \quad Y_2(s) = G_2(s)U_d(s) \tag{11}$$

With $Y_1(s)$, $Y_2(s)$ and $U_d(s)$, the Laplace transform respectively of y_1, y_2 and u_d. Experimental measurements of y_1 and y_2 allow computing these two transfer functions.

Transfer functions are numerically estimated as SISO systems with poles and zeros as:

$$G_{1,2}(s) = \frac{K}{s^{n_0}} \frac{\Pi_{i_z=1}^{n_z}(1 - z_{i_z}^{-1}s)}{\Pi_{i_p=1}^{n_p}(1 - p_{i_p}^{-1}s)} \tag{12}$$

With $K \in R$ as the constant gain, $n_0 \in Z$ as the number of null poles (integrator if $n_0 > 0$, derivative if $n_0 < 0$, and with taking care of causality), $z_i \in C$ zeros and $p_i \in C$ poles of the system, $n_z \in N$ the number of zeros and $n_p \in N$ the number of poles. Numerical transfer function estimations consider a frequency range between 1 and 2000 Hz to include potential experimental spillover effects.

The feedback control law $C(s)$ is applied to these estimated functions to simulate the beam behavior. To reduce the time needed for outputs y_1 and y_2 measurements, the agent trains on a numerical representation of the system. One numerical RL episode runs 10 times faster than one experimental RL episode.

The algorithm aims at tuning the control law derivative parameter. In the objective of minimizing vibrations at the beam-free end, the reward R_t is based on the laser measurement y_2:

$$R_t = \frac{1}{10} \sum_{i_r=1}^{10} \left(\frac{1}{\sum_{i_k=i_1}^{i_2} y_2(i_k)^2}\right) \tag{13}$$

With $i_1 = 0.9(i_r - 1)N_y + 1$, $i_2 = 1.1 \times i_r \times N_y$, $10 \times N_y \in N$ the signal y_2 length and $i_r \in N$ signal sample index.

Figure 8 shows reward variations with derivative value D. The reward function is computed with time dependency to consider both control performances and stability.

According to the reward function, the optimal parameter estimation achieves the maximum value of the reward.

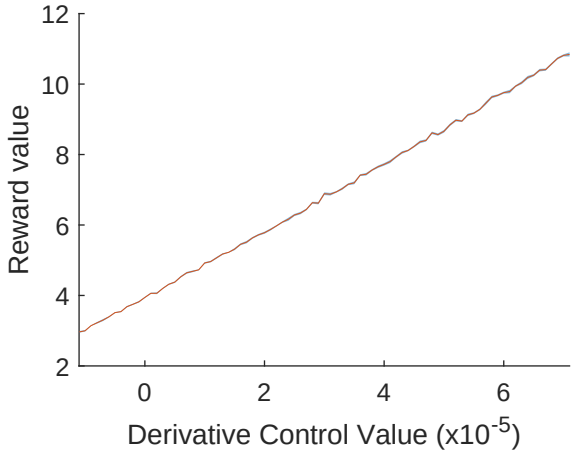


Figure 8: Average reward values variations with derivative D control values (red line) and the 0.95 confidence interval (blue area) for 10 executions.

As an input, the RL algorithm takes continuous derivative values around the stable domain: $D = \{-1 \times 10^{-5}, ..., 7 \times 10^{-5}\}$. Mean over time of y_1 and y_2 absolute values are environment observations S_1 and S_2.

5 Results and Discussions

A derivative feedback control applies on the piezoelectric patch system $G_1(s)$ to reduce free-beam end vibrations.

Derivative control value tuning done by the RL algorithm needs data to be estimated. Since the algorithm parameters θ and ϕ are set randomly, 10 independent training run on the system. Figure 9 and Table 1 show the different training results with the algorithm parameters defined in Appendix B.

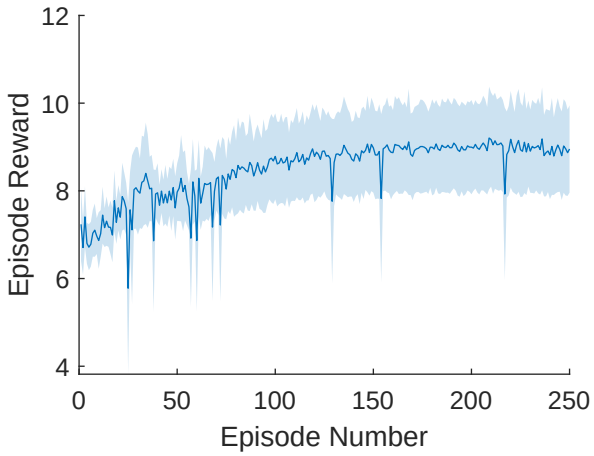


Figure 9: Average reward values variations with Episode number (dark blue line), the 0.95 confidence interval (blue area) for D tuning (10 Trainings).

Table 1: Training Results

Training	Real Reward	Expected Reward	Absolute Error Reward	$D_{opt} \times 10^{-5}$	$D_{opt}std$
1	11.4043	9.8996	1.5046	7.0000	0.7389
2	6.8019	6.9282	0.1262	3.0284	0.0041
3	7.6028	8.1113	0.5085	3.7546	0.0007
4	8.9178	8.6256	0.2922	4.8198	0.0005
5	10.6305	10.3559	0.2746	7.0000	0.1055
6	9.1947	9.3549	0.1603	5.3210	0.0018
7	**10.7293**	**10.7644**	**0.0351**	**7.0000**	**0.1231**
8	7.8461	7.2981	0.5479	4.1012	0.0012
9	7.1317	7.5333	0.4016	3.3272	0.0005
10	11.4273	10.2298	1.1975	7.0000	0.5041
Mean	9.3699	8.5889	0.8441	5.2824	0.3121

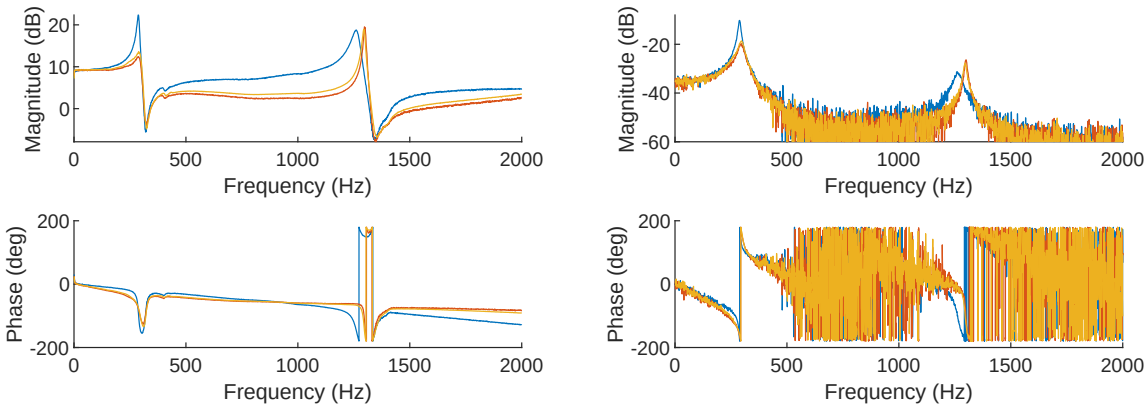
At the end of the training, the algorithm maximizes the reward and finds an optimal control value in the stability domain. The D value estimated by the different training does not lead to the same value after 250 episodes because of the randomness of the process. Some unfinished training leads to a local minimum. The D value with the best performance is the one with the best training end for both the real R_t and the expected R_V rewards.

Table 2 presents the D values obtained with the best training and an average with all training. Measurements are made on the experimental setup to estimate control performances. Figures 10a and 10b evaluate transfer functions $G_1(s)$ and $G_2(s)$ with control off and on. Figure 10a shows a decreasing phase implying a small delay in the experimental setup. However, the algorithm still finds an efficient and stable control value.

$\|H\|_2$ minimization and RL tunning methods lead to a control performance of 10 dB vibrations attenuation at the free beam end considering the first mode as in Fig. 11.

Table 2: Case continuous action

Method	$D \times 10^{-5}$	RMS_{y_2} [mm]	$\|G_2\|_2$	Δ_{dB}
None	0.00	0.0020	10.76	-
$\|H\|_2$	7.00	0.0013	5.93	-10.6
$TRPO_{avg}$	5.29	0.0014	6.59	-9.6
$TRPO_{best}$	7.00	0.0013	5.93	-10.6



(a) $G_1(s)$. (b) $G_2(s)$

Figure 10: Transfer Functions with control off (blue), control on with $D = 7 \times 10^{-5}$ (red) and control on with $D = 5.88 \times 10^{-5}$ (orange) computed from $y_1(t)$ and $y_2(t)$ experimental measurements

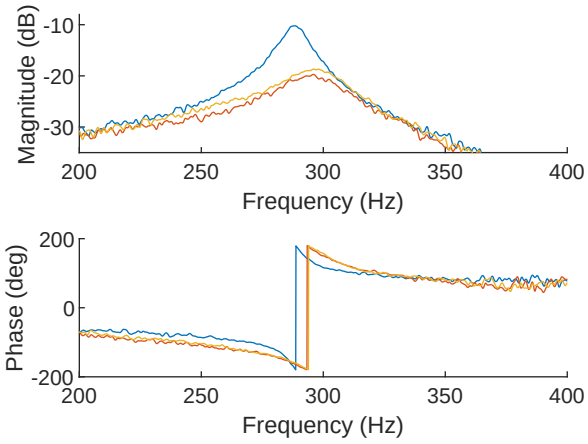


Figure 11: Figure 10b zoom between 200Hz and 400 Hz

6 Conclusion

As a preliminary approach, this study shows the RL algorithm performances in an experimental context for tuning controller parameters. The TRPO method finds a derivative control value to minimize cantilever beam-free end vibrations. The proposed method can tune an optimal derivative value D with 12% confidence in a 250 samples training of 3.5 seconds/sample. Depending on the initial setup, the method can be time-consuming. It is possible to improve the training by setting the training hyperparameters (Appendix B). Future works will introduce stability issues in the IA tuning process. This algorithm offers possibilities to implement on complex distributed systems with multi-parameters control methods to tune.

Acknowledgments

This work was supported by the LABEX CeLyA (ANR-10-LABX-0060) of Université de Lyon, within the program "Investissements d'Avenir" operated by the French National Research Agency (ANR).

A Experimental setup dimensions

Table 3 resumes the experimental setup dimensions.

Table 3: Experimental setup characteristics

Materials	Beam Aluminium			Piezoelectric Patch Pz26		
	Name	Value	Unit	Name	Value	Unit
Dimensions	L_b	105	[mm]	L_p	50	[mm]
	l_b	52	[mm]	l_p	50	[mm]
	h_b	3	[mm]	h_p	1	[mm]
	e_p	3	[mm]			
Laser sensitivity	0.1			[mm/V]		

B Experiment Algorithm Parameters

Table 4 resumes the algorithm parameters set during the training.
*In this study, the agent only interact once with the environment before changing the policy.

Table 4: Algorithm Characteristics

Name	Symbol	Value
Experience Horizon	N	512*
Mini Batch Size	M	128*
Entropy Loss Weigth	w	0.01
Number of Epoch	k	3
Average Estimate Method		"gae"
GAE Factor	λ	0.95
Conjugate Gradient Damping	δ_g	0.1
KL-Divergence Limit	δ	0.01
Number Iteration Conjugate Gradient	N_{Cg}	10
Number Iteration Line Search	n	10
Conjugate Gradient Residual Tolerance	C_{gr}	1e-8
Normalized Advantage Method		"none"
Advantage Normalizing Window	N_a	-
Learning Rate	L_R	0.01
Gradient Threshold	N_{th}	inf
Gradient Threshold Method		"l2norm"
L2RegularizationFactor	$l2$	1e-4
Training Algorithm		"adam"
Sample Time	t_s	1 (Event Base)
Dicount Factor	δ_f	0.99

References

[CWD03] M. Collet, V. Walter, and P. Delobelle. Active damping of a micro-cantilever piezo-composite beam. 260:453–476, 2003.

[DGKF89] J.C. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis. State-space solutions to standard h/sub 2/ and h/sub infinity / control problems. *IEEE Transactions on Automatic Control*, 34(8):831–847, 1989.

[DT95] Scott D.Snyder and Nobuo Tanaka. Active control of vibration using a neural network. *IEEE*, 6:819–828, 1995.

[JH04] Ratneshwar Jha and Chengli He. A comparative study of neural and conventional adaptive predictive controllers for vibration suppression. 13:811–818, 2004.

[LSO09] Luyu Li, Gangbing Song, and Jinping Ou. Adaptive fuzzy sliding mode based active vibration control of a smart beam with mass uncertainty. *STRUCTURAL CONTROL AND HEALTH MONITORING*, pages n/a–n/a, 2009.

[MCK15] Mohit, Deepak Chhabra, and Suresh Kumar. Active vibration control of the smart plate using artificial neural network controller. 2015:1–20, 2015.

[SB92] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning:An Introduction*. 1992.

[SFH+05] G. E. Stavroulakis, G. Foutsitzi, E. Hadjigeorgiou, D. Marinova, and C. C. Baniotopoulos. Design and robust optimal control of smart beams with application on vibrations suppression. 36:806–813, 2005.

[SLA+15] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1889–1897, Lille, France, 07–09 Jul 2015. PMLR.

[SW17] Ann-Kathrin Schug and Herbert Werner. Active vibration control of an aluminum beam — an experimental testbed for distributed vs. centralized control. 2017.

[Tie69] Tiersten. *Linear Piezoelectric Plate Vibrations*. 1969.

[VR07] C M A Vasques and J Dias Rodrigues. Active vibration control of a smart beam through piezoelectric actuation and laser vibrometer sensing: simulation, design and experimental implementation. *Smart Materials and Structures*, 16(2):305–316, jan 2007.