CIRP Journal of Manufacturing Science and Technology 29 (2020) 115–129

Contents lists available at ScienceDirect

CIRP Journal of Manufacturing Science and Technology

journal homepage: www.elsevier.com/locate/cirpj

Validation of a hardware-in-the-loop simulator for investigating and actively damping regenerative chatter in orthogonal cutting

G.N. Sahu[a], S. Vashisht[a], P. Wahi[a,b], M. Law[a,*]

[a] Machine Tool Dynamics Laboratory, Indian Institute of Technology Kanpur, Kanpur, 208016, India
[b] Department of Mechanical Engineering, Indian Institute of Technology Kanpur, Kanpur, 208016, India

ARTICLE INFO

Article history:
Available online 13 April 2020

Keywords:
Hardware-in-the-loop
Machine tool chatter
Active damping
Delay estimation
Digital control

ABSTRACT

Machine tool productivity and part surface quality are fundamentally limited by chatter occurring due to regenerative type self-excited vibrations. Efficacy of the existing high-fidelity chatter models to help avoid chatter in real industrial settings remains limited due to the uncertainties in machine tools. To help validate models without the vagaries of the uncertainties and without damaging real machine tool systems, this paper discusses the use of a mechatronic hardware-in-the-loop (HiL) simulator that offers a platform for investigation on chatter. The HiL simulator is a hybrid system of hardware and software components substituting the actual cutting tool-workpiece interaction through an actuator exciting a flexure by an emulated real time cutting force. For the simulator to faithfully represent the physics of the process-machine interactions, experimentally characterized stability behavior with the HiL must necessarily follow model predictions. However, the transfer between the hardware and the software layers of the HiL simulator involves inevitable delays that result in diverging stability behavior. We hence present systematic investigations to identify and compensate the delay in the hardware layer on account of the actuator and the transducers, as well as delays in the software layer on account of signal conditioning. The validated HiL simulator is then used as an experimental platform to investigate the effectiveness of four different control strategies for an active damping system. Being a non-destructive, cost-effective and repeatable platform, the HiL simulator opens up further possibilities for validation of high-fidelity chatter models.

Introduction

Increased material removal rates with good part surface quality is the prime demand in the metal cutting industry. However, chatter in the cutting process resulting from the regenerative type self-excited vibrations is the main hindrance to machine tool productivity. Regenerative chatter results from the interactions of the cutting process with the dynamics of the machine tool system. These closed-loop interactions can lead to large amplitude unstable vibrations that deteriorate manufactured part quality, tool life, and damage the machine tool system. Consequently, there has been a sustained interest and emphasis on modelling of chatter to understand and suppress it. A succinct overview of the advances in understanding of chatter to mitigate it for various machining processes can be found in Refs. [1–7].

Despite the many significant strides in the understanding of chatter, efficacy of models to guide the avoidance of chatter in real industrial settings has remained limited, in part due to the uncertainties in machine tools. Uncertainties occur due to position and speed dependent dynamics of the machine tool, cutting tool wear and speed dependent material behavior of the workpiece being cut [7]. Chatter models that account for uncertainties can be validated only by performing a large number of actual cutting tests in controlled laboratory experiments. However, actual chatter tests require testing at the limiting case of stability which may eventually damage the machine tool system.

In the context of doing experiments to validate models without the vagaries of the uncertainties and without damaging the real machine tool systems, this paper presents a hardware-in-the-loop (HiL) simulator that offers a platform for investigation on chatter. A HiL simulator employs a physical layer which consists of a flexure representing a machine, an actuator substituting the cutting tool and a sensor communicating with a virtual layer which emulates the cutting process dynamics. This combination of physical and virtual prototyping facilitates testing that is non-destructive, cost-effective, repeatable, rapid, and safe. These advantages make the HiL simulator an attractive alternative to investigate chatter. For much of the same reasons, HiL simulators have been made good

* Corresponding author.
E-mail address: mlaw@iitk.ac.in (M. Law).

https://doi.org/10.1016/j.cirpj.2020.03.002
1755-5817/

use of in the aerospace, automotive, marine, civil, and defense industries, as is summarized in Ref. [8].

The use of HiL simulators in machine tools has been shown to be effective for investigating the performance of real CNC systems coupled to virtual machine tool models [9–11], for investigating chatter [12–21], and since these simulators offer a convenient and repeatable experimental platform, the simulators have also found use in testing of actuators and active vibration control strategies in machine tool systems [12–14,16,20,22–24]. HiL simulators for investigations on chatter have also incorporated the influence of dynamics changing with speed and position by using realistic machines and spindles in their investigations [15,19].

Since the primary purpose of a HiL simulator is to represent the physics of the process-machine interactions during chatter, the experimentally characterized stability behavior with the HiL simulator must necessarily follow model predictions. However, since the HiL simulator has actuators and measurement transducers, and involves signal conditioning, and performs real-time computations on a real-time computer, how/if any/all of the hardware and software elements potentially contribute to divergence of the HiL simulator's stability behavior from that predicted by the model, must be understood and characterized. Though Ganguli et al. [12,13] were amongst the first to use HiL simulators for investigations on chatter, they apparently did not suffer from any divergent behavior. However, the systematic analysis by Mancisidor et al. [14,16] showed that chatter experiments on the HiL simulator do indeed diverge from model predictions. They [14,16] identified the delay caused by the contact-type electromagnetic actuator, and a small delay caused by the controller as the main sources of divergence. Recent work reported in Ref. [17–19] showed that with the use of high bandwidth non-contact actuators and sensors, the delay caused by hardware elements is negligible in comparison to delay caused by signal conditioning and the computational algorithm.

Given that HiL simulators have abundant potential for experimentation on chatter, this paper aims to report systematic investigations on how each of the key hardware and software elements and parameters govern and limit the performance of a HiL simulator. Like others [12–21] before us, we also focus our attention on a HiL simulator for investigating regenerative chatter in orthogonal cutting, which once validated comprehensively may be extended to other machining operations. However, our work differs from earlier reported work in the following four significant ways:

i We systematically identify the total delay in the system contributed by the actuator, the response transducer(s), and by the signal conditioning and processing parameters. We present two methods to identify delay: a method based on estimation of the negative phase provided by each element at a fixed-frequency, and another method based on estimating the true time delay through linear interpolation of the measured phase characteristics of each element across the range of frequencies of interest.

ii We use two methods for response measurements: a non-contact eddy current type displacement sensor, and a contact-type accelerometer in combination with hardware integrator that overcomes the challenges faced by real-time numerical integration in the software layer, offering a low-cost alternative to the laser sensors used in Refs. [12–14,16–19].

iii We present a simplified method to derive a phase lead compensation filter to account for delays present in the hardware and software layers.

iv Our validated HiL simulator facilitates comprehensive investigations of four different control strategies for active damping to suppress chatter.

Above ideas are demonstrated on an in-house developed HiL simulator. The remainder of the paper is structured as follows: at first, a brief overview of the stability model for orthogonal cutting process that the HiL simulator has to emulate is provided. In the subsequent section, we describe the experimental setup for the HiL simulator and detail the key hardware and software elements. This section incudes details about the National Instruments (NI) LabVIEW based digital control, as well as the custom-made GUI. In the section after, we first discuss how the experimentally characterized stability behavior with the HiL simulator diverges from model predictions due to inevitable delays in the hardware and software layers, and then we systematically discuss the identification of the total delay in the system. This section also includes discussions on introducing the estimated delay in the theoretical model, as well as discussions on compensation methods to account for the delay so as to validate the HiL simulator. The final section on active damping discusses the use of the HiL simulator to test four different active control strategies to mitigate chatter, which is followed by the main conclusions.

Theory of regenerative chatter in orthogonal cutting

Theoretical background on the stability of regenerative chatter in an orthogonal cutting process that is to be emulated on the HiL simulator is established in this section based on the models proposed in Ref. [1–3]. An orthogonal cutting process is characterized by a single point cutting tool performing turning operation on a rotating flexible work piece as shown in Fig. 1(a).

The workpiece being flexible vibrates under the action of cutting forces. This results in a wavy surface on the workpiece which is to be removed in the subsequent rotation. However, on account of the present structural vibration, the surface generated in the succeeding rotation is also wavy — see Fig. 1(b). These vibrations result in a dynamic chip thickness:

$$h_d(t) = h_0 + x(t-T) - x(t), \tag{1}$$

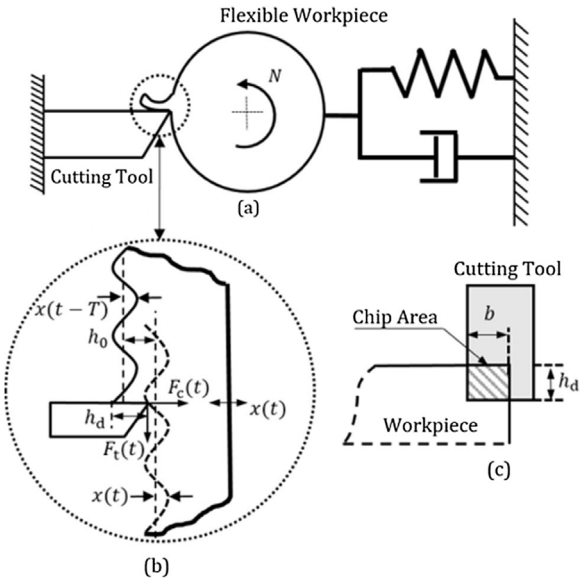

Fig. 1. Schematic of an orthogonal cutting process: (a) single point cutting tool performing turning operation on a flexible work piece; (b) cut surfaces due to the previous and present vibrations; (c) top view of the cutting area.

wherein h_0 is the static component of the chip thickness, $x(t)$ are the present vibrations, and $x(t-T)$ are the vibrations from the previous revolution. $T = 60/N$ is the time period of one revolution in seconds and N is the spindle RPM. The dynamic chip thickness leads to a dynamic cutting force $F_c(t)$ which is approximated by the product of an empirical coefficient K_f and the chip area as:

$$F_c(t) = K_f b h_d(t). \quad (2)$$

Approximating the flexibility of the workpiece by a single degree of freedom system with mass m, damping c, and stiffness k, the resulting governing equation of motion under excitation can be written in the form of a delay differential equation as:

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = F_c(t) \equiv K_f b[h_0 + x(t-T) - x(t)]. \quad (3)$$

Merritt [3] proposed that Eq. (3) can be represented by a feedback loop, as shown in Fig. 2, stability analysis of which can be done using linear control theory. To derive the feedback loop, Laplace transformations are applied to Eqs. (1–3) which results in:

$$h_d(s) = h_0(s) + X(s)(e^{-Ts} - 1) \quad (4)$$

$$F_c(s) = K_f b\left[h_0(s) + X(s)\left(e^{-Ts} - 1\right)\right]$$

$$(ms^2 + cs + k)X(s) = F_c(s). \quad (5)$$

From Eq. (5) structural dynamics, i.e., $\phi(s) = X(s)/F_c(s)$ can be derived. Eq. (4) and (5) can be combined together in a closed-loop representation as shown in Fig. 2.

Further, a transfer function between the static chip thickness $h_0(s)$ as the reference input quantity and the dynamic chip thickness $h_d(s)$ as the output or controlled quantity can be shown to be:

$$\frac{h_d(s)}{h_0(s)} = \frac{1}{1 + (1 - e^{-Ts})K_f b\phi(s)}. \quad (6)$$

The stability of the system can be inferred from the characteristic equation, i.e., the denominator of Eq. (6):

$$1 + (1 - e^{-Ts})K_f b\phi(s) = 0. \quad (7)$$

If the structure dynamics (m, c and k) are known, the stability of the closed loop system is characterized by the cutting conditions i.e., the depth of cut b and the spindle RPM N. We check the stability using the Nyquist stability criterion suggested by Tlusty [2] and detailed in Ref. [25]. The Nyquist stability method offers considerable ease in obtaining stability behavior of a delayed system with multiple dynamics which generally occurs in a mechatronic system, and is thence preferred in the present work over other classical analytical methods reported in Ref. [1–3]. The stability criterion analyzes the characteristic equation for different combinations of $[b, N]$ to check whether a particular combination is stable or not on the basis of the Nyquist plot.

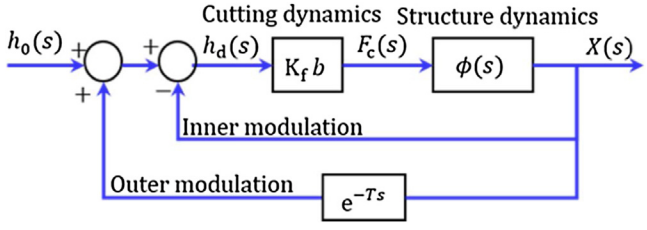

Fig. 2. Regenerative closed loop for the orthogonal cutting process. Adapted and modified from Ref. [3].

The theoretical stability analysis of regenerative chatter in orthogonal cutting serves as a benchmark against which the HiL simulator is validated. Details of the key hardware and software components of such a HiL simulator are discussed next.

The hardware-in-the-loop (HiL) simulator

The HiL simulator is implemented as a hybrid system of a hardware layer and a software layer. An overview of the two layers is schematically shown in Fig. 3. The hardware layer consists of a flexure, an actuator, and a response measurement transducer. The flexure is designed to behave like a single degree of freedom system which can approximate the dynamics of a machine, a tool, and/or a flexible workpiece. In this sense, the flexure is not representative of a real machine tool system, in which the dynamics can also change with speed and position — as was presented in earlier other work [15,19]. The response of the flexure to the actuator force is measured by a response transducer. Though we separately use two response transducers (an eddy-current sensor and an accelerometer), Fig. 3 shows a HiL simulator using an accelerometer. The measured response is transferred to the software layer through an analog to digital converter (ADC). Passing through a set of digital filters, the response signal is used to compute the cutting force signal as dictated by the theory of orthogonal cutting process. The computed force signal is sent to the actuator's amplifier through a digital to analog converter (DAC). The main objective of the experiment is to characterize the stability of the established closed-loop between the hardware and the software layer.

The hardware layer

The flexure

A simple bench type design is adopted for the flexure which emulates the flexible workpiece. The bottom plate of the flexure is fastened securely on to a rigid table — see Fig. 3. The plates are cut to dimension from 16 mm thick mild steel plates, and are welded at the interfaces. The design provides modularity in terms of actuation and response measurement location. The setup also provides ample space to mount another actuator — on the plate opposite to that of where the actuator applies the cutting force, to investigate active damping strategies.

Since the flexure is connected with the shaker through a stinger, and since the shaker is fixed, unlike in other HiL setups [16], to ensure that the stinger does not influence the flexure's response, we investigated the dynamics of the flexure with the shaker connected to it using varying lengths of stingers. Dynamics were measured by exciting the flexure by the shaker using a sinusoidal chirp signal with a frequency range of 30 Hz–700 Hz and measuring the input load cell signal and the output acceleration. Measured dynamics were characterized by frequency response functions (FRFs). FRF of the flexure measured with a modal hammer and without the flexure being connected to the shaker through the stinger were found to be similar to the measured FRFs of the flexure connected to the shaker with different stingers. This confirmed that the length of the stinger, the boundary conditions of the shaker, and the moving mass of the shaker have a very weak influence on the dynamics of the flexure. We settled on using a stinger with a diameter of 2.5 mm and a length of 150 mm.

The measured FRFs for excitation at location 1, and response at locations 2 and 3, i.e., H_{21} and H_{31} are shown in Fig. 4. First two dominant bending modes of vibrations of the flexure are shown at $f_{n1} = 112.6$ Hz and $f_{n2} = 530$ Hz respectively, in an inset of Fig. 4, which shows that location 2 is an antinode only for the first bending mode of interest, and a node for all other higher frequency

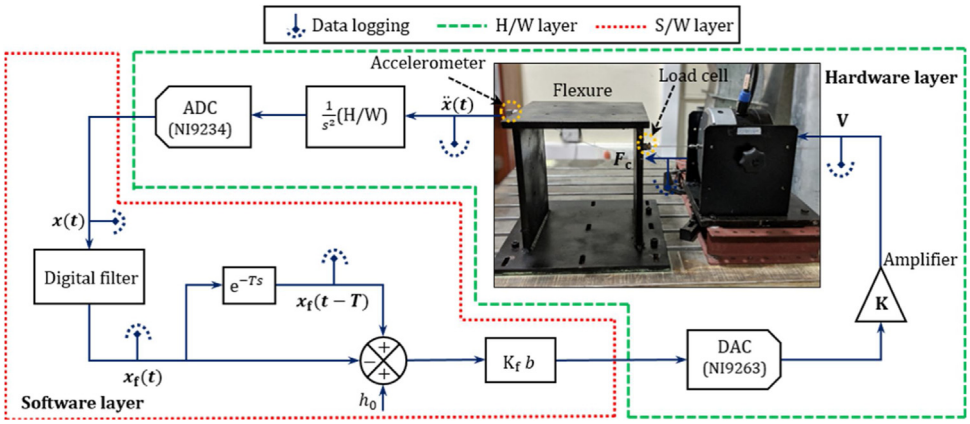

Fig. 3. Hardware and software layers of the HiL simulator for investigating chatter in orthogonal cutting.

modes. Therefore, location 2 is used for response measurement and location 1 is used for excitation in further investigation of the HiL simulator. As evident in Fig. 4, the response H_{21} is dominated by a single mode at 112.6 Hz that has a damping ratio $\zeta = 1.6\,\%$, and a stiffness $k = 9.4 \times 10^6$ N/m.

The actuator

An electromagnetic shaker (Make: B&K, Model number: 4824) is used as the actuator to excite the flexure. The shaker can apply a maximum force of 100 N. The shaker applies the force to the flexure through a steel stinger. A load cell (Make: Dytran, Model number: 1051V4) is mounted at the flexure-stinger junction to monitor the cutting force applied by the shaker — see Fig. 3. The shaker, driven by its power amplifier (Make: B&K, Model number: 2732) needs a voltage signal as the control input. The force/voltage dynamic characteristics of the shaker coupled to the flexure is measured in an open loop configuration using a sine chirp excitation between 30 Hz and 300 Hz as shown in Fig. 5. The suspension frequency of the shaker is calculated to be 21 Hz, and since it is high-pass filtered, it does not have any influence on the stability behavior of the HiL simulator. Since regenerative chatter occurs in the vicinity of the natural frequency of the system at 112.6 Hz, hence the shaker characterization coupled with the flexure is carried out up to 300 Hz. It is evident from the magnitude and phase plot of Fig. 5 that a force drop and change in phase occurs near the natural frequency of the flexure which is mainly caused by the non-linear armature dynamics of the shaker coupled with the flexure as discussed in Ref. [26].

As the chatter occurs in vicinity of the natural frequency of the system, a single F/V ratio at $f_n = 112.6$ Hz, i.e., 26.8 N/V is chosen for all the chatter experiments in the HiL simulator, making the voltage to force gain ratio, $g_{VN} \cong 0.04$ V/N. Such approximations of voltage to force gain ratios at f_n have also been used in earlier reported work [16]. The voltage to force gain ratio (g_{VN}) is used in the software layer to convert the calculated regenerative cutting force to voltage value which is then transferred to the power amplifier of the shaker as shown in Fig. 10. Furthermore, the shaker provides a negative phase between the input voltage signal and the output actuation force which is mainly attributed to an inevitable shaker delay, and this trend of negative phase is consistent with the results reported in Ref. [16]. This delay due to the shaker coupled to the flexure is separately investigated in the section *Estimation of total delay*.

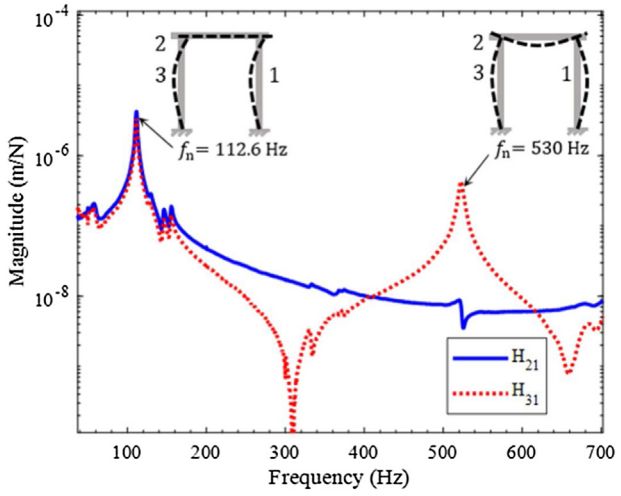

Fig. 4. Frequency response and bending mode shapes of the flexure.

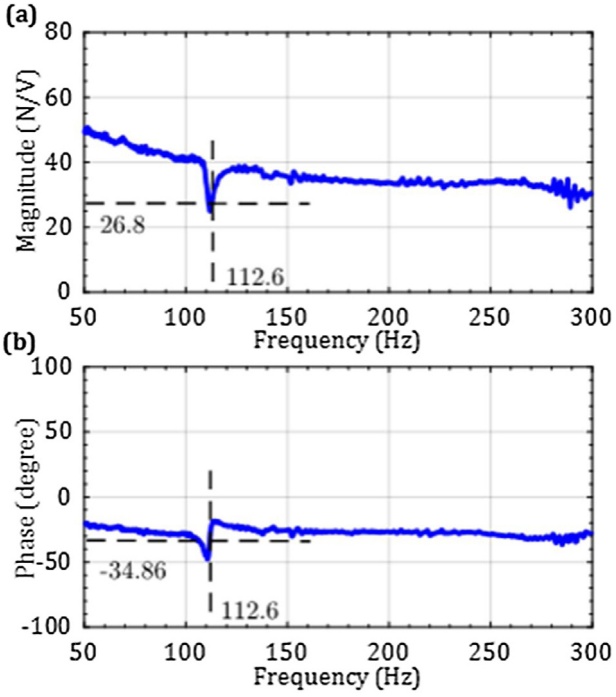

Fig. 5. Measured frequency characteristics of the shaker coupled with the flexure.

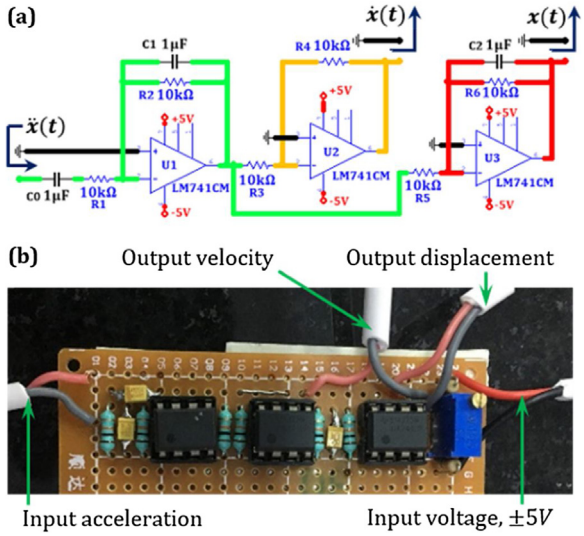

Fig. 6. (a) Circuit diagram of the hardware integrator, designed in NI Multisim, (b) fabricated hardware integrator.

The response measurement transducers

The orthogonal cutting model dictates the use of the displacement signal of the flexure to compute the cutting force. The displacement signal is obtained by two methods: using a non-contact eddy current displacement sensor (Make: KAMAN; Model number: KD2306-2S); and performing a double integration operation on the measured signal of a contact type acceleration sensor (Make: MEGGITT; Model number: 44A16-1032; IEPE type). A comparative analysis of these transducers is presented in subsequent sections.

The double integration operation on the accelerometer response can be done numerically or using a hardware (H/W) based integrator. Real-time numerical integration in a LabVIEW environment (software layer) not only taxes the speed of computation in one iteration but also is very sensitive to external noise and DC offsets. Therefore, an in-house hardware based integrator is developed, details of which are shown in Fig. 6.

To integrate acceleration signals from the IEPE based accelerometer, a capacitor of $C0 = 1\,\mu F$ is placed for decoupling the constant DC current. Three Op-Amps, namely U1, U2 and U3 are deployed in series for the double integration operation — see Fig. 6. The output of U2 gives the velocity signal $\dot{x}(t)$, and of U3 gives the displacement signal $x(t)$. U1 performs the first integration but provides a phase shift of 180° which is taken care of by U2, which does not have a capacitor in feedback and thus acts only as an inverter. A feedback resistor is also added for each of the three Op-amps to avoid output voltage saturation. For any single integration operation, the values of the feedback capacitor and the resistor decide the corner frequency of the integrator, i.e. the frequency above which proper integration happens. The corner frequency for the fabricated integrator is 100 Hz.

Appropriate calibration of $x(t)$ and $\dot{x}(t)$ signals measured using the accelerometer with the hardware integrator is vital for the measurement to be useful. Calibration was performed by using the previously calibrated eddy current sensor, such that the response amplitudes with both measurement methods approximate each other. Gain calibration of the hardware integrator was performed by exciting the flexure with the actuator in an open-loop configuration using a sine chirp excitation between 30 Hz and 300 Hz and comparing the response of the flexure measured using the eddy current sensor with the response measured using the accelerometer with the hardware integrator. Furthermore, displacement, velocity, and acceleration signals are also used for the testing of various control strategies of active damping devices.

The software layer

The software layer (S/W) is developed in the NI LabVIEW environment. The real-time computations are performed on a NI cRIO-9036 module with an onboard Field Programmable Gate Array (FPGA) Kintex-7 70T. For efficient working of the software layer, it is divided into three major parts as shown in Fig. 7. The target device for the data logging is the Host computer.

The FPGA module

The FPGA code contains an analog input/output module and filters as shown in Fig. 8. To execute the code chronologically from analog input (AI) to the analog output (AO) module, a *Flat sequence structure* is used. The analog signals coming through an ADC (NI9234) are sampled at a default sampling frequency of 10.2 kHz. Sensitivity of performance to different sampling rates is discussed subsequently. A Butterworth high pass and low pass filter is applied to the acquired signals to filter out the suspension frequency of the shaker and to eliminate any electrical interference. The filtered signals are then sent to the real-time (RT) code, wherein regenerative chatter and active damping loops are implemented. In addition, the load cell signal from the force provided by the shaker is monitored in the FPGA.

Characteristics of digital filters

A first order high pass filter with a cutoff frequency of 40 Hz and second order low pass filter with a cutoff frequency of 700 Hz are used in conjunction for digital filtering of the signal. The cutoff frequencies are chosen to be higher than the effective measurement range of the accelerometer and to retain only the lower structural modes of interest. As the chatter occurs in vicinity of the natural frequency (f_n) of the system, hence, the steady state phase and gain provided by the filter at $f_n = 112.6$ Hz are chosen, i.e., -6 dB and $\theta_{filter} = 58.6°$ respectively as shown in Fig. 9 which implies that although the filters reduce the amplitude of the displacement signal, they provide a phase lead, i.e., a left shift of the input signal on time axis. The reduced amplitude of the filtered signal was compensated by an appropriate digital gain $g_{filter} = 1.98$

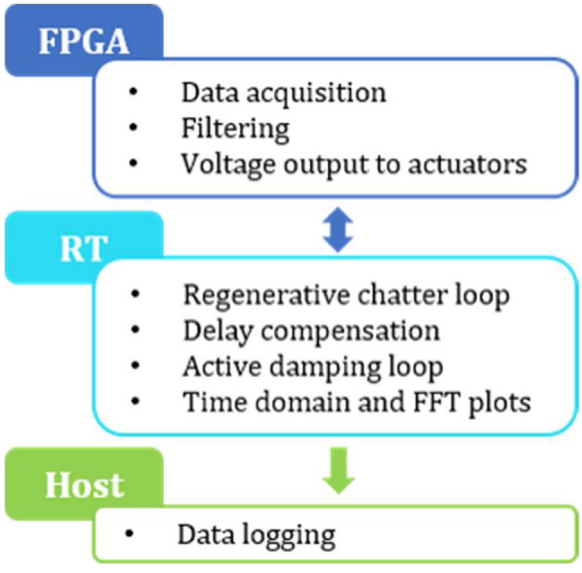

Fig. 7. Overview of the software layer.

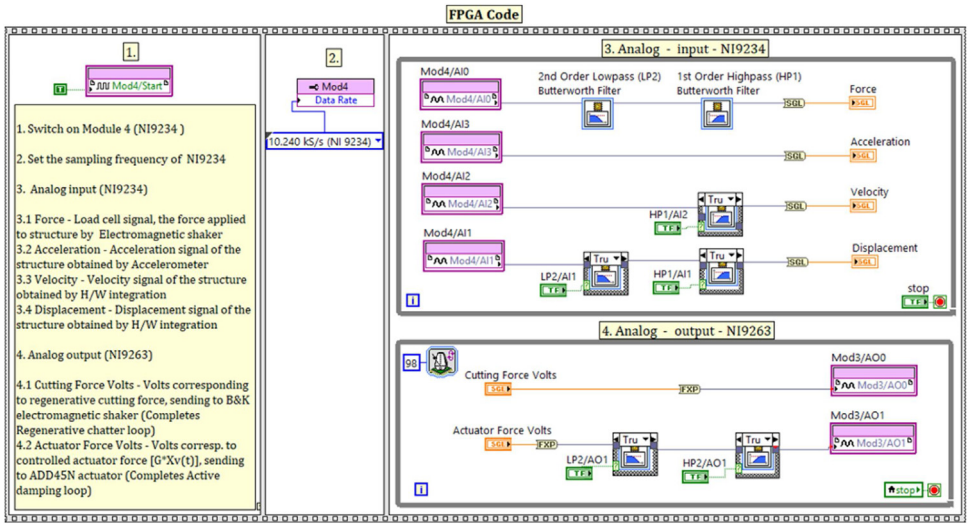

Fig. 8. FPGA code showing the signal acquisition and conditioning modules.

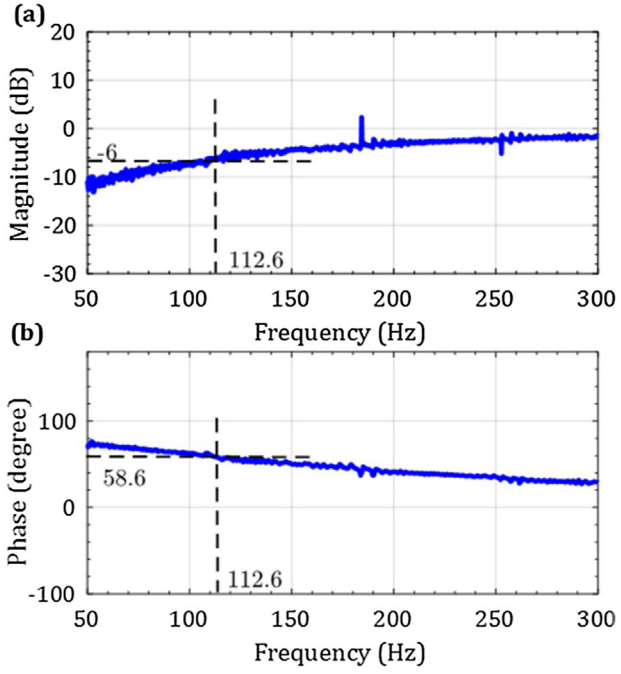

Fig. 9. Frequency characteristics of the digital filters: first order high pass filter with a cut-off frequency of 40 Hz working in conjunction with a second order low pass filter with a cut-off frequency of 700 Hz.

in the RT code. Phase characteristics of the filter used is separately investigated in the section *Estimation of total delay*.

The real-time (RT) module

Variables are accessed in the RT code from the FPGA reference and used for further calculations. Mathematical formulation of orthogonal regenerative chatter is written in the *Control & Simulation Loop* — see Fig. 10. The runtime of the single iteration of the loop is set at 100 μs (corresponding to a sampling rate of 10.2 kHz). Displacement signals corresponding to the previous revolution, $x(t-T)$ are approximated as delayed displacements obtained by using the *Transport delay* block within LabVIEW. Following this, the regenerative cutting force is calculated and is converted into voltage by multiplying it by g_{VN} as discussed previously. Subsequently, voltage corresponding to the cutting force is then sent to the analog output module in the FPGA which is further sent to the amplifier of the shaker through the DAC. A fixed step size for the Runge-Kutta 4th Order (RK-4) solver is used as the solver for the *Control and Simulation Loop*.

The graphical user interface (GUI)

To ease performing experiments, a graphical user interface (GUI) was built, as shown in Fig. 11. The GUI displays the information about the sensitivity of sensors and actuators, timing of different loops present in the RT code, and the cutting conditions to emulate. Additionally, it provides control of the digital filters. For the HiL experiments a workpiece material of steel is assumed and an empirical coefficient, $K_f = 1384$ MPa is selected. The time series data and frequency spectrum of vibration data are logged, and can be loaded in other environments for post-processing. A control of delay compensation filter is provided in the GUI to compensate for the delay in the system. Furthermore, a control is provided to test different control strategies of active damping.

Experimental characterization of chatter with the HiL simulator

Experiments are conducted sequentially for a pair of the simulated depth of cut and a specified spindle speed. The simulated depth of cut is incremented in steps of 0.01 mm, and specified spindle speed in steps of 50 RPM. To understand the stability behavior of the orthogonal cutting process, an initial force that acts as a perturbation to the system is provided by applying the static component of the force that occurs due to the static chip thickness of h_0, to result in a force of $K_f b h_0$. The simulated depth of cut b is then increased at a specified spindle speed N and the response is monitored to detect if the system is stable or not. For the case of a stable depth of cut, the flexure responds to the initial perturbation, and then the transients slowly die down. However, for the case of an unstable depth of cut, due to regenerative effects, after the response due to the initial perturbation, the response $x(t)$ starts to grow with time and the observed oscillation frequency (chatter frequency) ω_c for that case is recorded. Also, since the shaker is unable to apply a purely static force, the measured force has a mean of zero, i.e., only the dynamic part of the force is acting on the flexure. However, due to the closed-loop interaction of the process (software layer) and the flexure representing the machine (hardware layer), the force too exhibits a growing tendency for the unstable case.

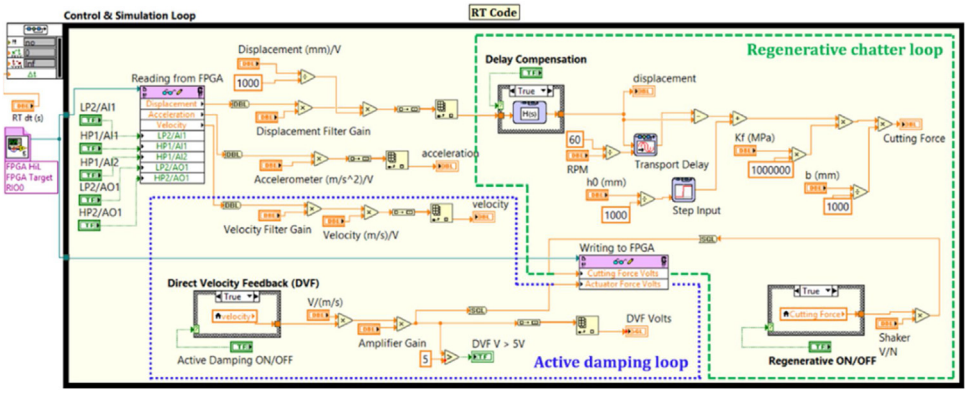

Fig. 10. Control and simulation loop in the RT code.

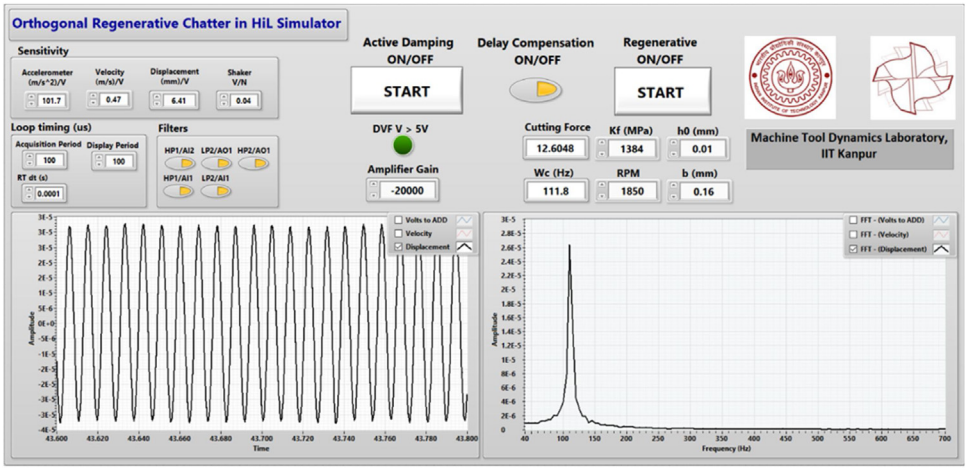

Fig. 11. Graphical user interface for experimentation on the HiL simulator.

Experimentally obtained stability lobe diagram with the corresponding chatter frequencies are compared with the theoretical results as shown in Fig. 12. Theoretical results are obtained using the measured FRF (H_{21}) of the flexure and Eq. (7). Experiments were conducted with the contact type accelerometer transducer with a hardware integrator. As is evident from Fig. 12, experimentally characterized stability behavior with the HiL simulator diverges significantly from theoretical predictions. Earlier work on a HiL simulator [14,16] for chatter in orthogonal cutting reported a similar divergence which was attributed to delay in the system. The delay in the HiL simulator acts like negative damping [27], and hence lowers the stability limit. In addition, the chatter frequency is also observed to be lower than the natural frequency of the flexure, which is typical for systems with delay. For the effective utilization of the HiL simulator to study the physics of regenerative chatter, systematic identification of the total delay in the system is necessary so as to compensate it.

Estimation of total delay

Separate experiments were conducted for the identification of delays caused by the key elements in the hardware and software layers. Key elements are considered to be the actuator, the integrator, and the digital filter. Inputs to and outputs from each of the key elements were recorded by providing a sinusoidal chirp signal ranging from 30 Hz to 300 Hz. Relative phase of the output with respect to the input for every element investigated was used to estimate the delay due to that element. Fig. 13 schematically shows the displacement acquisition path highlighting the input and output signals for the key elements.

Even though the flexure has its own phase response, that phase does not contribute to the divergent behavior, which is mainly attributed to the actuation, acquisition, and signal conditioning elements. Hence, the phase of the structure is neglected in the estimation of delay.

Having evaluated the phase of each element, we estimate the total delay using two methods. The first method, referred to as the '*Delay estimation using fixed frequency phase information*', simply sums the phase provided by each element at the frequency of interest, which is taken to be the natural frequency in this case, due to the fact that chatter occurs in the vicinity of it. This phase, if negative, implies an aggregate right shift of the input signal on the time axis, and can thus be converted to an appropriate time delay. The other method, referred to as the '*Delay estimation by linear interpolation*', simply factors the variation of phase over a range of frequencies of interest and estimates the delay from the slope of a linear interpolation of the phase-frequency behavior of the different elements.

Delay estimation using fixed frequency phase information

The measured phase response for the shaker, the digital filter and two response measurement techniques, i.e., the accelerometer with a hardware integrator and the eddy current sensor are shown in Fig. 14.

As discussed above, delay is estimated at a fixed frequency, i.e. at the natural frequency ($f_n = 112.6$ Hz) in this case. As is evident

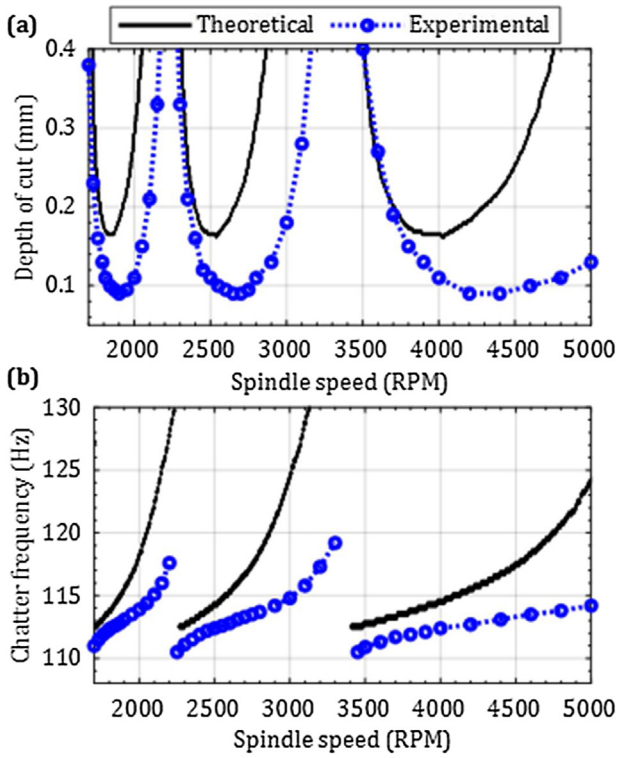

Fig. 12. Theoretical and experimental (a) stability lobe diagram (b) chatter frequencies.

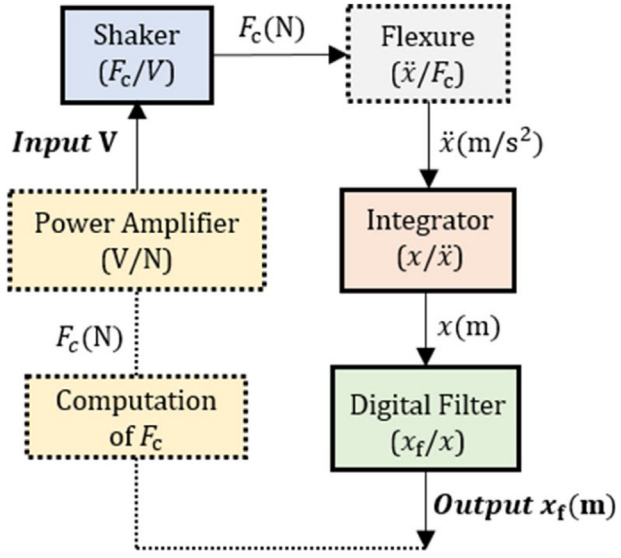

Fig. 13. Block diagram showing input and output for key hardware and software elements in the displacement acquisition path.

from Fig. 14(a), the phase provided by the shaker is $\theta_{\text{shaker}} = -34.86^\circ$, and as is evident from Fig. 14(b), the filter provides a phase of $\theta_{\text{filter}} = 58.6^\circ$. These phase contributions are independent of the response transducer. Since we use two different methods of measuring response: an accelerometer with a hardware integrator, and a non-contact eddy current sensor, the phase for each is evaluated separately.

The phase provided by the accelerometer with a hardware integrator is ideally expected to be 180°, however as is evident from Fig. 14(c), the hardware based integration leads to a reduced phase. Assuming the acceleration to be sinusoidal i.e., $\ddot{x}(t) = A\sin(\omega t)$ the displacement signal can be written as:

$$x(t) = \frac{A}{\omega^2}\sin(\omega t + \theta_{\text{H/w Int.}}) \equiv \frac{A}{\omega^2}\sin(\omega t + 180^\circ + \theta_{\text{integrator}}). \quad (8)$$

The phase contribution from hardware integrator in the total phase can thus be deduced to be:

$$180 + \theta_{\text{integrator}} = \theta_{\text{H/w Int.}} \equiv 108.8^\circ \rightarrow \theta_{\text{integrator}} = -71.20^\circ. \quad (9)$$

Since all elements of the HiL simulator are in series, the total phase is simply a sum of the phase contributed by individual elements. For the case of the accelerometer with a hardware integrator, a total phase is evaluated to be -47.5°, see Eq. (10a). This negative total phase, implies an aggregate right shift of the input signal on the time axis and thus it can be seen as a time delay. The total delay is calculated at the natural frequency of the flexure using Eq. (10b). An additional delay of $\tau_{\text{sampling}} = 100\,\mu s$ is also added to take into account the time it takes to complete one iteration of the control loop.

$$\theta_{\text{total}_{\text{H/W Int.}}} = \theta_{\text{shaker}} + \theta_{\text{filter}} + \theta_{\text{integrator}} \cong -47.5^\circ \quad (10a)$$

$$\tau_{\text{total}_{\text{H/W Int.}}} = -\frac{\theta_{\text{total}_{\text{H/W Int.}}}(\text{rad})}{\omega_n(\text{rad/sec})} + \tau_{\text{sampling}} \cong 1.3 \text{ ms} \quad (10b)$$

The total delay estimated with the accelerometer as the response transducer is ~ 1.3 ms. This delay is slightly larger than the delay reported in Refs. [14,16], and is thought to be larger due to our use of the contact type accelerometer transducer with a hardware integrator.

For the case of the eddy current transducer, the phase, θ_{Eddy} is evaluated relative to the hardware integrator based displacement measurement done simultaneously in the same experiment. This phase is shown in Fig. 14(d). The use of an eddy current transducer results in much larger total negative phase — see Eq. (11a), which shows the sum of all phases with this transducer to be ~ −159.3°, which when converted to a time delay, evaluated again at the same fixed frequency ($f_n = 112.6$ Hz) results in a large time delay of ~ 4.1 ms, see Eq. (11b).

$$\theta_{\text{total}_{\text{Eddy}}} = \theta_{\text{shaker}} + \theta_{\text{filter}} + \theta_{\text{integrator}} + \theta_{\text{Eddy}} \cong -159.3^\circ \quad (11a)$$

$$\tau_{\text{total}_{\text{Eddy}}} = -\frac{\theta_{\text{total}_{\text{Eddy}}}(\text{rad})}{\omega_n(\text{rad/sec})} + \tau_{\text{sampling}} \cong 4.1 \text{ ms} \quad (11b)$$

The time delay estimated with an eddy current transducer is significantly larger than the delay on account of using a contact type accelerometer transducer with a hardware integrator, and this is expected to further lower the experimentally characterized stability behavior on the HiL simulator — as is indeed confirmed in discussions in the following main section.

Delay estimation by linear interpolation

As opposed to evaluating the phase and delay at only a fixed frequency – the natural frequency, we also evaluate the delay over a range of frequencies of interest. The range of interest is taken to be between 115–130 Hz – the range corresponding to the theoretical chatter frequencies – see Fig. 12.

For a linear phase system, the true time delay should be independent of the frequency and is expected to follow the following form:

$$\text{Phase(rad)} = -\tau\,\omega(\text{rad/s}) + \text{Intercept(rad)} \quad (12)$$

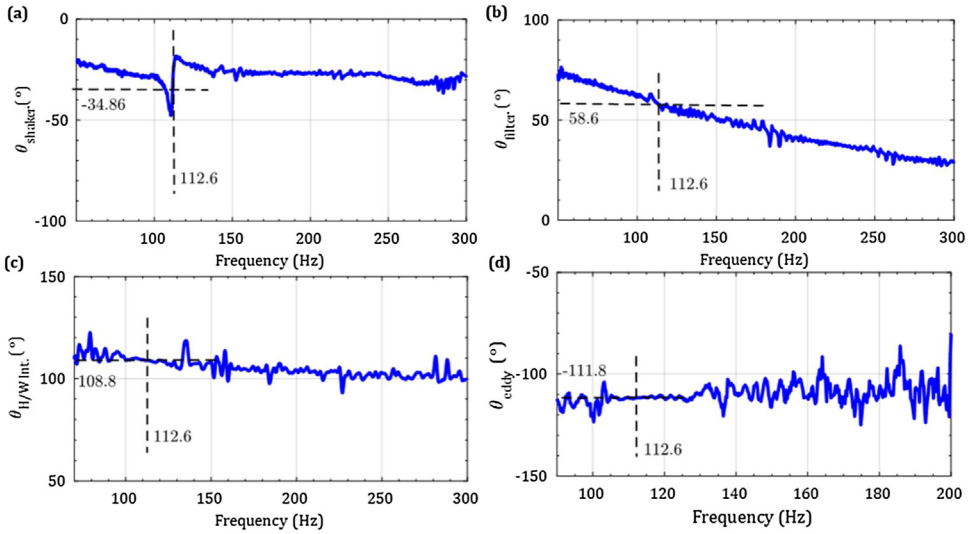

Fig. 14. Measured phase response for: (a) shaker [F_c/V]; (b) digital filter [x_f/x]; (c) hardware integrator [$x/\ddot{x}$]; and (d) eddy current sensor [$x_{eddy}/x_{H/w\,Int.}$].

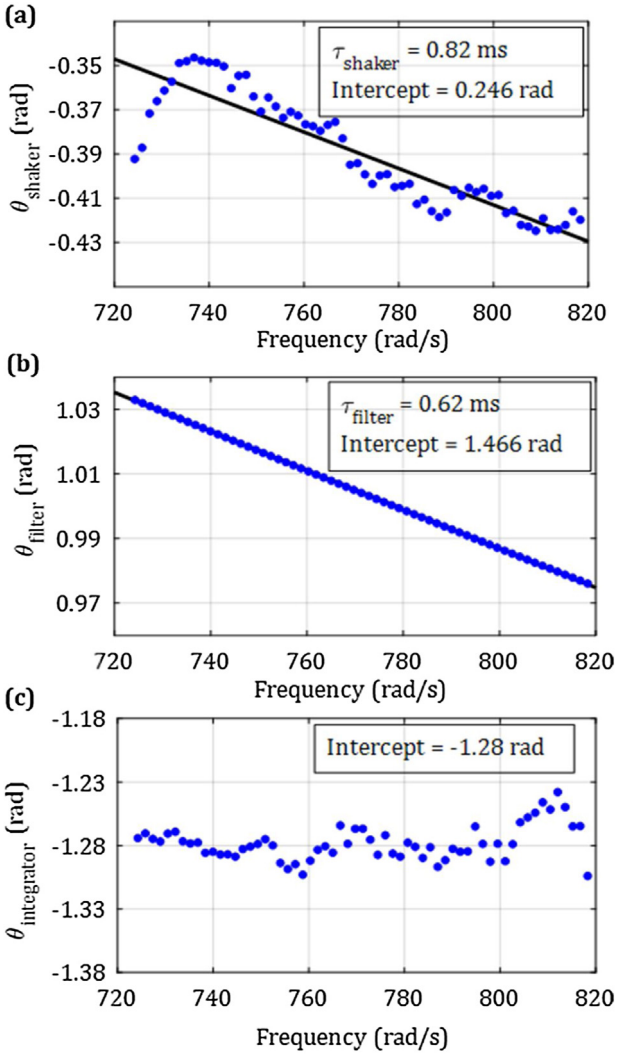

Fig. 15. Linear interpolation on the measured phase response for: (a) shaker; (b) digital filter; and (c) hardware integrator.

The slope from this linear equation gives us as an estimate of delay due to each of the elements, and the intercept represents an additional constant phase provided by the respective component. The measured phase-frequency behavior for the actuator, the filter, and the hardware integrator is shown in Fig. 15 for the frequency range of interest, i.e. from 115 to 130 Hz. Since the eddy current sensor was observed to result in much larger undesirable time delays, results with it are not discussed herein. Fig. 15 shows the phase and frequencies in radians and rad/s only for being consistent with Eq. (12). A line is fit to the data in Fig. 15(a) and Fig. 15(b), i.e. for the case of the actuator and the filter. The slopes, i.e. the time delays due to these components along with a constant phase shift for these components are listed on the figure. The hardware integrator has no slope, but just a constant negative phase line.

The total time delay from the linear interpolation of the phase-frequency behavior for all elements considered is found to be $\tau_{total} \cong 1.54$ ms, which includes the sampling delay of $\tau_{sampling} = 100\,\mu$s. The linear interpolation technique also gives an intercept value which is summed up to be $\text{Intercept}_{total} = \theta \rightarrow 0.432$ rad $\rightarrow 24.75°$. The delay estimated from linear interpolation accounting for the variation of phase over a range of frequencies of interest (evaluated to be ~ 1.54 ms) is thus not very different than the delay evaluated at a fixed-frequency (evaluated to be ~ 1.3 ms). These delays are first accounted for in the theoretical model — as is discussed next, and are subsequently compensated.

Accounting for the delay in the theoretical model

The delays, if identified correctly, when introduced into the theoretical model, are expected to make stability predictions with a delayed theoretical model match the experiments on the HiL simulator that has delays. Results for three theoretical models with delays are compared with two experimental results in Fig. 16. Of the three theoretical models with delays, one account for delays identified using the fixed-frequency method, and another one accounts for delays identified using the linear interpolation method. These results are compared with the experimental case with a contact type transducer, i.e. the accelerometer with a hardware integrator. Yet another model that accounts for delays identified using the fixed-frequency method is compared with experimental results with the non-contact type eddy current transducer.

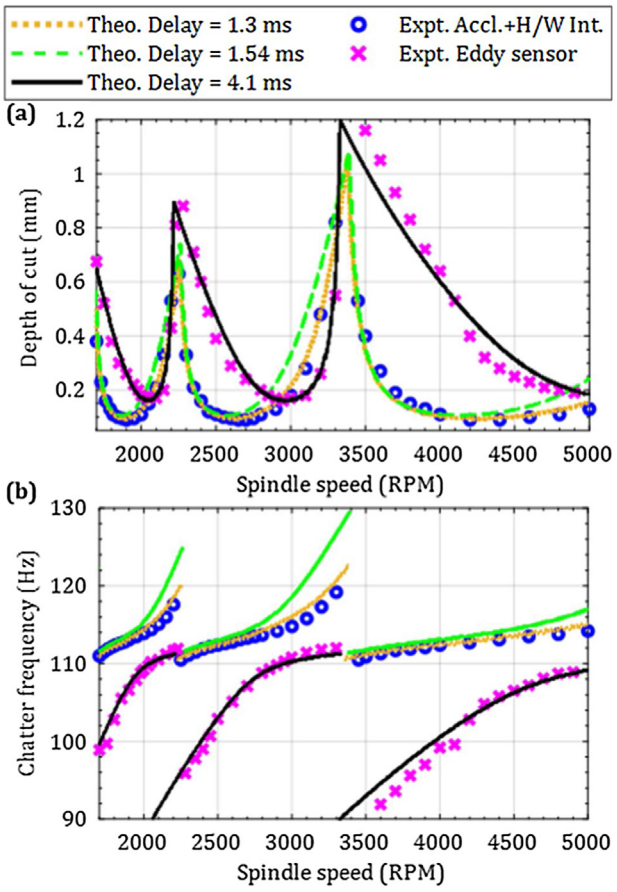

Fig. 16. Delayed theoretical and experimental (a) stability lobe diagram (b) chatter frequencies for accelerometer with a hardware integrator and eddy current sensor based response measurement.

The characteristic equations for stability are modified to account for delays as shown in Eq. (13). Eq. (13a) lists the characteristic equation with a delay identified using the fixed-frequency method; Eq. (13b) lists the characteristic equation with a delay identified using the interpolation method, and includes the constant phase in addition to the delay, and Eq. (13c) shows the characteristic equation for the case of the delay identified for the eddy current sensor.

$$1 + e^{-\tau_{\text{total}_{\text{H/W Int.}}} s}(1 - e^{-Ts})K_f b\phi(s) = 0. \tag{13a}$$

$$1 + e^{-\left(\tau_{\text{total}} - \frac{\theta}{\omega}\right)s}(1 - e^{-Ts})K_f b\phi(s) = 0. \tag{13b}$$

$$1 + e^{-\tau_{\text{total}_{\text{Eddy}}} s}(1 - e^{-Ts})K_f b\phi(s) = 0. \tag{13c}$$

Using these characteristic equations, Eq. (13a–13c), theoretically obtained stability lobes and chatter frequencies are generated for a delayed system, and are compared with experiments in Fig. 16.

Comparisons in Fig. 16 show that the delayed theoretical model compares well with experimentally characterized stability on the HiL simulator, verifying that the delay estimation is indeed correct. A comparison of the two methods of delay estimation show that there is not much of a difference in the delayed stability behavior between the two, however, the fixed-frequency method of delay estimation matches up better with the experimental observations, and hence for planned compensation only the delay estimates from the fixed-frequency approach will be used. The results also suggest that use of the eddy current sensor, on account of its larger delay, results in a lower stability envelope, and also much lower chatter frequencies than the case of using a contact-type transducer. We thence recommend avoiding the use of eddy current sensors for HiL simulators for chatter or any other closed loop system unless and until the delay caused by the eddy current sensor can be characterized correctly for it to be compensated suitably.

Before compensating the estimated delay, we digress for a discussion on the influence of the sampling rate on the stability behavior of the HiL simulator.

Influence of sampling rate on transport delay block

To understand the influence of sampling rates, experiments were conducted with two different sampling rates (f_s) of 25.6 kHz and 10.2 kHz. The RT code was suitably updated to modify the iteration time steps to 40 μs and 100 μs, corresponding to f_s of 25.6 kHz and 10.2 kHz. Experimental results are compared in Fig. 17 with the theoretically obtained stability limits for a model with no delay. As is evident, for the case of $f_s = 25.6$ kHz, there appear additional pseudo stability lobes in between the lobes for the case of $f_s = 10.2$ kHz. This peculiar behavior contradicts the results obtained in Ref. [17] which discusses that increasing the sampling rate decreases the divergence of the HiL simulator's stability from a continuous system.

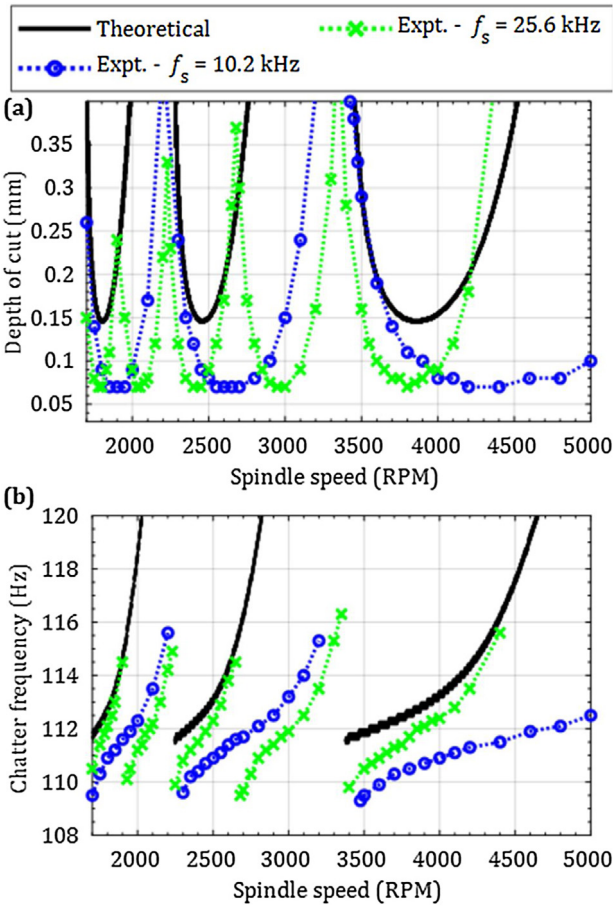

Fig. 17. Comparison of theoretical stability behavior with no delay with experimentally characterized stability on the HiL simulator for two different sampling rates.

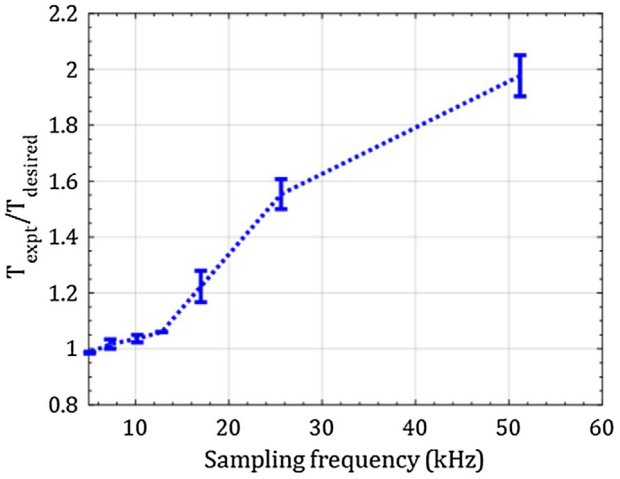

Fig. 18. Characteristic behavior of transport delay block for different sampling rates.

As is evident from Fig. 17, an increase in sampling rates does not appear to change the delay in the system on account of the hardware and software elements as the minimum stability limit is the same for both cases. Instead, sampling appears to influence the characteristic delay $T = 60/N$ in the regenerative system. Since the delay is obtained by using a *Transport delay* block in the RT code within LabVIEW, the performance of the *Transport delay* block is thought to degrade with an increase in the sampling rate. This is indeed confirmed by separately monitoring the input $(x(t))$ to and output from $(x(t-T))$ the delay block. The experimentally characterized delay T_{expt}, normalized with $T_{\text{desired}} = T$, when compared for increasing sampling rates, clearly diverges from unity, as shown in Fig. 18.

As is evident from the Fig. 18, at $f_s = 25.6$ kHz, the *Transport delay* block is ineffective as it gives an output of $x(t-1.5T)$ which naturally distorts the stability behavior. For f_s less than 12.4 kHz the transport delay block appears effective, with errors less than 2 %. Based on these observations, we recommend that the *Transport delay* block within LabVIEW should not be used for sampling frequencies greater than 12 kHz for a HiL simulator for chatter.

Since the *Transport delay* block is introduced to account for the delay between the current and the previous vibration response, i.e., to estimate the dynamic chip thickness, alternative methods that store the vibration response from all previous revolutions and use that information to estimate the dynamic chip thickness may be used [17]. However, storing all previous data can be computationally expensive, and may result in increasing the computation delay time. Hence, yet another alternative for use of the *Transport delay* block could be the use of an efficient surface location storage algorithm that stores data of only the surface resulting from all the previous vibrations of tool along with the relative revolution number, thereby lowering the computational cost [21].

Compensation of delay

The delay in the system can be compensated by using predictive methods or by using a conventional low order phase lead filter. Predictive type delay compensation methods such as Polynomial based methods, or the Smith Predictor method are not suitable for the present case since these predictive methods need a pre-generated reference signal on the basis of which the compensated signal is to be derived [27–29] — which is not available in a HiL simulator to study self-excited vibrations. A first order phase lead compensator is also not very effective for reshaping frequency characteristics of a delayed system since the magnitude of such a compensator linearly increases with frequency, which may amplify high-frequency noise, thus corrupting the signal, and rendering the compensator of less use.

Due to the difficulties with aforementioned methods, we instead compensate for the delay present in the system using an adaptive phase lead compensator which provides necessary phase at the natural frequency $(\mathrm{f_n})$ and rejects high frequency noise, as was likely also done in [16]. From the delay identified in the previous section based on the fixed-frequency method, the total phase to compensate at the natural frequency of the structure is:

$$\theta_{\text{compensation}} = \tau_{\text{total}_{\text{H/W Int.}}} \times \mathrm{f_n} \equiv -52.7^{\circ} \tag{14}$$

The compensator is desired to give a 0 dB magnitude and phase lead of $\sim 53^{\circ}$ near the natural frequency of the flexure. The compensation filter is designed in the frequency domain by the Bode plot technique. Due to the ineffectiveness of a simple first order phase lead compensator, the transfer function was constructed by incrementally increasing the number of poles and zeros to achieve the desired characteristics near the natural frequency. A second order numerator with complex zeros and a tuning parameter ζ_c, along with a third order denominator with real poles is chosen for the transfer function of the compensator as seen in Eq. (15):

$$C(s) = \frac{\mathrm{K}\left[\left(\frac{s}{z_1}\right)^2 + 2\zeta_c\left(\frac{s}{z_1}\right) + 1\right]}{(s+\mathrm{p_1})(s+\mathrm{p_2})(s+\mathrm{p_3})} \tag{15}$$

Parameters of the compensation filter are given in Table 1, and the asymptotic approximation of the same is shown in Fig. 19. Since we are manually tuning the order of the filter, other solutions are also possible. An effective alternative approach could have been to setup a least-squares problem with an objective of reducing the difference between the real parts of the compensated response and the original response of the system, and thereby resulting in a suitable fitted structure to, and, order for the compensation filter.

To check the effectiveness of the designed compensator, we compare in Fig. 20 the real part of the original FRF of the flexure $(Re[\phi(s)])$ with the FRF of the delayed system $\left(Re\left[\mathrm{e}^{-\tau_{\text{total}_{\text{H/W Int.}}} s} \times \phi(s)\right]\right)$, and that of the compensated system $Re\left[C(s) \times \mathrm{e}^{-\tau_{\text{total}_{\text{H/W Int.}}} s} \times \phi(s)\right]$. As is evident from Fig. 20, the minimum value of the real part is lowered further in case of a delayed system which proportionately decreases the stability limit but the real part of the compensated system matches up well with original system.

Seeing that the compensation filter appears effective, it is implemented by using a *Continuous Transfer Function* in the RT code inside the software layer and is placed ahead of the response signal as shown in Fig. 10. Experiments are repeated on the HiL simulator with the compensation filter implemented, and stability results are compared in Fig. 21.

As is evident from Fig. 21, the experimentally compensated system's stability compares well with theoretically predicted stability behavior without delay, suggesting that the HiL simulator

Table 1
Specifications of the delay compensation filter.

Parameters	Values
K	1.84e + 9
$\mathrm{p_1}$	100 rad/s
$\mathrm{p_2}$	3000 rad/s
$\mathrm{p_3}$	6000 rad/s
$\mathrm{z_1}$	250 rad/s
ζ_c	0.35

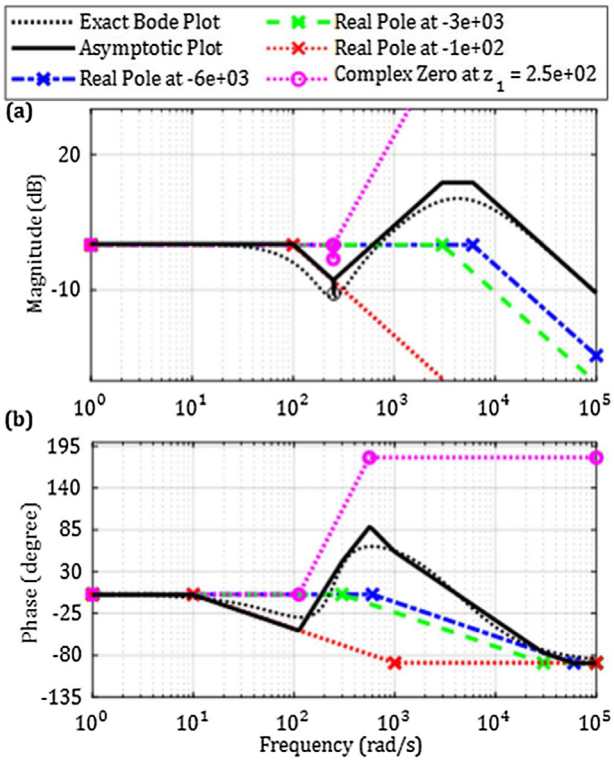

Fig. 19. Dynamic characteristics of the compensation filter (a) magnitude (b) phase.

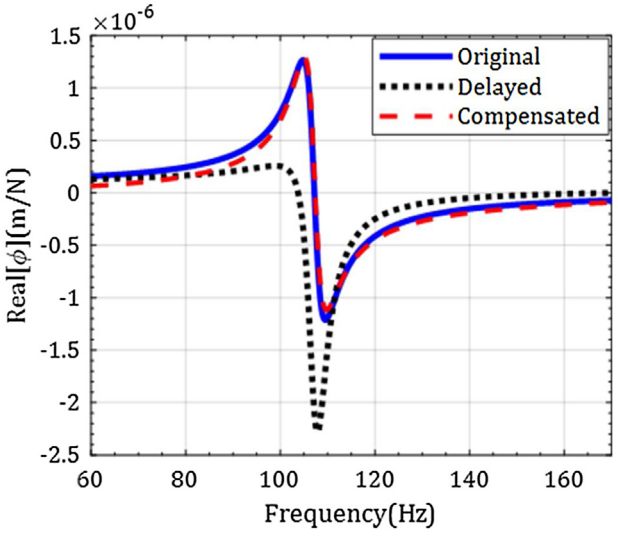

Fig. 20. The real part of the FRF for the original, delayed and compensated system.

is validated. This validated HiL simulator is used as a platform to investigate the effectiveness of different control strategies for active vibration control as is discussed next.

Active damping control strategies

HiL simulators offer a convenient experimental platform to test different active damping control strategies to increase the dynamic stiffness of the structure (representing a machine) using a sensor, actuator and a controller in a closed-loop. Actuator design and the control strategy influence the effectiveness of an active damping system. Similar to previous work [12–14,16,20,22–24,30–32], the present study focuses on the use of four different control strategies

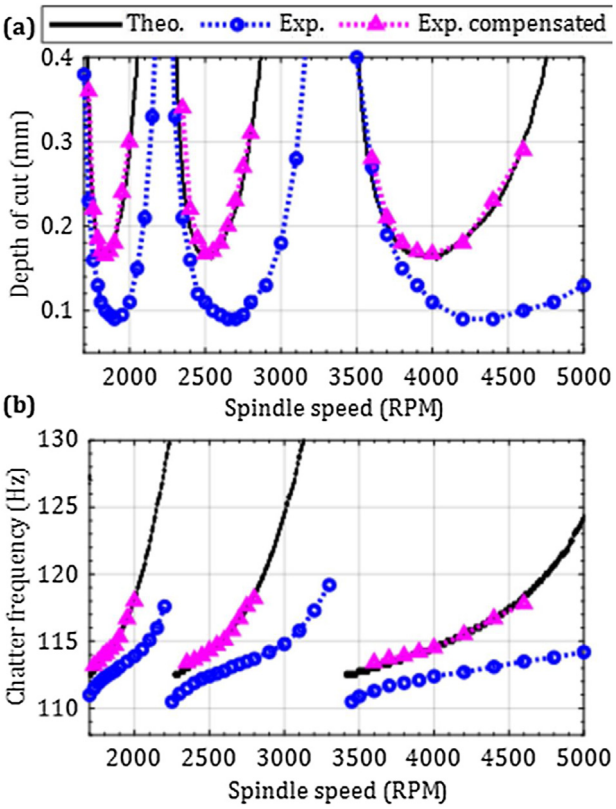

Fig. 21. Theoretical and experimental results with and without compensation of delay.

for the active damping device and their effect on the stability boundary of the HiL simulator for orthogonal cutting. We investigate non-model based control strategies with direct acceleration feedback (DAF), direct velocity feedback (DVF), direct position feedback (DPF), and with delayed position feedback (DelPF).

The experimental setup of an active damping system is shown in Fig. 22. The acceleration sensor measures the response of the flexure to a dynamic excitation by the main actuator. The response is fed back to the controller which computes the command signal for the actuator to apply the active damping force F_a at location 3 while the main actuator applies the cutting force F_c at location 1 (for locations, please refer to Figs. 3 and 4). For the present HiL setup, even though the second mode that occurs at 530 Hz is observable at location 3, i.e., the location where the active damping device applies a force, since the dynamic stiffness of the second mode of the flexure at 530 Hz is one order of magnitude higher than the dynamic stiffness of the primary dominant mode at 112.6 Hz, the minimum stability limit corresponding to the second mode is ~10 times that of the first mode at 112.6 Hz mode. And, since excitation and destabilization of the system due to the higher-frequency mode requires force excitation levels beyond what the primary exciter can provide, we were unable to excite and/or destabilize the system being dominated by the higher-frequency mode in the HiL experiments. Hence, the 530 Hz mode does not have any influence on the stability behavior of HiL simulator while testing all four control strategies.

The same acceleration sensor was used in the regenerative chatter and active damping loop. For the DPF, DelPF, and the DVF control strategies, the accelerometer along with the hardware integrator was used. A proof mass electromagnetic actuator (Make: Micromega Dynamics; Model number: ADD45N; with $F_{max} = 45$ N) is used for active damping.

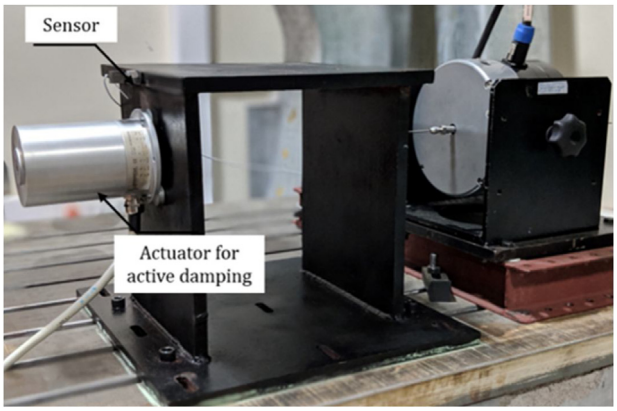

Fig. 22. HiL setup with the active damping device.

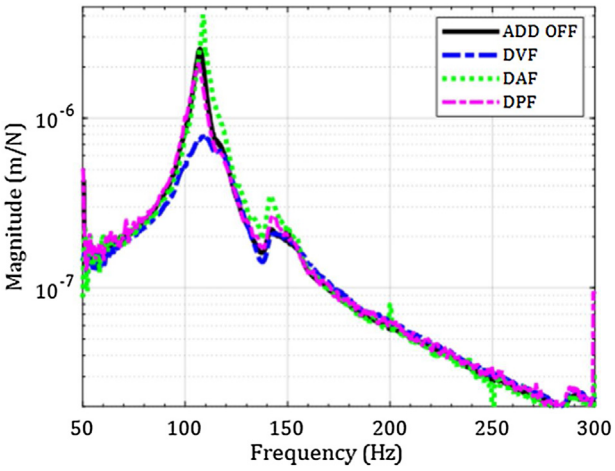

Fig. 23. Comparison of the response of the flexure to different control strategies.

Table 2
Different control laws, active damping force and control gains.

Control laws	Active damping force (F_a)	Control gains
DPF	$g_a K_{DPF} x(t)$	$K_{DPF} = 30000$ [A/m]
DVF	$g_a K_{DVF} \dot{x}(t)$	$K_{DVF} = 600$ [A/m/s]
DAF	$g_a K_{DAF} \ddot{x}(t)$	$K_{DAF} = 0.18$ [A/m/s^2]
DelPF	$g_a K_{DelPF} x(t-T)$	$K_{DelPF} = 28800$ [A/m]

The controller of the active damping system is designed in LabVIEW as shown earlier in Fig. 10. The active damping force F_a is governed by the feedback of a particular state of the flexure i.e. response of vibrations, multiplied by a control gain K. For stable operation of the active damping system, the control gain of the controller for each strategy is determined prior to the experiments. Model-based root locus plots were used to guide gains to be selected during the active damping experiments for the all control strategies investigated. Gains were then sequentially increased in the experiments. The stopping criterion for increasing gains was not until instabilities were observed experimentally, but until a gain corresponding to the active damping device applying a force of ~30 N was reached. The ~30 N force corresponds to a signal of 7 V being supplied to the actuator. This cut-off voltage was selected to be deliberately lower than full dynamic voltage range of the active damping device to ensure that the active damping device does not reach its force saturation limit of 45 N, and that there is no damage of the device. The gains thus obtained are tabulated in Table 2. It should be noted that the gains hence selected may not be optimal. Subsequently, the actuator force, as directed by the controller, can be expressed as:

$$F_a = g_a \times K \times feedback\,state \tag{16}$$

wherein $g_a (= 20\,N/A \equiv 4\,N/V)$ is the actuator force constant [33].

Four control laws using different feedback states are described in Table 2.

The active damping control strategies i.e. DPF, DVF, DAF were tested for their effectiveness in improving the dynamic stiffness of the flexure prior to integrating them in regenerative chatter experiments. Tests were performed by switching on (Control ON) the active damping control loop in LabVIEW while the flexure was excited by a sine sweep signal through the main actuator, and the response was measured. The H_{21} FRF thus obtained, called $\phi_{Control_{ON}}(s)$ is shown in Fig. 23, which is compared with the response when the control is switched off (ADD OFF), called $\phi_{Control_{OFF}}(s)$. Amongst the three control laws, the DVF control strategy shows the maximum improvement in the dynamic stiffness of the flexure. The delayed position feedback (DelPF) control law, as suggested in [31] was not investigated at this stage, since it is possible to implement this only with emulated machining conditions, as described next.

Experiments were subsequently undertaken to investigate the influence of active damping on regenerative chatter. Results with all four different control laws are presented in Fig. 24. The figure also shows the theoretical stability boundaries obtained with/without active damping. For DPF, DAF, DVF control strategies, the theoretical boundaries with active damping were obtained by the following modified characteristic equation:

$$1 + e^{-\tau_{total_{H/W\ Int.}} s}(1 - e^{-Ts}) K_f b \phi_{Control_{ON}}(s) = 0, \tag{17}$$

wherein $\phi_{Control_{ON}}(s)$ corresponds to the measured FRFs with the controller being ON.

The delayed position feedback strategy aims at decreasing the regenerative effect, the characteristic equation for which can be formulated as:

$$1 + e^{-\tau_{total_{H/W\ Int.}} s}\left(1 - \left(1 - \frac{K_{DelPF} g_a}{K_f b}\right) e^{-Ts}\right) K_f b \phi_{Control_{OFF}}(s) = 0. \tag{18}$$

Investigations with active damping were carried out on an uncompensated system, therefore the characteristic equations Eqs. (17) and (18) include a delay term in it, like the one in Eq. (13), to account for all the delays in the system.

Results in Fig. 24 show that experimentally characterized stability behavior of the HiL with active damping compares well with model predictions for every case investigated. Results suggest that the DPF strategy marginally improves (raises) the minimum stability limit, and that the DAF strategy lowers the stability limit, and that the DVF control strategy gives the maximum improvement in the minimum stability. These results confirm earlier reported trends with different control strategies in Refs. [12,13,16,31,32].

Interestingly however, results for the DelPF strategy do not follow the trend reported in Ref. [31] but match the trend reported in Ref. [20]. We observe that for the DelPF strategy, there are some values of RPM at which the control loop is unstable for all depth of cuts – for example, see results in the RPM range of 2000–2200 RPM. This loss of stability is because of an additional delay present in the HiL setup apart from the inherent delay (T) present in the chatter loop, and because of using a sub-optimal controller gain. For effective use of DelPF control law, delay must be compensated, and the controller gain should be optimally selected, i.e., $K_{DelPF} g_a = K_f b$, as also observed by Ref. [20]. However, since in

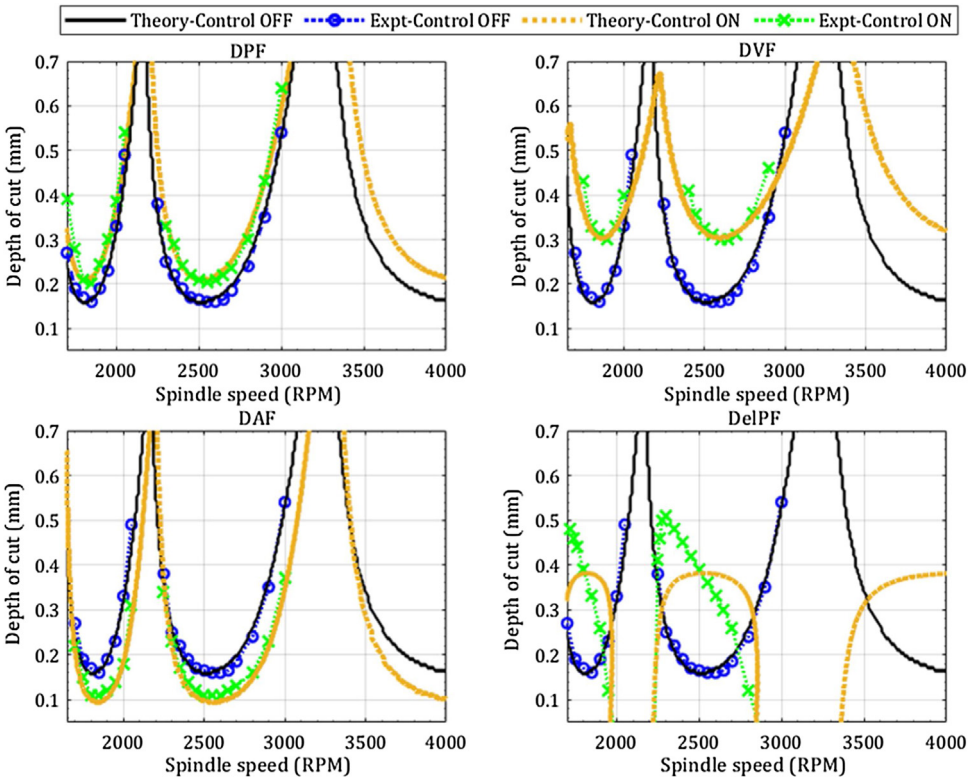

Fig. 24. Experimental and theoretical results showing the effectiveness of four active damping control strategies to improve the stability of the orthogonal cutting process. DPF-Direct position feedback; DVF-Direct velocity feedback; DAF-Direct acceleration feedback; DelPF-Delayed position feedback.

an active damping control system there will always be some additional delay, unless the delay is characterized correctly, and then compensated suitably, the DelPF strategy may prove ineffective. The validated HiL simulator thus proves useful to investigate control strategies for active damping of machine tool systems.

Conclusions

This paper presents a mechatronic Hardware-in-the-Loop (HiL) simulator to study regenerative chatter stability in an orthogonal cutting process. The HiL simulator employs a hardware layer which consists of a flexure representing a machine, and an actuator that provides the pseudo cutting force calculated in real time in the software layer. Response of the flexure is measured with two different sensors, a contact-type accelerometer with an in-house fabricated hardware integrator, and a non-contact type eddy-current sensor. The communication between the hardware and the software layers involves delays that result in the experimentally characterized stability with the HiL setup diverging from theoretical predictions. Through systematic experimental characterization of all the key components of the hardware and software layers, we identify the total delay in the mechatronic system.

We demonstrate two methods to estimate the delay using the phase-frequency characteristics of each of the elements and find that both methods are comparable. We also find that the delay with an accelerometer with a hardware integrator is less when compared to a non-contact type eddy current displacement sensor. We show that by using a hardware integrator we can circumvent challenges faced by real-time numerical integration in the software layer, thus offering a low-cost alternative to the laser sensors used in HiL setups by other researchers.

In the software layer, we observe peculiar loss of stability behavior with increasing sampling rates, and recommend that for effective use of a HiL setup, calibration of the *Transport delay* block in LabVIEW is necessary. We compensate the delay using an adaptive phase lead filter, and observe that the experimentally characterized stability behavior on a compensated HiL setup matches up well with theoretical predictions.

The validated HiL simulator is thence used to investigate the effectiveness of four different control strategies for active damping of vibrations. Results suggest that amongst the four non-model based control strategies tested, direct velocity feedback strategy is the most effective in improving the stability limit while the delayed position feedback which has been proven effective otherwise, is not recommended unless and until the mechatronic delay is characterized correctly and then compensated suitably.

Since the HiL setup presented herein facilitates testing of chatter models without the vagaries of uncertainties in machine tools, the setup serves as a convenient platform to test other advanced chatter models such as including nonlinear force models, process damping models, speed dependent cutting coefficients, etc. Furthermore, since the HiL simulator makes possible testing that is non-destructive, cost-effective, repeatable, rapid, and safe, it makes for an excellent pedagogical tool in laboratories to motivate future researchers interested in understanding the complexities of machine tool chatter.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Government of India's Impacting Research Innovation and Technology (IMPRINT) initiative through project number IMPRINT 5509. The authors would also like to acknowledge Mr. Srijan Bharati's help in developing the hardware components of the simulator.

References

[1] Tobias, S.A., Fishwick, W., 1958, Theory of Regenerative Machine Tool Chatter. The Engineer, 205/February (7): 199–203.
[2] Tlusty, J., 2000, Manufacturing Processes and Equipment. Prentice Hall.
[3] Merritt, H.E., 1965, Theory of Self-Excited Machine-Tool Chatter: Contribution to Machine-Tool Chatter Research—1. ASME Journal of Engineering for Industry, 87/4: 447–454. http://dx.doi.org/10.1115/1.3670861.
[4] Altintas, Y., Weck, M., 2004, Chatter Stability of Metal Cutting and Grinding. CIRP Annals – Manufacturing Technology, 53:619–642. http://dx.doi.org/10.1016/S0007-8506(07)60032-8.
[5] Quintana, G., Ciurana, J., 2011, Chatter in Machining Processes: A Review. International Journal of Machine Tools and Manufacture, 51:363–376. http://dx.doi.org/10.1016/j.ijmachtools.2011.01.001.
[6] Siddhpura, M., Paurobally, R., 2012, A Review of Chatter Vibration Research in Turning. International Journal of Machine Tools and Manufacture, 61:27–47. http://dx.doi.org/10.1016/j.ijmachtools.2012.05.007.
[7] Munoa, J., et al, 2016, Chatter Suppression Techniques in Metal Cutting. CIRP Annals – Manufacturing Technology, 65:785–808. http://dx.doi.org/10.1016/j.cirp.2016.06.004.
[8] Fathy, H.K., et al, 2006, Review of Hardware-in-the-Loop Simulation and its Prospects in the Automotive Area. Modeling and Simulation for Military Applications, 6228. http://dx.doi.org/10.1117/12.667794. 62280E.
[9] Pritschow, G., Röck, S., 2004, "Hardware in the Loop" Simulation of Machine Tools. CIRP Annals – Manufacturing Technology, 53:295–298. http://dx.doi.org/10.1016/S0007-8506(07)60701-X.
[10] Röck, S., 2011, Hardware in the Loop Simulation of Production Systems Dynamics. Production Engineering, 5:329–337. http://dx.doi.org/10.1007/s11740-011-0302-5.
[11] Witt, M., Klimant, P., 2015, Hardware-in-the-Loop Machine Simulation for Modular Machine Tools. Procedia CIRP, 31:76–81. http://dx.doi.org/10.1016/j.procir.2015.03.015.
[12] Ganguli, A., et al, 2005, Active Damping of Chatter in Machine Tools — Demonstration with a "Hardware-in-the-Loop" Simulator. Proceedings of the Institution of Mechanical Engineers Part I: Journal of Systems and Control Engineering, 219:359–369. http://dx.doi.org/10.1243/095965105X33455.
[13] Ganguli, A., et al, 2006, Simulation and Active Control of Chatter in Milling via a Mechatronic Simulator. Journal of Vibration and Control, 12:817–848. http://dx.doi.org/10.1177/1077546306064708.
[14] Mancisidor, I., et al, 2013, Design of a Bench Hardware-in-the-Loop System for the Study of Chatter in Turning. 2013 European Control Conference (ECC) (July 17), pp.791–796. http://dx.doi.org/10.23919/ECC.2013.6669683. IEEE.
[15] Matsubara, A., et al, 2015, Evaluation of Dynamic Stiffness of Machine Tool Spindle by Non-Contact Excitation Tests. CIRP Annals, 64/January (1): 365–368. http://dx.doi.org/10.1016/j.cirp.2015.04.101.
[16] Mancisidor, I., et al, 2015, Hardware-in-the-Loop Simulator for Stability Study in Orthogonal Cutting. Control Engineering Practice, 44:31–44. http://dx.doi.org/10.1016/j.conengprac.2015.07.006.
[17] Miklos, A., et al, 2017, The Development of High Speed Virtual Milling Test. ASME 2017 Dynamic Systems and Control Conference American Society of Mechanical Engineers, .
[18] Miklós, Á, et al, 2018, Hardware-in-the-Loop Experiment Of Turning. Procedia CIRP, 77:675–678. http://dx.doi.org/10.1016/j.procir.2018.08.179.
[19] Stepan, G., et al, 2019, On Stability of Emulated Turning Processes in HIL Environment. CIRP Annals, 4–7. http://dx.doi.org/10.1016/j.cirp.2019.04.035.
[20] Mancisidor, I., et al, 2019, Delayed Feedback Control for Chatter Suppression in Turning Machines. Mechatronics, 63/November: 102276. http://dx.doi.org/10.1016/j.mechatronics.2019.102276.
[21] Sahu, G., et al, 2019, Investigating Nonlinear Finite Amplitude Chatter Instabilities Using a Hardware-in-the-Loop Simulator. 10th Int Cong On Mach, UTIS (Antalya, Turkey), pp.15–21.
[22] Huyanan, S., Sims, N.D., 2007, Vibration Control Strategies for Proof-Mass Actuators. Journal of Vibration and Control, 13:1785–1806. http://dx.doi.org/10.1177/1077546307080031.
[23] Kaliński, K.J., Galewski, M.A., 2014, Vibration Surveillance Supported by Hardware-in-the-Loop Simulation in Milling Flexible Workpieces. Mechatronics, 24:1071–1082. http://dx.doi.org/10.1016/j.mechatronics.2014.06.006.
[24] Abele, E., et al, 2008, Development and Design of an Active Work Piece Holder Driven by Piezo Actuators. Production Engineering, 2:437–442. http://dx.doi.org/10.1007/s1174lease 0-008-0123-3.
[25] Mahdi, E., 2010, Chatter Stability of Turning and Milling with Process Damping [Ph.D. Dissertation]. University of British Columbia.
[26] Varoto, P.S., de Oliveira, L.P.R., 2002, On the Force Drop off Phenomenon in Shaker Testing in Experimental Modal Analysis. Shock Vibration, 9:165–175. http://dx.doi.org/10.1155/2002/675674.
[27] Horiuchi, T., et al, 1999, Real-Time Hybrid Experimental System with Actuator Delay Compensation and its Application to a Piping System with Energy Absorber. Earthquake Engineering & Structural Dynamics, 28:1121–1141. http://dx.doi.org/10.1002/(SICI)1096-9845(199910)28:10<1121.
[28] Hashemi, S.R., Gh, M.M., 2014, Polynomial-Based Time-Delay Compensation for Hardware-in-the-Loop Simulation of a Jet Engine Fuel Control Unit. International Journal of Automation and Control Engineering, 8:323. http://dx.doi.org/10.1504/ijaac.2014.065449.
[29] Qi, C., Gao, F., et al, 2016, Smith Predictor Based Delay Compensation For a Hardware-in-the-Loop Docking Simulator. Mechatronics, 36:63–76. http://dx.doi.org/10.1016/j.mechatronics.2016.04.005.
[30] Zaeh, M.F., et al, 2017, Automatic Tuning of Active Vibration Control Systems Using Inertial Actuators. CIRP Annals – Manufacturing Technology, 66:365–368. http://dx.doi.org/10.1016/j.cirp.2017.04.051.
[31] Munoa, J., et al, 2013, Chatter Suppression in Ram Type Travelling Column Milling Machines Using a Biaxial Inertial Actuator. CIRP Annals – Manufacturing Technology, 62:407–410. http://dx.doi.org/10.1016/j.cirp.2013.03.143.
[32] Mancisidor, I., et al, 2014, Optimal Control Laws for Chatter Suppression Using Inertial Actuator in Milling Processes. 11th Int Conf High Speed Mach (September 2014, Prague, Czech Repub), .
[33] Operation and Maintenance Manual, Active Damping Device, ADD-45 N, Micromega Dynamics..