Journal of Vibration Engineering & Technologies
https://doi.org/10.1007/s42417-024-01461-6

ORIGINAL PAPER

# Smart Automatic Modal Hammer: Predictor–Corrector Approach for Accurate Excitation of Dynamical Systems

**Mohammad Nasr[1] · Aryan Singh[1] · Keegan J. Moore[1,2]**

Received: 3 April 2024 / Revised: 31 May 2024 / Accepted: 2 June 2024


## Abstract

**Purpose** This research introduces an innovative solution that revolutionizes the study of linear and nonlinear dynamical systems—a smart automatic modal hammer. With its affordability and intelligent capabilities, this automatic modal hammer becomes an invaluable tool for research and industry, enabling repeatable strikes with precise force control. This system's significance becomes particularly evident when studying nonlinear systems, which heavily rely on the excitation level for their dynamics. By offering a cost-effective design this proposed system proves to be robust in accelerating research on nonlinear dynamics, providing researchers with an efficient and accessible means to delve deeper into these complex systems.
**Methods** The proposed design integrates a commercial modal hammer, commonly used in modal testing, and a stepper motor. This stepper motor is enhanced with an encoder and servo driver, all expertly controlled by a Raspberry Pi.
**Results** What sets this system apart is its clever utilization of regression models to acquire knowledge of the intrinsic relationship between the applied force and hammer velocity precisely during the impact. This acquired knowledge is the foundation for controlling the motor's behavior, ensuring consistent and accurate excitation of the structure with the desired force.
**Conclusion** The capabilities of the proposed automatic modal hammer are demonstrated using a linear two-story tower and a model airplane wing with a nonlinear vibration absorber.

**Keywords** Automatic modal hammer · Modal analysis · Nonlinear dynamics

## Introduction

The study of dynamic structures through experimentation has significantly enhanced our comprehension and management of vibration phenomena [1–3]. Conventional approaches involve the application of external forces to stimulate the structure and the resulting response is then measured using a variety of techniques, such as accelerometers and digital image correlation [4]. In real-world scenarios, excitations arise due to environmental conditions that the system operates in as well as due to moving parts in the system itself. However, within laboratory settings, excitations are generated using tools like modal impact hammers, electromagnetic shakers, and piezoelectric actuators. Among these tools, the use of modal impact hammers poses challenges. Specifically, they require manual operation by a user and, thus, lack repeatability and precision for both impact location and amplitude. Unfortunately, exciting a structure at precise force levels multiple times and at the same location is quite difficult and often requires dozens of repeat tests. This lack of reproducibility and force control undermines the comprehension of highly nonlinear structures, given that their dynamics are contingent on the imposed excitation. The reason is that nonlinearity introduces dependence on amplitude (or system energy), such that the response of a system differs based on the amplitude of the excitation. This amplitude-dependence results in extremely complex phenomena including bifurcations, jumps in the amplitude of the response, internal resonances, modal interactions, and chaos [3, 5–8]. Furthermore, the dependence on amplitude renders the typical approach of ensemble averaging useless because doing so removes changes in the response due to amplitude, resulting in physically meaningless data.

The involvement of humans in utilizing modal impact hammers constrains the reproducibility and precision of

✉ Keegan J. Moore
kmoore@gatech.edu

[1] Department of Mechanical and Materials Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588, USA

[2] Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA

Published online: 14 June 2024

force control during excitations. To overcome these challenges, researchers have explored diverse approaches and mechanisms to automate the process of exciting dynamical systems. Automatic modal hammers (AMHs) fall into two main categories: those that use linear action and those that employ rotational systems [9]. The first linearly actuated AMH emerged as an innovation at NASA designed for non-destructive testing of foam insulation attached to metal [10]. This system can quantify the impact force and duration of impact using a load cell; the collected data is transmitted as an electric signal to other system components for signal conditioning and electronic analysis, ultimately providing a visual display on a screen. However, it was developed for non-destructive testing purposes and not specifically for subjecting other structures to vibration testing. Additional patented designs utilizing solenoids and spring mechanisms providing fixed amplitude impacts are documented in references [11–14]. Norman et al. [15] introduced a solenoid-based AMH equipped with a load cell and an impact tip for striking the experimental structure. Despite its commendable repeatability, this design is limited to low amplitude impacts and is unsuitable for studying nonlinear systems requiring a wide range of forces to explore the underlying dynamics. Notably, a drawback of their system is its requirement to be installed on the structure itself, a configuration that introduces unwanted alterations to the local dynamics of the specimen. In a recent development, Maierhofer et al. [8, 16] introduced a new design called AMimpact. At the heart of AMimpact lies a magnetic linear actuator, operating based on the principles of reluctance forces. When an electric current flows through its coils, a piston equipped with a force sensor moves forward to impact the target. To reset its position, a spring mechanism has been incorporated. The concept behind this system is to deactivate the current just before the impact occurs. However, a significant drawback of AMimpact is its nonlinear relationship between force and piston position, necessitating considerable effort from the user to precisely position the hammer at the correct distance from the target. To address these limitations, an enhanced version known as AMimpact Pro was developed [16]. This upgraded system features a more sophisticated actuator designed to adjust the peak impulse of the impact. Additionally, it incorporates an advanced control system to account for the unknown distance to the target. AMimpact Pro offers a force range spanning from 0 to 150 N, making it particularly well-suited for applications involving small-scale structures. Nevertheless, its utility diminishes when dealing with medium and large-scale structures or nonlinear systems. Furthermore, it does not exhibit good repeatability in its performance.

Among the first rotary AMH, Bediz et al. [17] introduced the Impact Excitation System designed for modal testing of miniature structures. The Impact Excitation System facilitates consistent, high-bandwidth, and controlled-force excitation to small structures. It comprises a flexure-based body housing an instrumented impact tip with a force sensor and an electromagnetic mechanism for initial deflection and release. Despite its success, the Impact Excitation System is unsuitable for testing medium- or large-scale structures. More recently, several rotary automatic modal hammers have become commercially accessible and are progressively gaining traction in research applications. Among these, the WaveHitMAX, developed by gfa tech GmbH, stands out as the first smart automatic modal hammer tailored for fully autonomous testing [18, 19]. Notably, the WaveHitMAX streamlines the force search process within five iterations, achieving less than a 10% deviation from the desired force value. The WaveHitMAX system guarantees complete automation, precision, and reproducibility when exciting a test subject without the possibility of double hits. Users retain the flexibility to define the hit count, impact force, and temporal intervals between impacts. Moreover, the hammer autonomously configures prerequisites like zero-point calibration. While the WaveHitMAX system demonstrates a commendable margin of error within 10% when striking a specimen, its drawbacks, including its relatively high cost and limited error tolerance, prevent it from delivering an ideal solution. Additionally, WaveHitMAX necessitates using additional weights to achieve greater forces, maxing out at 2000 N when a 60 g additional weight is added to the hammer for use with plastic and metal tips. Furthermore, when it comes to vibration tests, the achievable force range for the rubber soft tip falls short, ranging from 80 to 280 N, which may prove insufficient for certain testing scenarios. NV Tech Design GmbH introduced the scalable automatic modal hammer (SAM) highlighted in [20]. The primary aim of the SAM hammer is to characterize the dynamic properties of nonlinear structures via iterative experimental modal analysis, displaying consistency in terms of force amplitude. The SAM series encompasses two distinct models: SAM1, designed for a narrow amplitude force range of 5–100 N, and SAM2, for more substantial amplitudes ranging from 80 to 2000 N. Although SAM exhibits remarkable precision in repeatability and reproducibility, it notably lacks a force search feature, necessitating a trial-and-error approach to attain the desired force level. Another significant drawback of these commercially available automatic modal hammers is their substantial cost.

In our previous work [21], we proposed a cost-effective solution to address the expense associated with commercially available systems. By utilizing Arduino technology and a stepper motor, our alternative system achieves a wide force range from 3 to 2500 N at a total cost of approximately $250, excluding the modal hammer. This cost consists of $150 for the motor and driver from eBay, $50 for manufacturing the mounting components for the hammer, and $50

for an Arduino UNO. The resulting system exhibits exceptional repeatability as quantified by a maximum error of 4% from nominal and mean force values. Nonetheless, this system's major limitation is the absence of a direct correlation between the excitation force and motor parameters. This lack prevents the user from targeting specific force values directly and, instead, requires a trial-and-error approach that proves both time-intensive and potentially impractical.

To confront this challenge, this research focuses on devising a cost-effective system that empowers users to input a desired force and reliably impact the structure with an error below 5%. To address this issue, we explore the potential of Scikit-Learn regression models [22]. In "The Proposed System and Methodology", we discuss the system components, measurement, data preprocessing, and the methodology for iterating to the desired force value. In "Experimental Results", we present the experimental results of the proposed automatic modal hammer using a linear two-story tower and a model airplane wing with a nonlinear vibration absorber. "Concluding Remarks" delivers concluding remarks.

## The Proposed System and Methodology

### System Setup and Components

In this section, we thoroughly explore the integral components of the system, comprising the Step Motor and its corresponding driver, the shaft mount, the microprocessor unit, and the measurement equipment. Furthermore, we delve into the specifics of the stepper motor driving method, elucidate the data collection process, and expound upon the proposed methodology. The cost of the SAMH system, excluding DAQ and modal hammer, is approximately $350. This cost consists of $140 for a Raspberry Pi 4B from Amazon, $150 for the motor and driver from eBay, $50 for manufacturing the mounting components for the hammer, and $1 for the serial communication module from Amazon. The software used for the SAMH includes MATLAB and Python in our implementation; however, the MATLAB codes could be replaced with Python versions. Lastly, we present a schematic representation of the entire system in the concluding subsection.

### Stepper Motor and Motor Driver

The proposed system employs a NEMA34 12-Nm stepper motor (model 86J18156EC-1000-LS-14) with a preinstalled encoder, as depicted in Fig. 1a. This motor, in conjunction with its closed-loop driver, is conveniently accessible on popular eCommerce platforms, such as Amazon where it was purchased for this research [23]. Stepper motors provide precise control over shaft position [24], which is crucial for accurately controlling the movement of the hammer during excitation. However, challenges arise in open-loop configurations due to issues like missed steps and stalling [25, 26]. In our case, when the motor is subjected to sudden loads from hammer impacts, missed stepping and shaft misalignment are prone to happen. To address these challenges, the stepper motor is enhanced with an encoder and a hybrid servo drive, depicted in Fig. 1b, creating a closed-loop control system. The encoder measures shaft position and compensates for miss stepping and misalignment, while the servo driver regulates velocity and acceleration. Moreover, the servo driver features the capability of operating the stepper motor in full-step, half-step, or micro-step mode. In our

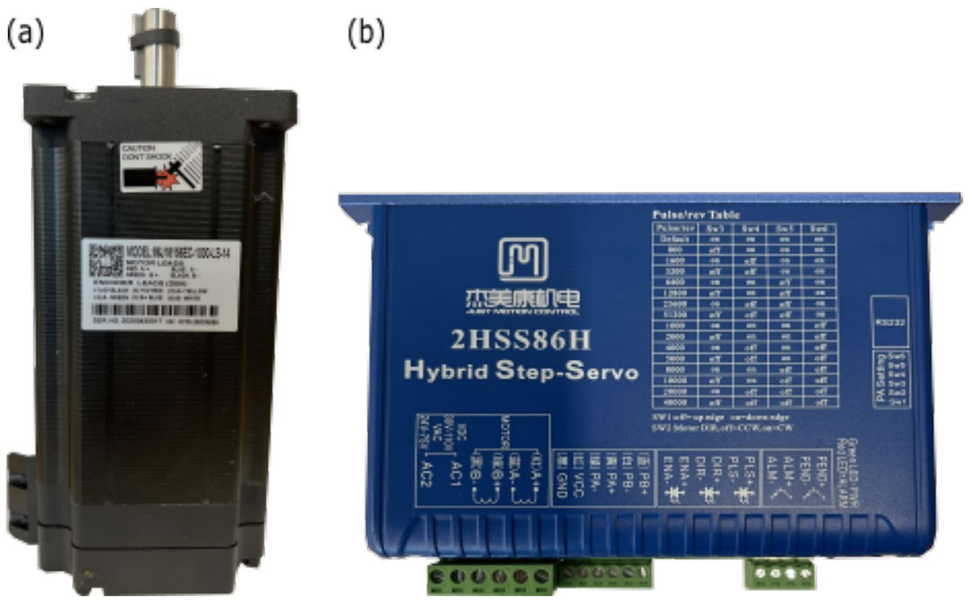


**Fig. 1** **a** The NEMA 34 stepper motor used in this research and **b** the hybrid step-servo driver used for controlling the motor

previous work [21], we established that segmenting a single step into five micro-steps (equivalent to 0.36° or 1000 steps/rev) provides an optimal balance between precision and torque for the application of an automatic modal hammer.

### The Shaft Mount

To connect the modal hammer to the motor, a custom mount was designed and manufactured, which is depicted in Fig. 2. The part attached to the motor shaft is called the mount base Fig. 2a, while the connecting part is called the mount top, Fig. 2b. The CAD models for the shaft mount were included in the supplementary materials of our previous work [21]. As the stepper motor used in this research differs from the one in our previous work, there was a need to modify the diameter of the mount base.

### Micro Processor, Circuit Connections, and Measurements

The motion of the motor is programmed and controlled using a Raspberry Pi 4 (shown in Fig. 3a), which enables the control of the amplitude of the impact exerted by the hammer. In our research, we used the GPIO pins of the Raspberry Pi to interface with the motor driver unit, which enables precise control of motor motion to achieve smooth and reliable strikes on the specimen. Additionally, we establish a serial communication link between the Raspberry Pi and MATLAB running on a laptop using the 10Gtek 3.3 V 5.5 V FT232RL Mini USB to TTL Serial Converter Adapter Module (Fig. 3b). This versatile module serves as a reliable bridge between the USB port of a computer and a TTL serial interface, commonly employed with Arduino boards and other microcontrollers. It is powered by the dependable FT232RL chip, ensuring efficient bi-directional data transfer, and supports a voltage range of 3.3–5.5 V. The use of the serial communication link enables us to call the motor program on the Raspberry Pi using MATLAB, such that MATLAB can be used to close the loop between motor motion and the measurement of the resulting impact force. The force exerted on the structure and the resulting response is measured using a National Instruments (NI) CompactDAQ (CDAQ) system with an NI 9215 module shown in Fig. 3c and d, respectively. The measurements are performed at a sampling rate of 128,000 Hz to ensure accurate capture of the applied impact force; however, the sampling rate can be changed based on the desired level of accuracy. The duration of the measurement is determined based on the response of the structure and varies for each system tested in this research. The modal hammer used in this research is a PCB Piezotronics modal hammer (model 086C03).

## Stepper Motor Driving Method

One of the most common operations of a stepper motor is to move a specified relative distance. This move is typically defined by a specific number of motor steps in either the clockwise or counterclockwise direction. To execute this operation efficiently, stepper motors typically follow a trapezoidal velocity versus time profile. The shape of this profile depends on factors such as the number of steps and the desired acceleration, deceleration, and peak speed [27]. To achieve acceleration in a stepper motor, it is essential to periodically modify the current speed from the initial value to reach the desired target speed. In the case of a stepper motor, each pulse from the pulse-width modulation (PWM) corresponds to one step. It is important to note that without

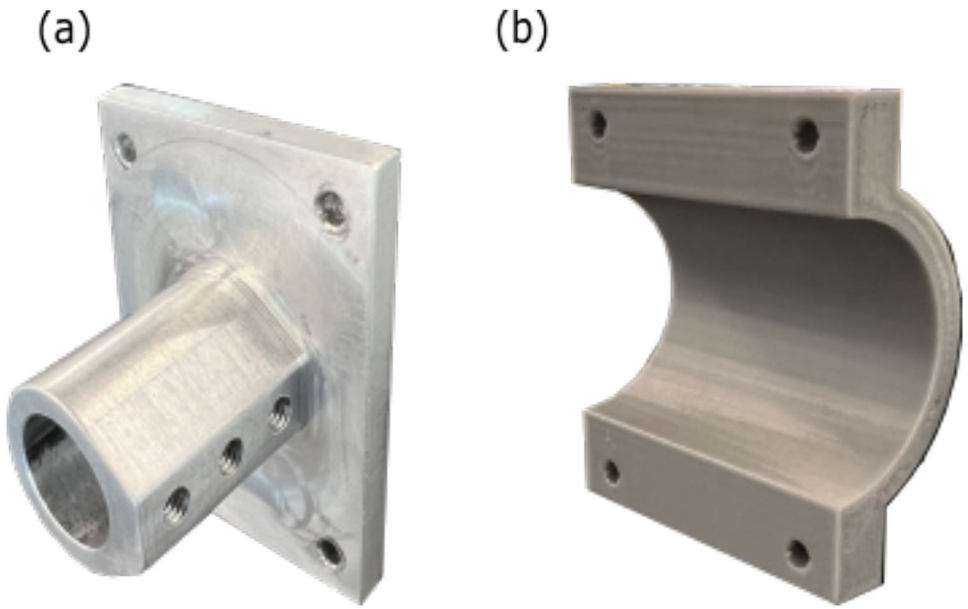


**Fig. 2** **a** The bottom half of the mount that connects to the motor shaft and **b** the top half of the mount that locks the hammer in place

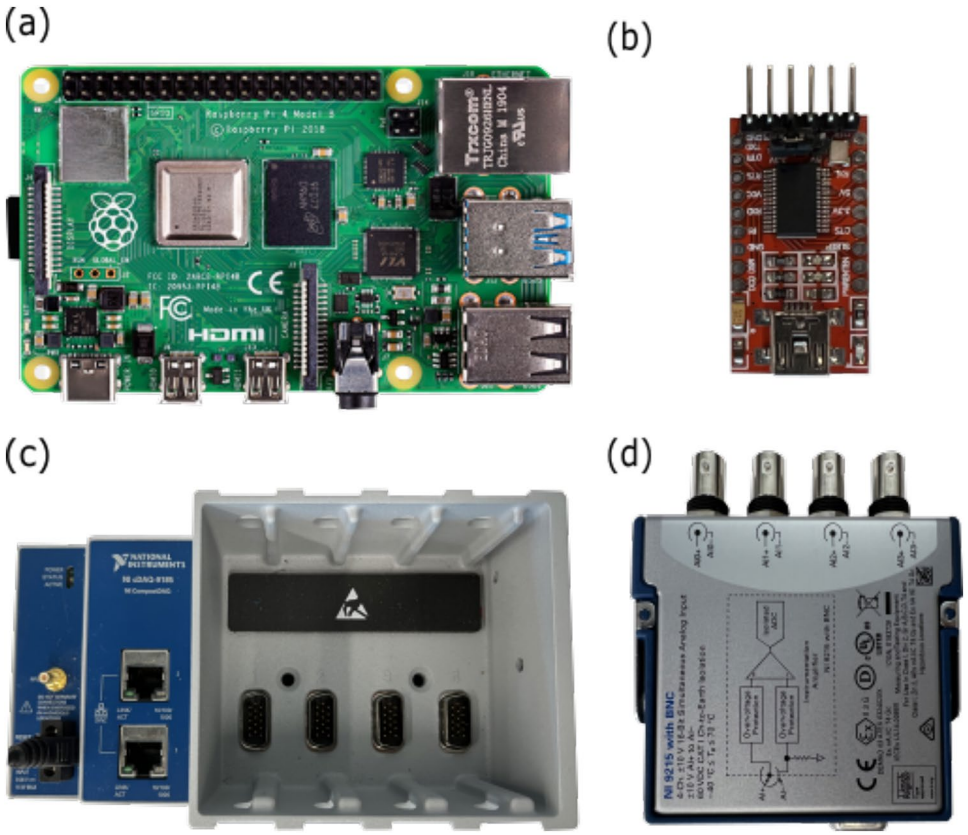


**Fig. 3** **a** The Raspberry Pi4 used for sending signals to the motor driver and **b** the serial adapter module used for serial communication between the Raspberry Pi and MATLAB on a separate computer. **c** The NI CDAQ chassis (model 9185) and **d** the NI 9215 module used to measure the hammer and accelerometer responses

adjusting the frequency of the PWM pulses, the stepper motor will move at a constant speed. However, by changing the frequency of the pulses per step, the stepper motor can accelerate. Specifically, a higher pulse frequency results in a higher acceleration rate [28].

The AccelStepper [29] library has become a popular choice for smooth and precise control of stepper motors. Specifically designed for Arduino and compatible microcontrollers, this software library can control stepper motors with acceleration and deceleration capabilities. It offers a user-friendly interface that facilitates control over various aspects of stepper motor movement, including speed, direction, and acceleration/deceleration profiles. However, as the AccelStepper library is not available for Python users, we developed a Python alternative, pyVelStepper, to achieve similar capabilities for controlling stepper motors with acceleration and deceleration. PyVelStepper implements acceleration and deceleration by changing the velocity per step. By specifying acceleration and deceleration rates alongside the number of steps, pyVelStepper incrementally changes the velocity per step to accelerate or decelerate the motor motion. Users can define different acceleration and deceleration rates, combined with the number of steps, to craft velocity profiles that align precisely with their needs. For scenarios necessitating constant velocity, users can set the max_velocity parameter to their desired speed and utilize the constant velocity function. It is important to note that the script was designed for use with a Raspberry Pi microprocessor. If users are utilizing a different microprocessor, they will need to adjust the pin initialization settings accordingly. In our application, the motor must change its direction of motion immediately after the impact and return to the initial position with deceleration. As a result, acceleration rate, deceleration rate, and number of steps remain the same. pyVelStepper can be found on GitHub [30].

## Dataset Preparation and Data Preprocessing

To address the absence of suitable datasets for our system, we took the initiative to create our own dataset for training a model. Based on the principles of impulse theory, we hypothesized that the force exerted on the structure is related to the hammer mass and velocity. Our modal hammer was equipped with two different tips, each varying in stiffness. In the initial stage, we focused on collecting data using the plastic tip, which possessed the highest stiffness. Additionally, we predetermined the number of steps the hammer would travel to strike the structure to be 200 steps.

To carry out the data collection process, we devised a loop on the Raspberry Pi. This loop enabled consecutive strikes on the structure, with each iteration adjusting the acceleration to achieve different velocities, and subsequently

different forces. Simultaneously, we developed MATLAB code running on a separate system to measure the force using the signal transmitted from the NI data acquisition system (DAQ). The communication between the Raspberry Pi and MATLAB was established through a UART (Universal Asynchronous Receiver/Transmitter) serial communication protocol facilitated by a serial module mentioned above. A signal was transmitted to the Raspberry Pi via the serial connection to initiate the loop, while another signal from the Raspberry Pi triggered the NI DAQ measurement. The measured force signal from the NI system was transferred to MATLAB, where it was further processed. Finally, MATLAB relayed the force magnitude back to the Raspberry Pi for storage, alongside the corresponding hit velocity, in an Excel file. The collected data is used for training models using Scikit-Learn regression models.

## Proposed Methodology: Model and Method of Successive Approximation

After collecting a dataset of 288 motor velocity-force pairs for the black tip and 249 pairs for the white tip using the two-story tower discussed in "Application to the Two-Story Tower", we analyzed and modeled the relationship between the motor velocity and the resulting impact force. The two datasets are presented in Fig. 4 as triangles for the white tip and squares for the black tip. The data pairs for the white tip are tightly clustered indicating a strong relationship between the impact force and the motor velocity. The tight clustering of the data for the white tip also illustrates the high repeatability of the system for the white tip. In contrast, the data for the black tip are only tightly clustered for forces below ~ 650 N, and above this force value, the data set becomes relatively sparse with a wide distribution of motor velocities for a given impact force. This is likely because the black tip is made from rubber and undergoes significant deformation during impact at high forces, unlike the white tip, which is made from hard plastic and does not deform significantly during impact.

Based on the observed trends in each dataset, we chose to model each relationship using polynomials instead of a more complicated model, such as a neural network. To this end, we employed a polynomial regression model using the Scikit-Learn polynomial feature and linear regression algorithm [22] to capture the underlying relation within each dataset. The optimal degree of the polynomial for each regression model was determined manually by assessing the performance of each model by evaluating two Scikit-Learn regression metrics: the R-squared value and the mean absolute error (MAE). These metrics provide insights into the accuracy and robustness of each model in predicting the motor velocity based on the applied force. Upon comparing the results, we determined that a second-degree polynomial and a fourth-degree polynomial regression yielded the best

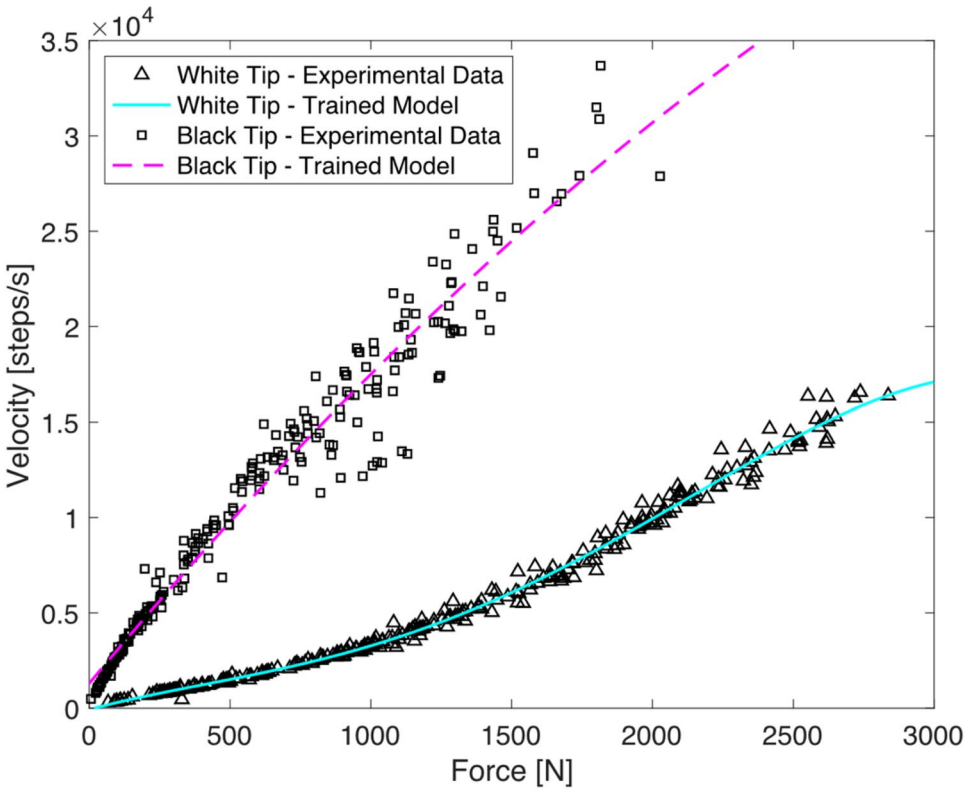


**Fig. 4** The force vs. velocity data distribution using the black tip

fit to the data with the least number of terms for the black and white tips, respectively. The strong performance of these models in accurately predicting hit velocity for a given impact force is demonstrated by their high R-squared (with 1 representing the perfect score) and low MAE values. Specifically, the identified model for the white tip produced an R-squared value of 0.992 and an MAE of 245.8, whereas the identified model for the black tip resulted in an R-squared value of 0.972 and an MAE of 916.0. Note that the higher MAE for the black tip is expected due to the sparser distribution observed in the data set (see Fig. 4). The resulting models are depicted in Fig. 4 as the solid line for the white tip and the dashed line for the black tip.

The trained models are implemented into the proposed system through the Raspberry Pi, such that an analyst inputs the desired force, and the model then translates that into the corresponding motor velocity needed to achieve the desired impact. Subsequently, the acceleration required for the motor to achieve the necessary velocity is computed for a fixed number of steps to the specimen. During the testing of the performance of the models (using the two-story tower—see "Application to the Two-Story Tower"), we observed that the error for some forces was above the desired error range of $\pm 5\%$. To address this issue, we implemented a successive approximation approach [31] to iterate until the measured force resides within the chosen error range. Specifically, for each force outside of the desired range, we compute the error percentage concerning the desired force and adjust the velocity by the same percentage until the measured force falls within the desired error range. Mathematically, this process is expressed as

$$v_{i+1} = v_i\left(1 + \frac{F_{Desired} - F_{Measured}}{F_{Desired}}\right), \tag{1}$$

where $i$ represents the iteration. Although this iterative approach is relatively simple, it enables the system to produce a force within the desired error bounds in six or less iterations as seen in the experimental results in "Experimental Results".

## Operation of the Proposed SAMH

This subsection discusses the operational aspects of the system. To minimize the number of programs and systems the user interacts with, the system is operated entirely using MATLAB, which is needed anyway to perform the measurements with the NI CDAQ. To this end, MATLAB is used to establish a direction SSH connection between the operating PC and the Raspberry Pi, such that MATLAB can trigger the operation of the motor script on the Raspberry Pi through the SSH connection. At the start of the MATLAB script, the user inputs the sampling rate, the number, type, and sensitivities of the measurement channels, and the desired accuracy for the applied force. Next, the code asks the user if system calibration is needed. The system is considered calibrated when the hammer is in contact with the structure without substantially deforming it. Further user input includes hammer tip choice, experiment count, and measurement duration. With these inputs set, the user specifies the desired force amplitude, and the system begins the task of exciting the structure.

The significance of serial communication is that it provides the ability to exchange data, commands, and control parameters between the Raspberry Pi and MATLAB. Specifically, the measured force and desired force are passed from MATLAB to the Raspberry Pi. The commands include the ones that the user sends through MATLAB to calibrate the hammer with the structure before testing begins. Finally, the control parameters are Boolean values that control the motor behavior during the test process, such as the timing between each hammer strike. The control parameters allow us to halt the motion of the hammer while the measurements are performed before applying the next strike. Furthermore, they can activate the measurements on MATLAB and terminate running scripts on both ends once the system converges. The establishment process involves matching the serial communication port and data exchange rate (Budd rate) so that both systems can communicate effectively.

The system is excited as follows. The motor velocity, and thus hammer velocity, progressively changes from an initial minimum to the predicted final impact velocity. This minimum velocity, which depends on the motor's specifications, is set at 100 steps/s in our system. The hammer accelerates and reaches this final velocity precisely at the last step before striking the target. During impact, the hammer's direction reverses to prevent a double strike and then decelerates to zero as it returns to the starting position. The severity of the hammer's rebound varies with the force of the impact; it is minimal at lower forces but can be quite pronounced under stronger impacts, potentially leading to motor shaft misalignment. The motor used in this research is equipped with a shaft encoder that allows the driver to detect and compensate for any misalignment. This ensures that the motor shaft remains accurately calibrated.

Based on the measured force, the system iterates on the inputs to the motor until the desired force level is achieved, then the measured force and acceleration responses are saved to a file specified by the user. Note that in this work, we set the maximum number of iterations to 20, but only six iterations were needed at most for the system to produce a force within the desired range. However, the user can define the maximum number of iterations to fit their needs. A flowchart of the system's components and how

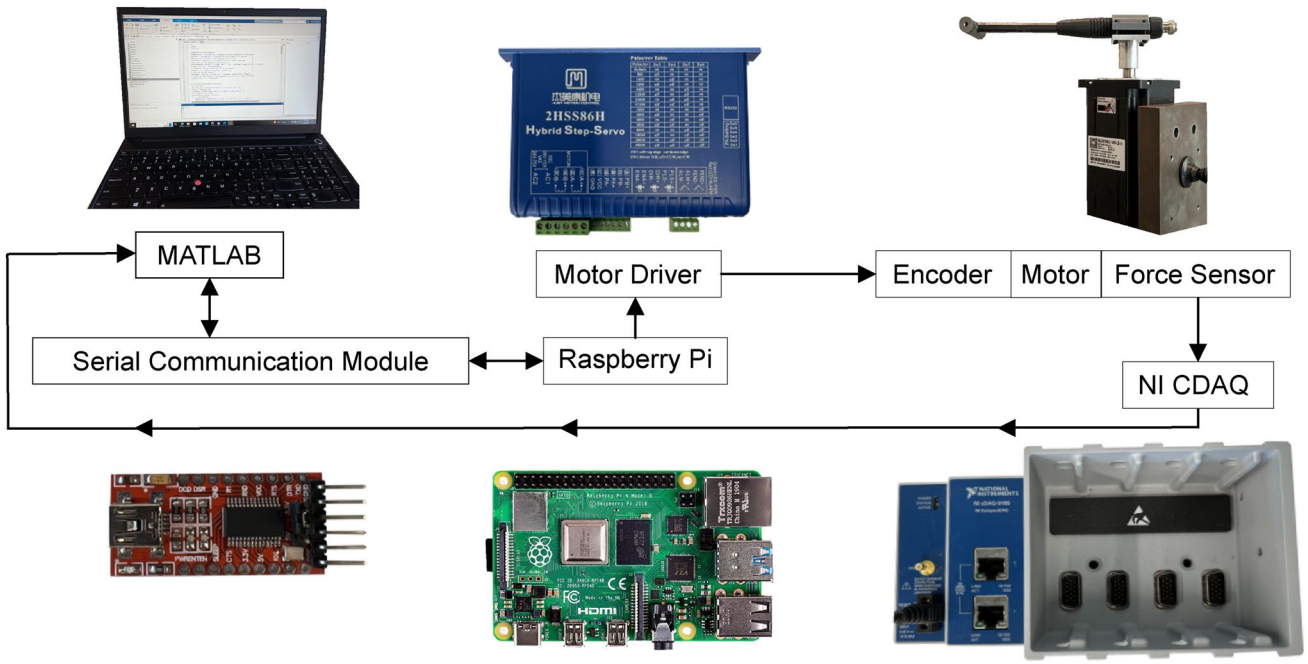


**Fig. 5** Flowchart representation of the proposed system

they connect is shown in Fig. 5. The full MATLAB and Raspberry Pi codes are available on GitHub [30].

## Experimental Results

In this section, we assess the performance of the proposed SAMH using two different test structures: a two-story tower and a model fighter jet wing with a nonlinear vibration absorber installed on the trailing edge of the wing tip. For all systems, the applied impact force and resulting accelerations were measured using the NI CDAQ system shown in Fig. 5 through MATLAB at a sampling rate of 128,000 Hz for a total duration of 10 s. Despite low-frequency vibration responses in both structures, a high sampling rate was selected to accurately capture peak force. Ensuring convergence requires precise peak force measurement. However, the sensitivity of the system to this parameter is discussed in "Sensitivity of SAMH to Error Bound and Sampling Rate". For all studies, the operational force range of the system was set to 60–3000 N for the white tip and 40–2500 N for the black tip.

The evaluation involves tracking strike counts to reach force targets, presenting the force sequence, and calculating the final error relative to the desired force. We also present measured responses from the two-story tower and the nonlinear vibration absorber installed on the model wing alongside the associated impulsive forces. The system's repeatability in terms of hitting the same force using the iterated motor parameters is also assessed using the two-story tower. Finally, in "Sensitivity of SAMH to Error Bound and Sampling Rate", we evaluate the system's sensitivity to the error bound chosen by the user and the sensitivity to the sampling rate. These results demonstrate the performance of the proposed SAMH system across a diverse range of systems and highlight its strengths and limitations.

### Application to the Two-Story Tower

We consider the behavior of the linear two-story tower in Fig. 6. We use this system to examine the number of iterations needed for the force to land within error bounds of 5%, the repeatability of the impacts using the iterated motor parameters, and the overall performance of the SAMH for both the white and black tips. We first consider the iteration process for eight difference force amplitudes ranging from 70 to 3000 N for the white tip in Table 1 and from 40 to 2400 N for the black tip in Table 2. These tables present the desired force, the number of hits needed for convergence, the final percent error, and the sequence of forces.

The results in Tables 1 and 2 show that the proposed SAMH system can converge to the desired force within six iterations at most and often requires only two iterations on average for the white tip and only 3 for the black tip. The number of iterations required by the SAMH is significantly less than the typical number required when the force is applied manually by the user. In general, it is difficult to quantify the number of manual hits necessary to achieve a

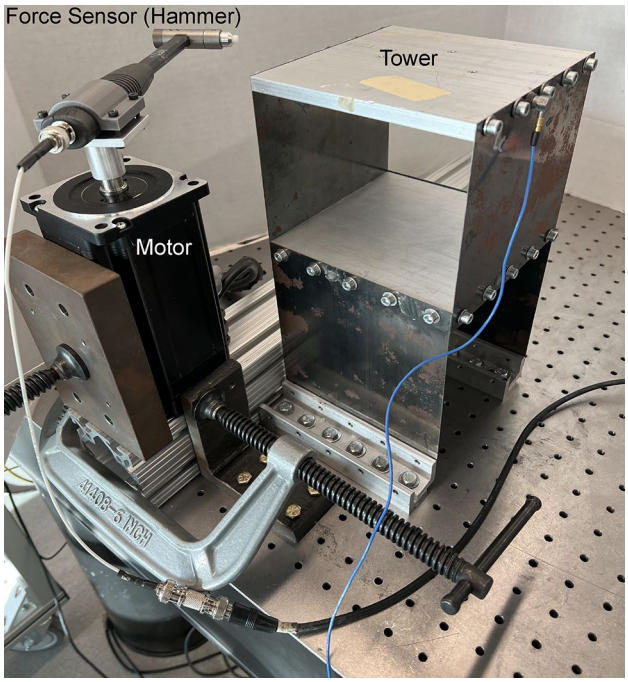


**Fig. 6** The experimental two-story tower with the AMH equipped with the white plastic tip

desired force within a 5% error bound because this depends on the user experience and their ability to iterate the hits. However, in the author's experience, the number of manual hits is typically at least 10 for an experienced user and substantially more for an inexperienced user. One application where the use of a manual impact hammer can prove especially tedious is when comparisons between the response of a structure under different configurations at the same loading amplitude need to be made. For example, [32] compares the free-response behavior of a nonlinear model aircraft in three different configurations, and 200 attempts were necessary to excite each configuration at the five different desired forces used for comparison. As such, even if the SAMH takes five iterations to achieve the desired force (Fig. 7), this represents a substantial improvement on the number needed if the force were applied manually. Furthermore, the proposed SAMH performs the iterations automatically without user input, such that the proposed SAMH not only reduces the time needed for testing but also frees up the user to work on something else in parallel. We note that a low number of iterations is expected for the tower because the models were identified using data captured on the tower. The other subsections in this section discuss the performance of the proposed SAMH when applied to systems not used in the training process.

**Table 1** Experimental results for the two-story tower for the white tip

| Desired force [N] | Number of hits | Final error [%] | Force sequence [N] |
|---|---|---|---|
| 70 | 5 | 2.8 | 47 → 50 → 60 → 64 → 68 |
| 400 | 2 | 3.5 | 315 → 414 |
| 800 | 1 | 4.6 | 763 |
| 1200 | 1 | 2.8 | 1166 |
| 1600 | 1 | 2.4 | 1561 |
| 2000 | 1 | 4.8 | 1904 |
| 2400 | 1 | 2.5 | 2339 |
| 3000 | 3 | 4.8 | 2723 → 2768 → 2876 |
| Average | 1.9 | 3.5 | – |

**Table 2** Experimental results for the two-story tower for the black rubber tip

| Desired force [N] | Number of hits | Final error [%] | Force sequence [N] |
|---|---|---|---|
| 40 | 4 | 5 | 52 → 32 → 43 → 38 |
| 50 | 3 | 0.4 | 57 → 47 → 52 |
| 400 | 2 | 0 | 348 → 400 |
| 800 | 3 | 1.5 | 738 → 845 → 812 |
| 1200 | 2 | 1.9 | 1092 → 1223 |
| 1600 | 2 | 1.2 | 1334 → 1580 |
| 2000 | 3 | 2.8 | 1421 → 1721 → 1945 |
| 2400 | 6 | 2.3 | 1621 → 1887 → 1925 → 1942 → 2193 → 2346 |
| Average | 3.1 | 2.3 | – |

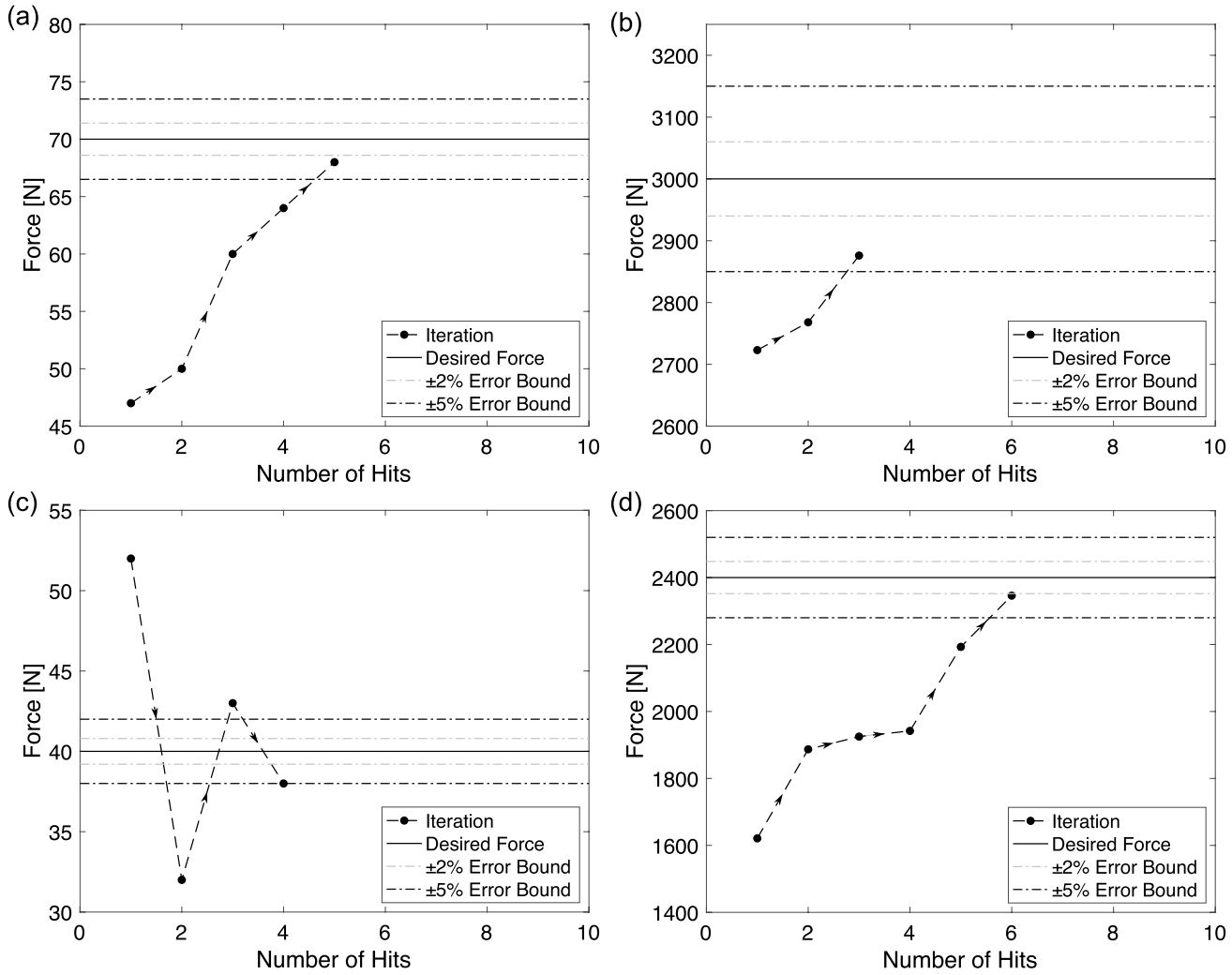


**Fig. 7** Example results of the iteration process for the linear two-story tower: white tip for desired forces of **a** 70 N and **b** 3000 N; and the black tip for desired forces of **c** 40 N and **d** 2400 N

To illustrate the resulting forces and response of the two-story tower, we depict the force and displacement time series along with the continuous wavelet transform (CWT) spectra of displacement for a desired force of 1200 N in Fig. 8a and b for the white tip and black tip, respectively. The force time series shows that only a single impact is achieved for both tips and that the peak force occurs at the same time due to the preciseness of the SAMH. The displacement time series were computed by numerically cumulatively integrating the acceleration twice: once to obtain the corresponding velocity and a second time to obtain the displacement. After computing the velocity, a high-pass, third-order Butterworth filter with a cutoff frequency of 4 Hz was applied to remove low-frequency errors introduced by the numerical integration. The filtered velocity signal was then integrated numerically to obtain the displacement, and the high-pass filter was applied again to remove low-frequency errors. The displacements show that the response is dominated by an oscillation at a single frequency, which correlates with the fact that the configuration of the tower causes the second floor to behave as a single-degree-of-freedom oscillator. The CWT spectra provide a time–frequency representation of the displacement signal content, where darker shading represents higher energy content at a particular time and frequency. Both CWT spectra shown have had their amplitudes normalized separately to range from 0 to 1, such that pure black shading represents an amplitude of 1. In both CWT spectra, a single band of frequency content appears, and the fact that this band is horizontal implies that the response is linear. That is, the response exhibits no dependence on time or, more accurately, no dependence on the instantaneous mechanical energy in the system that is observed for nonlinear systems [7, 33]

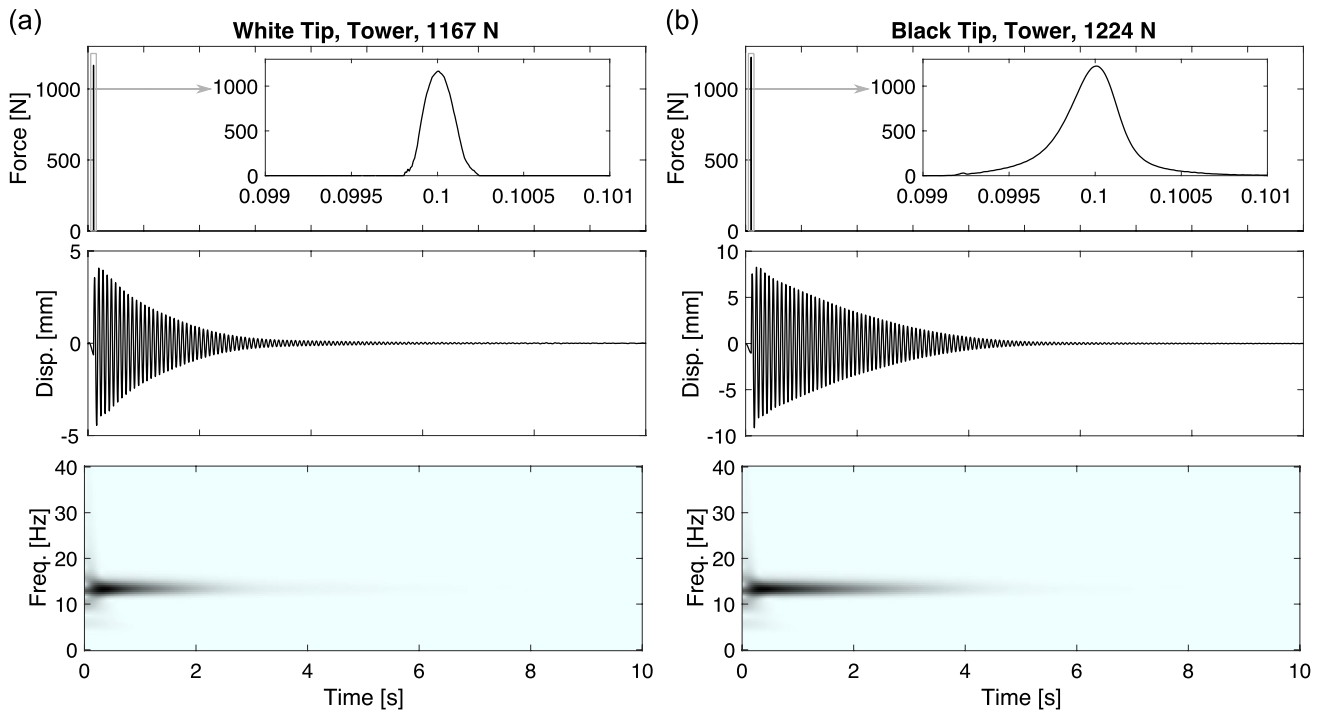


**Fig. 8** The final applied force and the displacement response and corresponding WT spectra of the tower for **a** the white tip and **b** the black tip

We note that the displacement response of the tower for the black tip is larger than that for the white tip despite the applied forces being within only 57 N (~5%) of each other. The reason for this is that the pulse produced by the black tip is wider than that produced by the white tip, such that the resulting work done on the tower for the black tip is higher than that for the white tip. The work done is computed by numerically cumulatively integrating the product of the measured force and the measured velocity (integrated from the acceleration as described in the previous paragraph). The work done by the white tip is 41.6 J, whereas the work done by the black tip is 187.5 J, which is just over 4.5 times more than the white tip. The difference in pulse width is due to the deformation that the black tip undergoes, which concentrates the energy that it imparts on the system in a narrower frequency range. Note that when the SAMH returns to its starting location, the motor typically overshoots and, as a result, the motor oscillates until it settles on the starting location. This introduces a small oscillatory force in the hammer measurement, which was removed manually before calculating the work done by each tip. We note that this oscillation does not affect the accuracy of the applied force because it occurs after the hammer has struck the system. Thus, no attempt was made to remove it; however, one could likely be implemented if desired. Lastly, we note that because the work done by the hammer can be computed using the experimental measurements directly, it may be possible to train a system to map work done to motor velocity, such that specific energy regimes could be excited instead of forcing regimes.

One question regarding the performance of the proposed SAMH is how repeatable the impacts are after the iteration process has achieved a force within the error bounds of the desired force. To answer this question, we performed a repeatability study using the final motor velocities that the system converged to for each desired force shown in Tables 1 and 2. The repeatability study was performed as follows: For each desired force, the system starts with the impact velocity (i.e., the velocity at the time of impact) predicted by the models. Next, the system iterates by adjusting impact velocity using Eq. 1 until convergence to the desired force within the desired accuracy is achieved. To evaluate the system's repeatability, we fixed the impact velocity to converged velocity, then impacted the structure 15 times using this setting.

We depict the results of the repeatability study for the white tip in Fig. 9 and for the black tip in Fig. 10, where the measured forces are shown as solid circles, the mean of the measured forces as a solid square, and the desired force as an asterisk on the number line. Additionally, we include the error ranges of the measured forces computed using the desired force. For the plastic tip, we present clusters of impacts centered around force levels of 60 N, 800 N, 1600 N, 2000 N, and 2400 N, each comprising 15 individual strikes. The results show that the system is consistently repeatable up to forces of 2000 N, after which we observed a notable

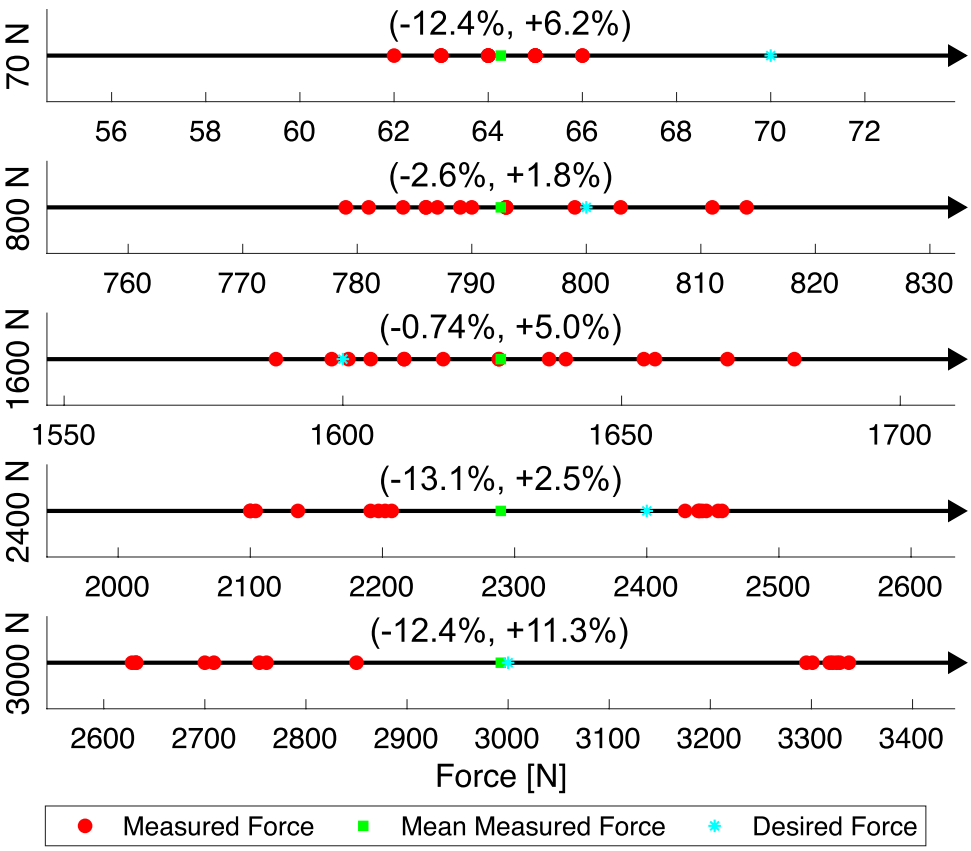


**Fig. 9** Repeatability analysis for the white tip performed on the two-story tower

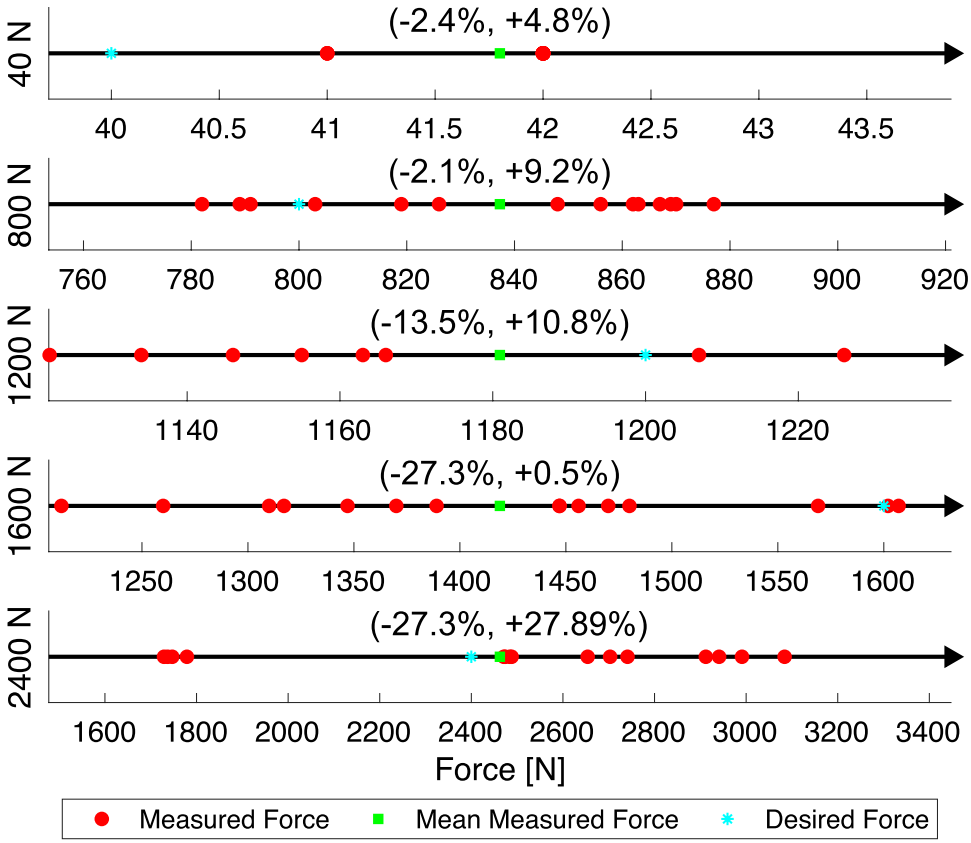


**Fig. 10** Repeatability analysis for the black tip performed on the two-story tower

increase in repeatability error (as seen in the results for 2400 and 3000 N). For the black tip, we conducted impacts at force levels around 40 N, 800 N, 1200 N, 1600 N, 2000 N, and 2400 N. The results show that the system is not as consistent as for the plastic tip; however, substantial deviations were only observed for forces above 1600 N. This behavior is explained in part because the softness of the black tip causes it to deform substantially at high impacts, whereas the plastic tip does not. Additionally, a substantial acceleration is needed to achieve such high forces with the black tip, and this overwhelms the motor driver used to control the motor's motion.

The results of the repeatability study reveal that it is better to repeat the iteration process for higher forces rather than reusing the motor parameters identified from a single iteration process. For low-amplitude forces, the same motor parameters can be used as only a small deviation is observed. However, for consistent accuracy, the iteration process should be employed with the previously identified motor parameters as initial conditions. Thus, to overcome the issues with repeatability, the user should run the iteration process each time using the previous motor parameters as starting conditions for the iteration. This approach effectively solves the repeatability concerns, making it possible to consistently achieve the desired force levels accurately.

## Application to a Model Fighter Jet Wing

To evaluate models' robustness and their ability to generalize over unseen structures we now consider the dynamics of a model fighter-jet wing featuring a nonlinear vibration absorber installed at the wing tip at the trailing edge. This configuration has been previously explored in [34]. A comprehensive description of the design and construction of the experimental specimen is documented in [35, 36]. The wing is designed to mimic the geometry of a General Dynamics F-16 Fighting Falcon fighter jet, an aircraft developed for the United States Air Force. The model wing itself is crafted from a flat steel plate with a thickness of 0.0046 m, a root length of 0.581 m, a span of 0.451 m, and a tip length measuring 0.229 m. The wing is bolted to a steel L-bracket using 3/8″-20 UNC bolts. The bracket is bolted on top of an optical table using 1/4″-20 UNC bolts. The wing's total mass is 6.7 kg, equating to a density of 7804 kg/m3. The technical drawings can be found in our previous work supplementary materials [21]. The nonlinear vibration absorber is built using a T-shaped steel mass connected to an anchor on the wing through a thin steel flexure, which acts as a linear spring and damper. The anchor is directly bolted to the wing and acts as an extension to the wing. In addition to the thin steel flexure, two thick steel flexures are attached to the anchor but not the absorber. Instead, the thick steel flexures are spaced away from the absorber, such that the absorber must displace a specific distance (called the clearance) to engage them. This type of nonlinear is a piecewise linear system and is commonly called clearance nonlinearity or free-play nonlinearity. Figure 11 presents pictures of the experimental setup including the wing and SAMH, and two views of the vibration absorber.

The model fighter jet wing determines how the SAMH performs for a structure not used to train or create the underlying models. To this end, we considered the performance of

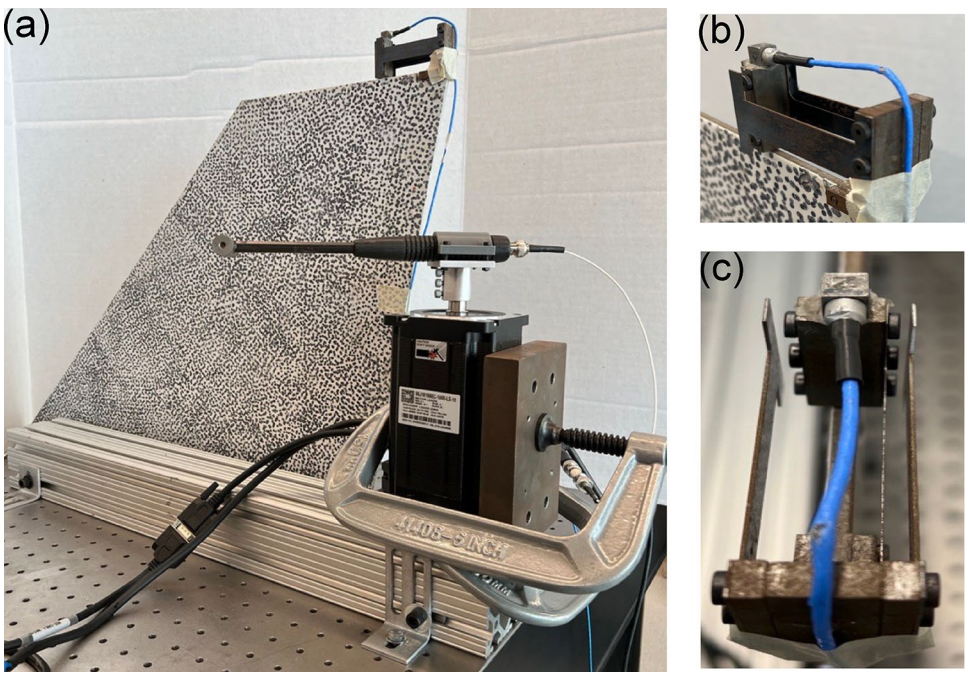


**Fig. 11** **a** The model fighter jet wing equipped with a nonlinear vibration absorber. **b** Close-up view of the nonlinear vibration absorber. **c** Zoomed-in view of the nonlinear vibration absorber showing the clearance nonlinearity

**Table 3** The experimental results for the model airplane wing and the black rubber tip

| Desired force [N] | Number of hits | Final error [%] | Force sequence [N] |
|---|---|---|---|
| 40 | 3 | 0.0 | 57 → 26 → 40 |
| 400 | 2 | 3.7 | 378 → 415 |
| 800 | 2 | 4 | 739 → 832 |
| 1200 | 3 | 4 | 992 → 1132 → 1250 |
| 1600 | 4 | 1.9 | 1255 → 1409 → 1469 → 1569 |
| 2000 | 6 | 2.4 | 1331 → 1685 → 1555 → 1613 → 1733 → 1951 |
| 2400 | 7 | 3.4 | 1600 → 1637 → 2234 → 2822 → 1902 → 2702 → 2483 |
| 3000 | 5 | 0.1 | 1386 → 1588 → 2473 → 3235 → 2995 |

**Table 4** The experimental results for the model airplane wing and the plastic tip

| Desired force [N] | Number of hits | Final error [%] | Force sequence [N] |
|---|---|---|---|
| 70 | 2 | 2.8 | 34 → 68 |
| 400 | 2 | 3.2 | 323 → 387 |
| 800 | 3 | 4.5 | 624 → 747 → 764 |
| 1200 | 4 | 1.5 | 965 → 1103 → 1134 → 1181 |
| 1600 | 5 | 1.6 | 1302 → 1507 → 1498 → 1509 → 1626 |
| 2000 | 4 | 0.9 | 1637 → 1777 → 1867 → 2019 |
| 2400 | 3 | 4.4 | 1939 → 2148 → 2293 |
| 3000 | 6 | 3.6 | 2273 → 2444 → 2711 → 2771 → 2846 → 2891 |

the SAMH system at the same force levels as those surveyed in the two-story tower experiments for both the white and black tips. We present the results of these experiments in Tables 3 and 4. Just as what was found for the two-story tower, the force sequences shown in Tables 3 and 4 reveal that the SAMH models typically underestimate the motor velocity needed to achieve the desired force – the only case where the model overpredicts the motor velocity is for desired forces of 40 N for the black tip. Across all forces, the average final error is only 2.43% for the black tip and only 2.81% for the white tip, which indicates that the SAMH can converge to the desired force within ~ 2% for both tips. While the operational force range for the black tip extends from 40 to 2500 N, SAMH demonstrates potential convergence even at higher forces, as illustrated in Table 3 for a force of 3000 N. To visually inspect the iteration process, we depict the iteration process for the lowest and highest forces for each tip in Fig. 12. Specifically, Fig. 12a and b present the iteration process for 70 N and 3000 N for the white tip, respectively, and Fig. 12c and d show the process for 40 N and 2400 N for the black tip, respectively. For the white tip, the plots show how the model underpredicts the motor velocity and how the iterative process marches forward to the desired force. For the black tip, the plots show that the model either under or overpredicts the velocity initially and how the iteration process ends up oscillating until it converges to the desired force. The fact that oscillations occur for the black tip implies that the system is approaching the stability limits as observed for the two-story tower.

We depict the force and displacement time series of the nonlinear vibration absorber along with the CWT spectra of displacement for a desired force of 1600 N in Fig. 13a and b for the white tip and black tip, respectively. The force time series shows that only a single impact is achieved for both tips and that the peak force occurs at the same time due to the preciseness of the SAMH. The same integration and filtering technique used to compute the displacements for the two-story tower is applied here to compute the displacement of the nonlinear vibration absorber. The displacements and the CWT spectra show that the responses can be separated into regions of high-frequency and low-frequency content. The high-frequency regions correspond to times when the displacement of the absorber is large enough to engage the thick steel flexures on the outside, whereas the low-frequency regions correspond to times when the thin steel flexure is engaged by the absorber. The CWT spectra capture this behavior and indicate that the system is strongly nonlinear due to the large changes in frequency content over time.

## Sensitivity of SAMH to Error Bound and Sampling Rate

In this section, we investigate the sensitivity of the SAMH and the iteration process to acceptable error and sampling

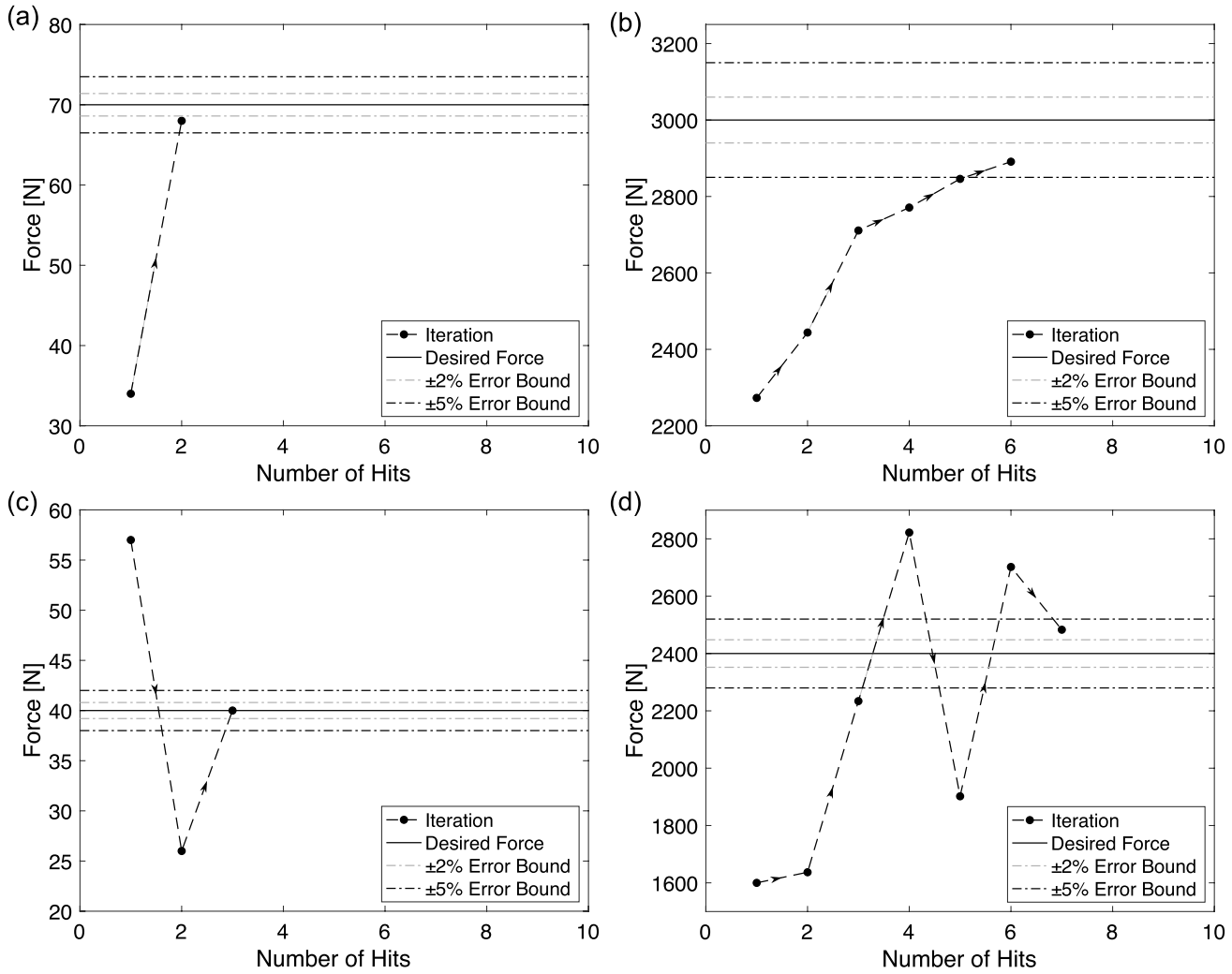


**Fig. 12** Example results of the iteration process for the model fighter jet wing with nonlinear vibration absorber: white tip for desired forces of **a** 70 N and **b** 3000 N; and the black tip for desired forces of **c** 40 N and **d** 2400 N

rate specified by the user. For all previous results, the acceptable percentage error is chosen by the user, and the SAMH system halts the iteration process whenever the error between the desired force and the measured force falls below that value. Note that all previous measurements were performed using an acceptable error percentage of 5%. The model fighter jet wing is used to test the sensitivity of the SAMH system for acceptable percentage errors of 0.5%, 1.5%, 2.5%, and 5% for forces of 70, 800, 1600, and 3000 N for the white tip and forces of 40, 800, 1600, and 2400 N for the black tip. The results are presented in Fig. 14 a and b for the white and black tips, respectively, where the dots represent the actual data collected in the experiments and the surface is a thin-plate spline fit to the data to show trends. In general, the number of hits increases as the acceptable percentage error approaches zero. However, there are some cases, such as 800 N for the white tip and 40 N for the black tip, where the number of hits remains low even for low percentage errors.

In some of the testing, the SAMH system was unable to achieve the desired force within 20 hits in one round of testing but was able to converge in a second round of testing. One case where this occurred is for 70 N with an acceptable percent error of 0.5% for the white tip. To illustrate this effect, we tasked the SAMH system with realizing a force of 70 N with an acceptable percent error of 0.1% and present the results of the process in Fig. 15a. In total, 36 iterations were required to achieve a force amplitude within the specified acceptable percent error range of $\pm 0.1\%$. The results demonstrate that the system can achieve the desired force even with such an extremely small acceptable percent error range, but the oscillations indicate that the algorithm is not able to converge. This inherent uncertainty is influenced by

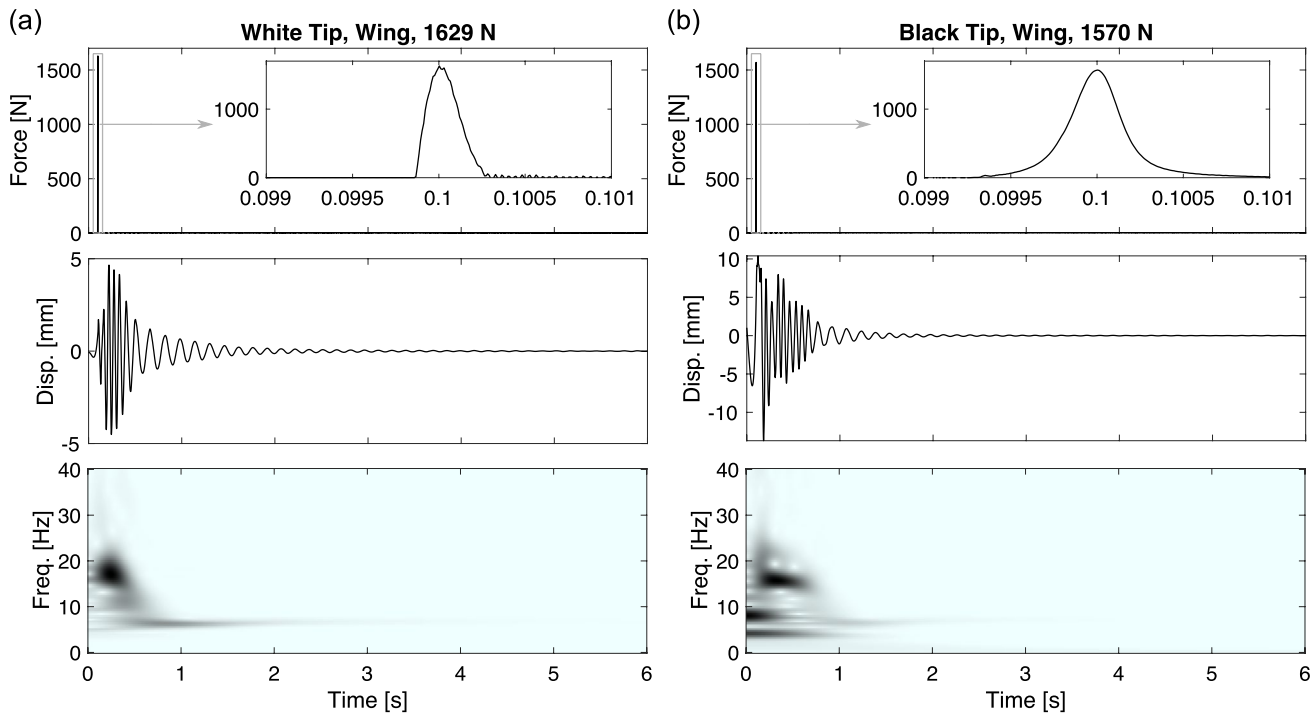


**Fig. 13** The final applied force and the displacement response and corresponding WT spectra of the tower for **a** the white tip and **b** the black tip

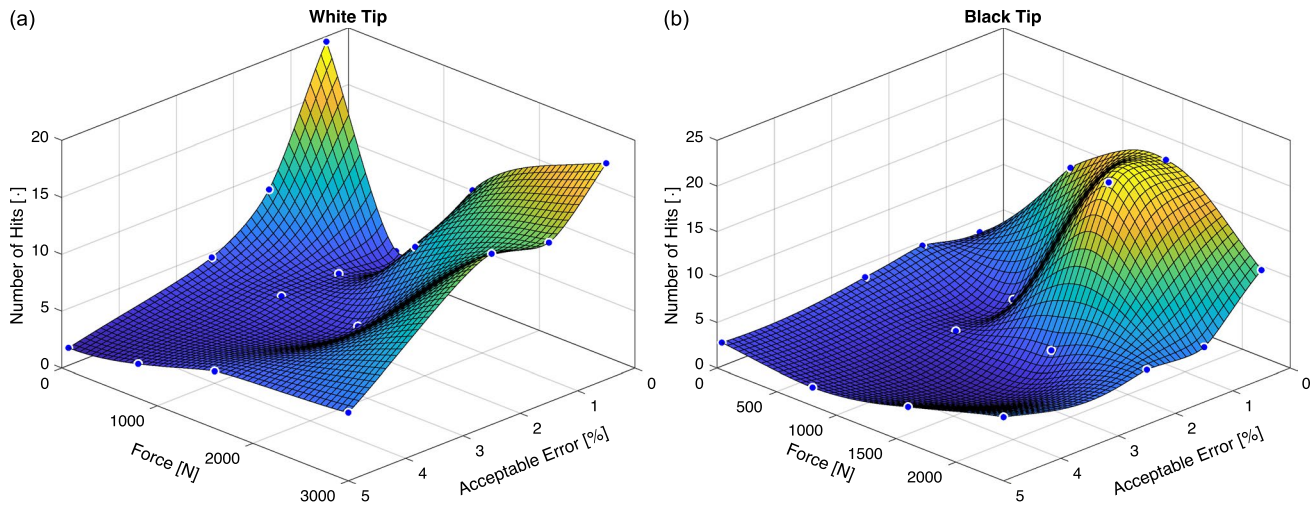


**Fig. 14** Sensitivity analysis on acceptable error for the **a** white plastic tip and **b** the black rubber tip

several factors, including the specific structure under examination and the initial calibration.

In addition to this, we also investigated the effect of a brand-new black tip compared to the black tip that was used for the model training process for a desired force of 40 N and an acceptable percent error range of $\pm 0.1\%$. The results of this comparison are presented in Fig. 15b, which shows that the SAMH converges much faster for the tip that was used for the training process than for the brand-new tip. For the new tip, the system overshoots the initial force, then undershoots before converging to the desired force after a total of 17 hits (11 are needed to reach the desired force within $\pm 5\%$). We note that this behavior was only observed for the black rubber tip and that the SAMH performed the same for used and new white plastic tips. Nevertheless, the results demonstrate that changing the hammer configuration also plays a significant role in the accuracy of the model and the iteration process. Understanding and accounting for

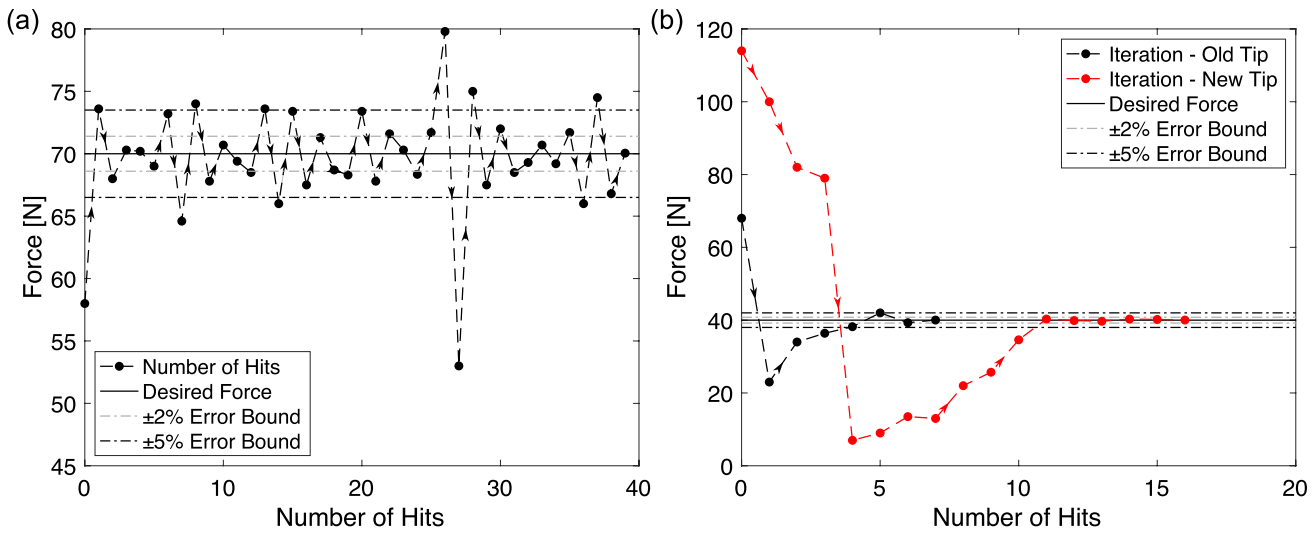


**Fig. 15** **a** An example of the iterative process for the model fighter jet wing for the white tip for a desired force of 70 N and an acceptable error of 0.1%. **b** A comparison of the iteration process for used and brand-new black tips for a desired force of 40 N and an acceptable error of 0.1%

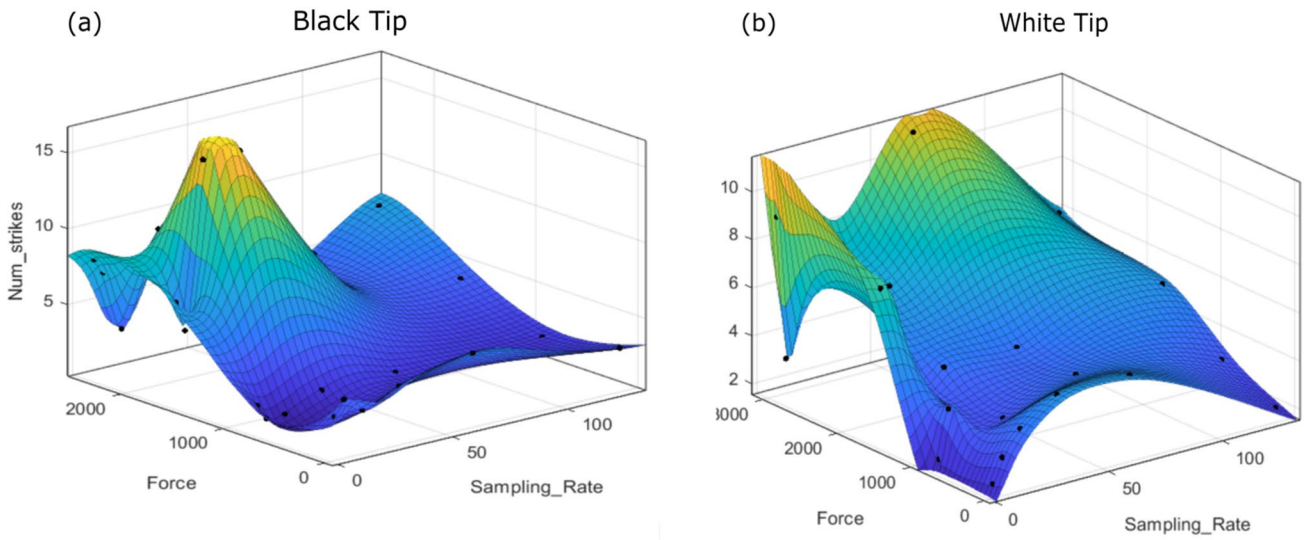


**Fig. 16** Sensitivity analysis on sampling rate for the **a** black rubber tip and **b** white plastic tip

these variabilities is key to harnessing the capabilities of the system effectively. Thus, future work should investigate other algorithms for iterating the hammer to the desired force that can assimilate new data and delete old data to account for the aging of the hammer tips.

To investigate SAMH's sensitivity to sampling rate, we evaluated its performance on a model fighter jet wing, decreasing the rate from 128 to 4 kHz for both tips. The results shown in Fig. 16, indicate that while SAMH consistently converges to the desired force, the number of iterations required may slightly increase at lower sampling rates. This increase is expected due to the reduced accuracy in signal reconstruction and force measurements at lower sampling rates.

## Concluding Remarks

This research focused on developing an economically viable smart automatic modal hammer (SAMH) designed to empower users to specify their desired force and consistently strike experimental structures while maintaining an error margin of less than 5%. The proposed system uses a stepper motor with an encoder and a motor driver that enables micro-stepping control facilitated by a Raspberry Pi 4. The

measurements were performed using an NI CDAQ system controlled with MATLAB. A serial communication link was established between the Raspberry Pi and MATLAB, such that the loop between the motor motion and the measurements can be closed. In addition, we developed a Python script, known as pyVelStepper, to control stepper motors with acceleration and deceleration capabilities. This script empowers Python users to control stepper motor motion with acceleration and deceleration. The fundamental operational principle of our system revolves around regression models trained on data collected using both the plastic and black tips on the two-story tower structure. These models establish an inherent connection between the impact force and the velocity of the hammer during impact. The SAMH uses the models to predict the motor velocity needed to achieve the desired force, then iterates on velocity until the measured force falls within the acceptable error range of the desired force. The operational range for the SAMH was shown to be 60 N to 3000 N for the plastic tip and 40 N to 2500 N for the black tip.

Our initial experimentation focused on a linear two-story tower, which was used to produce the training data for the models and served as a valuable reference point for evaluating the performance of the SAMH. The results demonstrated that the SAMH can consistently reach the desired force within a 5% error in less than five total hits on average. However, we found that the SAMH was only repeatable for a limited range of forces, such that the iteration process should be repeated at high force levels using the previously identified motor parameters as a starting point. To investigate the performance of the SAMH on a structure not used in the training, we applied it to excite a model fighter jet wing with a nonlinear vibration absorber installed on the wing tip at the trailing edge. The results of this application showed that the SAMH can accurately impact a structure within the desired error percentage across the same operational range as for the structure used in the training.

Lastly, we examined the performance of the SAMH with respect to the acceptable percent error. The results revealed that, on average, as the error percentage decreases the number of hits grows rapidly. This result is expected as, theoretically, the number of hits should grow to infinity at an acceptable percent error of 0%. However, the results also showed that sometimes the system could converge to the desired force for a low percent error faster than expected, such that there is some evidence that the SAMH itself is a chaotic system. The authors attribute the chaotic behavior observed in the system performance to the current operation of the motor. In pyVelStepper, we change the velocity of the motor per step leading to acceleration and deceleration. Generally, this is not an optimized approach to operate stepper motors with acceleration and deceleration. Particularly, the acceleration and deceleration should be calculated in time (steps/s$^2$) and synchronized with the clock timing of the microprocessor. It's believed that this operational inefficiency contributes to the repeatability issue discussed in the paper. Therefore, developing a deeper understanding of stepper motor control is crucial for addressing this challenge. In future studies, efforts will focus on mitigating this issue. A sensitivity analysis of the sampling rate revealed that SAMH convergence to the desired force remains unaffected by the changing sampling rate. However, the number of strikes may increase as the sampling rate decreases. This increase is expected because lower sampling rates compromise the force measurement accuracy.

The SAMH may also be limited in application to systems that do not return to their original starting position. For example, if a system is suspended by bungee cords, or slides around during the motion, then the rest position can change such that the hammer can no longer hit the structure. Specifically, the system cannot recalibrate itself if the target structure moves or does not return to the original starting position. Systems with bistable elements, such as snap-through bifurcations, may also suffer from this limitation. This limitation could be overcome by implementing a robotic arm with a computer vision system that adjusts the position of the hammer to account for changes in the equilibrium position. However, this dramatically increases the cost of the system by requiring a robotic arm and computer vision system. Nevertheless, the proposed SAMH system emerges as a cost-effective and precise automatic modal hammer. Its ability to consistently deliver accurate results equips researchers in experimental nonlinear dynamics with a valuable tool, streamlining their investigations and contributing to the advancement of this critical research domain.

**Funding** Funding was provided by Division of Civil, Mechanical and Manufacturing Innovation (Grant no. 2237715).

## Declarations

**Conflict of interest** On behalf of all authors, the corresponding author states that there is no conflict of interest.

## References

1. Silva JMM (1999) Modal analysis and testing. Springer, Dordrecht. https://doi.org/10.1007/978-94-011-4503-9
2. Ewins DJ (2000) Modal testing: theory, practice, and application. Research Studies Press, Taunton
3. Avitable P (2018) Modal testing: a practitioner's guide. Wiley, Hoboken
4. Sutton MA, Orteu JJ, Schreier H (2009) Image correlation for shape, motion and deformation measurements: basic concepts, theory and applications. Springer, New York
5. Nayfeh AH, Mook DT (1995) Nonlinear oscillations. Wiley, Hoboken

6. Vakakis AF (2018) Passive nonlinear targeted energy transfer. Philos Trans R Soc Lond Math Phys Eng Sci 376:20170132. https://doi.org/10.1098/rsta.2017.0132
7. Kerschen G, Peeters M, Golinval JC, Vakakis AF (2009) Nonlinear normal modes, part i: a useful framework for the structural dynamicist. Mech Syst Signal Process 23:170–194. https://doi.org/10.1016/j.ymssp.2008.04.002
8. Hill TL, Cammarano A, Neild SA (2017) Identifying the significance of nonlinear normal modes. Proc R Soc A. https://doi.org/10.1098/rspa.2016.0789
9. Maierhofer J, Mahmoudi AE, Rixen DJ (2020) Development of a low cost automatic modal hammer for applications in substructuring. In: Linderholt A, Allen MS, Mayes RL, Rixen D (eds) Dynamic substructures. Springer International Publishing, Cham, pp 77–86. https://doi.org/10.1007/978-3-030-12184-6_9
10. Redmon JW (1984) Impacting device for testing insulation, US4470293A. https://patents.google.com/patent/US4470293A/en. Accessed 18 Mar 2021
11. Evans HM (1985) Method and apparatus for the non-destructive testing of materials, US4519245A. https://patents.google.com/patent/US4519245A/en. Accessed 18 Mar 2021
12. Cawley P, Adams RD (1985) Testing of structures by impact, US4542639A. https://patents.google.com/patent/US4542639A/en. Accessed 18 Mar 2021
13. Sirica EG (1990) Impact calibration tool, US4967587A. https://patents.google.com/patent/US4967587A/en. Accessed 18 Mar 2021
14. Rioux D (2003) Impulse generator for profiling system, US20030174580A1. https://patents.google.com/patent/US20030174580A1/en. Accessed 18 Mar 2021
15. Norman PE, Jung G, Ratcliffe C, Crane R, Davis C (2012) Development of an automated impact hammer for modal analysis of structures. Air Vehicles Division, Defence Science and Technology Organisation Victoria (Australia). https://apps.dtic.mil/sti/citations/ADA560339. Accessed 17 Mar 2021)
16. Maierhofer J, Rixen DJ (2021) Development of an electrodynamic actuator for an automatic modal impulse hammer. In: Linderholt A, Allen M, D'Ambrogio W (eds) Dynamic substructures, vol 4. Springer International Publishing, Cham, pp 189–199. https://doi.org/10.1007/978-3-030-47630-4_18
17. Bediz B, Korkmaz E, Burak Ozdoganlar O (2014) An impact excitation system for repeatable, high-bandwidth modal testing of miniature structures. J Sound Vib 333:2743–2761. https://doi.org/10.1016/j.jsv.2014.02.022
18. Herfert D, Lemke A (2022) WaveHit: the first smart impulse hammer for fully automatic impact testing. In: Walber C, Stefanski M, Seidlitz S (eds) Sensors and instrumentation, aircraft/aerospace, energy harvesting & dynamic environments testing, vol 7. Springer International Publishing, Cham, pp 139–145
19. WaveHit—Home | The First Smart Impulse Hammer (n.d.). https://wave-hit.com/. Accessed 18 Mar 2021
20. Blaschke P, Schneider S, Kamenzky R, Alarcón DJ (2017) Non-linearity identification of composite materials by scalable impact modal testing. In: Wee Sit E, Walber C, Walter P, Seidlitz S (eds) Sensors and instrumentation, vol 5. Springer International Publishing, Cham, pp 7–14. https://doi.org/10.1007/978-3-319-54987-3_2
21. Singh A, Moore KJ (2022) An open-source, scalable, low-cost automatic modal hammer for studying nonlinear dynamical systems. Exp Tech 46:775–792. https://doi.org/10.1007/s40799-021-00516-7
22. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: machine learning in Python. J Mach Learn Res 12:2825–2830
23. StepMotor & Driver Aamazon Link (n.d.) https://www.amazon.com/Closed-Stepper-Driver-2HSS86H-86J18156EC-1000/dp/B0851NCR6N
24. Stepper motor, Wikipedia (2020) https://en.wikipedia.org/w/index.php?title=Stepper_motor&oldid=995319214. Accessed 23 Dec 2020
25. Moon S, Kim Y, Lim Y, Kim DH (2012) Missing step detection in a high speed micro stepping motor using current feedback. In: 12th International conference on control, automation and systems 2012
26. Stika IC, Kreindler L, Sarca A (2013) A robust method for stepper motor stall detection. In: 8th international symposium on advanced topics in electrical engineering (ATEE), 2013. https://doi.org/10.1109/ATEE.2013.6563391
27. StepMotor Operating Principals (n.d.) https://www.st.com/resource/en/application_note/an2044-operating-principals-for-practispin-stepper-motor-motion-control-stmicroelectronics.pdf
28. Applying acceleration and deceleration profile to bipolar steppermotors (n.d.) https://www.ti.com/lit/pdf/slyt482
29. AccelStepper: AccelStepper library for Arduino (n.d.) http://www.airspayce.com/mikem/arduino/AccelStepper/index.html. Accessed 18 Mar 2021
30. https://github.com/KeeganJMoore/SAMH_2024 (n.d.)
31. Kantorovitch L (1939) The method of successive approximation for functional equations. Acta Math 71:63–97. https://doi.org/10.1007/bf02547750
32. Brown J, Mustafa M, Moore KJ (n.d.) Vibration mitigation of a model aircraft with high-aspect-ratio wings using two-dimensional nonlinear vibration absorbers. Int J Non-Linear Mech (**submitted**)
33. Noël JP, Kerschen G (2017) Nonlinear system identification in structural dynamics: 10 more years of progress. Mech Syst Signal Process 83:2–35. https://doi.org/10.1016/j.ymssp.2016.07.020
34. Singh A, Moore KJ (2020) Characteristic nonlinear system identification of local attachments with clearance nonlinearities. Nonlinear Dyn 102:1667–1684. https://doi.org/10.1007/s11071-020-06004-8
35. Whitican SM, Copeland TJ (2016) Structural testbed design and testing with controlled nonlinearities. Sound Vib 50:6–10
36. Moore KJ, Mojahed A, Bergman LA, Vakakis AF (2019) Local nonlinear stores induce global dynamical effects in an experimental model plane. AIAA J 57:4953–4965. https://doi.org/10.2514/1.J058311