This full-text paper was peer-reviewed at the direction of IEEE Instrumentation and Measurement Society prior to the acceptance and publication.

A Novel Torque Sensing Approach to Eliminate Stiction in Haptic Devices with Hybrid Motor/Brake Actuation

Milan Djordjevic[1], Jacob Horne[1], Samuel Lovett[1], Colin Gallacher[2], Antoine Weill-Duflos[2], and Carlos Rossa[1]
[1]*Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada*
[2]*Haply Robotics, Montréal, QC, Canada*

***Abstract*—Hybrid haptic devices combining brakes and motors enable precise force rendering in applications spanning teleoperation to virtual reality simulation. Brakes can provide a strong resistance to motion, while motors can simulate interactions with elastic or animated elements. However, a common issue arises when the user pushes against an elastic element and reverses motion. Upon reversal, the actuator must apply a force in the direction of the user's motion. Since brakes can only oppose a force, they instead resist the user, creating an unwanted resistance known as stiction. Solutions to this problem often involve bulky rotary torque sensors or imprecise strain gauges to detect the user-applied force and deactivate the brake accordingly. However, these solutions introduce substantial inertia and friction to the actuator, are expensive, and not suitable for portable devices.**

This paper introduces a novel torque sensing approach that embeds a miniature 1-DOF *force sensor* in a rotary hybrid actuator without adding any inertia or friction. A new control algorithm is proposed to determine an optimal partition of a virtual environment torque between the brake and motor, and disengage the brake before stiction occurs. The proposed approach is validated through a series of experimental testing on a 1-DOF hybrid actuator prototype. The results show that proposed approach successfully eliminates stiction while preventing oscillations around zero velocity. Compared to conventional instrumentation methods, this implementation offers a more cost-effective, compact, and reliable alternative to the design and control of hybrid actuators.

I. Introduction

Haptic devices create cutaneous or kinaesthetic stimuli to emulate the sensation of interacting with virtual objects or teleoperated environments, ranging from vibrations in smartphones to multi-degree-of-freedom mechanical devices that apply forces to the user's hand. Examples include teleoperated robots, force-feedback joysticks, surgical trainers, haptic knobs, and steer by wire systems [1], [2].

In most applications, a haptic device either restrains the user's motion when interacting with an inert object, or restores energy when simulating elastic deformations. A commonly used performance benchmark is the device's dynamic range, that is, the ratio of the maximum to minimum impedance it can simulate [3]. On one end of the spectrum, the ideal device is able to simulate a rigid object (of infinite impedance), such as a virtual wall. In practice, the latter is approximated as a spring of some stiffness, and the desired actuator torque is made proportional to the actuator's displacement into the wall. On the other end, the haptic device must allow the user to move freely with minimal resistance. Several different actuation techniques have been proposed to maximize the range of impedance the device can emulate, including passive, active, or hybrid actuation, depending on the actuator that creates the force feedback [4].

Active devices use electric motors to dissipate energy and restore energy to the user. While easy to control, they suffer from an inherent limitation when simulating high impedance environments [5]. A high simulation impedance often translates into a high feedback control gain. Combined with imperfect discretization of position and time, high control gains lead to instability, limiting the maximum impedance the device can simulate. To mitigate this issue, a model that includes some energy dissipation is often placed between the actuator's controller and the virtual environment [6], however it can alter the user's perception of the simulation model.

Passive haptic devices use passive actuators, such as brakes or controllable dampers, to provide force feedback. Since these actuators can only provide a torque against the user's motion (i.e., they can only dissipate energy), passive actuators are inherently stable and can emulate environments of higher impedance than their active counterparts. Combined with a larger torque per volume ratio than electric motors, these characteristics make passive actuators an optimal candidate for a haptic device [7], [8]. However, brakes cannot actively aid the user in pushing in their intended direction of motion, limiting their use [9].

Hybrid haptic devices combining passive and active actuators can leverage the active actuator's ability to restore energy, and the stability and increased impedance of passive actuators. However, when a brake and a motor are connected in parallel and both actuators are engaged, the brake always opposes the motor's and user's torque. If the brake is commanded to apply a desired torque τ_d determined by a virtual environment, Karnopp's model gives the brake torque τ_b as [10]:

$$\tau_b = -\min(|\tau_e|, |\tau_d|) \begin{cases} \text{sign}(\dot{\theta}) & if \quad \dot{\theta} \neq 0 \\ \text{sign}(\tau_e) & if \quad \dot{\theta} = 0 \end{cases} \tag{1}$$

where $\dot{\theta}$ is the angular speed of the brake and τ_e is the sum of all external torques applied to it. Fig. 1 illustrates

the implications of this model in a hypothetical 2 degree-of-freedom motor/brake haptic device simulating a virtual wall. In (a) the user applies force f_h towards the wall. Upon reaching it in (b), the brake and the motor apply forces f_b and f_m to simulate contact with the wall. When the user reverses the direction of the applied force to move away from the wall in (c), the brake force changes direction as it opposes $f_h + f_m$, thereby preventing the user from exiting the wall. This unwanted resistance to motion felt by the user is often referred to as "stiction".

Several approaches have been proposed for partitioning a desired torque τ_d between the brake and motor and to avoid the problem above. The simplest approach is to engage both actuators only when the desired torque opposes the actuator's speed, that is $-\dot{\theta}\tau_d > 0$, and use the motor otherwise. In the first case, the controller can define a percentage of the desired torque to be transferred to each actuator as $\tau_m = c\tau_d$ and $\tau_b = (1-c)\tau_d$ where $0 \leq c \leq 1$ is a constant [11]. Alternatively, the stiffness of the virtual environment can be estimated based on the observed rate of change of the desired torque as a function of position. The motor and brake torques can then be determined such that the motor simulated stiffness does not exceed its stability limit, while the brake takes on the difference [12]. An issue with this approaches occurs when $\dot{\theta} = 0$. To determine if the user is reversing motion when the speed is zero, accordion to (1), the direction of τ_e must be measured. Otherwise, the user must apply a torque sufficiently large to overcome the brake's stiction torque before the actuator moves.

Solutions to detect the user's intent to reverse motion and measure τ_e at zero velocity often involve the use of strain gauges mounted on the actuator's shaft [13], [14], or a rotary torque sensor mounted in series between the user and the actuator [15], [16], [17]. Strain gauges are generally imprecise, have issues with longevity, and are prone to thermal expansion causing measurement errors [18]. Rotary torque sensors placed on the actuator's shaft are expensive and add significant inertia, volume, weight, and friction to the actuator.

Other solutions to eliminate stiction are mechanical in nature. An example is a brake connected to an overrunning clutch that only blocks motion in one direction, while allowing it to spin freely in the other [9]. When the user reverses motion, the brake's torque is not transmitted to the user. A complete actuator requires two brakes and two clutches, each adding inertia and unwanted backlash to the actuator. Another solution is to use a spring between the brake and motor to estimate the user's input torque based on the measured deformation of the spring [19]. This arrangement limits the maximum stiffness the brake can emulate to that of the spring, decreasing the actuator's dynamic range.

In this paper we introduce a new solution to measure the torque acting on the brake at zero velocity without adding compliance, unwanted inertia, or friction to the actuator. Unlike conventional approaches using expensive and bulky rotary torque sensors or unreliable strain gauges, we propose a two step solution that embeds a miniature 1-DOF *load cell/force sensor* in the actuator's housing, and a control algorithm to determine when the brake must be turned off while avoiding oscillations around zero velocity. The load cell adds no inertia or friction to the device, and is shown to be as reliable as a torque sensor. The accompanying control law determines when the brake or motor should be used to create a torque calculated by an unknown virtual environment model, while allowing the motor to provide additional torque should the brake reach saturation. To the best of our knowledge, no similar solution has been proposed for hybrid haptic devices, despite the simplicity of the proposed implementation.

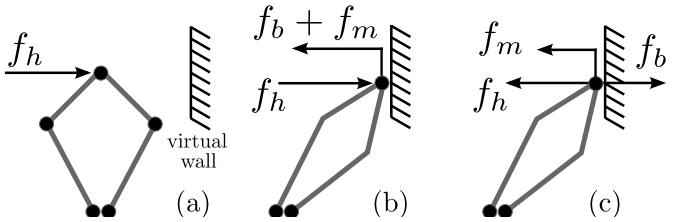

Fig. 1: Stiction effect when simulating a virtual wall with a motor/brake actuator. The user applies force f_h (a). Upon reaching the wall, the actuator applies f_a with both the brake and motor (b). When the user reverses the force to exit the wall, the brake force changes direction (c) opposing the user's and motor's torque.

This paper is organized as follows. Section II introduces the the proposed sensor design concept, and the associated control law that shares a desired torque between the brake and the motor. Section III details the built actuator prototype and the experimental results used to evaluate the design. The results confirm that the proposed actuator and control algorithm can effectively eliminate stiction. This is followed by a discussion of the results and limitations of the design in Section IV.

II. Sensor and Controller Design

The proposed system comprises two parts: the addition of a force sensor within the brake housing, and a control law that leverages the force sensor readings to partition the desired virtual environment torque between the brake and motor.

A. Brake Torque Sensing using a 1-DOF Force Sensor

The proposed design uses a 1-DOF force sensor to estimate the brake torque acting on a cylindrical, rotary brake. To illustrate the concept, picture the user holding the outer drum/housing of a rotary cylindrical brake in their hand. If an external torque τ_e is applied to the brake's shaft while the brake is turned off, and there is no off-state friction between the shaft and the housing, then no torque is transferred to the user as the shaft rotates freely. When the brake is commanded to provide a torque τ_d, a controllable amount of friction is added between the brake shaft and the outer drum. Some of the input torque is then transferred from the shaft to the housing, that is τ_d or τ_e, whichever is smaller. The torque is felt by the user holding the brake, as per the model given in (1).

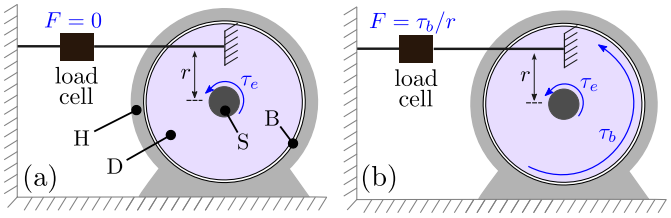

Fig. 2: The brake drum D is enclosed in a bearing B within the housing H. A load cell is mounted on the drum at a distance r from the shaft S subjected to an external torque τ_e. In (a) The brake is disengaged, no torque is transferred to the drum, and the load cell measures $F = 0$. In (b) the brake is engaged and the load cell reads $F = \tau_b/r$, where τ_b is the brake torque transferred to the drum, as given by (1).

Fig. 2 illustrates how this concept is implemented. The outer drum D of a rotary brake is connected to the actuator housing H through a bearing B, allowing the brake's outer drum to rotate. The outer drum is then connected to one end of a 1-DOF load cell, with the other end connected to a platform fixed to the housing, such that the drum is no longer free to rotate. When the brake is disengaged, its inner shaft S turns freely, as shown in 2 (a). Since the external torque τ_e acting on the shaft is not transferred to the drum, the measured force is $F = 0$ (assuming no off-state friction). When the brake is engaged, friction between the shaft and drum increases. This controllable coupling transfers some of the torque τ_e to the drum, as in Fig. 2(b). Therefore, the force sensor measures force $F = \tau_b/r$, where r is the distance between the mounting point of the load cell on the drum and the centre of the shaft. A schematic showing how this arrangement is implemented in practice can be seen in Fig. 4.

The force measured by a sensor mounted in this configuration is prone to noise from elastic deformations of the mounting structure, hysteresis, and unwanted backlash. In addition, a force offset due to pre-tension or pre-compression of the sensor may be present. Therefore, the control algorithm should rely on detecting significant *variations* of the applied torque, rather than on direct precise measurements of τ_b.

B. Control Law

In the control algorithm we assume that a virtual environment with an unknown simulation model calculates a desired torque τ_d to be created by the actuator as a function of its position θ and/or speed $\dot{\theta}$. When the user attempts to move the actuator against the direction of τ_d, both the brake or the motor may be used to create the torque. If the user applies a torque in the same direction as τ_d, the brake must be turned off as only the motor can provide energy to the user.

Let the power in the actuator be defined as a function of the desired virtual environment torque τ_d as:

$$P = -\tau_d \dot{\theta} \tag{2}$$

so that when $P > 0$, the actuator must dissipate energy, so that the user is moving against the desired torque. When $P < 0$, the actuator is restoring energy to the user, that is, the actuator moves in the direction of the virtual environment torque. The standard power flow algorithm delegates the desired torque to the brake if $P \geq 0$ and to the motor if $P < 0$ but reaches an indetermination when $\dot{\theta} = 0$ (and $P = 0$) [16].

The proposed control law must leverage the force reading to deal with the condition $P = 0$. For clarity, this will be presented in two steps: We first explain how the algorithm activates the brake or motor when $P \neq 0$ (Step 1), and then, a condition for $P = 0$ that uses the force readings is added to the controller (Step 2).

Step 1: The brake is primarily used to provide the desired torque when $P > 0$. If the desired torque exceeds the maximum (saturation) torque of the brake τ_{bm}, the motor compensates for the difference between the desired and maximum brake torque. The amount of torque the brake cannot generate when saturated is:

$$\tau_{bs} = \max\left(0, |\tau_d| - \tau_{bm}\right). \tag{3}$$

Therefore, the commanded brake torque τ_b and motor torque τ_m may be defined as:

$$\begin{bmatrix} \tau_b \\ \tau_m \end{bmatrix} = \begin{cases} \begin{bmatrix} \tau_d - \tau_{bs} \\ \{\tau_{bs} + \tau_b h(t)\} \operatorname{sign}(\tau_d) \end{bmatrix} & \text{if } P > 0 \\ \\ \begin{bmatrix} 0 \\ \tau_d \end{bmatrix} & \text{if } P < 0 \end{cases} \tag{4}$$

where

$$h(t) = e^{-t\beta} \tag{5}$$

approximates the transient, overdamped time response of the brake, t is the elapsed time after activation, and β is the inverse of the brake's time constant. While $P > 0$, the implications of this formulation are: **1)** If $\tau_{bs} = 0$ (the brake is not saturated), the desired torque is sent to the brake. At $t = 0$, $h(0) = 1$ since the brake's torque cannot immediately reach the desired torque due to its transient, the desired torque is initially taken on by the motor; **2)** As time progresses $h(t) \to 0$, therefore the motor torque decreases to zero while the brake torque increases at the same rate; **3)** When $\tau_{bs} > 0$ the motor takes on the difference between the desired and maximum brake toque, while also compensating for the response time of the brake; and **4)** By setting $\beta \to \infty$, the motor does not compensate for the brake's transient, which could be necessary to avoid instability in high stiffness environments. When $P < 0$ in (4), the brake is turned off and only the motor is activated.

Step 2: The condition above is insufficient when transitioning between positive and negative power ($P = 0$). Therefore, another condition must be imposed to detect the user's intention to reverse motion based on the force readings.

In normal operation, when the user intends to reverse motion, a sudden decrease in the measured force will occur. Therefore, the controller does not need to rely on accurate force measurements, but rather on variation of the force

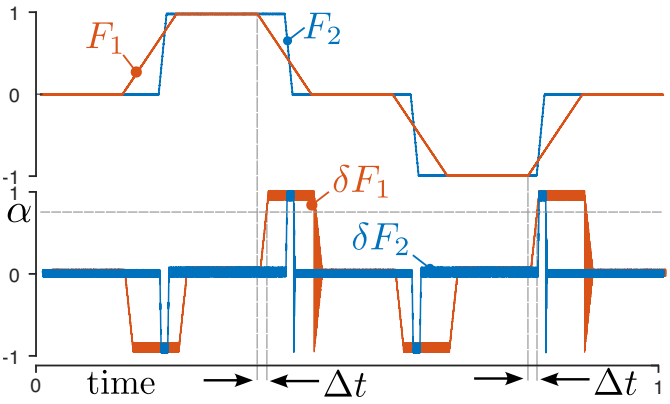

Fig. 3: Estimated δF for forces $F_1(t)$ and $F_2(t)$ with a 5% Gaussian noise. δF_1 and δF_2 are calculated with $t_0 = t_1$ and $t_0 = t_2$, as in (6), with $t_2 >> t_1$. The greater t_0, the smaller the signal to noise ratio of δF, and the greater the delay Δt between a decrease of $|F(t)|$ and the time δF crosses α.

magnitude when $\dot{\theta} = 0$. Let $k(t)$ be the accumulated change in the measured force within the past t_0 seconds, such that:

$$k(t) = \int_{t-t_0}^{t} \left(F(t) - F(t - \delta t)\right)\, dt \tag{6}$$

where δt is a delay term. We can now define a function δF that increases sharply when the magnitude of the measured force decreases within the time window $[t-t_0, t]$. While many formats can be used, a suitable candidate is:

$$\delta F = -k(t)\text{sign}(F(t)). \tag{7}$$

These two functions work similarly to the filtered derivative of $-F(t)$: It tends to zero when $F(t)$ is relatively constant, is negative when the force increases, and positive when the force decreases. The key feature is that the moving window integration averages noise to almost zero, and the rate at which δF increases is faster than the rate at which $F(t)$ decreases.

Fig. 3 shows the normalized δF for force inputs $F_1(t)$ and $F_2(t)$, to which a 5% Gaussian noise is added. The triangular waveform $F_1(t)$ represents the worst case scenario, where the user slowly decreases the magnitude of the force, before changing direction. In practice, this change is expected to occur much faster, closer to $F_2(t)$. The corresponding estimated δF_1 and δF_2 are shown, using $t_0 = t_1$ and $t_0 = t_2$, respectively in (6), with $t_2 >> t_1$. The result shows that increasing t_0 decreases noise in δF, but it increases the delay Δt between the point where the force $F(t)$ begins to decrease and when dF crosses a predefined threshold α.

Therefore, the underlining assumption is that the condition $\delta F > \alpha$ indicates the user has decreased the applied force sufficiently to indicate the intention of reversing motion. This condition can now be added to (4) to account for $P = 0$ as:

$$\begin{bmatrix} \tau_b \\ \tau_m \end{bmatrix} = \begin{cases} \begin{bmatrix} \tau_{vd} - \tau_{bs} \\ \{\tau_{bs} + \tau_b h(t)\}\,\text{sign}(\tau_d) \end{bmatrix} & \text{if } P \geq 0 \quad \textbf{and} \\ & \delta F < \alpha \\ \begin{bmatrix} 0 \\ \tau_d \end{bmatrix} & \text{if } P < 0 \quad \textbf{or} \\ & \delta F > \alpha \end{cases} \tag{8}$$

The added condition ensures that when the user decreases the applied torque while $P = 0$, the change in force is detected with the formulation above. The condition $\delta F < \alpha$ triggers the controller to disengage the brake when the uses decreases force before trying to exit the wall. With the brake off, the user is free to move in the direction of the desired torque. Once $\delta F < \alpha$, the condition $P < 0$ ensures that the controller does not switch the brake back on, preventing chattering.

III. Experimental Validation

To validate the proposed system, the 1-DOF hybrid actuator shown in Fig. 4 was built. The actuator uses a 24 V magnetic particle brake from Placid Industries with a maximum torque of 110 mNm and $\beta = 5$, and a 24 V DC motor with an experimentally determined torque constant of 34 mNm/A and peak torque of 150 mNm. An AMT 102 CUI 2422 quadrature encoder with 2048 pulses per revolution is attached to the actuator's shaft to measure the angular displacement.

A 1-DOF load cell (FUTEK LSB 201) connected to a Wheatstone bridge amplifier (Sparkfun SEN-13879) was used as the force sensor. The exploded CAD model of the actuator in Fig. 4 (bottom) shows how the load cell was integrated into the actuator following the principle described in Sec. II. The back of the brake's outer drum (shown in purple) is connected to a custom-made attachment (green) that slides into a bearing. The load cell (in red) is screwed on one end to the attachment part, and on the other end to the fixed actuation housing that holds the brake and bearing. The bearing reduces mechanical noise and friction between the brake and the housing, increasing the accuracy of the force sensor reading. Fig. 4 (bottom) also shows how the motor and brake shafts are attached via two couplers. The user interacts with the handle placed at the centre of the shaft.

The brake torque is controlled proportional to the applied voltage using a H-bridge amplifier. Magnetic hysteresis and off-state torque are neglected. The torque of the DC motor is controlled via a digital PID controller using current-sensing feedback. The control algorithm and data collection are done in Simulink Desktop Real Time with a Humusoft MF634 data acquisition card running at 1 kHz. At the beginning of each test, the force sensor readings are averaged over 2 seconds and the average is removed from subsequent readings to remove measurement off-set. The following experiments are conducted to evaluate the proposed design and control algorithm:

In **Experiment 1** the frequency response of the sensor is evaluated between 0.1 Hz to 200 Hz by applying a sinusoidal torque input with the motor, while measuring the force and

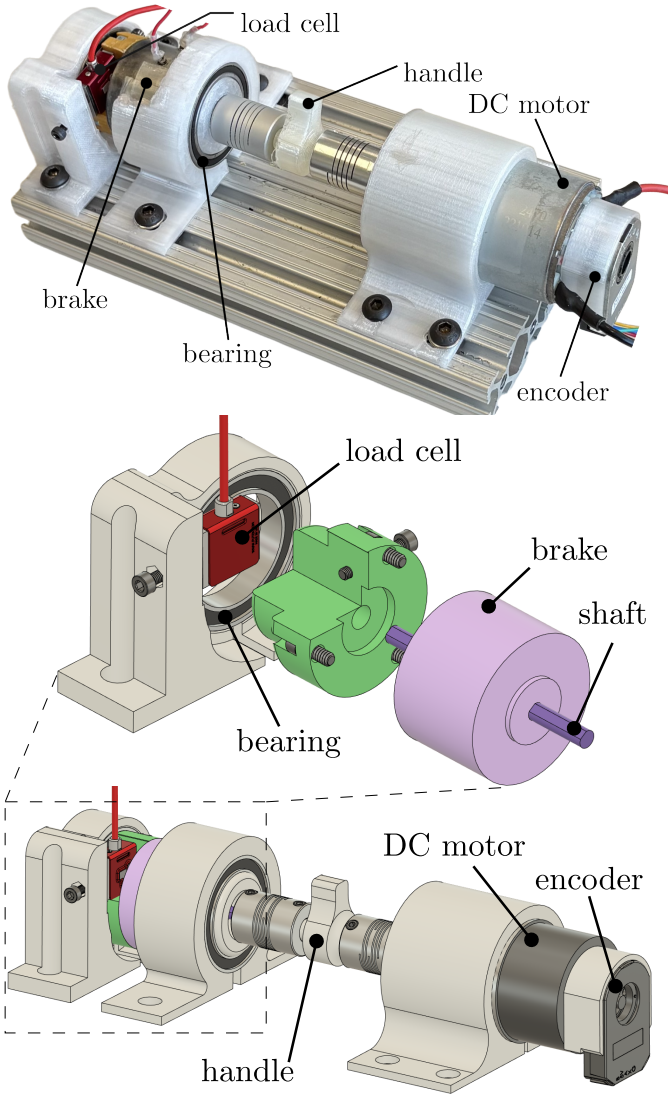

Fig. 4: Prototype of the hybrid actuator with a brake and motor connected in parallel (top) and exploded CAD model (bottom) showing how the load cell is attached to the brake drum. A rotary encoder is connected to the back of the motor.

keeping the brake on. The test indicates a cut-off frequency of 140 Hz, which includes the frequency response of the mechanical link between the sensor and the brake. This bandwidth is well above the required range for haptic interactions [20].

Experiment 2 evaluates the system's ability to detect a decrease in force and disengage the brake. The brake is turned on to its maximum torque, while the motor applies a square torque input. The observed δF, the time Δt the signal takes to cross the threshold α, along with the signal-to-noise ratio of δF are calculated. We set $\alpha = 0.9\max(\delta F)$, determined empirically to provide the best performance when $t_0 = 0.03$ sec. This test is repeated 32 times for different values of t_0, ranging from 10 ms to 350 ms. The results, seen in Fig. 5, confirm the trend observed when simulating (7). A large t_0

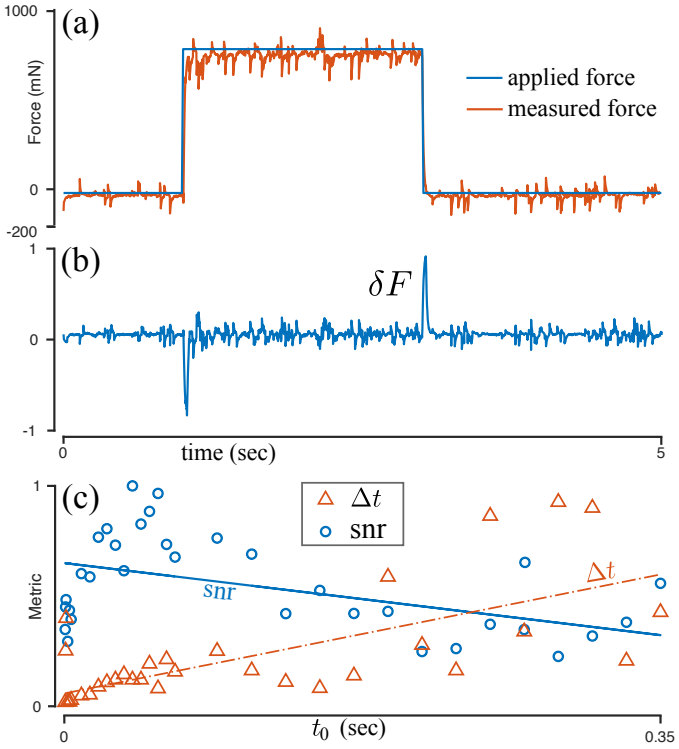

Fig. 5: Load cell response to a step torque input (a), and the normalized δF for $t_0 = 30$ ms in (b). In (c) the measured signal to noise ratio of δF and delay Δt to cross the threshold $\alpha = -0.9\max(\delta F)$ (for $t_0 = 0.03$) for 32 window lengths t_0 in (6). The lines represent the first order model fit.

improves the signal to noise ratio of δF, but increases the time it takes to cross the threshold α.

Experiment 3 simulates a virtual wall. The user moves the actuator's handle into the wall before reversing motion. The results for two stiffnesses of 400 mNm/rad and 800 mNm/rad are shown in Fig. 6 (a-b) and (c-d), respectively. As it can be seen, the algorithm creates the required τ_d and disengages the brake when the user reduces the applied torque before reversing motion. Fig. (c) and (d) show that if the condition $\delta F > \alpha$ is triggered, it prompts the controller to turn off the brake. In both simulations, the brake is turned off before a change in position is detected and before the input force changes sign, indicating that the user does not experience stiction. When the brake turns off, the motor increases torque to push the user out of the wall, while following the desired torque. In (c), when the brake reaches saturation, the motor compensates for the difference $\tau_d - \tau_b$.

IV. Conclusion

Hybrid actuation can increase the dynamic range of haptic devices. Yet, stiction created when the brake and motor are engaged concurrently remains a critical issue. Solutions often involve rotary torque sensors connected to the actuator's shaft and control algorithms that rely on precise readings of the user's input torque. In this paper, we propose an alternative

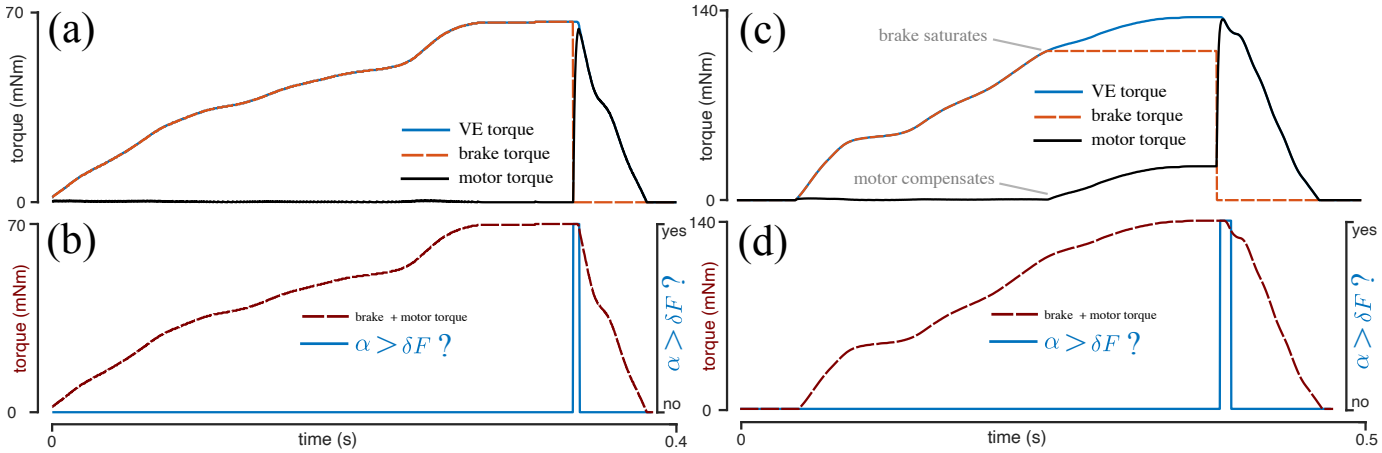

Fig. 6: Experimental results simulating a virtual wall of 400 mNm (a-b), and 800 mNm/rad with brake saturation (c-d).

sensing and control strategy that uses a low-cost load cell in a solution that does not add friction or inertia to the actuator.

Experimental results show that the proposed solution can eliminate stiction when simulating a virtual wall without creating chattering. The control algorithm is independent of the virtual environment, as it only requires the desired torque, the speed of the shaft, and the estimated braking torque. Tuning of the controller parameters is required so that any sudden changes in the input force below a predefined threshold at $P = 0$ indicates the user's intention to reverse motion, which is often the case during haptic interaction. If the threshold is set too low, the brake may disengage too early, leading to chattering. The control algorithm can easily be expanded to turn off the brake proportionally to the user's input force. However, this would require a more robust sensor with a complete model of actuator dynamics built in the controller.

A limitation of any hybrid motor/brake actuator is that the maximum torque it can provide to oppose the user's torque is greater than the torque in the direction of the user's motion. If the desired torque is higher than the maximum motor torque, the user will experience a torque discontinuity when reversing exiting a virtual wall or spring. However, this discontinuity has been shown not to affect the perception of the virtual environment [16]. Future work will focus on redesigning the actuator in a compact format to enable applications with multi-degree-of-freedom. The control algorithm will be extended to take into account the kinematic structure of the device to control the direction of the desired end effector force.

References

[1] T. A. Kern, C. Hatzfeld, and A. Abbasimoshaei, *Engineering haptic devices*. Springer Nature, 2023.

[2] A. Lelevé, T. McDaniel, and C. Rossa, "Haptic training simulation," *Frontiers in virtual reality*, vol. 1, p. 3, 2020.

[3] J. E. Colgate and J. M. Brown, "Factors affecting the z-width of a haptic display," in *International Conference on Robotics and Automation*, pp. 3205–3210, IEEE, 1994.

[4] A. Mohand-Ousaid, G. Millet, S. Régnier, S. Haliyo, and V. Hayward, "Haptic interface transparency achieved through viscous coupling," *The Int. Journal of Robotics Research*, vol. 31, no. 3, pp. 319–329, 2012.

[5] A. Mashayekhi, S. Behbahani, F. Ficuciello, and B. Siciliano, "Analytical stability criterion in haptic rendering: The role of damping," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 2, pp. 596–603, 2018.

[6] R. J. Adams and B. Hannaford, "Stable haptic interaction with virtual environments," *IEEE Transactions on robotics and Automation*, vol. 15, no. 3, pp. 465–474, 1999.

[7] M. Łacki and C. Rossa, "Design and control of a 3 degree-of-freedom parallel passive haptic device," *IEEE Transactions on Haptics*, vol. 13, no. 4, pp. 720–732, 2020.

[8] M. Lacki and C. Rossa, "On the feasibility of multi-degree-of-freedom haptic devices using passive actuators," in *Inter. Conference on Intelligent Robots and Systems*, pp. 7288–7293, IEEE, 2019.

[9] C. Rossa, J. Lozada, and A. Micaelli, "Design and control of a dual unidirectional brake hybrid actuation system for haptic devices," *IEEE transactions on haptics*, vol. 7, no. 4, pp. 442–453, 2014.

[10] M. S. Alkan, H. Gurocak, and B. Gonenc, "Linear magnetorheological brake with serpentine flux path as a high force and low off-state friction actuator for haptics," *Journal of intelligent material systems and structures*, vol. 24, no. 14, pp. 1699–1713, 2013.

[11] M. Antolini, O. Köse, and H. Gurocak, "A first order transfer function to balance the workload in brake-motor hybrid actuators," in *IEEE Haptics Symposium*, pp. 509–514, 2014.

[12] C. Rossa, J. Lozada, and A. Micaelli, "Stable haptic interaction using passive and active actuators," in *IEEE International Conference on Robotics and Automation*, pp. 2386–2392, IEEE, 2013.

[13] S. Aiguo and F. Liyue, "Multi-dimensional force sensor for haptic interaction: A review," *Virtual Reality & Intelligent Hardware*, vol. 1, no. 2, pp. 121–135, 2019.

[14] L. Zhang, S. Guo, H. Yu, and Y. Song, "Performance evaluation of a strain-gauge force sensor for a haptic robot-assisted catheter operating system," *Microsystem Technologies*, vol. 23, pp. 5041–5050, 2017.

[15] T.-B. Kwon and J.-B. Song, "Force display using a hybrid haptic device composed of motors and brakes," *Mechatronics*, vol. 16, no. 5, pp. 249–257, 2006.

[16] C. Rossa, M. Anastassova, A. Micaelli, and J. Lozada, "Perceptual evaluation of the passive/active torque and stiffness asymmetry of a hybrid haptic device," in *International Conference, EuroHaptics, Versailles, France, Proceedings, Part I 9*, pp. 55–60, Springer, 2014.

[17] H. G. Gang, S.-B. Choi, and J. W. Sohn, "Experimental performance evaluation of a mr brake-based haptic system for teleoperation," *Frontiers in Materials*, vol. 6, p. 25, 2019.

[18] D. Marioli, P. Rolla, and A. Taroni, "Strain gauge transducers: an evaluation of accuracy limits," *Measurement*, vol. 10, no. 3, pp. 98–104, 1992.

[19] F. Conti and O. Khatib, "A new actuation approach for haptic interface design," *Inter. J. of Robotics Research*, vol. 28, no. 6, pp. 834–848, 2009.

[20] R. Kilchenman and M. Goldfarb, "Force saturation, system bandwidth, information transfer, and surface quality in haptic interfaces," in *IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1382–1387, IEEE, 2001.