Effects of Velocity Processing on Passivity Controllers for Physical Human-Robot Interaction

Xinliang Guo [*,**] **Jonathan C. van Zanten** [***]
Vincent Crocher [*] **Arno H.A. Stienen** [***] **Ying Tan** [*]
Denny Oetomo [*]

[*] *Human Robotics Laboratory, Department of Mechanical Engineering, The University of Melbourne, Parkville, Australia (e-mail: xinliangg@student.unimelb.edu.au, {vcrocher, yingt, doetomo}@unimelb.edu.au)*
[**] *Graeme Clark Institute, The University of Melbourne, Australia*
[***] *Department of BioMechanical Engineering, Faculty of Mechanical Engineering, Delft University of Technology, Delft, Netherlands. (e-mail: {J.C.vanZanten, A.H.A.Stienen}@tudelft.nl)*

Abstract:
Physical human-robot interaction (pHRI) systems are widely used in assistive robotics for safe and responsive collaboration. They often employ admittance control to convert human forces into robot motion commands and incorporate passivity controllers to ensure stable interaction. Velocity signals are critical for defining motion and contributing to energy estimation in passivity control, but how different velocity processing methods affect system performance across passivity controllers remains unclear. This work experimentally investigates common velocity processing methods in interaction with state-of-the-art passivity controllers. Results show that low-pass filtering velocity commands improves performance but increases delay, whereas state-space fusion of encoder and accelerometer measurements achieves the best balance. A conservatively configured ultimately passive controller (UPC) provides stable interaction across conditions, offering practical insights for designing stable and high-performance pHRI systems.

Keywords: Physical human-robot interaction, admittance control, passivity controller, velocity signal processing

1. INTRODUCTION

Physical human-robot interaction (pHRI) systems have demonstrated broad applications in healthcare and manufacturing (Mehrholz et al., 2020; Cherubini et al., 2016). Many such systems rely on either impedance or admittance control schemes to regulate the robot dynamic response by rendering a digital virtual environment (VE), through which the human user experiences motion and force feedback (Keemink et al., 2018). When admittance control is adopted, the user applies force to the robotic device and the VE converts this measured force into the desired motion for the servomotor, enabling a natural and intuitive pHRI. In this process, achieving both stability and nominal system performance is essential for safe and high-quality interaction, but this remains challenging as improving one often degrades the other (Colgate and Schenkel, 1994).

While many control strategies exist for pHRI, passivity-based approaches are widely used because they provide a sufficient condition for stable interaction without requiring detailed human or robot models (Miller et al., 2000). For example, Hannaford and Ryu (2002) and Ryu et al. (2004) introduced a time-domain passivity method that employs a passivity observer (PO) to monitor energy exchange and a passivity controller (PC) to inject an energy-dissipating element (*i.e.* virtual damping) when needed to maintain system passive. This approach forms the basis of most subsequent passivity control developments in pHRI and is therefore referred to as the "classic PC". More recently, Ferraguti et al. (2019) proposed an energy-tank formulation that stores and releases energy as needed, combined with a device oscillation detector, to preserve passivity and stable interaction.

However, passivity approaches are inherently conservative, as they maintain stability at the cost of pHRI performance (Buerger and Hogan, 2006). To address this limitation, the ultimately passive controller (UPC) was proposed, allowing non-passive transients while ensuring system passivity at steady-state, thereby enabling a balance between stability and performance (Guo et al., 2025). This is achieved through switching between a UPC nominal controller for nominal performance and a conservative controller for passivity enforcement, based on PO-estimated energy and the UPC's tuned energy bounds.

When PCs are implemented on admittance devices, system performance may be influenced by the velocity signal

⋆ The work has been partially supported by the Australian Research Council's Linkage Projects funding scheme (LP220100417). Support from the Graeme Clark Institute's travel support scheme is also acknowledged.

configurations in the admittance control scheme. Velocity signals not only drive the robot motion in the control loop, modulating how admittance control is implemented, but they are also required for the computation of real-time energy exchange through the PO for PCs actions. Still, different velocity processing methods (*e.g.* filters, state-space fusion) used within the control loop lead to varied signal qualities and introduce distinct loop delays, thereby affecting VE rendering, system responsiveness, and the accuracy of energy estimation. All these factors can influence the overall pHRI performance under different PCs, but their combined impact has not yet been assessed. Therefore, this work investigates the effect of different velocity signal processing methods on pHRI system performance in an admittance control scheme, through an experimental evaluation of selected state-of-the-art passivity controllers: the classic PC, the energy-tank PC, and the UPC.

2. METHODS

2.1 Experimental Setup

Device and Characteristics The experiment was conducted on the Shoulder Elbow Perturbator (SEP) (Hankamp Rehab, Enschede, Netherlands) as shown in Figure 1. As described in van Zanten et al. (2025), the SEP is a one-degree-of-freedom rotational robot with a maximum range of motion of 160° and an arm of $0.3m$ connected to the motor output shaft via a load cell.

The admittance control framework and PCs were implemented in MATLAB Simulink 2024b and converted to C-code to be built into the Beckhoff TwinCAT 3.1 software. The system operated at $2kHz$ for both control and data acquisition. The load cell (Futek LCM200, Futek, USA) was used to measure the rotational interaction torque and the motor encoder provided kinematic measurements, which were transformed into linear force and kinematics at the robot arm endpoint in task space. An accelerometer (ADXL335, Analog Devices, USA) was also mounted at the endpoint of the robot arm to provide an additional kinematic measurement.

When the SEP device executed velocity commands, a motor drive delay of $2ms$ was identified. This delay was measured as the time between sending a velocity command to the motor and capturing the corresponding velocity from the encoder measurement, with no embedded filters or processing applied in this procedure.

VE Rendering As noted in Keemink et al. (2018), it is more challenging for an admittance device to achieve high transparency (*i.e.* low virtual mass and damping of the VE), especially when the human operator (HO) suddenly exhibits high stiffness. To evaluate system performance under such a demanding condition, this work employed the following approximated discrete-time virtual mass-damper model with sampling period T_s for digital VE at the k-th sampling instant:

$$v_{ve}[k] = \frac{f_m[k]T_s + m_{ve}v_m[k]}{m_{ve} + b_{ve}T_s}, \tag{1}$$

where v_{ve} is the desired velocity for VE rendering, f_m is the measured linear interaction force, and v_m is the measured linear velocity, all of which are expressed at the robot arm endpoint in the direction of rotation. The characteristics of VE are determined by the selected virtual mass m_{ve} and virtual damping coefficient b_{ve}. In the case where no PC was implemented, the velocity command v_d executed was simply equal to v_{ve}.

Importantly, when the SEP device was rendering an arbitrary VE, high-frequency chattering around $140Hz$ was observed and persisted even when the rendered transparency was decreased, regardless of the HO's behaviour. This chattering was likely due to quantisation effects in the fast-sampling small-scale robot, and sensor noise. Therefore, appropriate velocity processing methods were required in the control loop to enable stable VE rendering.

Energy Observing A digital PO was implemented to estimate the energy flow of the pHRI system. The energy was first calculated using the measured force f_m and velocity v_m as:

$$E[k] = \sum_{n=0}^{k} f_m[n]v_m[n]T_s + E[0]. \tag{2}$$

Then, based on the computed $E[k]$ and the VE command $v_{ve}[k]$ to be applied, the future energy (one-step prediction) was estimated as:

$$\hat{E}[k+1|k] = E_{all}[k] + f_m[k]v_{ve}[k]T_s. \tag{3}$$

Here, $\hat{E}[k+1|k]$ served as the PO's estimation outcome.

2.2 Velocity Processing Methods

To enable the system to render a stable VE for the HO's compliant haptic interaction behaviour at low stiffness, three commonly used velocity processing methods were successively implemented in the control loop.

Low-Pass Filters The first method applied a $12th$-order FIR low-pass filter with cut-off frequency $10Hz$ to the velocity commands v_d to remove high-frequency components (*i.e.* the observed $140Hz$ chattering) before they were executed by the servomotor. Another $4th$-order FIR low-pass filter with cut-off frequency $20Hz$ was applied to the encoder velocity measurements to mitigate quantisation effects and obtain v_m values. However, these additional filters introduced extra delays of $3ms$ and $1ms$ respectively, leading to a total loop delay of $6ms$.

Sensor Fusion with State-Space Model This method built a discrete state-space model and fused the encoder measured angular position θ_{enc} and the accelerometer measured and transformed angular acceleration $\ddot{\theta}_{acc}$ into an estimated angular velocity using a state-space model:

$$x[k+1] = Ax[k] + Bu[k],$$
$$y[k] = Cx[k], \tag{4}$$

where $u[k]$ is the input vector formed by $\left[\theta_{enc}[k] \; \ddot{\theta}_{acc}[k]\right]^T$, and $y[k]$ is the output vector. The state vector $x[k]$ is

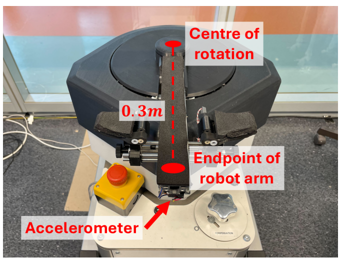


Fig. 1. The rotational robot used in the experiment.

defined as $\left[\hat{\theta}[k]\ \hat{\omega}[k]\ \hat{\alpha}[k]\right]^T$, representing the estimated angular position, velocity, and acceleration, respectively. The matrices A, B, and C were tuned with pre-experiment calibration data of the SEP device to achieve a balance between noise suppression and drift reduction in the estimated states, as follows:

$$A = \begin{bmatrix} 0.8787 & 4.6935 \times 10^{-4} & 8.9474 \times 10^{-8} \\ -14.9714 & 0.9962 & 4.1970 \times 10^{-4} \\ -923.8519 & -0.2360 & 0.9990 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.1213 & 3.0391 \times 10^{-8} \\ 14.9714 & 7.6654 \times 10^{-6} \\ 923.8519 & 9.5380 \times 10^{-4} \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

The estimated angular velocity $\hat{\omega}$ was converted into linear velocity v_m at the robot arm endpoint for subsequent use in VE rendering. The implementation of additional kinematic measurements by the accelerometer and this state-space model introduced only one sample delay of $0.5ms$, leading to a total loop delay of $2.5ms$.

Prior Velocity Command This method followed the admittance control model used in Keemink et al. (2018) and van Zanten et al. (2025), which is a commonly used approach that decouples the desired velocity generation from real-time velocity measurements, under the assumption that the desired velocity v_d is executed immediately. Using this method, the velocity command from the previous sample, represented by $v_d[k-1]$, was used in place of the measured velocity $v_m[k]$ for VE rendering. It is noted that this method did not introduce any additional delay beyond the motor drive delay which was measured as $2ms$.

2.3 Passivity Controllers Implementation

Three passivity control algorithms, namely classic PC, energy tank, and UPC, were implemented on the SEP device to evaluate their effectiveness in stabilising potentially unstable pHRI systems. The classic PC implementation followed Ryu et al. (2004). It relied on the implemented PO to monitor the passivity status and calculated the required PC virtual damping online to dissipate excess energy when needed. For safety considerations, the velocity command was set to saturate at $1m \cdot s^{-1}$ when the PC was working.

For the energy tank approach, the following parameters were defined after the calibration procedure provided in Ferraguti et al. (2019): oscillation detection threshold $\epsilon = 0.8m \cdot s^{-2}$; forgetting factor $\beta = 0.1$; upper bound of inertia variation $\Delta M = 0.1kg$; tank storage upper bound $\bar{T} = 0.2J$ and lower bound $\delta = 0.02J$.

Following Guo et al. (2025), the UPC parameters were chosen to allow switching between the nominal controller with $\{m_{PC,N} = 0, b_{PC,N} = 0\}$ and the conservative controller with $\{m_{PC,C} = 0, b_{PC,C} = 10Ns \cdot m^{-1}\}$. The UPC's upper and lower bound of energy were selected to be $\bar{E} = 0.05J$ and $-\underline{E} = -0.05J$ respectively.

In the subsequent experimental evaluation, when the default configurations of the passivity controllers showed insufficient ability to stabilise the pHRI system, the authors adjusted the configurations as needed when possible. The adapted parameters are reported as results in Section 3.

2.4 Procedure and Evaluation

Since undesired oscillations in the pHRI system can occur when the admittance device intends to be very transparent and the HO suddenly exhibits high stiffness, the experiment was conducted with the HO holding the endpoint of the robot arm and performing a sudden movement along the rotational axis, followed by a sudden stop with persistent high arm stiffness (*i.e.* maximum arm co-contraction).

In the experiment, stable interaction and highly transparent system behaviour were expected. Stability was assessed based on the absence of sustained oscillations or high-frequency chattering. To evaluate the overall system performance in rendering the more "transparent" possible behaviour, an effective virtual mass m_e was defined as the overall mass value to be rendered in the system, and a resulting effective virtual damping b_e was calculated as:

$$b_e[k] = \frac{f_m[k]T_s - m_e[k]v_d[k] + m_e[k]v_m[k]}{v_d[k]T_s}. \tag{6}$$

Here, a smaller m_e and b_e value indicate a more transparent system where the overall rendered behaviour is closer to the intended VE.

3. RESULTS

3.1 Low-Pass Filters

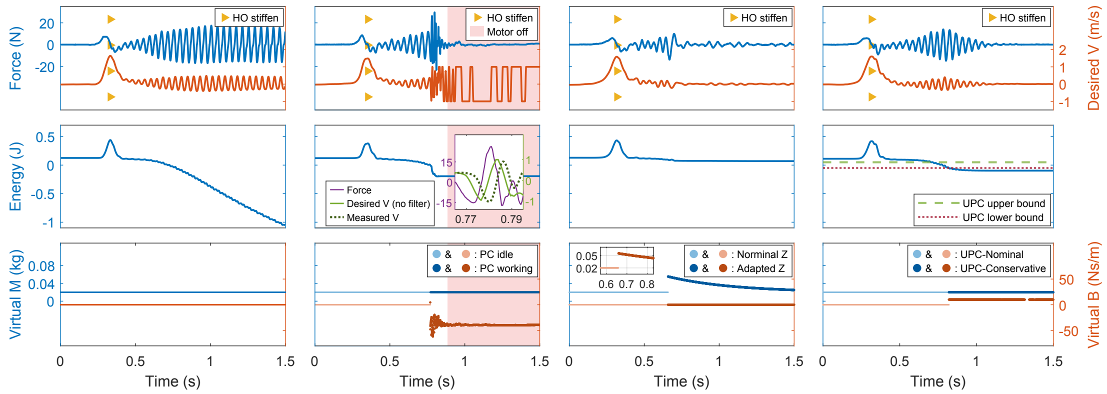
VE Rendering Behaviour A highly transparent VE could be rendered with $\{m_{ve} = 0.02kg, b_{ve} = 0.02Ns \cdot m^{-1}\}$ using this velocity processing method. Lower values of m_{ve} and b_{ve} were not further attempted for safety considerations, as this VE mass was already much lighter than the robot arm mass. The rendered VE provided stable haptic interaction with the HO's compliant behaviour at low arm stiffness, as shown in the first $0.4s$ of system performance in Figure 2.a. After that, in response to the HO's high stiffness behaviour in such a transparent VE with no PC implemented, the robot arm continued oscillating, with the PO-estimated energy dropping to negative values and maintaining this trend which reflected instability.

Passivity Controllers Effectiveness When the classic PC was used with this velocity processing method and rendered VE (Figure 2.b), it was unable to stabilise the system, and the intense oscillations caused the motor to shut down, as indicated by the pale pink area in the plots. The $6ms$ loop delay associated with the low-pass filters led to the interaction force and PC velocity being out of phase, as shown in the inserted graph, making a precise online calculation of the PC virtual damping impossible.

The tank in its default configuration stabilised the system (Figure 2.c), using the parameters in Table 1. The HO's high stiffness behaviour elicited robot arm oscillations, leading to measured tracking error exceeding pre-defined oscillation detection threshold ϵ, which in turn triggered the tank to implement incremental mass and damping for stabilisation. The PO-estimated energy was not used for tank implementation but provided for comparison.

Table 1. Tank parameters for three configurations: default, state-space-model update (State Sp), and prior-command update (Prior Cmd).

Params	Unit	Default	State Sp	Prior Cmd
ϵ	$m \cdot s^{-2}$	0.8	0.6	0.8
β	-	0.1	0.1	0.1
ΔM	kg	0.1	0.05	0.1
$\bar{T}$	J	0.2	0.2	0.2
δ	J	0.02	0.02	0.02
η	-	1	500	100



(a) VE of $m_{ve} = 0.02kg$, $b_{ve} = 0.02Ns \cdot m^{-1}$ (b) Classic PC (c) Tank (d) UPC

Fig. 2. Using low-pass filters, system performance under HO's high stiffness behaviour across passivity controllers.

The UPC was also able to stabilise the system (Figure 2.d). The robot arm oscillations resulted in the PO-estimated energy dropping below the UPC's lower bound of energy $-\underline{E} = -0.05J$, triggering the UPC to switch to its conservative controller for system stabilisation as described in Guo et al. (2025).

3.2 Sensor Fusion with State-Space Model

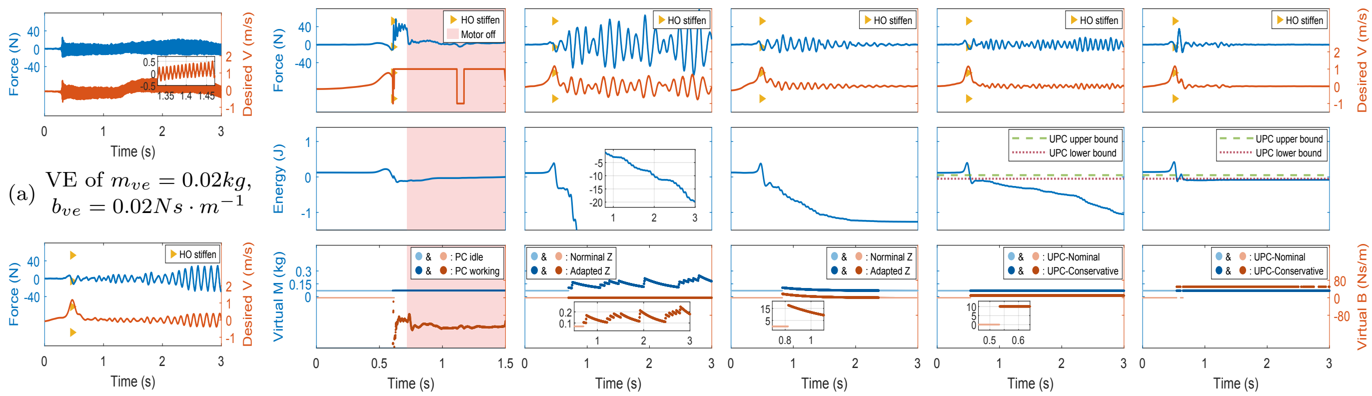
VE Rendering Behaviour When the VE parameters were set to $\{m_{ve} = 0.02kg,\, b_{ve} = 0.02Ns \cdot m^{-1}\}$ as for the low-pass filters method, the system performance without PC in Figure 3.a shows that high-frequency chattering occurred as soon as physical interaction took place between the HO and the robot arm, even without the HO exhibiting any high stiffness behaviour. The chattering had a frequency of approximately $110Hz$, as illustrated on the inserted graph zooming on the desired velocity.

Gradually increasing the VE parameters to $\{m_{ve} = 0.07kg,\, b_{ve} = 0.07Ns \cdot m^{-1}\}$ (Figure 3.b) enabled stable haptic interaction for the HO under compliant behaviour. Although oscillations occurred under high stiffness, no chattering was observed. Notably, this oscillation had a lower frequency and larger amplitude, which differed from the oscillation pattern observed in the system baseline configuration shown in Figure 2.a in Section 3.1.

Passivity Controllers Effectiveness Following the HO's high stiffness behaviour with a VE of $\{m_{ve} = 0.07kg,\, b_{ve} = 0.07Ns \cdot m^{-1}\}$, the classic PC was still unable to stabilise the system and led the motor to shut down, as shown in Figure 3.c. This performance remained similar to that of the system using low-pass filters, even though the loop delay had been reduced from $6ms$ to $2.5ms$.

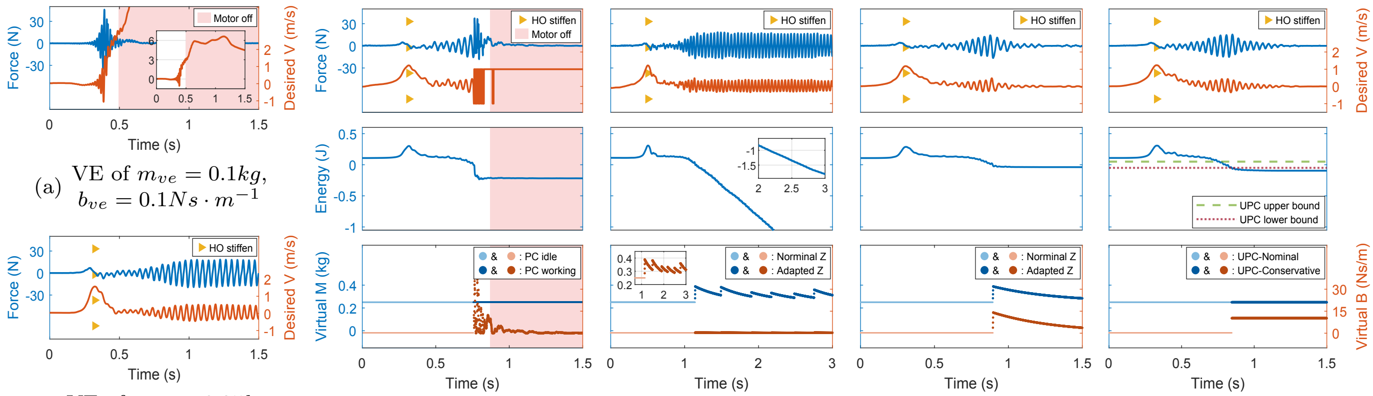
When the tank approach with the default configuration was used (Figure 3.d), it failed to stabilise the system. The HO's high stiffness continuously elicited robot arm oscillations, which in turn repeatedly triggered the tank to implement incremental mass and damping. To enable the tank to stabilise the system under this condition, its parameters were re-tuned to an updated version provided in Table 1. Specifically, the oscillation detection threshold ϵ was reduced for more sensitive detection and faster tank response, and the upper bound of inertia variation ΔM was halved to mitigate the inertia effect. More importantly, an additional tank parameter—the damping adding factor η—was introduced. This factor defines that the incremental damping implemented equals its default value multiplied by η ($\eta = 1$ in default configuration). The system performance shown in Figure 3.e demonstrates that the tank updated configuration successfully stabilised the system, though at the cost of a heavier transparency performance sacrifice represented by much larger b_e values.

Regarding the UPC, Figure 3.f shows that its default configuration was able to dissipate some excess energy, thereby suppressing the robot arm oscillation to some extent as indicated by the smaller oscillation amplitude compared to the no-PC condition (see Figure 3.b). How-



(a) VE of $m_{ve} = 0.02kg$, $b_{ve} = 0.02Ns \cdot m^{-1}$ (b) VE of $m_{ve} = 0.07kg$, $b_{ve} = 0.07Ns \cdot m^{-1}$ (c) Classic PC (d) Tank default config. (e) Tank updated config. (f) UPC $b_{PC,C} =$ 10 (default) (g) UPC $b_{PC,C} =$ 50 (updated)

Fig. 3. Using state-space model for sensor fusion, system performance under HO's high stiffness across conditions.



(a) VE of $m_{ve} = 0.1kg$, $b_{ve} = 0.1Ns \cdot m^{-1}$

(b) VE of $m_{ve} = 0.25kg$, $b_{ve} = 0.25Ns \cdot m^{-1}$

(c) Classic PC

(d) Tank default config.

(e) Tank updated config.

(f) UPC

Fig. 4. Using prior velocity command, system performance under HO's high stiffness across conditions.

ever, this was still not sufficient to fully stabilise the system under this oscillation pattern. When a more aggressive UPC conservative controller was implemented by increasing $b_{PC,C}$ from 10 to $50Ns \cdot m^{-1}$ to enhance excess energy dissipation, the results in Figure 3.g demonstrates the UPC's capability to stabilise the system with an increase in the resultant b_e values.

3.3 Prior Velocity Command

VE Rendering Behaviour The attempted VE started from $\{m_{ve} = 0.1kg, b_{ve} = 0.1Ns \cdot m^{-1}\}$ and the corresponding system performance is provided in Figure 4.a. These values were selected higher than those in Section 3.1 and 3.2 as this method assumes immediate velocity command execution, whereas the actual $2ms$ loop delay creates a high likelihood of out-of-phase phenomenon when the system is very transparent. It can be observed that any physical interaction between the HO and the robot arm induced out-of-phase phenomenon, resulting in strong instantaneous oscillations and eventual motor shutdown.

While gradually increasing the VE to $\{m_{ve} = 0.25kg$, $b_{ve} = 0.25Ns \cdot m^{-1}\}$, Figure 4.b shows robot arm oscillations as a result of the HO's high stiffness, but no high-frequency chatter was present.

Passivity Controllers Effectiveness When the VE was rendered at $\{m_{ve} = 0.25kg, b_{ve} = 0.25Ns \cdot m^{-1}\}$ under HO's high stiffness behaviour, the classic PC (Figure 4.c) failed to stabilise the system and led to motor shutdown, even with the minimum achievable delay of $2ms$ (*i.e.* the motor drive delay) in this velocity processing method.

The tank approach implemented with its default configuration still failed to stabilise the system. As shown in Figure 4.d, the robot arm oscillations induced by HO's high stiffness were not eliminated by the tank approach but repeatedly triggered its response of incremental mass and damping. This issue was resolved by implementing a damping adding factor η of 100, with the full set of parameters provided in Table 1. Using the updated tank configuration (Figure 4.e), the tank approach effectively stabilised the system, with b_e increasing as a result.

The UPC was evaluated using the parameters from its default configuration. As shown in Figure 4.f, the high stiffness induced oscillation caused the PO-estimated energy to drop to $-\underline{E} = -0.05J$, triggering a switch to the conservative controller of $b_{PC,C} = 10Ns \cdot m^{-1}$ to successfully stabilise the system.

4. DISCUSSION

This work conducted experiments to investigate the effects of selected commonly used velocity processing methods for an admittance control scheme on pHRI system performance. The experimental evaluation assessed VE rendering behaviour and the effectiveness of three selected PCs in stabilising a pHRI system. A summary of the results is provided in Table 2 and discussed in detail subsequently.

4.1 Velocity Processing Effects on VE Rendering

In this experimental investigation, when the HO exhibited compliant behaviour at low arm stiffness, the rendered VE was expected to provide stable interaction without an implemented PC, while attempting to maintain maximum transparency indicated by the lowest possible m_{ve} and b_{ve}.

When three different velocity processing methods were successively implemented in the control loop, the one using low-pass filtered velocity commands showed the best transparent VE performance, corresponding to the lowest selected m_{ve}. By reducing the robot bandwidth, it prevented high-frequency system chatter and provided a "cleaner" system for better rendering performance, although this method also introduced the longest loop delay of $6ms$ which resulted in the lowest responsiveness.

In comparison, adding an additional velocity measurement and fusing it using a state-space model successfully solved the measurement noise issue and greatly improved system responsiveness to a $2.5ms$ loop delay, but at the cost of VE performance as reflected by a higher achievable m_{ve}. This is because the device bandwidth was not not sharply reduced, so some noise and non-linearities still remained.

Table 2. Effects of velocity processing methods on pHRI performance across passivity controllers. (✓: successful stabilisation, ×: failed, †: previously successful in low sacrif. config.)

Passivity controllers	LP Filters	State Sp	Prior Cmd
Stable VE without PC $\{m_{ve}(kg), b_{ve}(Ns \cdot m^{-1})\}$	{0.02, 0.02}	{0.07, 0.07}	{0.25, 0.25}
Classic PC	×	×	×
Tank (default config.)	✓	×	×
UPC (low sacrif. config.)	✓	×	✓
UPC (high sacrif. config.)	✓†	✓	✓†

Finally, using the admittance control model that relies on prior velocity commands is the easiest to implement, as it does not require device characteristic identification or additional sensors. However, this method assumes immediate velocity command execution, whereas in practice the loop delay is difficult to eliminate on an admittance device. The system was thus highly sensitive to noise, easily leading to out-of-phase phenomenon when the VE was very transparent. This simple method should probably be used only when best VE performance is not required.

4.2 Velocity Processing Effects on Passivity Controllers

When the pHRI system operated in a VE with the highest possible transparency under any velocity processing method, the HO's high stiffness resulted in the robot arm to oscillate, at which point different passivity controllers were expected to stabilise the system. Unfortunately, the classic PC still failed to achieve stabilisation, even when the loop delay was reduced to the minimum achievable value of $2ms$. This is because the residual delays still make precise online calculation of the PC response impossible.

The effectiveness of the tank approach was affected by oscillation patterns arising from specific velocity processing method and VE parameters. For example, when the oscillation pattern had lower frequency and larger amplitude as observed in Section 3.2, the tank with default configuration demonstrated very limited effectiveness. On one hand, this pattern reduced measured tracking error during oscillation, making the detection threshold less sensitive and thus delaying tank response. On the other hand, this stabilisation relied largely on incremental mass, but adding mass without sufficient incremental damping caused the HO to experience a strong inertia effect rather than suppressing oscillations. Although the updated tank configurations—particularly the use of additional damping element for stabilisation—significantly improved effectiveness, they also introduced additional tuning work.

The UPC showed overall good behaviours across conditions. The only exception occurred with the less conservative implementation coupled with state-space sensor fusion in Section 3.2, where lower frequency and larger amplitude oscillations appeared. This was due to an optimistic selection of the UPC conservative controller damping $b_{PC,C}$ which was only $10Ns \cdot m^{-1}$, corresponding to a low transparency-performance sacrifice configuration in Table 2. Increasing $b_{PC,C}$ was a straightforward and effective solution, although it led to heavier transparency sacrifice. A high transparency-performance sacrifice configuration (see Table 2) was only experimentally tested in Section 3.2, but it is reasonable to infer its effectiveness for the methods in Section 3.1 and 3.3, as larger damping would enhance energy dissipation and system stabilisation. In practice, this implies that a larger stabilising damping in UPC is recommended for more reliable stability guarantee. This strategy maintains the practicality of the approach, while improving its generalisability when implemented with different velocity processing methods in admittance control and oscillation patterns might be unknown.

4.3 Limitations and Future Work

It is acknowledged that only selected velocity processing methods and state-of-the-art passivity controllers were evaluated in the experiment. Although this work provides some useful findings and practical implications for pHRI design on an admittance device, it remains preliminary. In addition, all haptic interactions in the presented results were performed by a HO. While this offers the advantage of reflecting real-world pHRI scenarios, and although each testing condition was repeated multiple times with consistent system performance observed, such a design still introduced some deviations in the HO's behaviour across trials. Future work could include evaluating more velocity processing methods and passivity controllers, as well as replacing the HO with physical stiff objects for more systematic and quantitative investigations.

REFERENCES

Buerger, S.P. and Hogan, N. (2006). Relaxing passivity for human–robot interaction. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4570–4575.

Cherubini, A., Passama, R., Crosnier, A., Lasnier, A., and Fraisse, P. (2016). Collaborative manufacturing with physical human–robot interaction. *Robotics and Computer-Integrated Manufacturing*, 40, 1–13.

Colgate, J. and Schenkel, G. (1994). Passivity of a class of sampled-data systems: application to haptic interfaces. In *Proceedings of 1994 American Control Conference*, 3236–3240.

Ferraguti, F., Talignani Landi, C., Sabattini, L., Bonfe, M., Fantuzzi, C., and Secchi, C. (2019). A variable admittance control strategy for stable physical human–robot interaction. *The International Journal of Robotics Research*, 38(6), 747–765.

Guo, X., Liu, Z., Crocher, V., Tan, Y., Oetomo, D., and Stienen, A.H.A. (2025). Ultimate passivity: Balancing performance and stability in physical human–robot interaction. *IEEE Transactions on Robotics*, 41, 2050–2066.

Hannaford, B. and Ryu, J.H. (2002). Time-domain passivity control of haptic interfaces. *IEEE Transactions on Robotics and Automation*, 18(1), 1–10.

Keemink, A.Q.L., van der Kooij, H., and Stienen, A.H.A. (2018). Admittance control for physical human–robot interaction. *The International Journal of Robotics Research*, 37(11), 1421–1444.

Mehrholz, J., Pollock, A., Pohl, M., Kugler, J., and Elsner, B. (2020). Systematic review with network meta-analysis of randomized controlled trials of robotic-assisted arm training for improving activities of daily living and upper limb function after stroke. *Journal of NeuroEngineering and Rehabilitation*, 17, 1–14.

Miller, B.E., Colgate, J.E., and Freeman, R.A. (2000). Guaranteed stability of haptic systems with nonlinear virtual environments. *IEEE Transactions on Robotics and Automation*, 16(6), 712–719.

Ryu, J.H., Kim, Y.S., and Hannaford, B. (2004). Sampled- and continuous-time passivity and stability of virtual environments. *IEEE Transactions on Robotics*, 20(4), 772–776.

van Zanten, J.C., Ter Welle, K., Haarman, C.J., Schouten, A.C., Van De Ruit, M., Mugge, W., and Stienen, A.H.A. (2025). System performance analysis of the shoulder elbow perturbator. In *2025 International Conference on Rehabilitation Robotics (ICORR)*, 346–351.