Open Source Dynamic Model of the HRX-1 Robot for Haptic Interaction

Aleksandar Pavlov[1], Joong-Ku Lee[2], Joshua Brown[3,4], Nicola Bailey[1], Jee-Hwan Ryu[2] and Ildar Farkhatdinov[5,4]

***Abstract*— Accurate dynamic modelling is critical for stable and safe human-robot interaction with haptic robots. This paper presents an open-source MATLAB Simulink framework for the HRX-1 haptic robot dynamic model. The model consists of a continuous dynamic plant with integrated discrete-time position, torque/current cascaded control loops acting within a virtual environment. This architecture enables realistic simulation of real hardware behaviour including sampled sensing and control delay. Evaluation of the model is conducted through simulated haptic interaction experimentation between a virtual wall environment and external virtual user force through passivity-based Z-width analysis in the stiffness-damping plane. Two sampling times/frequencies representative of the real system were used to conduct simulations demonstrating stable and unstable interaction. Results showed the stable interaction KB region is decreased significantly for decreased sampling frequency, introducing enough delay to violate passivity, inject energy and produce unstable behaviour.**

I. Introduction

Modelling the physical and dynamic behaviours of robotic systems is a fundamental requirement in the design, analysis, and control of modern robotic and mechatronic platforms. Accurate dynamic models enable the prediction of system responses under varying operational conditions, facilitating controller development, performance optimisation, and stability analysis. This requirement becomes particularly critical in applications where robots physically interact with humans, such as haptic interfaces, wearable robots, and collaborative manipulators. In such systems, robots actively exchange forces with the human body, thereby introducing stringent safety, compliance, and transparency constraints that must be addressed at both the mechanical design and control levels [1], [2].

Correct dynamic modelling supports the prototyping and validation of physically interactive control strategies, including impedance, admittance, and hybrid force–position controllers. These controllers are essential for ensuring safe and efficient operation in medical robotics, where devices are deployed for rehabilitation therapy, assistive augmentation, and clinical interventions such as robot-assisted surgery.

[1]Department of Engineering, King's College London, London, UK.
[2]Department of Civil and Environmental Engineering, Korea Advanced Institute of Science & Technology, Daejeon, Republic of Korea.
[3]Department of Surgery & Cancer, Imperial College of Science, Technology and Medicine, London, UK.
[4]Human Robotix Ltd., London, UK.
[5]School of Biomedical Engineering and Imaging Sciences, King's College London, London, UK. ildar.farkhatdinov@kcl.ac.uk



Fig. 1: The HRX-1 robot in its default wrist configuration

In these contexts, modelling inaccuracies may lead to unsafe force application, degraded therapeutic outcomes, or compromised surgical precision. Consequently, high-fidelity models that incorporate actuator dynamics, human–robot interaction forces, and soft-tissue mechanics are increasingly required [3], [4].

Furthermore, the use of appropriate computational models and physics-based simulators significantly reduces the cost, time, and risk associated with robotic system design and testing. Simulation environments enable rapid prototyping, controller benchmarking, and safety validation prior to hardware deployment. This is particularly advantageous in medical and rehabilitation robotics, where ethical, regulatory, and patient-safety considerations limit early-stage physical experimentation. In addition, the growing availability of open-source robotic simulators has facilitated wider system deployment, reproducibility of research, and collaborative development across institutions. Platforms such as Gazebo, MuJoCo and MATLAB provide researchers and clinicians with accessible tools for modelling, simulation, and user training, thereby accelerating innovation and translational adoption [5], [6].

This paper presents a reproducible open-source dynamic model of the HRX-1 robot [7] for simulation, control system development and haptic stability analysis. The core aims are to

- implement and exemplify modular control and a virtual

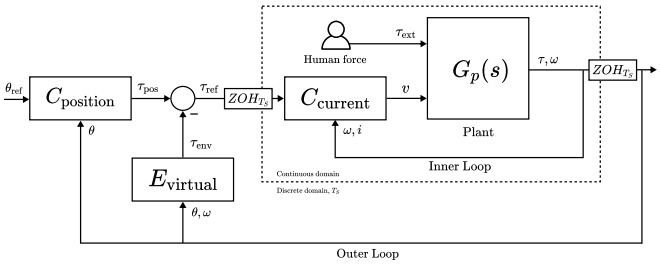


Fig. 2: Hybrid HRX-1 control architecture showing how sampled virtual wall feedback couples to the continuous plant.

environment with configurable discrete-time sampling.
- obtain linear time invariant (LTI) transfer functions using open-loop linearisation.
- simulate human-device interaction effectively within the virtual environment.
- observe and understand the effects of sampling time on haptic stability using Z-width in stiffness-damping plane.

II. Overview of HRX-1 robot

The HRX-1 robot (by Human Robotix Ltd[1]) is a single Degree of Freedom (DoF) torque feedback robot variously used for research into physical human-robot interaction [8], biomechanics [9], neuromechanics [10], and clinical motor assessment and rehabilitation [11], [12], [13], [14], [15]. It is powered by a 600W Maxon EC90 brushless DC motor, capable of generating up to 4 Nm of torque at its output shaft. It is controlled by a Maxon EPOS4 70/15 controller which enables closed-loop control of both current (proportional to torque) and velocity. Position is reported by a 6400 cpt encoder. It can be controlled either via MATLAB over a USB/serial connection (up to approximately 150 Hz control loop rate) or a TI LaunchPad-based real-time controller using a CAN communication interface (approximately 1 kHz control loop rate).

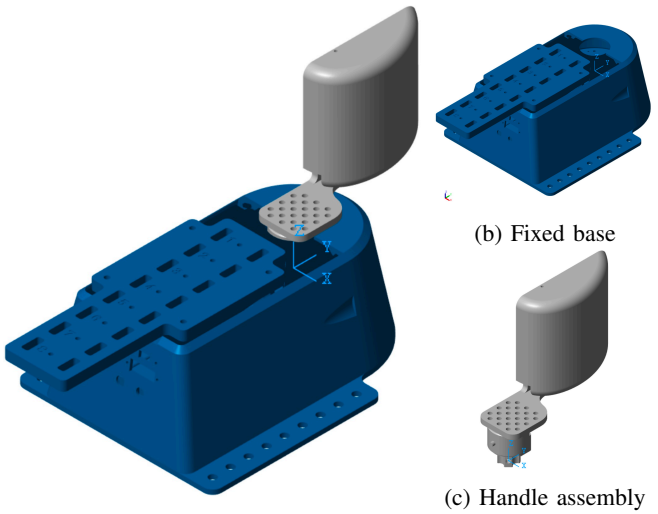
The robot is modular and can be fitted with attachments allowing it to operate on different joints (i.e. wrist, elbow, or ankle) as well as gearboxes and other mechanical elements to change its output performance or kinematics. The default configuration, shown in Figure 1 consists of an open-palm handle and forearm rest, and is intended to measure and manipulate the movement of the wrist.

III. Dynamic Model of HRX-1 robot

A. System Description and Parameters

To model the mechanics of the HRX-1 robot we need to take into account the moving parts which include a brushless direct current (DC) actuator that is fixed to a static desktop base (Figure 3b). The actuator's shaft is attached to a rigidly assembled wrist handle that rotates around the motor axis (Figure 3c). In the presented wrist configuration, all components comprising the handle are rigidly coupled

[1]https://www.humanrobotix.co.uk



(a) Full Assembly

(b) Fixed base

(c) Handle assembly

Fig. 3: HRX-1 Assembly: (a) full assembly, (b) Base, (c) Handle.

with their distributed mass properties being represented by a single lumped load inertia J_L, extracted from Computer-Aided Design model (CAD). The motor's rotor inertia is J_m, and the total inertia can be modelled as,

$$J = J_m + J_L. \tag{1}$$

Since the robot's base is static and is fixed to a tabletop during operation, the continuous-time nonlinear dynamics of the HRX-1 system/plant are isolated to the DC motor's rotor and wrist handle:

$$\begin{aligned} \dot{\theta} &= \omega, \\ \dot{\omega} &= \frac{1}{J}\left(K_t i - b\,\omega - \tau_{\text{ext}}\right), \\ \dot{i} &= \frac{1}{L}\left(v - Ri - K_e\omega\right). \end{aligned} \tag{2}$$

Here, θ is the measured motor-shaft angle and thus the angular position of the rigidly coupled handle; positive rotation is modelled as moving anticlockwise. The input τ_{ext} denotes the externally applied load torque and is subtracted from the motor torque. In the wrist configuration, the constrained operating range is approximately $-90°$ to $+90°$. The parameters for all equations are defined in Table I.

B. Virtual Environment Model

In this paper, we model interaction of the HRX-1 robot with a generic haptic environment represented by a spring-damper virtual wall with virtual stiffness K_{wall} and damping B_{wall}. The boundary θ_{wall} uses the same angular coordinate as θ, is adjustable and defaults to $0°$, at the centre of the constrained range. The virtual-environment feedback term τ_{env}, which is subtracted from the actuator torque command, is computed from the measured joint position and angular velocity:

$$\tau_{\text{env}} = \begin{cases} K_{\text{wall}}\big(\theta - \theta_{\text{wall}}\big) + B_{\text{wall}}\,\omega, & \theta \geq \theta_{\text{wall}}, \\ 0, & \theta < \theta_{\text{wall}}, \end{cases} \tag{3}$$

TABLE I: HRX-1 model parameters used throughout.

symbol	description	value
R	Motor terminal resistance [Ω]	0.28
L	Motor terminal inductance [H]	3.69×10^{-4}
K_t	Torque constant [$\mathrm{N\,m\,A^{-1}}$]	0.136
K_e	Back-EMF constant [$\mathrm{V\,s\,rad^{-1}}$]	0.136
J_m	Motor rotor inertia [$\mathrm{kg\,m^2}$]	5.1×10^{-4}
J_L	Load inertia (CAD) [$\mathrm{kg\,m^2}$]	3.692827×10^{-3}
J	Total inertia ($J = J_m + J_L$) [$\mathrm{kg\,m^2}$]	4.202827×10^{-3}
b	Viscous damping [$\mathrm{N\,m\,s\,rad^{-1}}$]	1.0×10^{-3}
$V_{\max}$	Drive saturation upper limit [V]	30
$V_{\min}$	Drive saturation lower limit [V]	-30
θ_{wall}	Virtual wall location [rad]	0 Default
K_{wall}	Virtual wall stiffness [$\mathrm{N\,m\,rad^{-1}}$]	50 Representative
B_{wall}	Virtual wall damping [$\mathrm{N\,m\,s\,rad^{-1}}$]	0.2 Representative
T_s^{fast}	Control sampling time (fast) [s]	1×10^{-3}
T_s^{slow}	Control sampling time (slow) [s]	10×10^{-3}
controller parameters		
$K_{p\theta}$	Position controller proportional gain [$\mathrm{N\,m\,rad^{-1}}$]	0.033534
$K_{i\theta}$	Position controller integral gain [$\mathrm{N\,m\,rad^{-1}\,s^{-1}}$]	0.009680
$K_{d\theta}$	Position controller derivative gain [$\mathrm{N\,m\,s\,rad^{-1}}$]	0.028561
K_{pi}	Current controller proportional gain [$\mathrm{V\,A^{-1}}$]	0
K_{ii}	Current controller integral gain [$\mathrm{V\,A^{-1}\,s^{-1}}$]	0.5

Here, the wall occupies the positive-angle side, with contact for $\theta \geq \theta_{\mathrm{wall}}$. Subtracting τ_{env} produces the restoring torque on the handle.

C. Control Architecture

Depending on the application, the HRX-1 system is controlled through cascaded position-torque control with the virtual environment force feedback acting parallel to the outer control loop. This architecture lends itself to high modularity between controllers and the virtual environment where the position controller and virtual environment can easily be excluded without affecting operation.

The control architecture consists of three subsystems:

- inner loop current/torque controller,
- virtual environment model,
- outer loop position controller,

with the sole input to the system being voltage supplied to the DC motor by the motor controller, v.

Outer Loop Position Control. When position control is enabled, the outer Proportional-Integral-Derivative (PID) controller uses the tracking error e_θ to generate the desired actuator torque τ_{pos},

$$\tau_{\mathrm{pos}} = K_{p\theta} e_\theta + K_{i\theta} \int e_\theta \, dt + K_{d\theta} \dot{e}_\theta \tag{4}$$

where $e_\theta = \theta_{\mathrm{ref}} - \theta$. The reference θ_{ref} is an arbitrary and adjustable target for this optional outer controller, measured in the same angular coordinate as θ. These angular conventions are illustrated in Figure 4.

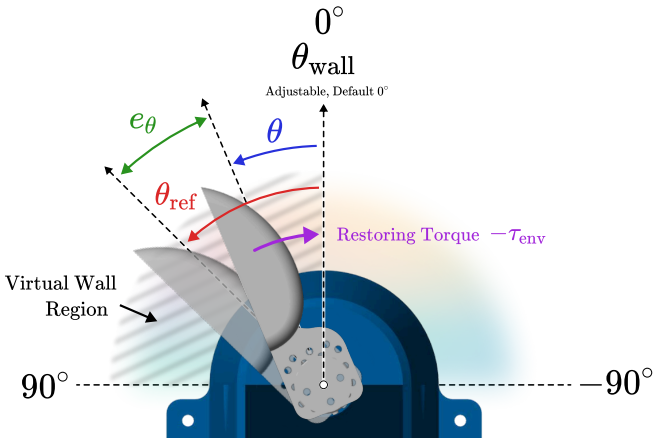


Fig. 4: Top-down schematic view of the HRX-1 robot wrist handle with notations, showing the measured handle angle θ and position reference θ_{ref} from 0°, their error e_θ, and the virtual-wall boundary θ_{wall}. The illustrated handle lies inside the positive-angle wall region, where $-\tau_{\mathrm{env}}$ provides the restoring torque.

Virtual Environment Torque Injection. The virtual environment generates the feedback term τ_{env} from measured joint state (θ, ω) input, as defined in section III-B.

Superposing the desired outer loop position control torque τ_{pos} and the opposing virtual environment feedback term τ_{env} computes the final actuator torque command τ_{ref} fed into the inner loop controller,

$$\tau_{\mathrm{ref}} = \tau_{\mathrm{pos}} - \tau_{\mathrm{env}} \tag{5}$$

Inner Loop Current/Torque Control: Once fed into the inner loop controller, the reference/desired torque τ_{ref} is converted into a reference current using the motor's torque constant K_t,

$$i_{\mathrm{ref}} = \frac{\tau_{\mathrm{ref}}}{K_t}. \tag{6}$$

The current error is regulated using PI-structured feedback with resistive-drop and back-EMF feedforward:

$$\begin{aligned} e_i &= i_{\mathrm{ref}} - i, \\ v^\star &= K_{pi} e_i + K_{ii} \int e_i \, dt + R i_{\mathrm{ref}} + K_e \omega, \\ v &= \mathrm{sat}_{[V_{\min}, V_{\max}]}(v^\star). \end{aligned} \tag{7}$$

The motor-current dynamics are first order. Integral action eliminates steady-state current error and therefore the torque error due to torque being proportional to current, while derivative action is omitted to avoid amplifying measurement noise. The feedforward terms $R i_{\mathrm{ref}}$ and $K_e \omega$ compensate for motor resistance and back-EMF, while the PI controller corrects the remaining current error. The plant was linearised using the Simulink Control Design Model Linearizer app, after which the controller gains reported in Table I were automatically tuned in Simulink using the resultant linear model. The tuned proportional gain was negligible and was

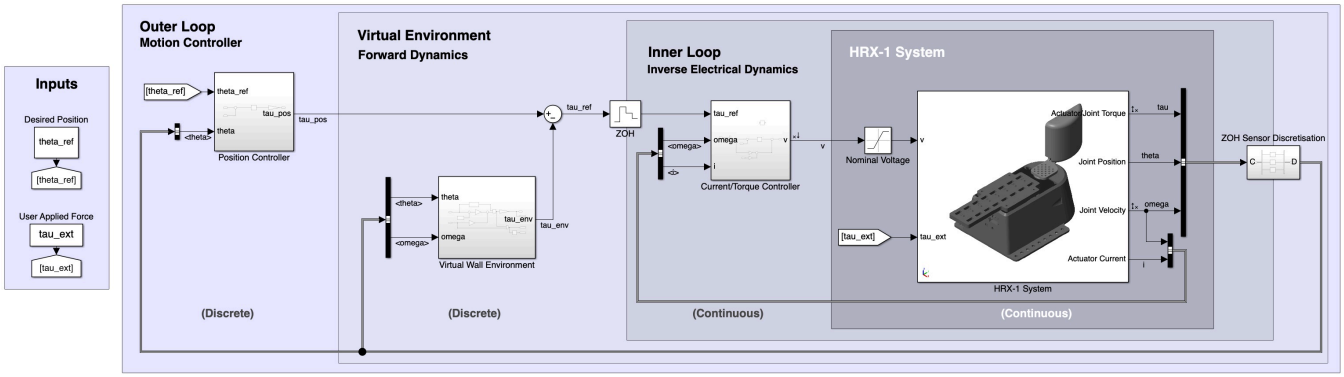


Fig. 5: Simulink implementation of the HRX-1 model, with discrete control and virtual environment blocks surrounding the continuous Simscape plant and inner loop. The model can be accessed via [7].

set to zero after confirming that this did not materially change the simulated response.

Control signal pathway. The simple illustration in (8) shows the pathway or flow from desired to final system position inclusive of the virtual environment:

$$\theta_{\text{ref}} \xrightarrow{\text{PID}} \tau_{\text{pos}} \xrightarrow{-\tau_{\text{env}}} \tau_{\text{ref}} \xrightarrow{\text{current control}} v \xrightarrow{\text{plant}} \theta \quad (8)$$

D. Discrete Time Implementation in MATLAB Simulink

The controller and virtual environment subsystems defined in section III naturally lend themselves to direct implementation in Simulink. However, the HRX-1 robot can be modelled as both a Simulink ODE model and directly using Simscape's Physical and Multibody libraries. The model can be downloaded from [7]. Both models were compared and resulted in a maximum error of 8.94×10^{-5} rad, showing good consensus between the modelling methods. The ODE model served as a benchmark for validation, while the Simscape-Multibody model was additionally used for effective 3D visualisation of the plant dynamics in Simulink's Mechanics Explorer.

To test the effects of real system sampling time, all subsystems outside the plant and inner loop are modelled in discrete-time while the plant itself with its internal current control is modelled in continuous-time, reflecting the real system. Two sampling times were considered for discrete-time: $T_s^{\text{fast}} = 1$ ms corresponds to the TI LaunchPad/CAN controller, while $T_s^{\text{slow}} = 10$ ms (100 Hz) lies within the MATLAB USB/serial control rate of up to approximately 150 Hz. The dynamic model was discretised using

- Zero-Order-Hold (ZOH) method applied at the continuous-discrete system interfaces: at the simulated sensor measurements and at the torque controller interface;
- Discrete integrator and differentiator implementations of the PI and PID controllers in the current and position controllers respectively.

These methods can be seen in Figure 5 where the subsystem named 'Sensor Discretisation' applied the ZOH method on the plant sensor signals, at the torque controller interface and also in the position controller subsystem, where the PID(z) subsystem internal to the controller denotes discrete implementations of the internal integrators and differentiators.

IV. Modelling Results

The following section presents results on (A) system dynamics identification, (B) haptic interaction with a virtual wall, and (C) haptic interaction stability evaluation through Z-width analysis. The last two simulations were conducted with sampling rates of 100 Hz and 1000 Hz to evaluate the quality of haptic rendering.

A. System Dynamics Identification

After open-loop linearisation was performed, the following LTI transfer functions describing the plant were obtained,

$$\begin{bmatrix}\tau(s)\\ \omega(s)\end{bmatrix} = G_p(s)\begin{bmatrix}v(s)\\ \tau_{\text{ext}}(s)\end{bmatrix} \quad (9)$$

where $\tau(s)$ and $\omega(s)$ are the plant output joint torque and angular velocity in the Laplace domain, respectively.

$$G_p(s) = \begin{bmatrix} G_{\tau,v}(s) & G_{\tau,\tau_{\text{ext}}}(s) \\ G_{\omega,v}(s) & G_{\omega,\tau_{\text{ext}}}(s)\end{bmatrix}$$

$$G_p(s) = \begin{bmatrix} \dfrac{368.6s}{D(s)} & -\dfrac{s^2+758.8s}{D(s)} \\ \dfrac{87690}{D(s)} & -\dfrac{237.9s+180500}{D(s)}\end{bmatrix} \quad (10)$$

$$\text{with } D(s) = s^2 + 759s + 12110, \quad (11)$$

$$\tau(s) = G_{\tau,v}(s)v(s) + G_{\tau,\tau_{\text{ext}}}(s)\tau_{\text{ext}}(s), \quad (12)$$

$$\omega(s) = G_{\omega,v}(s)v(s) + G_{\omega,\tau_{\text{ext}}}(s)\tau_{\text{ext}}(s). \quad (13)$$

The poles of the system are given by the roots of the characteristic polynomial or denominator $D(s)$, where both poles lie in the left half of the complex plane, indicating that the linearised continuous-time plant is internally stable. The poles $s_1 = -16.3$ and $s_2 = -742.7$ correspond to the slower mechanical and faster electrical dynamics, respectively.

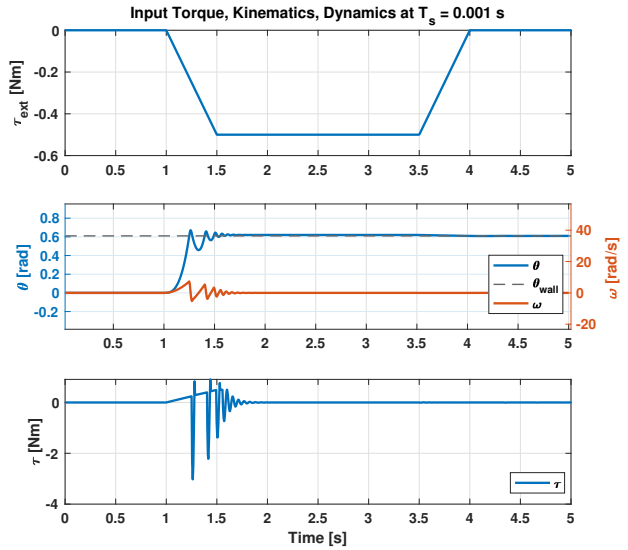


Fig. 6: At 1 ms sampling time, wall contact remains stable as the handle undergoes a damped rebound and remains bounded against the wall until the user input force is released.

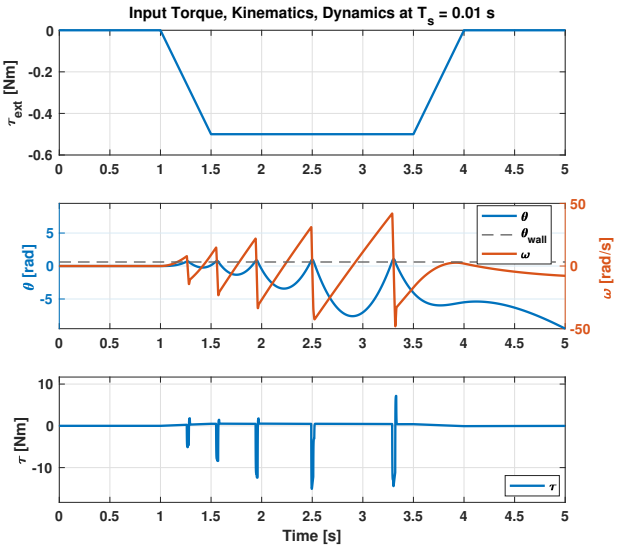


Fig. 7: At 10 ms sampling, the same interaction with the wall does not remain bounded, with oscillations growing after impact.

B. Haptic Interaction in Virtual Environment

To examine the interaction between the virtual wall and user force input, a signal input with a linear engaging and disengaging characteristic (Figure 6, Figure 7) was used with an absolute maximum of $0.5\,\text{Nm}$. This emulates a light smooth press into the wall, realistic for human interaction while remaining well within saturation limits on actuator response torque.

The virtual wall's location θ_{wall} was automatically moved near 0.6 such that contact occurred at half the input maximum at t=1.25 s, enabling visualisation of collision during acceleration into the wall.

A moderately stiff wall with K_{wall}=50 Nm/rad and B_{wall}=0.2 Nm·s/rad was chosen to demonstrate stable and unstable behaviour as shown in Figure 8 where stable contact is achieved in about 3 bounces for the stable case and a large divergence is observed for the unstable case. This representative configuration lies inside the passive-stable region for T_s^{fast} (1000 Hz) but does not remain in this region for T_s^{slow} (100 Hz). The divergence observed for the unstable case is a direct result of the delay introduced by sampled position and velocity, creating phase lag, reducing the available amount of damping to counteract the destabilising effect of sampling and causing the discrete wall to inject energy into the system [16].

C. Z-width Analysis

For Z-width evaluation, stability is determined using the passivity criterion commonly used in renderable impedances in sampled haptic systems [16]. In this paper, we use a numerical automated MATLAB method to compute the

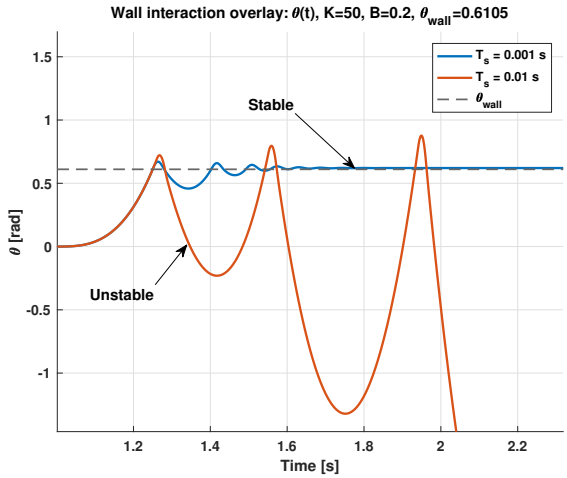


Fig. 8: For the same wall parameters and input force, the 1 ms sampling response converges to wall, while the 10 ms response becomes unstable and leads to divergent motion.

cumulative interaction energy at the human-device interface,

$$E(t) = \int_0^t P(t)\,d\tau$$
$$P(t) = \tau_{\text{ext}}(t)\omega(t) \tag{14}$$

where the human-device interface is represented by the applied external user torque $\tau_{\text{ext}}(t)$ and measured velocity $\omega(t)$, giving instantaneous power $P(t)$.

For each (B_{wall}, K_{wall}) coordinate and sampling time T_s, the interaction is determined to be passive if the energy does not drop below zero,

$$\min_{t\in[0,T]} E(t) \geq 0 \tag{15}$$

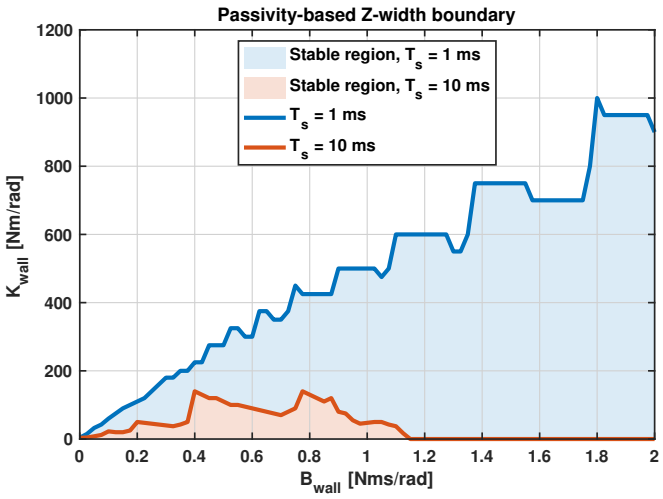


Fig. 9: Z-width boundary showing that faster sampling greatly increases the passive stiffness-damping region.

where simulation time T=5 s to allow for transient behaviour to settle.

Practically, the evaluation is done by running successive iterative simulations, performing the passive-stability check and saving the result to obtain the Z-width boundary map in Figure 9. Following the Z-width definition in [16], the boundary is reported as the maximum passively renderable stiffness for each damping value. Numerically, stiffness is increased from $K_{\text{wall}} = 0$ until the first passivity violation, and the prior value is recorded as $K_{\max}(B)$.

By applying this criterion, a substantial difference in stable region size for the T_s^{fast} and T_s^{slow} cases is observed, where T_s^{fast} remains passive for a much larger region of the KB plane while the T_s^{slow} case shows active (non-passive) energy generation at lower stiffness and damping values than the alternative case (see Figure 9). This is consistent with the convergence and divergence observed for the T_s^{fast} and T_s^{slow} cases respectively in Figure 8 and with the sampled-delay phenomenon that causes the discrete wall to become active, as discussed in Section IV-B. The smaller stable region at 10 ms results from sampling delay in the virtual-wall loop, rather than a fixed limit of the HRX-1 system. Faster sampling improves stability and passivity-based interaction control may provide further improvement but was not evaluated here.

V. Conclusion

This paper presented a digital dynamic model of the HRX-1 haptic robot with a cascaded position-torque control architecture and a virtual wall environment in parallel implemented in MATLAB. The model was discretised to emulate real system control loop sample rates and shared online [7].

Stability of the haptic interaction was explored using Z-width analysis. Results demonstrate that with increasing sample time, Z-width is significantly reduced. For T_s= 1 ms, the passive region covered a substantially larger portion of the tested stiffness–damping range, whereas for T_s= 10 ms, the passive region was significantly smaller due to energy injection into the system, as discussed in Section IV.

The modelling methods and results demonstrate a clear, extensible and modular architecture for haptic interface development and testing while highlighting the strong effect of discrete-time delay on haptic stability and rendering of high-stiffness virtual environments. This study evaluates the simulation framework and does not validate model fidelity against the physical HRX-1 where physical hardware validation remains future work.

Disclosure of Interests

J. Brown and I. Farkhatdinov are co-founders of Human Robotix Ltd.

References

[1] S. Haddadin, E. Croft *et al.*, "Physical human–robot interaction," in *Springer Handbook of Robotics*, 2nd ed., B. Siciliano and O. Khatib, Eds. Cham: Springer, 2017, pp. 1835–1874.

[2] A. Bicchi and G. Tonietti, "Fast and "soft-arm" tactics: Dealing with the safety–performance trade-off in robot arms design and control," *IEEE Robot. Autom. Mag.*, vol. 11, no. 2, pp. 22–33, 2004.

[3] N. Hogan, "Impedance control: An approach to manipulation," *J. Dyn. Syst. Meas. Control Trans. ASME.*, vol. 107, no. 1, pp. 1–24, 1985.

[4] H. Kazerooni, "Human–robot interaction via the transfer of power and information signals," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 2, pp. 450–463, 1990.

[5] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *IEEE/RSJ IROS*. IEEE, 2004, pp. 2149–2154.

[6] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *IEEE/RSJ IROS*, 2012, pp. 5026–5033.

[7] "HRX-1 dynamic model," https://github.com/aleccsseo/HRX-1-Dynamic-Model, [Accessed 22-06-2026].

[8] M. Cecamore, I. Gaponov, S. C. Miller, and I. Farkhatdinov, "Design and Validation of a Haptic Ankle Platform for Physical Human-Machine Interaction," in *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*. IEEE Computer Society, 2024, pp. 1301–1306.

[9] J. Brown, A. Yurkewich, M. Zardbani, L. Cazenave, E. Burdet, and I. Farkhatdinov, "Measuring Muscle Activation with the HRX-1 Wrist Manipulation Robot," in *2025 International Conference On Rehabilitation Robotics (ICORR)*. IEEE, 5 2025, pp. 1047–1052.

[10] B. Berret, D. Verdel, E. Burdet, and F. Jean, "Co-contraction embodies uncertainty: An optimal feedforward strategy for robust motor control," *PLOS Computational Biology*, vol. 20, no. 11, p. e1012598, 11 2024.

[11] L. Cazenave, M. Einenkel, A. Yurkewich, S. Endo, S. Hirche, and E. Burdet, "Hybrid Robotic and Electrical Stimulation Assistance Can Enhance Performance and Reduce Mental Demand," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 4063–4072, 2023.

[12] I. Farkhatdinov, J. Brown, A. Yurkewich, and E. Burdet, "Advancing human sensorimotor research through modular robotics: the case of the HRX-1 ," *Philos. Trans. R. Soc. B*, vol. 381, no. 1958, 2026.

[13] A. Dutta, K. Hatjipanagioti, M. Newsham, L. Leyland, L. Rickson, A. Buchanan, I. Farkhatdinov, J. Twamley, and A. Das, "Co-design and usability evaluation of an extended reality biofeedback platform for functional upper limb weakness," *JMIR XR and Spatial Computing (JMXR)*, 2025.

[14] X. Zhang, M. Cecamore, M. L. U. M. Recaman, T. Su, V. McClelland, and I. Farkhatdinov, "Validation of the HRX-1 wrist human-robot interface in passive and active sensorimotor tasks with eeg recording," in *2026 IEEE Conference on Telepresence*. IEEE, 2026.

[15] M. Graoui, R. Clarke, M. Cecamore, A. Pavlov, L. Lindenroth, and I. Farkhatdinov, "Design of a modular transmission to increase torque capacity of the HRX-1 robot for haptics and rehabilitation research," in *2026 IEEE Conference on Telepresence*. IEEE, 2026.

[16] J. E. Colgate and J. M. Brown, "Factors Affecting the Z-Width of a Haptic Display," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, San Diego, CA, 1994, pp. 3205–3210.