2024 Tenth Indian Control Conference (ICC)
9 - 11 December, 2024
IISER Bhopal, India

A Real-Time Hardware-in-the-loop Framework Using Software-Defined Radios for Emulating Re-configurable Networked Control Systems

Ayyappadas Rajagopal[1], *Student Member, IEEE*, and Shaikshavali Chitraganti[2], *Senior Member, IEEE*

***Abstract*— This experimental work introduces a real-time hardware-in-the-loop framework using software-defined radios, serving as a testbed for implementing networked control systems with real hardware. This work also considers an artificial channel modeler to emulate time varying channel uncertainties. The overall system enables the remote distribution of modules such as sensors, controllers, and estimators, mimicking real-world setups. One test case features a local plant with a sensor and a remote node containing both an estimator and a controller. This framework bridges the gap between theoretical research and practical implementation, providing a robust platform for pre-deployment testing.**

I. INTRODUCTION

Networked Control Systems (NCS) are expanding rapidly with the advent of latest technologies, especially with low-latency wireless communication standards, finding applications in areas like military, agriculture, unmanned navigation, and area surveillance. While providing benefits like teleoperation, scalability, and modularity, NCS are also susceptible to impairments, such as wireless channel distortions, which can result in packet drops [1], [2]. For example, in [3], the authors addressed intermittent measurement availability, while [4] examined state estimation combined with control despite packet drops. In [5], the authors tackled a remote estimation problem using an adaptive power strategy to transmit data over a time-varying packet-drop channel. More recently, [6] explored a recursive Bayesian estimation problem for linear systems with state-dependent packet dropouts. While most of the aforementioned works are primarily theoretical, effective testing of developed algorithms requires reliable testbeds in addition to simulations. Although some experimental studies exist, they are significantly fewer than theoretical ones, and comprehensive hardware testing under real-world channel conditions has been limited. For example, [7] describes a real-time HIL system for rotary inverted pendulums using deep Q-networks, but it lacks network communication. Similarly, [8] presents an inverted pendulum testbed with vision-based control and ethernet based network. Also, [9] covers a 2-DOF robot control HIL system with recurrent neural networks (RNNs), though it is not specifically an NCS. Other related HIL frameworks for NCS can be found in [10]- [12]. However, these works either do not fully utilize an HIL framework specific to NCS or rely on existing networks, which motivated our approach.

*This work was supported in part by *IIT Palakkad Technology IHub Foundation Technology Development Grant IPTIF/TD/IP/001.*

The authors are affiliated with the Department of Electrical Engineering, Indian Institute of Technology Palakkad, 678623, India. 122004004@smail.iitpkd.ac.in, shaik@iitpkd.ac.in

In contrast to the aforementioned results, our work introduces a novel HIL system with networking capabilities that can fully emulate an NCS, incorporating a channel condition emulator to emulate realistic environments. We also propose a theorem based on SINR for state estimation. This system provides an extensive analysis under varying channel conditions, providing insights into the performance and robustness of the actual system before deployment.

II. PROBLEM DESCRIPTION

A. Linear Discrete-time Systems

Consider a discrete-time stochastic linear system in closed-loop setting as

$$x_{k+1} = Ax_k + Bu_k + w_k \tag{1a}$$

$$y_k = Cx_k + v_k, \tag{1b}$$

where $k \in \mathbb{N}$ is the time index, $x_k \in \mathbb{R}^n$ is the state, $y_k \in \mathbb{R}^m$ is the output, $u_k \in \mathbb{R}$ is the control input, $w_k \in \mathbb{R}^n$ and $v_k \in \mathbb{R}^m$ are the process noise and measurement noises with covariance matrices $Q \geq 0$ and $R > 0$, respectively. The pairs (A, B) and $(A, \sqrt{Q})$ are assumed to be controllable, and (A, C) observable. All measurements are assumed to be encapsulated into fixed-length packets and transmitted using appropriate modulation schemes. The closed-loop system comprises a plant with sensors and actuators, and a remote estimator-controller node connected via wireless channels as shown in Fig. 1. Now, the measurement and control data

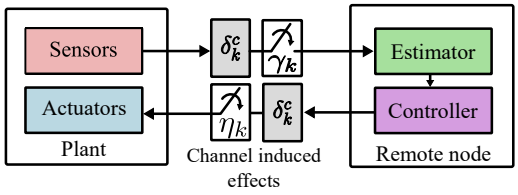


Fig. 1: System model

arrivals are represented by Bernoulli rvs γ_k and η_k as

$$\gamma_k = \begin{cases} 1, & \text{if measurements available at estimator} \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

$$\eta_k = \begin{cases} 1, & \text{if control available at the actuator} \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

with respective probabilities of success p_γ and p_η.

Remark 1. For the analysis, we assume a TCP-like protocol with acknowledged packet drops. This assumption supports the classic separation principle, allowing the controller and estimator to be designed independently [4].

B. Optimal State Estimation with Packet Drops

Now, the state estimates and error covariance equations can be written in recursive form as

$$\hat{x}_k^- = A\hat{x}_{k-1} + \eta_{k-1}Bu_{k-1} \tag{4}$$
$$P_k^- = g(P_{k-1}) \tag{5}$$
$$\hat{x}_k = \hat{x}_k^- + \gamma_k L_k z_k \tag{6}$$
$$P_k = h(P_k^-), \tag{7}$$

where $\hat{x}_k^-$ is state update and P_k^- is error covariance update for the k^{th} packet with $g(X) = AXA' + Q$. Also, $z_k = y_k - C\hat{x}_k^-$ and $h(X) = X - L_kCX$ with $L_k = P_k^-C' \times (CP_k^-C' + R)^{-1}$.

C. Optimal LQR Controller

A Linear Quadratic Regulator (LQR) controller is used to achieve optimal control by minimizing a cost function

$$J_k = \mathbb{E}[x_N'W_Nx_N + \sum_{k=0}^{N-1}(x_k'W_kx_k + \eta_k u_k'U_ku_k)|\mathbf{u}^{k-1}, x_0, P_0],$$

where $\mathbf{u}^{k-1} = \{u_{k-1}, u_{k-2}, \ldots, u_0\}$, $W_k \geq 0$ and $U_k > 0$ are assumed to be symmetric and known. The optimal control for this cost function can easily be obtained using the dynamic programming principle as follows

$$u_k = -K\hat{x}_k, \tag{8}$$

with controller gain $K = (B'S_{k+1}B + U_k)^{-1}B'S_{k+1}A$, where $S_k = A'S_{k+1}A + W_k - \eta_k A'S_{k+1}BK$.

D. Linear State Estimation and Control in the Presence of Channel dependent drops

The packet drops will depend on the SNR or in turn the channel condition, as explained in the following.

Theorem 1. *For system (1), let measurement and control packets modulated with suitable QPSK waveforms and traversing through an AWGN channel $\mathcal{N}(0, N_0)$ with arrivals based on (2) and (3), respectively. Now, for given values of* $\mathrm{SNR_m}$ *and* $\mathrm{SNR_c}$*, estimated states and covariance are*

$$\hat{x}_k^- = A\hat{x}_{k-1} + \eta_{k-1}Bu_{k-1} \tag{9}$$
$$P_k^- = g(P_{k-1}), \tag{10}$$
$$\hat{x}_k = \hat{x}_k^- + \gamma_k L_k z_k \tag{11}$$
$$P_k = h_\gamma(P_k^-), \tag{12}$$

with

$$\mathbb{E}[\gamma_k] = p_\gamma = 1 - 2\mathcal{Q}(\sqrt{\mu\mathrm{SNR_m}}),$$
$$\mathbb{E}[\eta_k] = p_\eta = 1 - 2\mathcal{Q}(\sqrt{\mu\mathrm{SNR_c}}),$$

where $\mathrm{SNR_m}$ *and* $\mathrm{SNR_c}$ *represent the SINRs for the measurement and control channels, respectively.*

Proof. Following similar lines of proof as in [3], [4] the estimation equations (9)-(12) can be directly obtained. Now, with the assumption of AWGN channel with QPSK modulation, the symbol error rate is given by $\mathrm{SER} = 2\mathcal{Q}(\sqrt{\mu\mathrm{SNR}})$. The detector is equipped with error detection mechanisms, ensuring that any error is detectable. When an error is detected, the packet is discarded. Therefore, the symbol error rate is equivalent to the probability of packet drops $Pr(\gamma_k = 0) = 2\mathcal{Q}(\sqrt{\mu\mathrm{SNR_m}})$ Now, the probability of arrival or the expected arrival rate for a fixed value of $\mathrm{SNR_m}$ can be written directly. □

III. Numerical Simulations

To validate the mathematical formulations, a linear discrete model of the 'Qube-servo 3' was simulated using MATLAB, chosen for its fundamental role in control systems and wide range of real-world applications. Linear discretized system matrices with a sampling time of 0.1 seconds result in (1), where the states are angular position and angular velocity, denoted as θ, ω. The system matrices are defined accordingly as

$$A = \begin{bmatrix} 1.0000 & 0.0665 \\ 0 & 0.3309 \end{bmatrix},\ B = \begin{bmatrix} 0.7965 \\ 15.9307 \end{bmatrix},\ C = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{13}$$

Noise vectors w_k and v_k are assumed to have respective covariances of $Q = 0.0001I$ and $R = 0.0001$. For the simulation, the control channel is assumed to be perfect. Also, for the LQR controller, the gain is found to be $K = [0.0597, 0.0237]$ with corresponding weighting matrices as

$$W_k = \begin{bmatrix} 0.0001 & 0 \\ 0 & 0.0001 \end{bmatrix}, U_k = \begin{bmatrix} 0.0001 \end{bmatrix}. \tag{14}$$

Now, Fig. 3 displays both the actual and estimated states of the servo system, demonstrating its temporal response. This stabilization was achieved by the optimal control applied. The reference position was set at 45^o and the average steady state error was found to be 0.1^o.

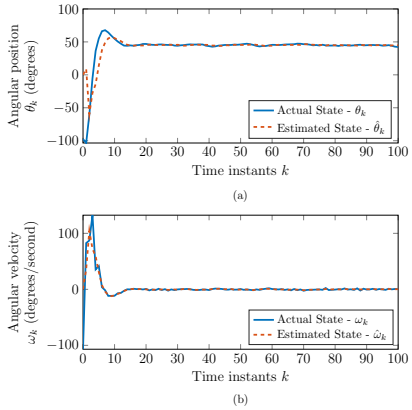


Fig. 3: Actual states and estimated states - one sample realization of (a) Position, (b) Angular velocity

IV. Experimental work

Before detailing the experimental setup, several important aspects related to the setup are discussed in the following.

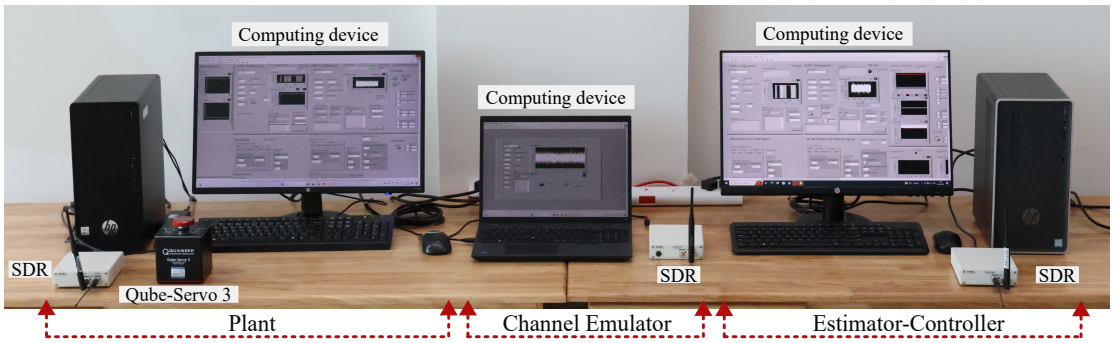


Fig. 2: Developed HIL System

A. Packet Encapsulation

The sensor measurement data is first converted into a binary bit stream and encapsulated into a packet format, as illustrated in Fig. 4. These additional bits facilitate packet recovery and differentiation at the receiver's end.

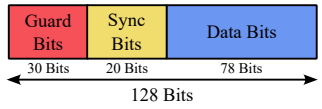


Fig. 4: Packet format for the transmitter and receiver

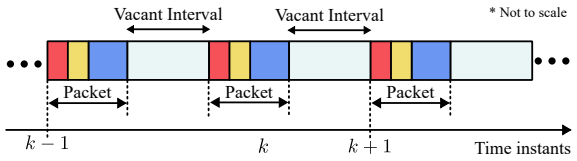


Fig. 5: Packet ordering

Now, to distinguish one packet from another, an additional vacant interval is introduced. As illustrated in Fig. 5, ans a deliberate delay is added after each packet transmission to separate them.

B. Delay Model and Sampling Rate

Delay is crucial in real-time closed-loop systems, arising from sensing, computation, or actuation. It should be minimal relative to the sampling period, as excessive delay reduces the maximum sampling rate and can destabilize the system.

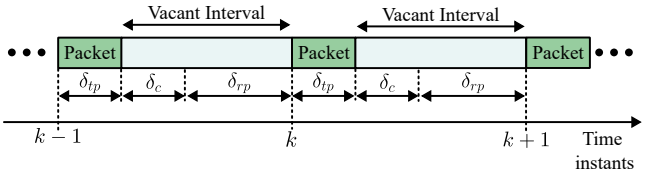


Fig. 6: Delay model used in analysis

The delay model for the system, depicted in Fig. 6, includes three components: transmitter processing delay (δ_{tp}), channel propagation delay (δ_c), and receiver processing delay (δ_{rp}). It is assumed that the round-trip delay for the channel has a maximum value δ_{max}, beyond which any delay is considered a packet drop, as demonstrated in [13].

C. Software defined Radios

SDRs are utilized for wireless communication at both the plant and remote sides, and are fully configurable through software. The configuration parameters for the SDRs are listed in Table I and II. The parameters are configurable and one can find comprehensive information about all the parameters in [14].

TABLE I: Specifications of Tx and Rx SDRs

Parameters	**Transmitter**	**Receiver**
Operating Frequency	1.2 & 2.5 GHz	1.2 & 2.5 GHz
Gain	20 dB	20 dB
Maximum power (P_{out}) / (P_{in})	20 dBm	-15 dBm
Maximum instantaneous real-time bandwidth	56 MHz	56 MHz
Message data type	Time stamped packets	Time stamped packets
Modulation / Demodulation scheme	BPSK	BPSK
Sampling rate	1M samples/s	1M samples/s

D. Channel emulator system

A channel emulator system is employed to introduce controllable distortions for testing the HIL system. As shown in

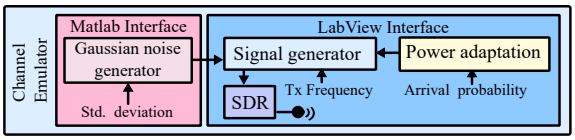


Fig. 7: Channel emulator

Fig. 7, the noise generator block produces Gaussian random variables, which are then used to generate signals with the desired frequency and power. The configuration parameters for the channel modeler are listed in Table II.

E. Experimental implementation of HIL Platform

The HIL framework used in this study, illustrated in Fig. 8, consists of two main nodes. One node acts as the plant, connected to the hardware platform and a sensor. The other node includes the estimator and controller blocks.

TABLE II: Specifications of Emulator SDR

Fading Channel Emulator	
Fading Frequency	1.2 GHz
Gain	20 dB
Maximum output power (P_{out})	40 dBm
Maximum instantaneous real-time bandwidth	56 MHz
Message data type	-NA-
Modulation scheme	-NA-

Two separate wireless channels are employed for transmitting measurement data and control data packets. Now, the actual experimental setup is shown in Fig. 2. The actual and

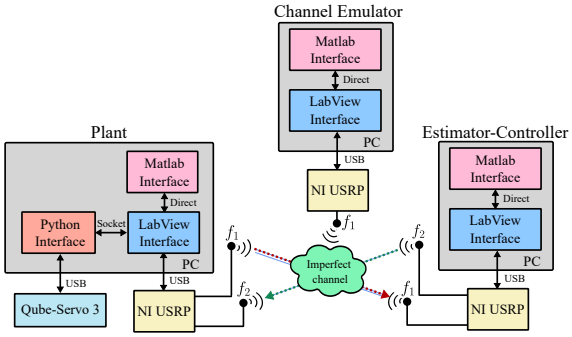


Fig. 8: HIL System Architecture

estimated states obtained using the experimental evaluation is shown in Fig. 9. It indicates that the system is stabilizing in a reasonable amount of time. Similarly, Fig. 10 depicts the

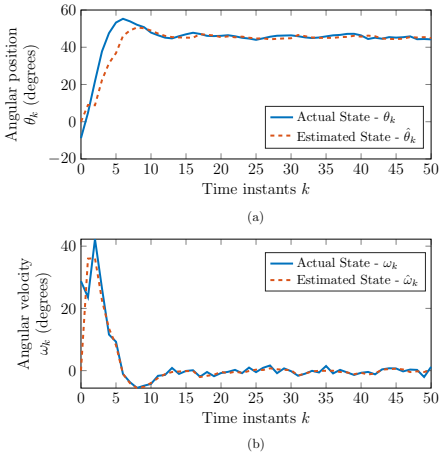


Fig. 9: Experimental results for actual states and estimated states - one sample realization

estimation error for both position and angular velocity. The bounded nature of these errors indicates that the estimation process is operating within acceptable limits.

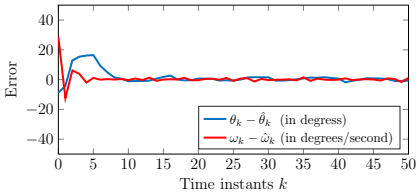


Fig. 10: Experimental results showing estimation errors - one sample realization

V. CONCLUSION

We introduce a novel HIL system for closed-loop testing of real plants from a networked control standpoint. Additionally, a versatile channel emulator is developed to introduce controllable distortions. This re-configurable system allows for comprehensive analysis of algorithms under channel uncertainties, advancing NCS evaluation and optimization.

REFERENCES

[1] F. Y. Wang and D. Liu, "Networked control system," *Theory and Applications*, 2008.

[2] M. M. Hamdan and M. M. Mahmoud, "Analysis and challenges in wireless networked control system: A survey," *International Journal of Robotics and Control Systems*, vol. 2, no. 3, pp. 492–522, 2022.

[3] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.

[4] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, 2007.

[5] J. Chakravorty and A. Mahajan, "Remote estimation over a packet-drop channel with markovian state," *IEEE Transactions on Automatic Control*, vol. 65, no. 5, pp. 2016–2031, 2020.

[6] Q. Liu, Z. Wang, H. Dong, and C. Jiang, "Recursive bayesian estimation for discrete-time systems with state-dependent packet dropouts: A cross-coupled method," *IEEE Transactions on Automatic Control*, vol. 69, no. 6, pp. 3705–3716, 2024.

[7] Y. Dai, K. Lee, and S. Lee, "A real-time hil control system on rotary inverted pendulum hardware platform based on double deep q-network," *Measurement and Control*, vol. 54, no. 3-4, pp. 417–428, 2021.

[8] C. Bachhuber, S. Conrady, M. Schütz, and E. Steinbach, "A testbed for vision-based networked control systems," in *Computer Vision Systems: 11th International Conference, ICVS 2017, Shenzhen, China, July 10-13, 2017, Revised Selected Papers 11*, pp. 26–36, Springer, 2017.

[9] U. Davalos-Guzman, C. E. Castañeda, L. M. Aguilar-Lobo, and G. Ochoa-Ruiz, "Design and implementation of a real time control system for a 2dof robot based on recurrent high order neural network using a hardware in the loop architecture," *Applied Sciences*, vol. 11, no. 3, p. 1154, 2021.

[10] V. Bhambhani, Y. Han, S. Mukhopadhyay, Y. Luo, and Y. Chen, "Hardware-in-the-loop experimental study on a fractional order networked control system testbed," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 9, pp. 2486–2496, 2010.

[11] J. H. Taylor and J. Slipp, "An integrated testbed for advanced wireless networked control systems technology," in *IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society*, pp. 2101–2106, IEEE, 2010.

[12] A. Tzes, G. Nikolakopoulos, and I. Koutroulis, "Development and experimental verification of a mobile client-centric networked controlled system," *European Journal of Control*, vol. 11, no. 3, pp. 229–241, 2005.

[13] V. Gupta, "On the effect of stochastic delay on estimation," *IEEE Transactions on Automatic Control*, vol. 56, no. 9, pp. 2145–2150, 2011.

[14] "NI USRP." `https://www.ni.com/en-in/shop/hardware/products/usrp-software-defined-radio-device.html`, 2023. [Online; accessed 15-June-2023].