Design of Networked Control Systems Using Passivity

Nicholas Kottenstette, *Senior Member, IEEE*, Joseph F. Hall III, *Member, IEEE*,
Xenofon Koutsoukos, *Senior Member, IEEE*, Janos Sztipanovits, *Fellow, IEEE*, and
Panos Antsaklis, *Fellow, IEEE*

Abstract—Real-life cyber physical systems, such as automotive vehicles, building automation systems, and groups of unmanned vehicles are monitored and controlled by networked control systems (NCS). The overall system dynamics emerges from the interaction among physical dynamics, computational dynamics, and communication networks. Network uncertainties such as time-varying delay and packet loss cause significant challenges. This paper proposes a passive control architecture for designing NCS that are insensitive to network uncertainties. We describe the architecture for a system consisting of a robotic manipulator controlled by a digital controller over a wireless network and show that the system is stable even in the presence of time-varying delays. Experimental results demonstrate the advantages of the passivity-based architecture with respect to stability and performance and show that the system is insensitive to network uncertainties.

Index Terms—Automation, cyber-physical system, discrete time systems, industrial control, Lyapunov methods, networked control systems, nonlinear control systems, passivity-based control, robot control, stability analysis, telerobotics, time-varying systems.

I. Introduction

THE heterogeneous composition of computing, sensing, actuation, and communication components has enabled a modern grand vision for real-world cyber physical systems (CPS). Real-world CPS, such as automotive vehicles, building automation systems, and groups of unmanned air vehicles, are monitored and controlled by networked control systems (NCS) and the overall system dynamics emerges from the interaction among physical dynamics, computational dynamics, and communication networks. This paper is inspired by the rapidly increasing use of NCS architectures in constructing real-world CPS that integrate computational and physical devices using

wireless networks, such as medical device networks, groups of unmanned vehicles, and transportation networks. NCS research has recently become a very active area, investigating problems at the intersection of control systems, networking, and computer science [1].

CPS are inherently heterogeneous not only in terms of their components but also in terms of design requirements. Besides functional properties, CPS are subject to a wide range of physical requirements, such as dynamics, power, physical size, and fault tolerance in addition to system-level requirements, such as safety and security. Heterogeneity does not go well with current methods of compositional design. The most important principle used in achieving multiobjective compositionality is the separation of concerns. Separation of concerns works if the design views are orthogonal, i.e., design decisions in one view does not influence design decisions in other views. Unfortunately, achieving compositionality for multiple physical and functional properties simultaneously is a very hard problem because of the lack of orthogonality among the design views.

Fig. 1 represents a simplified model-based design flow of a CPS composed of a physical plant and a NCS. In a conventional design flow, the controller dynamics is synthesized with the purpose of optimizing performance. The selected design platform (abstractions and tools used for control design in the design flow) is frequently provided by a modeling language and a simulation tool, such as MATLAB/Simulink [2]. The controller specification is passed to the implementation design layer through a "specification/implementation interface." The implementation in itself has a rich design flow that is expressed in two layers: system-level design and implementation platform design. The software architecture and its mapping on the (distributed) implementation platform are generated in the system-level design layer. The results—expressed again in the form of architecture and system models—are passed on through the next specification and implementation interface to generate code as well as the hardware and network design. This simplified flow reflects the fundamental strategy in platform-based design [3]. The design progresses along precisely defined abstraction layers. The design flow usually includes top-down and bottom-up elements and iterations (not shown in the figure).

Effectiveness of the platform-based design largely depends on how much the design concerns (captured in the abstraction layers) are orthogonal, i.e., how much the design decisions in the different layers are independent. Heterogeneity causes
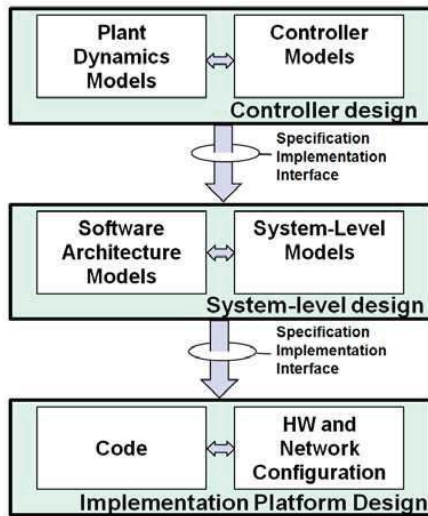
Fig. 1. Simplified CPS design flow.

major difficulties in this regard. The controller dynamics is typically designed without considering implementation side effects (e.g., numeric accuracy of computational components, timing accuracy caused by shared resource and schedulers, time varying delays caused by network effects, etc.) Timing characteristics of the implementation emerge at the confluence of design decisions in software componentization, system architecture, coding, and HW/network design choices. Compositionality in one layer depends on a web of assumptions to be satisfied by other layers. For example, compositionality on the controller design layer depends on assumptions that the effects of quantization and finite wordlength can be neglected and the discrete-time (DT) model is accurate. Since these assumptions are not satisfied by the implementation layer, the overall design needs to be verified after implementation—even worst—changes in any layer may require reverification of the full system.

An increasingly accepted way to address these problems is to enrich abstractions in each layer with implementation concepts. An excellent example for this approach is TrueTime [4] that extends MATLAB/Simulink with implementation-related modeling concepts (networks, clocks, schedulers) and supports simulation of networked and embedded control systems. While this is a major step in improving understanding of the implementation effects, it does not achieve orthogonality across the design concerns. A controller designer can now factor in implementation effects (e.g., network delays), but still, if the implementation changes, the controller may need to be redesigned.

Decoupling the design layers is a very hard problem and typically introduces significant restrictions and/or overdesign. For example, the timed triggered architecture orthogonalizes timing, fault tolerance, and functionality, but it comes at the cost of strict synchrony and static structure [5]. An approach for decoupling between the control design and implementation layers has been proposed recently in [6]. The approach allows the design of state-feedback controllers that minimize a quadratic performance bound for a given level of timing jitter using linear matrix inequality methods.

Our aim is to address fundamental problems caused by networks effects, such as time-varying delay, jitter, limited bandwidth, and packet loss in NCS. To deal with these implementation uncertainties, we propose a model-based design flow on top of passivity, a very significant concept from systems theory [7]. The inherent safety that passive systems provide is fundamental in building systems that are insensitive to implementation uncertainties. Passive systems have been exploited for the design of diverse systems such as smart exercise machines [8], teleoperators [9]–[14], digital filters [15], NCS [16]–[18], and complex nonlinear thermal- and chemical-based processes [19]–[21].

Our approach advocates a concrete and important transformation of model-based methods that can improve orthogonality across the design layers and facilitate compositional component-based design. By using multirate data sample-and-hold elements and imposing passivity constraints on component dynamics, the stability of the NCS is guaranteed in the presence of time delays and packet loss. This separation of concerns empowers the model-based design process to be applied for NCS. Information about the network effects needs not to be considered at the controller design layer because the theoretical guarantees for stability are independent of the networking implementation uncertainties.

The primary contributions of this paper are as follows:

1) we present a passive control architecture for a system consisting of a robotic manipulator controlled by a *digital controller* over a wireless network;
2) we provide analytical results that prove that our architecture ensures stability of the NCS in the presence of time-varying delays assuming that the communication protocols do not process duplicate transmissions;
3) we implement the passive control architecture on an experimental NCS consisting of two computer nodes that realize the robotic manipulator and the digital controller, respectively, and communicate over an ad hoc 802.1 lb wireless network subject to additional traffic induced by disturbance nodes;
4) we present experimental results that demonstrate the stable operation of the system in the presence of severe time-varying delays caused by network traffic generated by the disturbance nodes or by excessive computational load competing with the controller.

The work presented in this paper demonstrates that passivity can be exploited to account for the effects of network uncertainties, thus improving orthogonality across the controller design and implementation design layers and empowering model-driven development. Part of this paper has been presented in [22]. The main extensions are: 1) experimental implementation and evaluation of the passivity-based architecture using a NCS; 2) detailed design of the digital passive controller; and 3) theoretical analysis that includes the proofs of passivity and stability for the proposed architecture. It should be noted that passive structures offer additional advantages for robustness to finite length representations and saturation [15], but this paper focuses on network effects which is one of the most significant concerns in the development of CPS.

The rest of this paper is organized as follows. An overview of related work is presented in Section II followed by a summary of notation used and passivity definitions in Section III. Section IV presents the passive control architecture focusing on the technical details required for implementation. Analysis of our proposed NCS is provided in Section V. Section VI describes the implementation and presents detailed experimental results. Finally, Section VII presents the main directions of our future work.

II. RELATED WORK

Our overall approach to designing NCS that can tolerate time-varying delay and data loss is constructive in nature and relies on *passivity*-based networked control fundamentals [16]. Constructive approaches typically rely on a system or controller to be restricted to a given *sector* in order for the overall system trajectories to remain stable [23]–[25]. When passive systems are connected in either a parallel or negative feedback arrangement, the resulting system remains passive [23], [24]. In addition, the observable state variable trajectories for a passive system are Lyapunov-stable [25]. When time delays are introduced into a feedback arrangement involving two passive systems, the overall passive system structure is lost and its corresponding state trajectories may become unstable. As a result, those in the telemanipulation community who wanted to preserve stability for arbitrary fixed time-delays proposed using *wave variables* [26], [27]. Wave variables were originally introduced by Fettweis in order to circumvent the problem of delay-free loops and guarantee that the implementation of wave digital filters is both stable and realizable [15]. The wave variables that resulted from a bilinear scattering transformation allowed for a stable minimumphase continuous-time (CT) system to be mapped to a stable minimum-phase DT system. Stability is guaranteed because the wave variables allowed for the primitive DT components derived from their passive CT counterparts to remain passive while allowing for the overlying CT networking structure to remain unaltered so as to preserve stability. The use of wave variables for networked control has continued to advance.

In particular, [18] provides constructive conditions for CT plants and controllers that are interconnected with wave variables derived from a *generalized scattering transform* in order to maintain L_2^m-stability when subject to fixed time delays. It is further asserted that the results presented in [18] apply for the case in which the wave variables transmitted between the two CT systems are first compressed and converted to a DT wave variable, then transmitted over a network, received, and finally decompressed back to a CT wave variable. Additional details on compression/decompression techniques for wave variables are described in [14], [17], and [28]. Both [10], [17], [29] have shown that the DT wave variables can tolerate both arbitrary fixed delays and data loss in network control architectures. Finally, it has been shown that network control architectures consisting of DT wave variables can be subject to time-varying delays as long as duplicate data packets are not processed at the receiver [10], [16], [17].

The precise definitions chosen for passivity shall be presented from the input–output perspective similar to the definition for positive systems given in [23]. Systems that will satisfy this input–output passivity definition include positive real and dissipative dynamical systems [30]. When a dissipative dynamical system can be described by a Hamiltonian (the sum of kinetic and potential energy, $\mathcal{H} = \mathcal{T} + \mathcal{V}$), a passive mapping typically exists in which the Hamiltonian serves as storage function (β) [30]. This will be clearly illustrated in our discussion of the passive structure of robotic systems in relating the joint velocities to their corresponding motor torques. However, there are some limitations with the study of passive systems. For example, systems that consist of cascades of passive systems (such as two integrators in series) are not necessarily passive systems.

The conditions for stability in our networked digital control system require the digital controller and CT plant to be strictly output-passive. As a result, this limits us to initially controlling the velocity output of a robot in order to indirectly control its corresponding position. Such indirect control frameworks can be subject to position drift and require an additional drift-compensation algorithm such as those described in [31]. However, using the notion of a *passivity index* [19], we demonstrate how to design low-complexity analog filters to place in parallel with an asymptotically stable minimum-phase lineartime invariant stable system in order to render the combined system strictly output passive. Such an architecture allows us to achieve steady-state position control in our proposed framework. Finally, we have recently shown that our proposed framework is applicable to the control of a larger class of Lyapunov-stable systems that possess the same number of inputs and outputs and are interior conic or equivalently inside the sector $[a, \infty]$ in which $|a| < \infty$ [32]. Certain classes of stable non-minimum-phase systems are inside the sector $[a, \infty]$ in which $a < 0$ and can be controlled in our proposed framework.

III. PRELIMINARIES

This section presents the passivity and stability definitions as well as the notation used in this paper. We use the following notation for CT systems:

$$\langle G(u), u \rangle_{NT_s} \triangleq \int_0^{NT_s} G(u(t))^{\mathrm{T}} u(t) dt \quad (CT \text{ inner product})$$

$$\langle G(u), G(u) \rangle_{NT_s} \triangleq \|(G(u))_{NT_s}\|_2^2$$

in which $\|(G(u))_{NT_s}\|_2^2$ is the truncated squared *CT* 2-norm of the function $G(u)$. Similarly, we use the following notation for DT systems:

$$\langle G(u), u \rangle_N \triangleq \sum_{i=0}^{N-1} G(u[i])^{\mathrm{T}} u[i] \quad (DT \text{ inner product})$$

$$\langle G(u), G(u) \rangle_N \triangleq \|(G(u))_N\|_2^2$$

in which $\|(G(u))_N\|_2^2$ is the truncated *DT* squared 2-norm of the function $G(u)$. Note that, in order to distinguish *CT* from *DT* systems, the integral is taken to the limit NT_s where T_s is a real nonnegative number, while the summation is taken to $N - 1$ in which $N \in \{1, 2, \ldots, \}$.

We also denote $L_{2_e}^m(U)$ as the *extended* L_2^m space for the function $u(t) \in U$ in which $U \subset \mathbb{R}^m$ is the set of all possible functions which satisfies

$$\|(u)_{NT_s}\|_2^2 < \infty \text{ for a given } NT_s \geq 0.$$

In the limit as $NT_s \to \infty$, then $u \in L_2^m(U)$ is any function that satisfies

$$\int_0^\infty u^T(t)u(t)dt < \infty, \text{ or more compactly, } \|u\|_2^2 < \infty.$$

For the *DT* case, we denote $l_{2_e}^m(U)$ as the *extended* l_2^m space for the function $u[i] \in U$ in which $U \subset \mathbb{R}^m$ is the set of all possible functions which satisfies

$$\|(u)_N\|_2^2 < \infty \text{ for a given } N.$$

In the limit as $N \to \infty$, then $u \in l_2^m(U)$ is any function that satisfies

$$\sum_{i=0}^\infty u^T[i]u[i]dt < \infty, \text{ or more compactly, } \|u\|_2^2 < \infty.$$

Note also that $L_2^m(U) \subset L_{2_e}^m(U)$ for the *CT* case and $l_2^m(U) \subset l_{2_e}^m(U)$ for the *DT* case.

Let: 1) $\mathcal{X}_2^m(U)$ ($\mathcal{X}_{2_e}^m(U)$) denote the $L_2^m(U)$ and $L_{2_e}^m(U)$ spaces for the *CT* case, and $l_2^m(U)$ and $l_{2_e}^m(U)$ spaces for the *DT* case, respectively; and 2) $\mathcal{T}$ denote NT_s in which $NT_s \geq 0$ for the *CT* case and N such that $N \in \{1, 2, \ldots, \}$ for the *DT* case. The corresponding definitions of passivity for the *CT* and *DT* case can be described as follows.

Definition 1 ([7], [33], [34]): Let $G : \mathcal{X}_{2_e}^m(U) \to \mathcal{X}_{2_e}^m(U)$ then for all $u \in \mathcal{X}_{2_e}^m(U)$ and all $\mathcal{T}$: I.

1) G is *passive* if there exists a constant β such that (1) holds

$$\langle G(u), u \rangle_{\mathcal{T}} \geq -\beta. \tag{1}$$

2) G is *strictly output passive* if there exists some constants β and $\epsilon > 0$ such that (2) holds

$$\langle G(u), u \rangle_{\mathcal{T}} \geq \epsilon \|(G(u))_{\mathcal{T}}\|_2^2 - \beta. \tag{2}$$

3) G is *strictly input passive* if there exists some constants β and $\upsilon > 0$ such that (3) holds

$$\langle G(u), u \rangle_{\mathcal{T}} \geq \upsilon \|(u)_{\mathcal{T}}\|_2^2 - \beta. \tag{3}$$

Definition 2 ([33] Definition 1.2.3 [34]): Let $G : \mathcal{X}_{2_e}^m(U) \to \mathcal{X}_{2_e}^m(U)$ is said to be L_2^m-*stable* (l_2^m-*stable*) if

$$u \in \mathcal{X}_2^m(U) \implies y = G(u) \in \mathcal{X}_2^m(U)$$

and G is said to have *finite-L_2^m-gain* (*finite-l_2^m-gain*) if $\exists \gamma_q > 0, \beta_q$ s.t. for all $\mathcal{T}$

$$u \in \mathcal{X}_{2_e}^m(U) \implies \|(y)_{\mathcal{T}}\|_2 \leq \gamma_q \|(u)_{\mathcal{T}}\|_2 + \beta_q.$$

Any $G : \mathcal{X}_{2_e}^m(U) \to \mathcal{X}_{2_e}^m(U)$ which has *finite-L_2^m-gain* (*finite-l_2^m-gain*) is L_2^m-*stable* (l_2^m-*stable*).

The following theorem for CT systems is used in the proof of our main result (Theorem V-A) in which it is shown that the network control system architecture depicted in Fig. 2 is strictly output passive for any passive robot (plant).

Theorem 1 [33, Th. 2.2.14]: Let $G : L_{2_e}^m(U) \to L_{2_e}^m(U)$ be strictly output passive. Then G has *finite L_2^m-gain*.[1]

For an asymptotically stable LTI system $G : L_{2_e}^m(U) \to L_{2_e}^m(U)$ whose transfer function is denoted $G(s)$, a frequency-dependent measure known as the *passivity index* $\upsilon_F(G(s), \omega) = -(1/2)[G(j\omega) + G(-j\omega)]$ is defined such that $G(s) + \upsilon_F(G(s), \omega)$ is positive real (equivalently passive) [19].[2] If the passivity index is negative for all $\omega \in \mathbb{R}$, then $G(s)$ is strictly input passive such that $(G(s) + \upsilon)$ is positive real in which $\upsilon = \omega \in \mathbb{R} \sup \upsilon_F(G(s), \omega)$. In addition, if $G(s)$ is both asymptotically stable and strictly input passive, then it is strictly output passive [30, Prop. 5.2-x].

IV. PASSIVE CONTROL ARCHITECTURE

This section presents the proposed architecture depicted in Fig. 2. The NCS consists of a robotic system controlled by a *DT* controller through a wireless network. The robotic system is denoted by $G_{\text{robot}} : \tau(t) \mapsto \dot{\Theta}(t)$, where $\tau(t)$ is the input torque vector, and Θ and $\dot{\Theta}$ are the joint angles and angular velocities, respectively. The robotic system contains components for gravity compensation denoted by $g(\Theta)$ and negative velocity feedback denoted by ϵI. The robotic system is controlled by a multiple-input multiple-output (MIMO) digital controller that receives as input the velocity error $\dot{e}[i]$ and computes the torque vector $\tau_{\text{uc}}[i]$. The controller also employs the *inner product equivalent sampler* (*IPES*, T_s) and zero-order-hold (*ZOH*, T_s) blocks (as a pair we denote them as *IPESH*) required by our analysis, which relate the *DT* variables $(\dot{\Theta}_{sr}[i], \tau_{\text{uc}}[i])$ to the respective *CT* variables $(\dot{\Theta}_{sr}(t), \tau_{\text{uc}}(t))$. The communication between the robotic system and the controller is realized using wave variables. The bilinear transformation used to relate the torque and angular velocity vectors to the wave variables is denoted by b. In order to transmit the information from the robotic system, the *CT* wave variable $u_p(t)$ is converted to a *DT* wave variable $u_p[i]$ using a *wave variable sampler* denoted by (WS, T_s). Similarly, to transmit information from the digital controller, the *DT* wave variable $\upsilon_{ucd}[i]$ is converted to a *CT* wave variable $\upsilon_{ucd}(t)$ using a *wave variable hold* denoted (WH, T_s). Finally, the figure depicts a wireless network link with *DT* varying nonnegative integer delays denoted by $z^{-p(i)}$ and $z^{-c(i)}$ such that $u_{pd}[i] = u_p[i - p(i)]$ and $\upsilon_{ucd}[i] = \upsilon_{uc}[i - c(i)]$. In the remaining parts of this section, we define in detail all components used in the architecture of Fig. 2 and explain the design decisions necessary to ensure passivity.

A. Robotic System

Our control strategy takes advantage of the passive structure of a robotic system [37]. The robot dynamics, denoted by $G_{\text{robot}}(\tau(t))$ in Fig. 2, is described by

$$\tau = M(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta})\dot{\Theta} + g(\Theta). \tag{4}$$

---

[1] We only require the *CT* result, however, the analogous *DT* result is presented in [35, Th. 2].

[2] For the multi-input multi-output (MIMO) case, $\upsilon_F(G(s), \omega) = -1/2\lambda_{\min}[G(j\omega) + G^T(-j\omega)]$ in which $\lambda_{\min}[M]$ denotes the minimum eigenvalue of the matrix M. We only require the *CT* definition, however, the analogous *DT* definition is presented in [36].
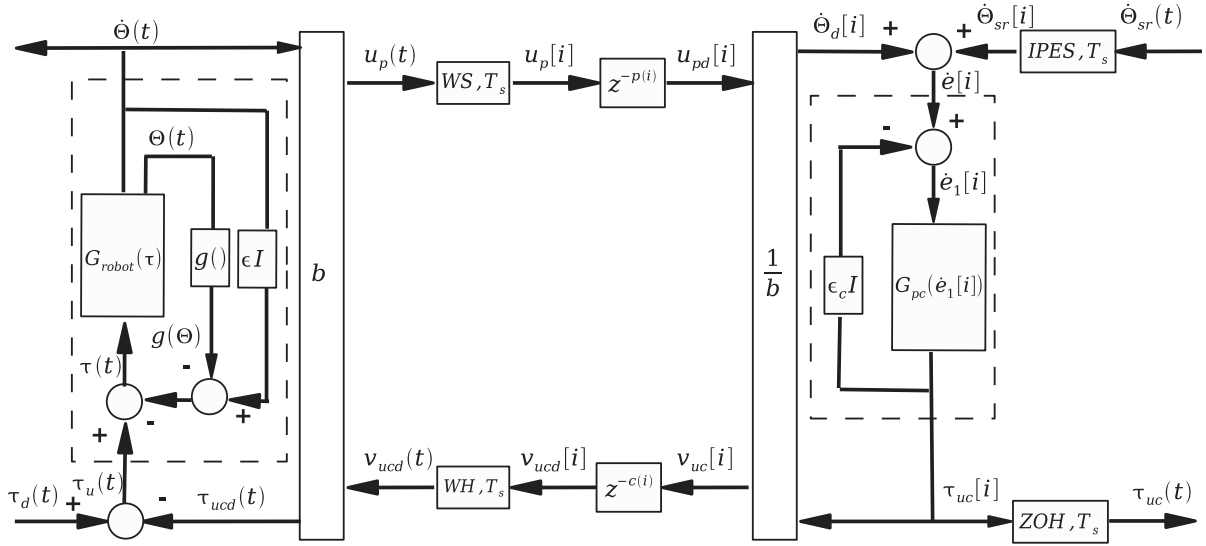
Fig. 2. Proposed wireless digital control architecture.

The state variables Θ denote the robot joint angles, τ is the input torque vector, $M(\Theta)$ is the mass matrix, $C(\Theta, \dot{\Theta})$ is the matrix of centrifugal and Coriolis effects, and $g(\Theta)$ is the gravity vector. The inertia matrix $M(\Theta) = M(\Theta)^{\mathrm{T}} > 0$ and the matrix C and $\dot{M}$ are related as follows:

$$-(\dot{M} - 2C) = (\dot{M} - 2C)^{\mathrm{T}} \implies x^{\mathrm{T}}(\dot{M} - 2C)x = 0, \ \forall x \in \mathbb{R}^n. \tag{5}$$

The skew-symmetric constraint in (5) is necessary to prove that the robot is passive.

Despite the complexity of robotic manipulators, simple control laws can be used in a number of cases. A fundamental consequence of the passivity property is that a simple independent joint *CT* proportional-derivative (PD) control can achieve global asymptotic stability for set-point tracking in the absence of gravity [38]. Our architecture uses a *DT* equivalent implementation of a PD controller that communicates with the robotic system via a wireless network. To compensate for gravity, we select as the control command $\bar{\tau}_u = \tau - g(\Theta)$. Then by choosing the supply rate $s(\bar{\tau}_u(t), \dot{\Theta}(t)) = \dot{\Theta}^{\mathrm{T}}(t)\bar{\tau}_u(t)$ and the corresponding storage function $V(\dot{\Theta}(t)) = (1/2)\dot{\Theta}^{\mathrm{T}}(t)M(\Theta(t))\dot{\Theta}(t)$, the robot can be shown to be a *lossless* passive system in which all supplied energy is stored as kinetic energy and the following relationship is satisfied: $\int_0^{NT_s} \dot{\Theta}(t)^{\mathrm{T}}\bar{\tau}_u(t)dt = (V(\dot{\Theta}(NT_s)) - V(\dot{\Theta}(0)))$ [30].[3] Because $V(0) = 0$, all feasible input trajectories $\bar{\tau}_u(t)$ in which $\dot{\Theta}(0) = \dot{\Theta}(NT_s) = 0$ result in the lossless expression $\int_0^{NT_s} \dot{\Theta}(t)^{\mathrm{T}}\bar{\tau}_u(t)dt = 0$. We shall use $\int_0^{NT_s} \dot{\Theta}(t)^{\mathrm{T}}\bar{\tau}_u(t)dt \geq -V(\dot{\Theta}(0))$ as our passivity expression for our robot in which $V(\dot{\Theta}(0))$ represents all the *available storage* energy that can be extracted from the robot at time $t = 0$.

For simplicity of discussion, we assume there is no friction at the robot's joints. We then render the robot strictly output passive by including negative velocity feedback [16] such that the control command τ_u shall be of the following final form: $\tau_u = \bar{\tau}_u + \epsilon\dot{\Theta} = \tau - g(\Theta) + \epsilon\dot{\Theta}, \ \epsilon \geq 0$. Direct substitution

of $\bar{\tau}_u = \tau_u - \epsilon\dot{\Theta}$ in the aforementioned passivity expression results in the strictly output passive expression (7) for $G: \tau_u \mapsto \dot{\Theta}$ when $\epsilon > 0$

$$\int_0^{NT_s} \left[\dot{\Theta}(t)^{\mathrm{T}}\tau_u(t) - \epsilon\dot{\Theta}^{\mathrm{T}}(t)\dot{\Theta}(t)\right]dt \geq V(\dot{\Theta}(NT_s)) - V(\dot{\Theta}(0)) \tag{6}$$

$$\int_0^{NT_s} \dot{\Theta}(t)^{\mathrm{T}}\tau_u(t)dt \geq \epsilon \int_0^{NT_s} \dot{\Theta}^{\mathrm{T}}(t)\dot{\Theta}(t)dt - V(\dot{\Theta}(0)). \tag{7}$$

If a linear friction model is assumed such that $\epsilon_{\text{friction}}$ is a positive real value and $\tau = \tau_{\text{no friction}} - \epsilon_{\text{friction}}\dot{\Theta}(t)$, then by including negative velocity feedback as described above it is sufficient to render the robotic system strictly output passive with respect to the idealized input $\tau_{\text{no friction}}$ and output $\dot{\Theta}(t)$. If nonlinear stiction type friction is considerable at the robot joints, it is desirable to address its effects that can lead to jerky motion. The proposed architecture is still applicable, for example, by considering the LuGre Model [39], [40]. The LuGre Model can be studied in a passivity-based framework. Necessary and sufficient conditions for passivity of the LuGre model are presented in [41]. Friction identification and compensation algorithms based on the LuGre Model have been successfully implemented to control robotic systems [42] in order to achieve improved performance and they could be incorporated in our framework.

In conclusion, the velocity-damped robot is an strictly output passive system which is L_2^m-*stable*. The robot's strictly output passive property allows us to interconnect an strictly output passive controller over a wireless network using wave variables such that the overall system remains strictly output passive and L_2^m-*stable*. The strictly output passive property is used in Theorem V-A to show that the NCS depicted in Fig. 2 is L_2^m *stable*.

B. Wave Variables

The information communicated between the robotic system and the controller is subject to delays as shown in Fig. 2. Wave variables allow us to transmit data between the passive system

---

[3]A lossless passive system satisfies Definition 1-1 and possesses a lossless property such that (1) holds with equality in which $\beta = 0$ for the set of all feasible input functions u in which $G(u(0)) = G(u(NT_s)) = 0$.

and the passive controller while preserving passivity even in the presence of time-varying delays. Since we have proposed a mixed-signal architecture with *CT* and *DT* components, we recall the *CT* and *DT* wave variable definitions.

Definition 3 ([27]): The *CT* wave variables $v_{ucd}(t)$, $u_p(t) \in \mathbb{R}^m$ depicted in Fig. 2 are related to the corresponding torque and velocity vectors $\tau_{ucd}(t)$, $\dot{\Theta}(t) \in \mathbb{R}^m$ as follows:

$$\frac{1}{2}\left(u_p^{\mathrm{T}}(t)u_p(t) - v_{ucd}^{\mathrm{T}}(t)v_{ucd}(t)\right) = \dot{\Theta}^{\mathrm{T}}(t)\tau_{ucd}(t). \quad (8)$$

The wave variable $v_{ucd}(t)$ and velocity measurement $\dot{\Theta}(t)$ determine the corresponding wave variable $u_p(t)$ and delayed control torque $\tau_{ucd}(t)$, which result from the following equation:

$$\begin{bmatrix} u_p(t) \\ \tau_{ucd}(t) \end{bmatrix} = \begin{bmatrix} -I & \sqrt{2b}I \\ -\sqrt{2b}I & bI \end{bmatrix} \begin{bmatrix} v_{ucd}(t) \\ \dot{\Theta}(t) \end{bmatrix}$$

in which $I \in \mathbb{R}^{m \times m}$ denotes the identity matrix and $0 < b < \infty$ is a real number.

Definition 4 ([43]): The *DT* input and output *wave variables* $u_{pd}[i]$, $v_{uc}[i] \in \mathbb{R}^m$ depicted in Fig. 2 are related to the corresponding discrete torque and velocity vectors $\tau_{uc}[i]$, $\dot{\Theta}_d[i] \in \mathbb{R}^m$ as follows:

$$\frac{1}{2}\left(u_{pd}^{\mathrm{T}}[i]u_{pd}[i] - v_{uc}^{\mathrm{T}}[i]v_{uc}[i]\right) = \tau_{uc}[i]^{\mathrm{T}}\dot{\Theta}_d[i]. \quad (9)$$

The wave variable $u_{pd}[i]$ and control torque $\tau_{uc}[i]$ determine the corresponding wave variable $v_{uc}[i]$ and delayed velocity $\dot{\Theta}_d[i]$, which result from the following equation:

$$\begin{bmatrix} v_{uc}[i] \\ \dot{\Theta}_d[i] \end{bmatrix} = \begin{bmatrix} I & -\sqrt{\frac{2}{b}}I \\ \sqrt{\frac{2}{b}}I & -\frac{1}{b}I \end{bmatrix} \begin{bmatrix} u_{pd}[i] \\ \tau_{uc}[i] \end{bmatrix}. \quad (10)$$

The received wave variables $u_{pd}[i]$, $v_{ucd}[i]$ are delayed versions of the transmitted wave variables $u_p[i]$, $v_{uc}[i]$ such that $u_{pd}[i] = u_p[i - p(i)]$ and $v_{ucd}[i] = v_{uc}[i - c(i)]$ in which $p(i)$, $c(i) \in \{0, 1, \ldots, N\}$ are the respective delays at time i. The *CT* wave variable transformation is denoted by the block with the symbol b immediately to the right of the robotic system. The *DT* wave variable transformation is denoted by the block with the symbol $1/b$ immediately to the left of the controller system.

C. Mapping DT to CT Controller Variables

The *inner product equivalent sampler* (*IPES*) and *ZOH* blocks are introduced in order to relate the *CT* robot's inputs and outputs to the *DT* controller's inputs and outputs. Specifically, the *IPES* at the input of the digital controller and *ZOH* at the output can be used to ensure that the overall system $G_{net} : [\dot{\Theta}_{sr}^{\mathrm{T}}(t), \tau_d^{\mathrm{T}}(t)]^{\mathrm{T}} \mapsto [\tau_{uc}^{\mathrm{T}}(t), \dot{\Theta}^{\mathrm{T}}(t)]^{\mathrm{T}}$ is (*strictly output*) passive. We refer to both *IPES* and *ZOH* devices as *IPESH*. We present a *noncausal* version of the *IPESH* which is based on the *causal* version of the *IPESH* presented in [16], [43, Definition 4] and earlier work in [12] and [44].

Definition 5: The *IPESH* consists of an *IPES* and *ZOH* element. The *IPES* is defined as follows:

$$x(t) = \int_0^t \dot{\Theta}_{sr}(\tau)d\tau$$

$$\dot{\Theta}_{sr}[i] = x((i + 1)T_s) - x(iT_s). \quad (11)$$

The *ZOH* is defined as follows:

$$\tau_{uc}(t) = \tau_{uc}[i] \qquad \forall t \in [iT_s, (i + 1)T_s). \quad (12)$$

$\dot{\Theta}_{sr}(t)$ denotes an appropriately scaled velocity profile for the robot to follow such that $\dot{\Theta}_{sr}(t) = -k_s\dot{\Theta}_r(t)$, in which $\dot{\Theta}_r(t)$ is the desired velocity trajectory to track and k_s is a positive real value which is determined to account for the scaling effects that result from the use of the wave variable sampler described in Section IV-B.

Lemma 1: The *IPESH* has the following properties.

1) The *CT* variables $(\dot{\Theta}_{sr}(t), \tau_{uc}(t))$ and the *DT* variables $(\dot{\Theta}_{sr}[i], \tau_{uc}[i])$ have equivalent inner products such that $\langle\tau_{uc}, \dot{\Theta}_{sr}\rangle_N = \langle\tau_{uc}, \dot{\Theta}_{sr}\rangle_{NT_s}$ holds $\forall N \geq 1$.

2) The *CT* variable $\tau_{uc}(t)$ and the *DT* variable $\tau_{uc}[i]$ have proportional 2-norms such that $\|(\tau_{uc})_{NT_s}\|_2^2 = T_s\|(\tau_{uc})_N\|_2^2$.

Proof:

1)
$$\begin{aligned} \langle\tau_{uc}, \dot{\Theta}_{sr}\rangle_N &= \sum_{i=0}^{N-1} \tau_{uc}^{\mathrm{T}}[i]\dot{\Theta}_{sr}[i] \\ &= \sum_{i=0}^{N-1}\sum_{j=1}^{m} \tau_{uc_j}[i]\dot{\Theta}_{sr_j}[i] \\ &= \sum_{j=1}^{m}\sum_{i=0}^{N-1} \tau_{uc_j}[i]\int_{iT_s}^{(i+1)T_s} \dot{\Theta}_{sr_j}(\tau)d\tau \\ &= \sum_{j=1}^{m}\sum_{i=0}^{N-1}\int_{iT_s}^{(i+1)T_s} \tau_{uc_j}(\tau)\dot{\Theta}_{sr_j}(\tau)d\tau \\ &= \sum_{j=1}^{m}\int_{0}^{NT_s} \tau_{uc_j}(\tau)\dot{\Theta}_{sr_j}(\tau)d\tau \\ &= \int_{0}^{NT_s} \tau_{uc}^{\mathrm{T}}(\tau)\dot{\Theta}_{sr}(\tau)d\tau \end{aligned}$$
$$\langle\tau_{uc}, \dot{\Theta}_{sr}\rangle_N = \langle\tau_{uc}, \dot{\Theta}_{sr}\rangle_{NT_s} \text{ holds } \forall N \geq 1. \quad (13)$$

2)
$$\begin{aligned} \|(\tau_{uc})_{NT_s}\|_2^2 &= \int_{0}^{NT_s} \tau_{uc}^{\mathrm{T}}(t)\tau_{uc}(t)dt \\ &= \sum_{j=1}^{m}\int_{0}^{NT_s} \tau_{uc_j}(t)\tau_{uc_j}(t)dt \\ &= \sum_{j=1}^{m}\sum_{i=0}^{N-1}\int_{iT_s}^{(i+1)T_s} \tau_{uc_j}[i]\tau_{uc_j}[i]dt \\ &= T_s\sum_{j=1}^{m}\sum_{i=0}^{N-1} \tau_{uc_j}[i]\tau_{uc_j}[i] \\ &= T_s\sum_{i=0}^{N-1} \tau_{uc}^{\mathrm{T}}[i]\tau_{uc}[i] \end{aligned}$$
$$\|(\tau_{uc})_{NT_s}\|_2^2 = T_s\|(\tau_{uc})_N\|_2^2. \quad (14)$$
∎

Note that the ordering of the *IPESH* is reversed from typical applications [12], [16], [43], [44] in which the *IPES* is located at the output of a *CT* system and the *ZOH* is located at the input to the *CT* system. In [43, Appendix E], we show that such traditional *IPESH* arrangements are indeed causal

and can be used to synthesize passive *DT* single-input single-output (SISO) LTI filters. The causal *IPESH* can be realized through the use of state-space observers for LTI systems [35], [45]. The difficulty in applying the *IPESH* in the causal framework is that it is not clear how to systematically apply it to *CT nonlinear* plants. Therefore, implementations that attempt to apply the *IPESH* to a *CT* plant are only approximate realizations [12], [44] in which some nonpassive behavior remains. The recent work of [46] shows promise of achieving improved *DT* mappings that preserve passivity-like properties for certain nonlinear *CT* systems, however, these results are still sampling-rate-limited.

In our proposed framework, we simply use the noncausal arrangement of the *IPESH* for analysis. The *IPESH* allows us to relate our digital control reference $\dot{\Theta}_{sr}[i]$ and digital control output $\tau_{uc}[i]$ back to the *CT* domain in order to show L_2^m-stability. By removing the need for a *ZOH* input to a nonlinear *CT* plant and an *IPES* from the plant's output, we eliminate the need for a complex nonlinear observer. Our framework can still precisely include a digital controller for the *CT* plant which is interconnected to wave variables.

D. Wave Variable Sampler and Wave Variable Hold

The *wave variable sampler and wave variable hold system* denoted by the pair WS,T_s and WH,T_s in Fig. 2 are use to convert *CT* to *DT* wave variables and vice versa. In order to ensure passivity of the overall control architecture, the sample-and-hold devices must satisfy the following inequality:

$$\int_0^{NT_s} \left(u_p^{\mathrm{T}}(t)u_p(t)v_{\mathrm{ucd}}^{\mathrm{T}}(t)v_{\mathrm{ucd}}(t) \right) dt$$
$$- \sum_{i=0}^{N-1} \left(u_p^{\mathrm{T}}[i]u_p[i] - v_{\mathrm{ucd}}^{\mathrm{T}}[i]v_{\mathrm{ucd}}[i] \right) \geq 0. \quad (15)$$

Equation (15) is used in the proof of our main result (Theorem V-A). A sufficient condition to satisfy (15) is to design the WS to satisfy the following inequality:

$$\sum_{i=0}^{N-1} u_p^{\mathrm{T}}[i]u_p[i] \leq \int_0^{NT_s} u_p^{\mathrm{T}}(t)u_p(t)dt \quad (16)$$

and for the WH to satisfy the following inequality:

$$\int_0^{NT_s} v_{\mathrm{ucd}}^{\mathrm{T}}(t)v_{\mathrm{ucd}}(t)dt \leq \sum_{i=0}^{N-1} v_{\mathrm{ucd}}^{\mathrm{T}}[i]v_{\mathrm{ucd}}[i]. \quad (17)$$

Sufficiency is proven by rearranging terms in (16) and (17) such that (18) and (19) result

$$\int_0^{NT_s} u_p^{\mathrm{T}}(t)u_p(t)dt - \sum_{i=0}^{N-1} u_p^{\mathrm{T}}[i]u_p[i] \geq 0 \quad (18)$$

$$- \int_0^{NT_s} v_{\mathrm{ucd}}^{\mathrm{T}}(t)v_{\mathrm{ucd}}(t)dt - \left(-\sum_{i=0}^{N-1} v_{\mathrm{ucd}}^{\mathrm{T}}[i]v_{\mathrm{ucd}}[i] \right) \geq 0. \quad (19)$$

Then, taking the sum of the respective left-hand and right-hand sides of (18) and (19) results in (15).

In the following, we present two *realizations* of a WS and one WH which satisfy (16) and (17), respectively. The first realization of the WS is a *nonlinear* WS which is easy to verify as satisfying (16) by direct substitution. However, we also present a *linear* WS and corresponding *linear* PH because they are easier to realize with analog electronics.

Denote each jth element of the column vectors $u_p(t), u_p[i]$ as $u_{p_j}(t), u_{p_j}[i]$ in which $j = \{1, \ldots, m\}$ and assume that $u_{p_j}(t) = 0$, if $t < 0$. The *nonlinear* WS is realized as follows:

$$u_{p_j}[i] = \sqrt{\int_{(i-1)T_s}^{iT_s} u_{p_j}^2(t)dt} \ \mathrm{sgn}\left(\int_{(i-1)T_s}^{iT_s} u_{p_j}(t)dt \right). \quad (20)$$

Lemma 2: The *nonlinear* WS defined by (20) satisfies the wave variable sampling condition (16).

Proof: Direct substitution of (20) into the left-hand side of (16) allows one to verify that (16) holds [47]. ∎

Denote each jth element of the column vectors $v_{\mathrm{ucd}}(t), v_{\mathrm{ucd}}[i]$ as $v_{\mathrm{ucd}_j}(t), v_{\mathrm{ucd}_j}[i]$ in which $j = \{1, \ldots, m\}$ and assume that $v_{\mathrm{ucd}_j}[i] = 0$, if $i < 0$. The *linear* WH is implemented as follows:

$$v_{\mathrm{ucd}_j}(t) = \frac{1}{\sqrt{T_s}}v_{\mathrm{ucd}_j}[i-1], \ t \in [iT_s, (i+1)T_s]. \quad (21)$$

Lemma 3: The *linear* WH defined by (21) satisfies the wave variable hold condition (17).

Proof: Direct substitution of (21) into the left-hand side of (17) allows one to verify that (17) holds [47]. ∎

Deriving a Linear WS From Data Reduction Methods Used in Telemanipulation Systems: It was easily shown how a *nonlinear* WS can satisfy (16) by direct substitution of terms. However, we shall present a less complex *linear* WS that satisfies (16). The *linear* WS does not require a nonlinear square root operation, which will simplify the analog circuit required for implementation. Our *linear* WS is inspired by the telemanipulation literature. The sufficient conditions given by (16) and (17) in order to implement a WS and WH for our architecture are similar to those required to send *CT* wave variables over a digital network between two *CT* robotic systems in a telemanipulation network. For example, in [17, Th. 3.1] the sufficient Conditions 1 and 2 required to achieve an asymptotically stable telemanipulation system required the slaves-received wave variable $\hat{u}_s(t)$ to be bounded by the original wave variable transmitted from the master $u_m(t)$ s.t. $\int_0^t \hat{u}_s^{\mathrm{T}}(s)\hat{u}_s(s)ds \leq \int_0^t u_m^{\mathrm{T}}(s)u_m(s)ds$ (Condition 1) and, conversely, the masters-received wave variable $\hat{v}_m(t)$ is to be bounded by the original wave variable transmitted from the slave $v_s(t)$ such that $\int_0^t \hat{v}_m^{\mathrm{T}}(s)\hat{v}_m(s)ds \leq \int_0^t v_s^{\mathrm{T}}(s)v_s(s)ds$ (Condition 2). In order to satisfy these two conditions, [17] proposed to use an identity *function generator* and a time-varying gain *signal reconstructor*.

The recent work of [14] describes elaborate compression/decompression techniques known as *energy-supervised data reconstruction* in order to satisfy Conditions 1 and 2. Finally, the work of [28] describes *passive encoder/decoder* algorithms that satisfy Conditions 1 and 2. Although Conditions 1 and 2 are too general for our framework, these algorithms can typically be modified with a scaling term in order to satisfy (16) and (17) or the weaker condition (15). For example, in [28] the authors propose a *passive interpolative*

downsampler to generate a *DT* wave variable $u_m[i]$ from its *DT* counterpart $u_m(t)$ as follows:

$$u_m[i] = \frac{1}{T_s} \int_{(i-1)T_s}^{iT_s} u_m(t)dt.$$

They then transmit $u_m[i]$ over a network to the slave and extrapolate it such that $\hat{u}_s(t) = u_m[i - p(i)]$, $t \in [iT_s, (i+1)T_s)$. For simplicity of discussion, if we assume that $p(i) = 0$ and u_m is a scalar, then we have

$$\int_0^{NT_s} \hat{u}_s^2(t)dt = \frac{1}{T_s} \sum_{i=1}^{N-1} \left(\int_{(i-1)T_s}^{iT_s} u_m(t)dt \right)^2$$

$$\leq \frac{T_s}{T_s} \sum_{i=1}^{N-1} \int_{(i-1)T_s}^{iT_s} u_m^2(t)dt \text{ (Schwarz inequality)}$$

$$\leq \sum_{i=0}^{N-1} \int_{iT_s}^{(i+1)T_s} u_m^2(t)dt$$

$$\leq \int_0^{NT_s} u_m^2(t)dt \text{ (Condition 1).}$$

The passive interpolative downsampler *must be rescaled* in order to satisfy (16), therefore we propose a *linear WS*.

A *linear WS* is realized as follows:

$$u_{p_j}[i] = \frac{1}{\sqrt{T_s}} \int_{(i-1)T_s}^{iT_s} u_{p_j}(t)dt \quad \forall j \in \{1, \ldots, m\}. \quad (22)$$

Lemma 4: The *linear WS* defined by (22) satisfies the wave variable sampler condition (16).

Proof: Substituting (22) into the left-hand side of (16) results in

$$\sum_{i=0}^{N-1} u_p^T[i]u_p[i] = \sum_{j=1}^m \sum_{i=0}^{N-1} u_{p_j}^2[i]$$

$$= \sum_{j=1}^m \sum_{i=1}^{N-1} \left(\frac{1}{\sqrt{T_s}} \int_{(i-1)T_s}^{iT_s} u_{p_j}(t)dt \right)^2$$

$$\leq \frac{T_s}{T_s} \sum_{j=1}^m \sum_{i=1}^{N-1} \int_{(i-1)T_s}^{iT_s} u_{p_j}^2(t)dt$$

$$\leq \int_0^{NT_s} u_p^T(t)u_p(t)dt. \qquad \blacksquare$$

Accounting for the Scaling Effects of WS and WH: The consequence of using the proposed *(non)linear WS* and *WH* interconnected to a digital controller is that $\dot{\Theta}(t) \neq \dot{\Theta}_d[i]$ at steady state. Intuitively, this is due to the identical scaling of $u_p[i]$ because of either the *nonlinear* or *linear WS* chosen, however, the relationships are also dependent on the properties of both the controller and plant. We can derive these relationships precisely in order to scale the controller set point for the *DT* case when using a *wave variable downsampler*.

By assuming that the system reaches a steady-state operating point, we can compare *CT* and *DT* variables for the SISO case. The steady-state relationships for the plant are denoted by the scalar gain $k_p = (\dot{\Theta}(t)/\tau_u(t))$ in which $\dot{\Theta}(t)$ and $\tau_u(t)$ remain at a constant steady-state value for some $t \geq T_s i_{\text{steady-state}}$ in which $i_{\text{steady-state}}$ is a nonnegative

integer. Denote $\dot{\Theta}(s)$ and $\tau_u(s)$ as the corresponding Laplace transforms for $\dot{\Theta}(t)$ and $\tau_u(t)$, respectively, with transfer function $G_p(s) = (\dot{\Theta}(s)/\tau_u(s))$. Then, if the plant is linear, $k_p = G_p(0) \leq (1/\epsilon)$. $k_p \leq (1/\epsilon)$ because the L_2^m-gain of our strictly output passive system is $(1/\epsilon)$. For simplicity of discussion, we derive the L_2^m-gain by assuming that $V(\dot{\Theta}(0)) = 0$. From (7), the following inequality holds: $\epsilon \|(\dot{\Theta})_{NT_s}\|_2^2 \leq \langle \dot{\Theta}, \tau_u \rangle_{NT_s} = |\langle \dot{\Theta}, \tau_u \rangle_{NT_s}| \leq \|(\dot{\Theta})_{NT_s}\|_2 \|(\tau)_{NT_s}\|_2$. Therefore, $\|(\dot{\Theta})_{NT_s}\|_2 \leq (1/\epsilon)\|(\tau)_{NT_s}\|_2$ holds.[4] The steady-state gain for the controller is denoted by the scalar $k_c = (\tau_{\text{uc}}[i]/\dot{e}[i])$ in which $\tau_{\text{uc}}[i]$ and $\dot{e}[i]$ remain constant for $i \geq i_{\text{steady-state}}$. A linear passive control law shall be considered such that $G_{\text{pc}} : \dot{e}_1 \rightarrow \tau_{\text{uc}}$ can be denoted by the transfer function $G_{\text{pc}}(z) = (\tau_{\text{uc}}(z)/\dot{e}_1(z))$ in which $\tau_{\text{uc}}(z)$ and $\dot{e}_1(z)$ are the respective z-transforms of the *DT* signals $\tau_{\text{uc}}[i]$ and $\dot{e}_1[i]$ in which $\dot{e}_1[i] = \dot{e}[i] - \epsilon_c \tau_{\text{uc}}[i]$. Therefore, $k_c = (G_{\text{pc}}(1)/1 + \epsilon_c G_{\text{pc}}(1))$. The passive controller will be designed to have infinite steady-state gain such that $G_{\text{pc}}(1) = \infty$, which results in $k_c = (1/\epsilon_c)$.

Lemma 5: Assume that the *CT* plant and controller subsystems depicted in Fig. 2 are SISO and they are not subject to additional data loss and the disturbance $\tau_d(t)$ and reference $\dot{\Theta}_{sr}(t) = -k_s \dot{\Theta}_r(t)$ are held constant; then the following steady-state relationship holds:

$$\dot{\Theta}(t) = k_s \frac{T_s}{\sqrt{T_s}} \frac{k_c k_p}{1 + k_c k_p} \dot{\Theta}_r(t) + \frac{k_p}{1 + k_p k_c} \tau_d(t).$$

Proof: If there is no data loss and the disturbance $\tau_d(t)$ and reference $\dot{\Theta}_r(t)$ are held constant, then the following steady-state relationships hold $\dot{\Theta}(t) = k_p \tau_u(t) \leq (1/\epsilon)\tau_u(t)$, $\tau_{\text{uc}}[i] = k_c \dot{e}[i] = (1/\epsilon_c)\dot{e}[i]$, $\dot{\Theta}_{sr}[i] = T_s \dot{\Theta}_{sr}(t)$, $\dot{\Theta}_{sr}(t) = -k_s \dot{\Theta}_r(t)$. Further, at steady state, we have

$$u_{\text{pd}}[i] = \sqrt{T_s} u_p(t) \qquad (23)$$

$$v_{\text{ucd}}(t) = \frac{1}{\sqrt{T_s}} v_{\text{uc}}[i] \qquad (24)$$

$$\begin{bmatrix} v_{\text{uc}}[i] \\ \dot{\Theta}_d[i] \end{bmatrix} = \begin{bmatrix} 1 & -\sqrt{\frac{2}{b}} \\ \sqrt{\frac{2}{b}} & -\frac{1}{b} \end{bmatrix} \begin{bmatrix} u_{\text{pd}}[i] \\ \tau_{\text{uc}}[i] \end{bmatrix} \qquad (25)$$

$$\begin{bmatrix} u_p(t) \\ \tau_{\text{ucd}}(t) \end{bmatrix} = \begin{bmatrix} -1 & \sqrt{2b} \\ -\sqrt{2b} & b \end{bmatrix} \begin{bmatrix} v_{\text{ucd}}(t) \\ \dot{\Theta}(t) \end{bmatrix}. \qquad (26)$$

Substituting (23) into (25), and (24) into (26) results in

$$\begin{bmatrix} v_{\text{uc}}[i] \\ \dot{\Theta}_d[i] \end{bmatrix} = \begin{bmatrix} \sqrt{T_s} & -\sqrt{\frac{2}{b}} \\ \sqrt{\frac{2T_s}{b}} & -\frac{1}{b} \end{bmatrix} \begin{bmatrix} u_p(t) \\ \tau_{\text{uc}}[i] \end{bmatrix} \qquad (27)$$

$$\begin{bmatrix} u_p(t) \\ \tau_{\text{ucd}}(t) \end{bmatrix} = \begin{bmatrix} -\sqrt{\frac{1}{T_s}} & \sqrt{2b} \\ -\sqrt{\frac{2b}{T_s}} & b \end{bmatrix} \begin{bmatrix} v_{\text{uc}}[i] \\ \dot{\Theta}(t) \end{bmatrix} \qquad (28)$$

respectively, which can be written in the following form:

$$\begin{bmatrix} \tau_{\text{ucd}}(t) \\ \dot{\Theta}_d[i] \end{bmatrix} = C_1 \begin{bmatrix} v_{\text{uc}}[i] \\ u_p(t) \end{bmatrix} + C_2 \begin{bmatrix} \tau_{\text{uc}}[i] \\ \dot{\Theta}(t) \end{bmatrix}$$

$$C_1 = \begin{bmatrix} -\sqrt{\frac{2b}{T_s}} & 0 \\ 0 & \sqrt{\frac{2T_s}{b}} \end{bmatrix}, \quad C_2 = \begin{bmatrix} 0 & b \\ -\frac{1}{b} & 0 \end{bmatrix} \qquad (29)$$

---

[4]The proof in [33, Th. 2.2.14] shows how to account for nonzero $V(\dot{\Theta}(0))$ in (7).

$$\begin{bmatrix} v_{\text{uc}}[i] \\ u_p(t) \end{bmatrix} = C_3 \begin{bmatrix} v_{\text{uc}}[i] \\ u_p(t) \end{bmatrix} + C_4 \begin{bmatrix} \tau_{\text{uc}}[i] \\ \dot{\Theta}(t) \end{bmatrix}$$

$$C_3 = \begin{bmatrix} 0 & \sqrt{T_s} \\ -\sqrt{\frac{1}{T_s}} & 0 \end{bmatrix}, \quad C_4 = \begin{bmatrix} -\sqrt{\frac{2}{b}} & 0 \\ 0 & \sqrt{2b} \end{bmatrix}. \quad (30)$$

Solving for the wave variables in terms of $\tau_{\text{uc}}[i]$ and $\dot{\Theta}(t)$ in (30) results in

$$\begin{bmatrix} v_{\text{uc}}[i] \\ u_p(t) \end{bmatrix} = (I - C_3)^{-1} C_4 \begin{bmatrix} \tau_{\text{uc}}[i] \\ \dot{\Theta}(t) \end{bmatrix}. \quad (31)$$

Substituting (31) into (29) results in

$$\begin{bmatrix} \tau_{\text{ucd}}(t) \\ \dot{\Theta}_d[i] \end{bmatrix} = [C_2 + C_1(I - C_3)^{-1} C_4] \begin{bmatrix} \tau_{\text{uc}}[i] \\ \dot{\Theta}(t) \end{bmatrix}. \quad (32)$$

Solving for (32) results in

$$\begin{bmatrix} \tau_{\text{ucd}}(t) \\ \dot{\Theta}_d[i] \end{bmatrix} = \begin{bmatrix} \sqrt{1/T_s} & 0 \\ 0 & \sqrt{T_s} \end{bmatrix} \begin{bmatrix} \tau_{\text{uc}}[i] \\ \dot{\Theta}(t) \end{bmatrix}. \quad (33)$$

Using (33), it is straightforward to show that

$$\dot{\Theta}(t) = k_s \frac{T_s}{\sqrt{T_s}} \frac{k_c k_p}{1 + k_c k_p} \dot{\Theta}_r(t) + \frac{k_p}{1 + k_p k_c} \tau_d(t). \quad \blacksquare$$

Therefore, in order for $\dot{\Theta}(t) = \dot{\Theta}_r(t)$ when $\tau_d(t) = 0$ and $k_p k_c >> 1$, $k_s = (1 + k_c k_p / k_c k_p \sqrt{T_s}) \approx (1/\sqrt{T_s})$. Note that our analysis reveals that the controller attenuates the steady-state disturbances τ_d such that $\dot{\Theta} \approx (1/k_c)\tau_d$ when $k_p k_c >> 1$ and $\dot{\Theta}_r = 0$, independent of T_s. Further, it should be noted that this result takes in to account the scaling effects of the *IPES*. For the case in which the *IPES* is not used and $\dot{\Theta}_{sr}[i] = \dot{\Theta}_{sr}(t)$ for $t \in [iT_s, (i+1)T_s)$, then $k_s = \sqrt{T_s}$.

E. Passive Digital Controller

Here we present a passive digital control law. We derive this control law by first presenting a *CT* passive control law that allows for two symmetric positive definite gain matrices. We then apply a causal *IPESH* transformation to this *CT* control law to derive a passive *DT* control law. We conclude our presentation with a minimal realization for the digital control law that accounts for the use of wave variables.

A passive *CT* PD controller is given by

$$\dot{e}_1(t) = (\dot{\Theta}_d(t) + \dot{\Theta}_{sr}(t) - \epsilon_c \tau_{\text{uc}}(t))$$
$$\tau_{\text{uc}}(t) = K_p e_1(t) + K_d(\dot{\Theta}_d(t) + \dot{\Theta}_{sr}(t) - \epsilon_c \tau_{\text{uc}}(t))$$

in which $\dot{\Theta}_d$, $\dot{\Theta}_{sr}$, $\tau_{\text{uc}}(t) \in \mathbb{R}^m$ and K_p and K_d are real symmetric positive definite matrices. Denote the input to the controller as $u(t) = (\dot{\Theta}_d(t) + \dot{\Theta}_{sr}(t) - \epsilon_c \tau_{\text{uc}}(t))$, the integrated error $e_1(t) = x(t)$ and respective output as $y(t) = \tau_{\text{uc}}(t) = y_p(t) + y_d(t)$ in which $y_p = K_p x(t)$ and $y_d(t) = K_d u(t)$. Passivity is verified by showing that the output consists of a parallel combination of two passive subsystems with outputs $y_p(t)$ and $y_d(t)$ and shared input $u(t)$. Since $y_d^T u = u^T K_d^T u \geq 0$ holds for all u, then the integral of $y_d^T u$ is nonnegative ($\langle y_d, u \rangle_{NT_s} \geq 0$). Therefore the system with output $y_d(t)$ is passive. Passivity of the second system with input $u(t)$ and output $y_p(t)$ is verified by selecting the Lyapunov function $V(x) = (1/2)x^T K_p x > 0$ in which its derivative $\dot{V}(x) = x^T K_p^T \dot{x} = y_p^T u$. Integrating $\dot{V}(x)$ results in

the passive expression $\langle y_p, u \rangle_{NT_s} \geq V(x(NT_s)) - V(x(0)) \geq -V(x(0))$. Summing the two passivity expressions results in $\langle y_p, u \rangle_{NT_s} + \langle y_d, u \rangle_{NT_s} = \langle y, u \rangle_{NT_s} \geq -V(x(0))$.

The "proportional" term K_p and "derivative" term K_d are with respect to the integrated velocity error term $e_1(t)$, and therefore our proposed control architecture includes an integrator, which results in an infinite steady-state gain. Although we may refer to our structure in terms of scalar terms in order to simplify the discussion, controllers can still exploit our matrix formulation.

As discussed, our state-space realization of the controller can be described by

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (34)$$
$$y(t) = Cx(t) + Du(t) \quad (35)$$

in which $A = 0$, $B = I$, $C = K_p = K_p^T > 0$, $D = K_d = K_d^T > 0\}$ (all matrices are in $\mathbb{R}^{m \times m}$, therefore $x(t)$, $u(t)$, and $y(t) \in \mathbb{R}^m$).

A discrete passive controller can now be synthesized following the steps presented in [35]. We first derive the resulting *DT* equivalent passive controller $G_{\text{pc}}:\dot{e}_1[i] \mapsto \tau_{\text{uc}}[i]$ by applying a causal *IPESH* transformation to the state-space realization (34)-(35) in which the *ZOH* is applied to the input such that $u(t) = u[i] \ \forall t \in [iT_s, (i+1)T_s)$ and *scaled IPES* is located at the output of the CT passive controller such that $y[i] = (1/T_s) \int_{iT_s}^{(i+1)T_s} y(t)dt$ with sampling period T_s. The resulting controller is of the following form:

$$x[i+1] = \Phi_{\mathbf{o}} x[i] + \Gamma_{\mathbf{o}} u[i]$$
$$y[i] = \mathbf{C_p} x[i] + \mathbf{D_p} u[i]$$
$$\Phi_{\mathbf{o}} = e^{A_o T_s},$$
$$A_o = \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ K_p & 0 \end{bmatrix} = \begin{bmatrix} I & 0 \\ T_s K_p & I \end{bmatrix}$$
$$\Gamma_{\mathbf{o}} = \int_0^{T_s} e^{A_o \eta} d\eta B_o,$$
$$B_o = \begin{bmatrix} B \\ D \end{bmatrix} = \begin{bmatrix} I \\ K_d \end{bmatrix} = \begin{bmatrix} T_s I & 0 \\ \frac{T_s^2}{2} K_p & T_s I \end{bmatrix} \begin{bmatrix} I \\ K_d \end{bmatrix}$$
$$= \begin{bmatrix} T_s I \\ \frac{T_s^2}{2} K_p + T_s K_d \end{bmatrix}$$
$$\mathbf{C_p} = \frac{1}{T_s} \mathbf{C_o}(\Phi_{\mathbf{o}} - I) = \frac{1}{T_s} \begin{bmatrix} 0 & I \end{bmatrix} \begin{bmatrix} 0 & 0 \\ T_s K_p & 0 \end{bmatrix} = \begin{bmatrix} K_p & 0 \end{bmatrix}$$
$$\mathbf{D_p} = \frac{1}{T_s} \mathbf{C_o} \Gamma_{\mathbf{o}} = \frac{1}{T_s} \begin{bmatrix} 0 & I \end{bmatrix} \begin{bmatrix} T_s I \\ \frac{T_s^2}{2} K_p + T_s K_d \end{bmatrix} = \frac{T_s}{2} K_p + K_d. \quad (36)$$

As depicted in Fig. 2, the variables $u[i] = \dot{e}_1[i] = (\dot{\Theta}_d[i] + \dot{\Theta}_{sr}[i] - \epsilon_c \tau_{uc}[i])$ and $y[i] = \tau_{uc}[i]$. Note that in our derivation we scaled $\mathbf{C_p}$ and $\mathbf{D_p}$ by $(1/T_s)$ in order to closely match the frequency response of the *DT* controller with that of the *CT* controller it is derived from. It should be noted that this is not a minimal realization for this controller. However, solving for

$$H(z) = \mathbf{C_p}(zI - \Phi_{\mathbf{o}})^{-1} \Gamma_{\mathbf{o}} + \mathbf{D_p}$$

$$= \begin{bmatrix} K_p & 0 \end{bmatrix} \begin{bmatrix} (z-I)^{-1} & 0 \\ T_s(z-I)^{-1}K_p(z-I)^{-1} & (z-I)^{-1} \end{bmatrix}$$

$$\times \begin{bmatrix} T_s I \\ \frac{T_s^2}{2}K_p + T_s K_d \end{bmatrix} + \mathbf{D_p}$$

$$= K_p(z-I)^{-1}T_s I + \mathbf{D_p}$$

results in a minimal controller $\Sigma = \{\Phi_{\mathbf{o}}, \Gamma_{\mathbf{o}}, \mathbf{C_p}, \mathbf{D_p}\}$ such that: $\Phi_{\mathbf{o}} = I \in \mathbb{R}^{m \times m}$, $\Gamma_{\mathbf{o}} = T_s I \in \mathbb{R}^{m \times m}$, $\mathbf{C_p} = K_p \in \mathbb{R}^{m \times m}$, and $\mathbf{D_p} = (T_s/2)K_p + K_d \in \mathbb{R}^{m \times m}$.

As depicted in Fig. 2, the following equality holds: $\dot{e}[i] = \dot{\Theta}_d[i] + \dot{\Theta}_{sr}[i] = \dot{e}_1[i] + \epsilon_c \tau_{uc}[i] \in \mathbb{R}^m$. Therefore, the feedback controller mapping $G_{sp} : \dot{e}[i] \mapsto \tau_{uc}[i]$ depicted in Fig. 2 is strictly output passive if $\epsilon_c > 0$. Denoting $G = (I + \epsilon_c \mathbf{D_p})$, the strictly output passive controller has a DT realization $\Sigma_{\mathbf{sp}} = \{\Phi_{\mathbf{sp}}, \Gamma_{\mathbf{sp}}, \mathbf{C_{sp}}, \mathbf{D_{sp}}\}$, which is described analogously to (36) in which

$$\Phi_{\mathbf{sp}} = \Phi_{\mathbf{o}} - \epsilon_c \Gamma_{\mathbf{o}} G^{-1} \mathbf{C_p} = I - \epsilon_c T_s G^{-1} K_p$$
$$\Gamma_{\mathbf{sp}} = \Gamma_{\mathbf{o}}(I - \epsilon_c G^{-1}\mathbf{D_p}) = T_s(I - \epsilon_c G^{-1}\mathbf{D_p})$$
$$\mathbf{C_{sp}} = G^{-1}\mathbf{C_p} = G^{-1}K_p$$
$$\mathbf{D_{sp}} = G^{-1}\mathbf{D_p} = G^{-1}\mathbf{D_p}.$$

Finally, the effects of the wave variables (Section IV-B) need to be considered, therefore, the final form of the digital control law depicted in Fig. 2 with inputs $(u_{pd}[i], \dot{\Theta}_{sr}[i])$ and outputs $(\tau_{uc}[i], v_{uc}[i])$ is as follows:

$$x[i+1] = \Phi_{\mathbf{fe}}x[i] + \Gamma_{\mathbf{fe}}\left(\sqrt{\frac{2}{b}}u_{pd}[i] + \dot{\Theta}_{sr}[i]\right)$$

$$\tau_{uc}[i] = \mathbf{C_{fe}}x[i] + \mathbf{D_{fe}}\left(\sqrt{\frac{2}{b}}u_{pd}[i] + \dot{\Theta}_{sr}[i]\right)$$

$$v_{uc}[i] = u_{pd}[i] - \sqrt{\frac{2}{b}}\tau_{uc}[i]$$

in which $G_1 = I + (1/b)\mathbf{D_{sp}}$, $\mathbf{C_{fe}} = G_1^{-1}\mathbf{C_{sp}}$, $\mathbf{D_{fe}} = G_1^{-1}\mathbf{D_{sp}}$, $\Phi_{\mathbf{fe}} = \Phi_{\mathbf{sp}} - (1/b)\Gamma_{\mathbf{sp}}\mathbf{C_{fe}}$ and $\Gamma_{\mathbf{fe}} = \Gamma_{\mathbf{sp}}(I - (1/b)\mathbf{D_{fe}})$.

V. ANALYSIS OF THE NCS

This section first presents the main results that provide sufficient conditions for our proposed networked control architecture to remain passive and stable. In addition, in order to characterize the applicability of the approach, we explain how minimum-phase LTI asymptotically stable systems that are not passive can be rendered strictly output passive, and thereby used in our proposed architecture.

A. Passivity Analysis

We prove that the NCS is L_2^m-stable and discuss practical networked delay conditions and weak synchronization requirements that satisfy the assumption of our framework. *Theorem 2:* For the wireless control architecture depicted in Fig. 2, which consists of the passive robot described by (4) and (5) and the passive digital controller described by (36), if the communication protocol ensures that both Condition 1)

$i - p(i) \neq j - p(j)$ and Condition 2) $i - c(i) \neq j - c(j)$ for all $j \neq i$ in which $j, i \in \{0, 1, \ldots, N-1\}$, then

$$\int_0^{NT_s} \dot{\Theta}^{\mathrm{T}}(t)\tau_{ucd}(t)dt \geq \sum_{i=0}^{(N-1)} \tau_{uc}^{\mathrm{T}}[i]\dot{\Theta}_d[i] \qquad (37)$$

always holds. Therefore, if $\epsilon_c = \epsilon = 0$, then the system depicted in Fig. 2 is passive. In addition, if $\epsilon_c > 0$, and $\epsilon > 0$, then the system is both strictly output passive and L_2^m stable.

Proof: The WS and WH satisfy (15), which can be compactly written as

$$\|(u_p)_{NT_s}\|_2^2 - \|(v_{ucd})_{NT_s}\|_2^2 \geq \|(u_p)_N\|_2^2 - \|(v_{ucd})_N\|_2^2. \quad (38)$$

Integrating both sides of (8) and substituting into (38) results in

$$\langle \dot{\Theta}, \tau_{ucd} \rangle_{NT_s} \geq \|(u_p)_N\|_2^2 - \|(v_{ucd})_N\|_2^2.$$

Condition 1) ensures that $\|(u_{pd})_N\|_2^2 = \sum_{i=0}^{N-1} u_p^{\mathrm{T}}[i - p(i)]u_p[i - p(i)] \leq \|(u_p)_N\|_2^2$; analogously Condition 2) ensures that $\|(v_{ucd})_N\|_2^2 = \sum_{i=0}^{N-1} v_{uc}^{\mathrm{T}}[i - c(i)]v_{uc}[i - c(i)] \leq \|(v_{uc})_N\|_2^2$. Therefore

$$\|(u_p)_N\|_2^2 - \|(v_{ucd})_N\|_2^2 \geq \|(u_{pd})_N\|_2^2 - \|(v_{uc})_N\|_2^2 \geq \langle \dot{\Theta}_d, \tau_{uc} \rangle_N$$

always holds. Therefore, we can satisfy (37), which can be more compactly written as

$$\langle \dot{\Theta}, \tau_{ucd} \rangle_{NT_s} \geq \langle \dot{\Theta}_d, \tau_{uc} \rangle_N. \qquad (39)$$

Recall that the gravity-compensated robot satisfies (7). Denote $V(\dot{\Theta}(0))$ as β_r for the robot and $\beta_c > 0$ as an appropriate controller-state-dependent Lyapunov function to account for potentially nonzero initial state conditions [48]. Then the robot satisfies

$$\langle \dot{\Theta}, \tau_u \rangle_{NT_s} \geq \epsilon \|(\dot{\Theta})_{NT_s}\|_2^2 - \beta_r \qquad (40)$$

and the controller satisfies

$$\langle \tau_{uc}, \dot{e} \rangle_N \geq \epsilon_c \|(\tau_{uc})_N\|_2^2 - \beta_c. \qquad (41)$$

We recall that

$$\tau_{ucd}(t) = \tau_d(t) - \tau_u(t) \qquad (42)$$
$$\dot{\Theta}_d[i] = \dot{e}[i] - \dot{\Theta}_{sr}[i]. \qquad (43)$$

Substituting (42) into the left-hand side of (39) and (43) into the right-hand side of (39) results in

$$\langle \dot{\Theta}, \tau_d \rangle_{NT_s} - \langle \dot{\Theta}, \tau_u \rangle_{NT_s} \geq \langle \dot{e}, \tau_{uc} \rangle_N - \langle \dot{\Theta}_{sr}, \tau_{uc} \rangle_N$$
$$\langle \dot{\Theta}, \tau_d \rangle_{NT_s} + \langle \dot{\Theta}_{sr}, \tau_{uc} \rangle_N \geq \langle \dot{\Theta}, \tau_u \rangle_{NT_s} + \langle \dot{e}, \tau_{uc} \rangle_N. \quad (44)$$

Substituting (40) and (41) into (44) results in

$$\langle \dot{\Theta}, \tau_d \rangle_{NT_s} + \langle \dot{\Theta}_{sr}, \tau_{uc} \rangle_N \geq \epsilon \|(\dot{\Theta})_{NT_s}\|_2^2$$
$$+ \epsilon_c \|(\tau_{uc})_N\|_2^2 - (\beta_r + \beta_c). \quad (45)$$

Next, we recall the first two properties listed for the *IPESH* in Section IV-C in which (13) is $\langle \tau_{uc}, \dot{\Theta}_{sr} \rangle_N = \langle \tau_{uc}, \dot{\Theta}_{sr} \rangle_{NT_s}$ and (14) is $T_s \|(\tau_{uc})_N\|_2^2 = \|(\tau_{uc})_{NT_s}\|_2^2$. Substituting (13) and (14) into (45) results in

$$\langle y, u \rangle_{NT_s} \geq \epsilon_s \|(y)_{NT_s}\|_2^2 - \beta_s \qquad (46)$$

in which

$$y = \left[\dot{\Theta}^{\mathrm{T}}, \tau_{\mathrm{uc}}^{\mathrm{T}}\right]^{\mathrm{T}}, \quad u = \left[\tau_d^{\mathrm{T}}, \dot{\Theta}_{sr}^{\mathrm{T}}\right]^{\mathrm{T}}$$

$$\epsilon_s = \min\left\{\epsilon, \frac{\epsilon_c}{T_s}\right\}, \quad \beta_s = \beta_r + \beta_c.$$

Therefore, (46) satisfies Definition 1-1 for passivity when $(\epsilon_c, \epsilon) \geq 0$ and either $\epsilon_c = 0$ or $\epsilon = 0 \implies \epsilon_s = 0$. Furthermore, (46) satisfies Definition 1-2 when $\epsilon_c > 0$ and $\epsilon > 0 \implies \epsilon_s > 0$ in order for the system to be strictly output passive, therefore from Theorem III the strictly output passive system is also L_2^m-*stable*. ∎

Our results apply for the more general case when the robotic and controller subsystems are replaced with arbitrary passive systems.

Corollary 1: For the network control architecture depicted in Fig. 2 in which the robot ($G_{\mathrm{robot}}(\tau(t))$) is replaced by any passive system satisfying Definition 1-1 (with gravity compensation disabled $g(\Theta(t)) = 0$) and the passive digital controller ($G_{\mathrm{pc}}(\dot{e}_1[i])$) satisfies Definition 1-1, if the communication protocol ensures that both Condition 1) $i - p(i) \neq j - p(j)$ and Condition 2) $i - c(i) \neq j - c(j)$ for all $j \neq i$ in which $j, i \in \{0, 1, \ldots, N-1\}$, then

$$\int_0^{NT_s} \dot{\Theta}^{\mathrm{T}}(t)\tau_{\mathrm{ucd}}(t)dt \geq \sum_{i=0}^{(N-1)} \tau_{\mathrm{uc}}^{\mathrm{T}}[i]\dot{\Theta}_d[i] \quad (47)$$

always holds. Therefore, if $\epsilon_c = \epsilon = 0$, then the system depicted in Fig. 2 is passive. In addition, if $\epsilon_c > 0$, and $\epsilon > 0$, then the system is both strictly output passive and L_2^m stable. Conditions 1) and 2) can be satisfied by communication protocols that prevent processing of duplicate transmissions of wave variables [10], [16], [17]. TCP is an appropriate protocol because it provides an unduplicated ordered stream of data unlike the user datagram protocol (UDP) protocol, which *can duplicate datagrams* due to the effects of having potentially multiple routes for the data to travel. Note that Conditions 1) and 2) *do not* require the data to be ordered or for all the data to arrive as is guaranteed by the TCP protocol. Therefore, an engineer can use the UDP protocol by augmenting the transmitted wave variables with an index and use either a table or choose to drop older out of order packets to enforce Conditions 1) and 2). For example, the summer-interpolation algorithm presented in [17] is suitable for transmitting wave variable data over UDP, as it ensures dropping old out-of-order data.

Furthermore, the controller can essentially be run as an asynchronous nonperiodic task in which it only needs to compute and send a new control command as new data is received from the plant [49]. An asynchronous nonperiodic task as defined in [49] requires the global digital clocks of the plant subsystem and the digital controller subsystem to be synchronized, however, the controller does not need to be executed in a synchronous manner as governed by a periodic schedule; instead, its execution schedule is determined by the arrival of sensor feedback data from the plant. The reason for the *relatively weak* global clock synchronization assumption becomes obvious when studying the proofs in [49] for stability because removing the synchronization assumption in order to

relate input–output relationships between the controller and plant subsystems would render the task of proving stability *ad absurdum*. There are many methods to enforce a global clock synchronization that should help reduce communication delays and jitter in order to improve system performance [50]–[52].

B. Passifying Asymptotically Stable LTI Systems

Although our results focus on a robotic system, our architecture is applicable to any systems that can be rendered strictly output passive. In the following, we show how minimum-phase LTI asymptotically stable systems that are not passive can be rendered strictly output passive, and thereby characterizing a large class of systems that can be used in the proposed NCS architecture. A direct result of Corollary 1 is that stable LTI passive systems with a corresponding square real-rational transfer function matrix $G_{\mathrm{robot}}(s)$ can be rendered strictly output passive when $\epsilon > 0$, which results in an L_2^m-stable system. However, if $G_{\mathrm{robot}}(s)$ is already strictly output passive, then the additional analog feedback loop can be eliminated by setting $\epsilon = 0$. We wish to consider the control of a minimum-phase LTI asymptotically stable system that is not passive and has the corresponding real-rational transfer function matrix $G(s)$. We show how to design a low-complexity asymptotically stable analog filter $H_*(s)$ such that $G_{\mathrm{robot}}(s) = G(s) + H_*(s)$ is rendered strictly output passive, and that it can be incorporated into our networked digital control framework in which $\epsilon = 0$.

For simplicity of discussion, we shall consider the control of a SISO LTI minimum-phase systems $G(s)$. The passivity index described in Section III can be used to design a low-complexity asymptotically stable high-pass filter $H_{\mathrm{hp}}(s)$ such that, if $\upsilon_f(G_{\mathrm{robot}}(s), \omega) < 0$ ($\upsilon_f(H_{\mathrm{hp}}(s), \omega) < -\upsilon_f(G(s), \omega)$) for all ω, then $G_{\mathrm{robot}}(s)$ is strictly output passive. Furthermore, if $H_{\mathrm{hp}}(s)$ is designed such that $|H_{\mathrm{hp}}(j0)| = 0$ and $|H_{\mathrm{hp}}(j\omega)|_{\lim \omega \to \infty} \neq 0$, then the output of $G(s)$ can be directly controlled by our digital controller at steady-state. A typical high-pass filter that may satisfy these conditions is of the form $H_{\mathrm{hp}}(s) = (\upsilon s/s + \omega_{\mathrm{hp}})$ in which υ, $\omega_{\mathrm{hp}} > 0$ and can be easily diagonalized for the MIMO case. If, instead, we choose to design an asymptotically stable band-pass filter $H_{\mathrm{bp}}(s)$ in which both $|H_{\mathrm{bp}}(j0)| = |H_{\mathrm{bp}}(j\omega)|_{\lim \omega \to \infty} = 0$ and the combined system $\upsilon_f(G_{\mathrm{robot}}(s), \omega) < 0$ for all ω except in the limit such that $\upsilon_f(G_{\mathrm{robot}}(s), \omega)_{\lim \omega \to \infty} = 0$, then we can only conclude that the combined system $G_{\mathrm{robot}}(s) = (G(s) + H_{\mathrm{bp}}(s))$ is passive and asymptotically stable; however, a necessary and sufficient test to determine if $G_{\mathrm{robot}}(s)$ is strictly output passive is to determine the feasibility of the following linear-matrix inequality (LMI), which results from the application of [48, Corollary 1].

Corollary 2: The real-rational transfer-function matrix $G_{\mathrm{robot}}(s)$ with a corresponding minimal state-space realization $\Sigma : \{A, B, C, D\}$ s.t. $\dot{x} = Ax + Bu$, $y = Cx + Du$ in which $x \in \mathbb{R}^n$, $y, u \in \mathbb{R}^m$, $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{m \times n}$, and $D \in \mathbb{R}^{m \times m}$ is strictly output passive iff there exists a real symmetric positive definite matrix $P = P^{\mathrm{T}} > 0$ and positive real constant $\bar{\epsilon} > 0$ s.t. the following LMI is satisfied:

$$\begin{bmatrix} A^{\mathrm{T}}P + PA + \bar{\epsilon}C^{\mathrm{T}}C & PB - \frac{1}{2}C^{\mathrm{T}} + \bar{\epsilon}C^{\mathrm{T}}D \\ \left(PB - \frac{1}{2}C^{\mathrm{T}} + \bar{\epsilon}C^{\mathrm{T}}D\right)^{\mathrm{T}} & \bar{\epsilon}D^{\mathrm{T}}D - \frac{1}{2}\left(D^{\mathrm{T}} + D\right) \end{bmatrix} \leq 0.$$
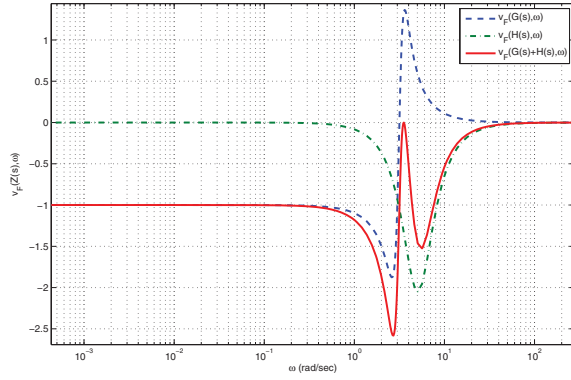
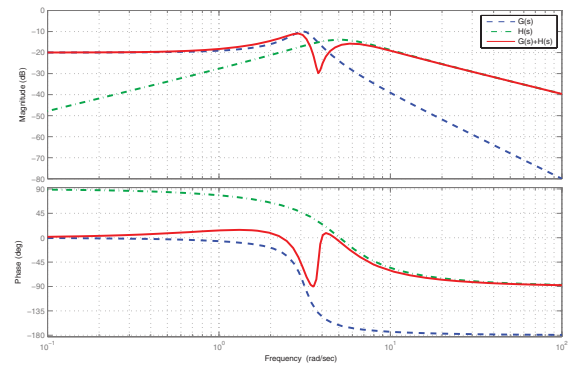Fig. 3. Passivity indexes for $G(s)$, $H(s) = H_{bp}(s)$, and $G_{robot}(s) = (H(s) + G(s))$.

Fig. 4. Bode plot for $G(s)$, $H(s) = H_{bp}(s)$, and $G_{robot}(s) = (H(s)+G(s))$.

Fig. 5. System response $x(t)$ as it tracks $x_r(t)$ for the mass–spring–damper system.

A typical band-pass filter that may render $G_{robot}(s)$ to be strictly output-passive is one of the form $H_{bp}(s) = k_{bp}(\omega_{bp}^2 s/s^2 + 2\zeta_{bp}\omega_{bp}s + \omega_{bp}^2)$ in which $k_{bp}, \omega_{bp} > 0$ and $0 < \zeta_{bp} \le 1$. This band-pass filter will be used in the digital control of a mass–spring damper system described in the following subsection.

Example: Position Control of a Mass–Spring Damper: For simplicity of discussion, we will focus on the SISO LTI case and consider the position $X(s)$ control of a cart of mass m which can exert a force $\tau_u(s)$ while attached to a wall by a spring with stiffness k and a damper with dampening coefficient c in which $(X(s)/\tau_u(s)) = G(s) = (\omega_n^2/k (s^2 + 2\zeta\omega_n s + \omega_n^2))$, $\omega_n = \sqrt{(k/m)}$ and $\zeta = (c/2\sqrt{km})$. It is well known that the force-to-velocity mapping of this mass–spring–damper system is passive, however, because of the stiffness of the spring, even our PD controller with integral action will be unable to achieve near-perfect tracking when trying to close the loop using only velocity feedback. Therefore, in order to achieve near-perfect tracking, we need to close the loop directly using position feedback. As described in Section V-B, we shall recover a strictly output mapping by augmenting the position output with a band-pass-filtered command output such that $\dot{\Theta}(s) = X(s) + H_{bp}(s)\tau_u(s) = (G(s) + H_{bp}(s))\tau_u(s) = G_{robot}(s)\tau_u(s)$. The band-pass filter will be of the form $H_{bp}(s) = k_{bp}(\omega_{bp}^2 s/s^2 + 2\zeta_{bp}\omega_{bp}s + \omega_{bp}^2)$.

We close the loop on $\dot{\Theta}(s)$ with our digital PD controller in order to get near-perfect tracking. For the case when $m = 1$ kg, $k = 10$ N/m, and $c = 1$ (N-s)/m, we have that $\zeta = 0.16$ and $\omega_n = 3.16$ in order to make the system strictly output passive so we choose $k_{bp} = (.41/k)$, $\zeta_{bp} = 0.5$, and $\omega_{bp} = 1.6\omega_n$. The resulting passivity indexes are plotted in Fig. 3 and the respective Bode plots are depicted in Fig. 4, which indicate that $G_{robot}(s)$ is passive and asymptotically stable. Solving the LMIs given in Corollary 2, we can verify that $G_{robot}(s)$ is strictly output passive in which $\bar{\epsilon} = 0.1977$.

Finally, we evaluate our control framework when $b = 1$, $T_s = 0.05$ s, $\epsilon_c = 1.0e - 5$, and $\epsilon = 0$ as the output $x(t)$ tracks the desired reference $x_r(t)$ depicted in Fig. 5. We compare our response $x(t)$ to the response of a *CT* controller implementation when $G_c(s) = (k_p + k_d s/s)$ and $H_{bp}(s) = 0$,
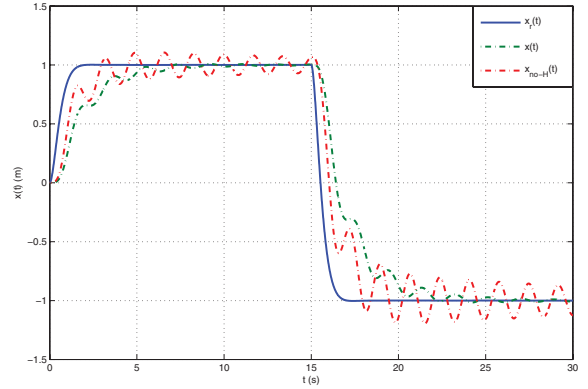
which is denoted $x_{no-H}(t)$. It is clear from Fig. 5 that a better performing system can be achieved when adding $H_{bp}(s)$ in parallel in order to achieve a strictly output passive system. Although not depicted, the system response for the ideal *CT* controller implementation with $H_{bp}(s)$ included is nearly identical to our digital controller implementation except for the T_s sample delay in the output of $x(t)$. Therefore with little loss in performance, the output of a class of asymptotically stable minimum-phase systems can be augmented with low-complexity analog filters $H_*(s)$ in order to create a strictly output passive system which can be integrated in to our proposed digital control framework. Corollary 1 ensures that L_2^m-stability is then independent of the sampling rate T_s chosen.

VI. EXPERIMENTAL EVALUATION

This section presents the experimental results for an NCS consisting of an asynchronous passive controller and a soft real-time simulated passive plant representing a robotic arm using an actual 802.1 1b wireless network. The controller is implemented in an asynchronous manner so that the reference input $\dot{\Theta}_{sr}[i]$ is buffered and processed as measurements from the plant $u_{pd}[i]$ arrive over the wireless network. The plant, a Simulink-based model that requires a variable time step solver, cannot be simulated in a hard real-time manner. However, we are able to pace the simulation in a soft real-time manner (the simulation can be paced such that the simulation time proceeds closely to the operating systems clock) such that the
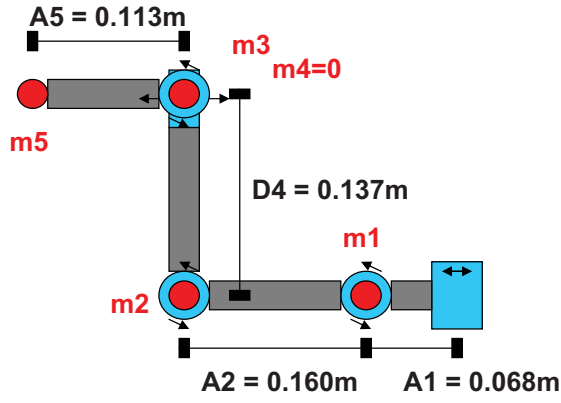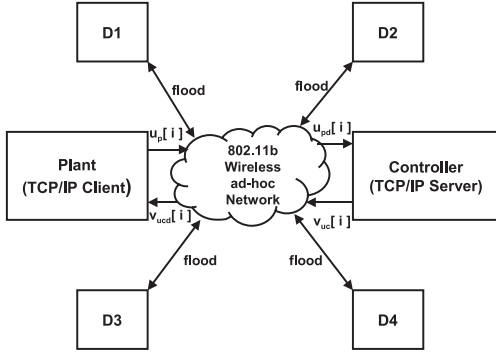
Fig. 6. Pioneer 3 arm.

Fig. 8. Robot performance with no network disturbance.

Fig. 7. Experimental setup.

experienced network delays correspond to delays an actual networked controlled plant would be subjected to.[5]

A. Robotic System Simulation

We consider the Pioneer 3 (P3) arm, which is a robotic manipulator built for the P3-DX and P3-AT ActivMedia mobile robots. The P3 arm has two main segments, the manipulator and the gripper. The manipulator has five degrees of freedom and the gripper has an additional one. Fig. 6 shows the home position of the P3 arm including the locations for the centers of gravity using the point-mass assumption. The dynamic model of the robotic arm is described by (4) and is derived using the Lagrangian approach for computing the elements of the mass matrix, Coriolis and centrifugal vector, and gravity vector [53]. The model is implemented as a Simulink block using the "Robotics Toolbox for MATLAB" [54] and includes gravity compensation and velocity damping as described in Section IV.

B. Passive Control Architecture

In order to choose an appropriate set of *CT* gains k_p and k_d, we focus our attention on joint 1 which is subject to the largest (changes of) inertia J as can be deduced from Fig. 6 such that $G_{pm}(s) = (1/Js)$. Similarly, we approximate the controller to be of the form $G_c(s) = (k_p + k_d s/s)$. Next, using basic loop

[5]We selected to simulate the plant because robotic arms such at the Pioneer 3 are controlled using simple servos and they do not provide feedback.
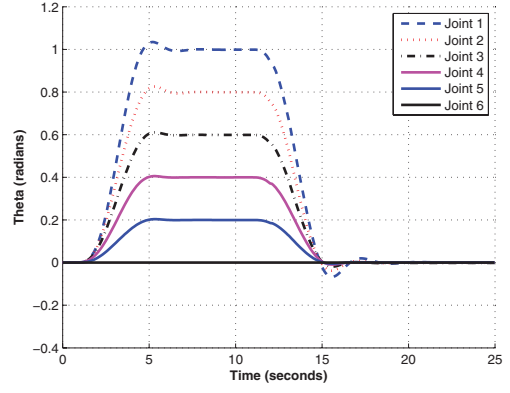
shaping techniques we desire the system to have a crossover frequency (ω_c s.t. $20 \log_{10}[|G_{pm}(j\omega_c)G_c(j\omega_c)|] = 0$ dB, in which $\omega_c = (\omega_n/N)$. $\omega_n = (\pi/T_s)$ is denoted as the Nyquist frequency. Therefore, the control gains can be computed based on a desired phase margin $0 < \phi \le 90$ (degrees) as follows: 1) $\tau = ((\phi - 40)/5\omega_c)$; 2) $k_p = (J\omega_c^2/(\tau\omega_c + 1))$; and 3) $k_d = k_p\tau$. Although the phase margin will never exceed 90°, you can still calculate appropriate gains for k_p and k_d for $\phi > 90$ using the above straight-line approximation. All simulations given are for $\phi = 80$ degrees, $N = 2$, and $J = 0.293$ kg-m^2. The remaining system parameters are as follows: $T_s = 0.1$ s, $\epsilon_c = 1.0e-6$, $\epsilon = 0.5$, $k_p = 8.02$, and $k_d = 4.1$. Because $\epsilon_c > 0$ and $\epsilon > 0$, Theorem V-A ensures that our system will remain L_2^m-*stable* for fixed time delays. In order to allow for time varying delays which can satisfy Conditions 1) and 2), we chose to use the TCP/IP protocol layers as detailed in Section VI-C.

C. NCS Setup

The experimental setup is shown in Fig. 7. The network is a wireless 802.1 1b ad hoc network with six wireless nodes. One node contains the passive controller written in C, another node contains a Simulink program which simulates the robotic arm. The controller and the plant use the TCP/IP Send and TCP/IP Receive blocks in Simulink to communicate with the controller. The remaining four nodes are used to send disturbance packets onto the network.

In order to evaluate the stability and robustness to time-varying network delays of the proposed architecture, we record the joint angles of the arm and the round-trip delays observed at the plant. The controller produces a trajectory for the robot to follow. The first stage moves the robotic arm from the zero home position to the position of [1 0.8 0.6 0.4 0.2 0] rad within 5 s. For the second stage, the robot remains in place for 5 s. In the third stage, the robot returns to the home position within 5 s.

During a simulation, the controller waits for a connection from the computer containing the passive robotic model. During this time, some or all of the disturbance machines send ping floods to the computer containing the passive controller. When the node containing the passive plant is able to send and receive data successfully, the plant model records
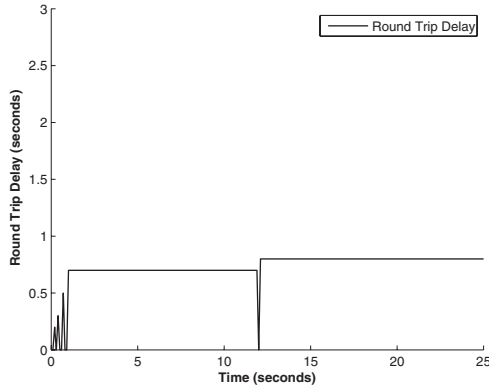
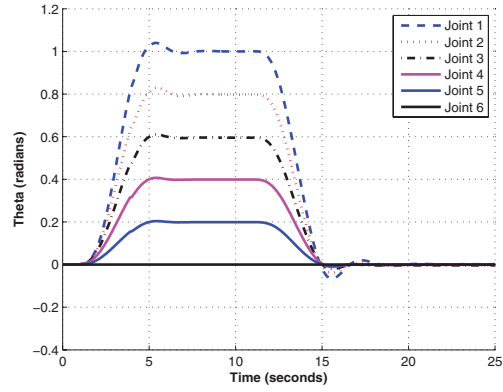Fig. 9.   Packet round-trip delay with no network disturbance.

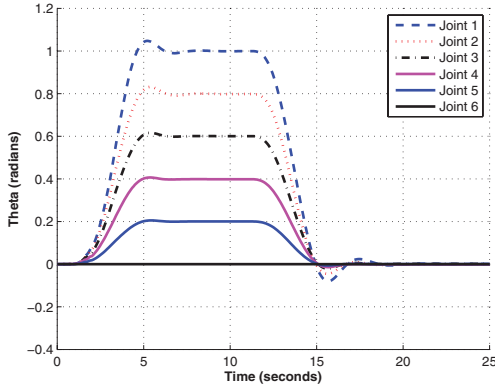Fig. 12.   Robot performance with two disturbance nodes.

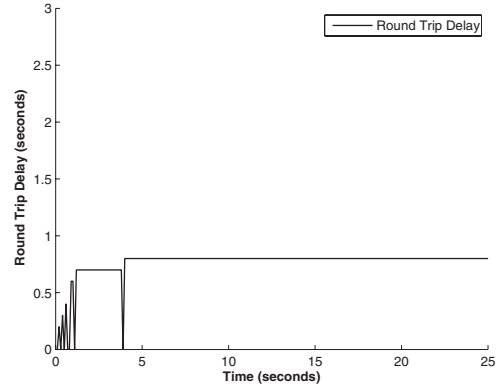Fig. 10.   Robot performance with one disturbance node.

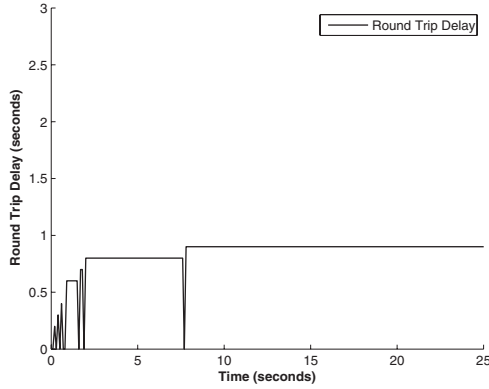Fig. 13.   Packet round-trip delay with two disturbance nodes.

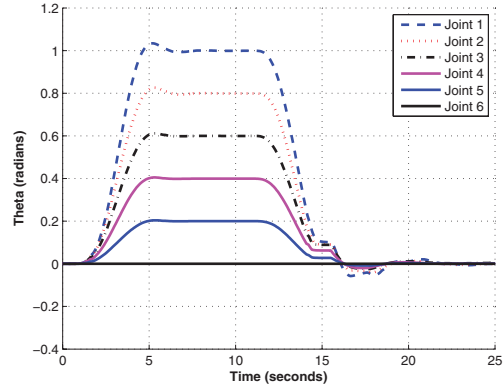Fig. 11.   Packet round-trip delay with one disturbance node.

Fig. 14.   Robot performance with three disturbance nodes.

the packet round-trip time. Specifically, the round-trip delay (Fig. 9) corresponds to the time difference when $u_p[i_{\text{sent}}]$ is sent ($t_{\text{sent}} = i_{\text{sent}}T_s$) and when the corresponding control command arrives back to the plant in the form of a wave variable $v_{\text{ucd}}[i_{\text{arrived}}]$ ($t_{\text{arrived}} = i_{\text{arrived}}T_s$), or in other words, $\Delta t_{\text{round trip}} = (i_{\text{arrived}} - i_{\text{sent}})T_s$.

Experiment 1 (Nominal Conditions): In experiment 1, the controller and plant operate and communicate with each other without any interference due to communication from the disturbance nodes. This experiment shows how the system behaves under nominal conditions. Fig. 8 displays the joint angles of the robotic arm that follow the reference trajectory

provided to the controller. The round-trip network delay, as seen in Fig. 9, is minimal and repeatable, and it has very little effect on the stability of the robot model. The delay is a product of internal processing of both the plant and the controller rather than network delay itself.

Experiment 2 (Network Disturbances): Figs. 10, 12, 14, and 16 show how the robotic model behaves under network disturbance with respective packet round-trip delays plotted in Figs. 11, 13, 15, and 17. During the experiment, each disturbance node outputs ping flood packets as fast as they come back or 100 times/s, whichever is more. When one node or two nodes send out ping floods, the robot behavior is
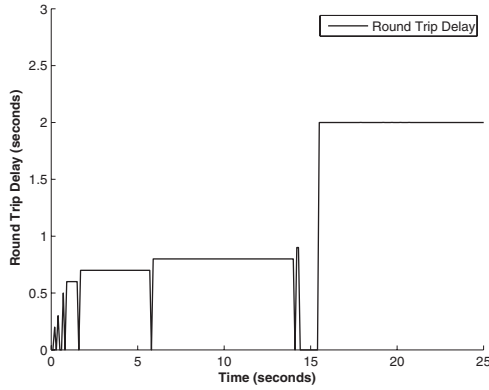
Fig. 15.   Packet round-trip delay with three disturbance nodes.
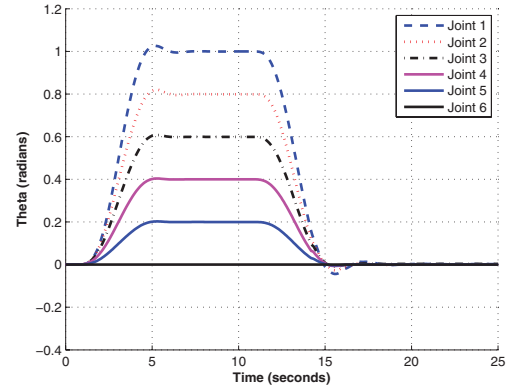

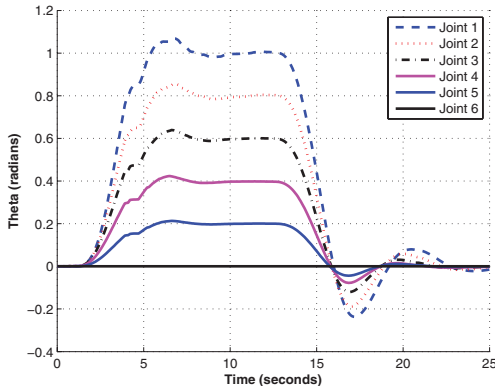Fig. 18.   Robot performance with 100% CPU load.

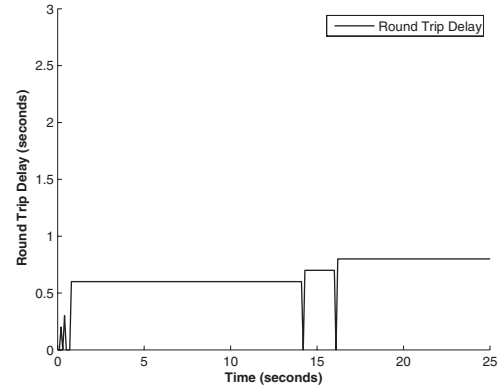Fig. 16.   Robot performance with four disturbance nodes.

Fig. 19.   Packet round-trip delay with 100% CPU load.

program uses the Cygwin/Unix low-level copy program "dd" to continuously write random numbers to a file. This process overloads the CPU of the controller node. Both programs have the same priority, and both share the same single core processor. Figs. 18 and 19 show how the robotic model behaves when the controlling computer is at 100 similar to the nominal case in experiment 1, and Fig. 18 also shows a similar performance to the system in experiment 1. Programs that increase the load on the CPU running on the controller computer had a negligible effect on degrading system performance.
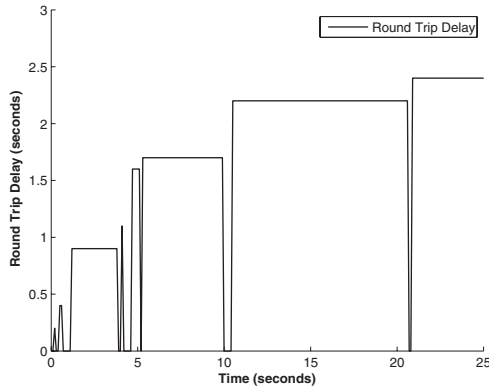

Fig. 17.   Packet round-trip delay with four disturbance nodes.

very close to the nominal case. However, when three and four disturbance nodes participate on the network, the controller computer has difficulty receiving messages from and sending messages to the plant computer. This case demonstrates the advantages of the passive control architecture. When the plant is unable to communicate with the controller, the robot simply stops and waits for the next packet from the controller. This can be seen in Figs. 14 and 16. These results show that in the face of crippling network traffic, the robot remains stable.

Experiment 3 (CPU Disturbances): In experiment 3, the disturbance nodes are silenced. In this experiment, the controller computer executes two programs simultaneously, the passive control program and a disturbance program. The disturbance

VII. CONCLUSION

This paper presented a passive control architecture that offers advantages in building CPS with guaranteed stability independent of network delay uncertainties provided the data transmitted over a given network is only processed once at the respective receiving controller or plant nodes. The proposed approach leads to orthogonality across the controller and implementation design layers and thus empowers model-driven development. We have presented an architecture for a system consisting of a robotic manipulator controlled by a digital controller over a wireless network and we have proved the NCS to be stable. The experimental results validated the significant advantages of the passivity-based architecture, especially in the presence of time-varying delays. Our current and future work focuses on methods that provide an effective

way to interconnect multiple passive systems and controllers as well as an integrated end-to-end tool chain for the model-based design of CPS based on passivity.

REFERENCES

[1] J. Baillieul and P. Antsaklis, "Control and communication challenges in networked control systems," *Proc. IEEE*, vol. 95, no. 1, pp. 9–28, Jan. 2007.

[2] *MATLAB/Simulink Language Technique Computer Version 7.6*, Math-Works, Natick, MA, 2008.

[3] A. L. Sangiovanni-Vincentelli, "Quo vadis sld: Reasoning about trends and challenges of system-level design," *Proc. IEEE*, vol. 95, no. 3, pp. 467–506, Mar. 2007.

[4] A. C. M. Ohlin and D. Henriksson, "True time 1.5-reference manual," Dept. Autom. Control, Lund Univ., Lund, Sweden, Jan. 2007.

[5] H. Kopetz, "The time-triggered architecture," *Proc. IEEE*, vol. 91, no. 1, pp. 92–126, Jan. 2003.

[6] J. Skaf and S. Boyd, "Analysis and synthesis of state-feedback controllers with timing jitter," *IEEE Trans. Autom. Control*, vol. 54, no. 3, pp. 652–657, Mar. 2009.

[7] C. Desoer and M. Vidyasagar, *Feedback Systems: Input-Output Properties*. New York: Academic, 1975.

[8] P. Li and R. Horowitz, "Control of smart machines, part 1: Problem formulation and non-adaptive control," *IEEE/ASME Trans. Mechatron.*, vol. 2, no. 4, pp. 237–247, Dec. 1997.

[9] G. Niemeyer and J.-J. E. Slotine, "Toward force-reflecting teleoperation over the internet," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1998, pp. 1909–1915.

[10] P. Berestesky, N. Chopra, and M. W. Spong, "Discrete time passivity in bilateral teleoperation over the internet," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr.–May 2004, pp. 4557–4564.

[11] D. Lee and P. Li, "Passive coordination control of nonlinear mechanical teleoperator," *IEEE Trans. Robot.*, vol. 21, no. 5, pp. 935–951, Oct. 2005.

[12] S. Stramigioli, C. Secchi, A. J. van der Schaft, and C. Fantuzzi, "Sampled data systems passivity and discrete port-hamiltonian systems," *IEEE Trans. Robot.*, vol. 21, no. 4, pp. 574–587, Aug. 2005.

[13] C. Secchi, S. Stramigioli, and C. Fantuzzi, *Control of Interactive Robotic Interfaces: A Port-Hamiltonian Approach*. New York: Springer-Verlag, 2007.

[14] S. Hirche and M. Buss, "Transparent data reduction in networked telepresence and teleaction systems. Part II: Time-delayed communication," *Presence: Teleoperators Virtual Environ.*, vol. 16, no. 5, pp. 532–542, 2007.

[15] A. Fettweis, "Wave digital filters: Theory and practice," *Proc. IEEE*, vol. 74, no. 2, pp. 270–327, Feb. 1986.

[16] N. Kottenstette and P. J. Antsaklis, "Stable digital control networks for continuous passive plants subject to delays and data dropouts," in *Proc. IEEE 46th Conf. Decision Control*, Dec. 2007, pp. 4433–4440.

[17] N. Chopra, P. Berestesky, and M. Spong, "Bilateral teleoperation over unreliable communication networks," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 2, pp. 304–313, Mar. 2008.

[18] S. Hirche, T. Matiakis, and M. Buss, "A distributed controller approach for delay-independent stability of networked control systems," *Automatica*, vol. 45, no. 8, pp. 1828–1836, 2009.

[19] J. Bao, P. Lee, F. Wangc, W. Zhou, and Y. Samyudia, "A new approach to decentralised process control using passivity and sector stability conditions," *Chem. Eng. Commun.*, vol. 182, no. 1, pp. 213–237, 2000.

[20] B. Ydstie, "Passivity based control via the second law," *Comput. Chem. Eng.*, vol. 26, nos. 7–8, pp. 1037–1048, 2002.

[21] K. Jillson and B. E. Ydstie, "Process networks with decentralized inventory and flow control," *J. Process Control*, vol. 17, no. 5, pp. 399–413, 2007.

[22] N. Kottenstette, X. Koutsoukos, J. Hall, P. Antsaklis, and J. Sztinapovits, "Passivity-based design of wireless networked control systems for robustness to time-varying delays," in *Proc. IEEE 29th Real-Time Syst. Symp.*, 2008, pp. 15–24.

[23] G. Zames, "On the input-output stability of time-varying nonlinear feedback systems part I: Conditions derived using concepts of loop gain, conicity and positivity," *IEEE Trans. Autom. Control*, vol. 11, no. 2, pp. 228–238, Apr. 1966.

[24] J. C. Willems, "Dissipative dynamical systems part I: General theory," *Soviet J. Opt. Technol.*, vol. 45, no. 5, pp. 321–351, 1972.

[25] D. J. Hill and P. J. Moylan, "Stability results for nonlinear feedback systems," *Automatica*, vol. 13, no. 4, pp. 377–382, 1977.

[26] R. J. Anderson and M. W. Spong, "Bilateral control of teleoperators with time delay," in *Proc. IEEE Conf. Decision Control Including Symp. Adaptive Process.*, May 1988, pp. 167–173.

[27] G. Niemeyer and J.-J. E. Slotine, "Telemanipulation with time delays," *Int. J. Robot. Res.*, vol. 23, no. 9, pp. 873–890, 2004.

[28] M. Kuschel, P. Kremer, and M. Buss, "Passive haptic data-compression methods with perceptual coding for bilateral presence systems," *IEEE Trans. Syst. Man. Cybern. A, Syst. Humans*, vol. 39, no. 6, pp. 1142–1151, Nov. 2009.

[29] C. Secchi, S. Stramigioli, and C. Fantuzzi, "Digital passive geometric telemanipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, Sep. 2003, pp. 3290–3295.

[30] W. M. Haddad and V. S. Chellaboina, *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*. Princeton, NJ: Princeton Univ. Press, 2008.

[31] N. Chopra, M. W. Spong, S. Hirche, and M. Buss, "Bilateral teleoperation over the internet: The time varying delay problem," in *Proc. Amer. Control Conf.*, Jun. 2003, pp. 155–160.

[32] N. Kottenstette, H. LeBlanc, E. Eyisi, and X. Koutsoukos, "Multi-rate networked control of conic (dissipative) systems," in *Proc. Amer. Control Conf.*, Jul. 2011, pp. 274–280.

[33] A. van der Schaft, *L2-Gain and Passivity in Nonlinear Control*. New York: Springer-Verlag, 1999.

[34] N. Kottenstette and P. Antsaklis, "Control of multiple networked passive plants with delays and data dropouts," in *Proc. Amer. Control Conf.*, Seattle, WA, Jun. 2008, pp. 3126–3132.

[35] N. Kottenstette and P. J. Antsaklis, "Digital control networks for continuous passive plants which maintain stability using cooperative schedulers," Dept. Elect. Eng., Univ. Notre Dame, Tech. Rep. ISIS-2007-002, 2007, pp. 1–16.

[36] J. Bao, K. Chan, W. Zhang, and P. Lee, "An experimental pairing method for multi-loop control based on passivity," *J. Process Control*, vol. 17, no. 10, pp. 787–798, 2007.

[37] R. Ortega and M. Spong, "Adaptive motion control of rigid robots: A tutorial," in *Proc. IEEE 27th Conf. Decision Control*, 1988, pp. 1575–1584.

[38] W. S. Levine, *Control System Applications*. Boca Raton, FL: CRC Press, 2000.

[39] H. Olsson, K. Astrom, C. Canudas, M. Gafvert, and P. Lischinsky, "Friction models and friction compensation," *Eur. J. Control*, vol. 4, no. 3, pp. 176–195, 1998.

[40] C. Canudas, H. Olsson, K. Astrom, and P. Lischinsky, "A new model for control of systems with friction," *IEEE Trans. Autom. Control*, vol. 40, no. 3, pp. 419–425, Mar. 1995.

[41] N. Barahanov and R. Ortega, "Necessary and sufficient conditions for passivity of the lugre friction model," *IEEE Trans. Autom. Control*, vol. 45, no. 4, pp. 830–832, Apr. 2000.

[42] K. Khayati, P. Bigras, and L.-A. Dessaint, "A multistage position/force control for constrained robotic systems with friction: Joint-space decomposition, linearization, and multiobjective observer/controller synthesis using lmi formalism," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1698–1712, Oct. 2006.

[43] N. Kottenstette, J. Hall, X. Koutsoukos, P. Antsaklis, and J. Sztipanovits, "Digital control of multiple discrete passive plants over networks," *Int. J. Syst. Commun. Control*, vol. 3, no. 2, pp. 194–228, 2011.

[44] J.-H. Ryu, Y. S. Kim, and B. Hannaford, "Sampled and continuous time passivity and stability of virtual environments," *IEEE Trans. Robot.*, vol. 20, no. 4, pp. 772–776, Aug. 2004.

[45] R. Costa-Castello and E. Fossas, "On preserving passivity in sampled-data linear systems by using state observers," in *Proc. Eur. Control Conf.*, 2007, pp. 5276–5281.

[46] S. Monaco, D. Normand-Cyrot, and F. Tiefensee, "From passivity under sampling to a new discrete time passivity concept," in *Proc. IEEE Conf. Decision Control*, Dec. 2008, pp. 3157–3162.

[47] N. Kottenstette, J. Hall, X. Koutsoukos, J. Sztipanovits, and P. Antsaklis, "Passivity-based design of wireless networked control systems subject to time-varying delays," Inst. Software Integrated Systems, Vanderbilt Univ., Nashville, TN, Tech. Rep. ISIS-08-904, 2008.

[48] N. Kottenstette and P. Antsaklis, "Relationships between positive real, passive dissipative, and positive systems," in *Proc. Amer. Control Conf.*, Baltimore, MD, 2010, pp. 409–416.

[49] N. Kottenstette and P. J. Antsaklis, "Wireless digital control of continuous passive plants over token ring networks," *Int. J. Robust Nonlinear Control*, vol. 19, no. 18, pp. 2016–2039, Dec. 2009.

[50] S. Johannessen, "Time synchronization in a local area network," *IEEE Control Syst. Mag.*, vol. 24, no. 2, pp. 61–69, Apr. 2004.

[51] H. Kopetz, A. Ademaj, P. Grillinger, and K. Steinhammer, "The time-triggered ethernet (TTE) design," in *Proc. IEEE Int. Symp. Object-Oriented Real-Time Distrib. Comput.*, May 2005, pp. 22–33.

[52] B. Sundararaman, U. Buy, and A. Kshemkalyani, "Clock synchronization for wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 281–323, 2005.

[53] J. J. Craig, *Introduction to Robotics: Mechanics and Control*. Reading, MA: Addison-Wesley, 1989.

[54] P. Corke, "A robotics toolbox for MATLAB," *IEEE Robot. Autom. Mag.*, vol. 3, no. 1, pp. 24–32, Mar. 1996.

Nicholas Kottenstette (M'00–SM'09) received the M.S. degree from the Mechanical Engineering Department, Massachusetts Institute of Technology, Cambridge, in 1997, and the Ph.D. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, in 2008.

He is the Director of Cyber-Physical Systems Research and Development, WW Technology Group (WWTG), Worcester, MA. His research develops constructive resilient networked control principles for cyber physical systems. The principles he has developed apply to nonlinear affine systems, including fixed-wing aircraft, quadrotor aircraft, robotic thermal semiconductor manufacturing, alternative energy generation, digital signal processing, and suspension systems. Prior to joining WWTG, he was a Research Scientist with Vanderbilt University, Nashville, TN, during which he co-authored numerous research papers published in journals and conference proceedings on networked control of robotic and aircraft systems. Prior to joining Vanderbilt, he was a Principal Software Engineer with the Advanced Technology Group, MKS Instruments, Andover, MA. While at MKS, he was a co-inventor of networked pressure and mass flow controllers for semiconductor manufacturing, which resulted in 14 U.S. patents.

Joseph F. Hall III (M'02) received the B.S. degree in engineering with electrical emphasis and the B.S. degree in computer science from Union University, Jackson, TN, in 2005, and the M.S. degree in electrical engineering from Vanderbilt University, Nashville, TN, in 2007.

His initial graduate work focused on cognitive control in humanoid robotics which culminated in a thesis focused on an internal rehearsal module for the central executive agent, which was designed and implemented in the Computer and Information Science Laboratory, Vanderbilt University, under the direction of Dr. P. Ratanaswasd. This internal rehearsal system allowed Intelligent SoftArm Control, the cognitive robot, to try certain behaviors internally and ascertain consequences before the robot executed the behavior in real life. His current research interests include robotic cognitive controls, robotic manipulator kinematics and dynamics, and (Linux-based) real-time digital control of robotic manipulators using passivity-based techniques.

Xenofon Koutsoukos (S'96–M'00–SM'08) received the Ph.D. degree in electrical engineering from the University of Notre Dame, Notre Dame, TN, in 2000.

He is an Associate Professor with the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN. He is also a Senior Research Scientist with the Institute for Software Integrated Systems, Vanderbilt University. He was a Research Staff Member with Xerox Palo Alto Research Center, Palo Alta, CA, from 2000 to 2002, working in the area of embedded collaborative computing. He has published numerous articles in journals and conference proceedings and is the co-inventor of four U.S. patents. His current research interests include cyber-physical systems, with emphasis on formal methods, distributed algorithms, diagnosis and fault tolerance, and adaptive resource management.

Dr. Koutsoukos was the recipient of the National Science Foundation Career Award in 2004 and the Excellence in Teaching Award in 2009 from the Vanderbilt University School of Engineering.

Janos Sztipanovits (M'86–SM'90–F'01) graduated from the Technical University of Budapest, Budapest, Hungary, in 1970. He received the Candidate of Technical Sciences degree from the Hungarian Academy of Sciences, Budapest, in 1980, and the Distinguished Doctor degree (Golden Ring of the Republic) in 1982.

He is currently the E. Bronson Ingram Distinguished Professor of engineering with the Electrical Engineering and Computer Science Department, Vanderbilt University, Nashville, TN. He is the Founding Director of the Institute for Software Integrated Systems. From 1999 to 2001, he worked as a Program Manager and Acting Deputy Director of the DARPA Information Technology Office. During the past two decades he has conducted research on model-integrated computing, structurally adaptive systems, and embedded software and systems. He has published over 160 research papers and co-authored two books.

Prof. Sztipanovits holds the Chair of the ACM Special Interest Group on Embedded Systems.

Panos Antsaklis (S'74–M'76–SM'86–F'99) received the Graduate degree from the National Technical University of Athens, Athens, Greece, and the M.S. and Ph.D. degrees from Brown University, Providence, RI.

He is the Brosey Professor of electrical engineering with the University of Notre Dame, Notre Dame, IN. His research addresses the problems of control and automation and examines ways to design control systems that will exhibit a high degree of autonomy. He has co-authored two research monographs on discrete-event systems, two graduate textbooks on linear systems, and has co-edited six books on intelligent autonomous control, hybrid systems, and networked embedded control systems. His current research interests include cyber-physical systems and problems in the interdisciplinary research area of control, computing, and communication networks, and on hybrid and discrete-event dynamical systems.

Prof. Antsaklis was the recipient of the Engineering Alumni Medal of Brown University in 2006. He is a fellow of the International Federation of Automatic Control and the American Association for the Advancement of Science. He is the Editor-in-Chief of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL.