Optimal Cross-Layer Design of Sampling Rate Adaptation and Network Scheduling for Wireless Networked Control Systems

Jia Bai, Emeka P. Eyisi, Fan Qiu, Yuan Xue, Xenofon D. Koutsoukos
Department of Electrical Engineering and Computer Science, Vanderbilt University
Email: {jia.bai, emeka.eyisi, fan.qiu, yuan.xue, xenofon.koutsoukos}@vanderbilt.edu

***Abstract*—Wireless Networked Control Systems (NCS) are increasingly deployed to monitor and control Cyber-Physical Systems (CPS). To achieve and maintain a desirable level of performance, NCS face significant challenges posed by the scarce wireless resource and network dynamics. We consider NCS consisting of multiple physical plant and digital controller pairs communicating over a multi-hop wireless network. The control objective is for the plants to follow the reference trajectories provided by the controllers. This paper presents a novel optimization formulation for minimizing the tracking error introduced due to discretization and packet delays and losses. The optimization problem maximizes a utility function that characterizes the relationship between the sampling rate and the capability of disturbance rejection of the control system. The constraints represent the wireless network capacity and the end-to-end delay requirements. The solution to this optimization problem leads to a joint design of sampling rate adaptation and network scheduling, which can be naturally deployed over the existing layered networking systems. Based on a passivity-based control framework, we show that the proposed cross-layer design can achieve both stability and performance optimality. Simulation studies conducted in an integrated simulation environment consisting of Matlab/Simulink and ns-2 demonstrate that our algorithm is able to provide agile and stable sampling rate adaptation and achieve optimal NCS performance.**

***Index Terms*—wireless networked control system; cross-layer design; sampling rate adaptation; network scheduling**

I. Introduction

The integration of physical systems through computing and networking has become a trend, known as Cyber-Physical Systems (CPS). Many real-world CPS such as automotive vehicles and distributed robotics, are monitored and controlled by Networked Control Systems (NCS), where information among sensors, controllers and actuators is exchanged via a communication network. NCS are increasingly deployed over wireless networks, as they provide great convenience in terms of deployment and mobility support [1], [2]. However in a wireless networking environment, the stability and performance of the control system are greatly affected by its limited and dynamic resource availability.

Three major approaches have been investigated in literature to address the challenges in designing wireless NCS. The first approach, independent of the network protocol design, investigates the design of the control layer with a goal of achieving the desired performance despite of the underlying network uncertainties (*e.g.*, [1], [3]). Alternatively, the network-centric approach focuses on reliable and timely packet deliveries, independent of the control system. Yet without the knowledge and support from the other components of the NCS, these approaches can hardly achieve both stability and optimal performance simultaneously (*e.g.*, [4], [5]). To ensure the stability and optimize the performance of a NCS, co-design of the control system and the networking system has been investigated where the operation points of these two systems are coordinated. Existing literature ([6], [7], [8]) either makes simplifying assumptions on the network models or involves too many interactions between the control and the networking systems, which prevents efficient layer abstraction and encapsulation, hindering broader adoption for real-world deployment.

In this paper, we consider NCS consisting of multiple physical plant and digital controller pairs communicating via a multi-hop wireless network, where the plants follow the reference trajectories provided by the controllers. The performance of the NCS is characterized by the tracking errors of the plants which are introduced from two sources: 1) discretization of the controller and the noise disturbance from the operating environment and 2) packet delays and losses caused by network congestion and dynamics. Both sources of error are related to the sampling rate of the control system. Intuitively, high sampling rates allow frequent state updates and provide NCS with better capability to reduce the effect of environmental disturbances. On the other hand, high sampling rates increase the network load, which increases the possibility of packet loss and delay [9] and deteriorates the tracking error.

We transform the NCS performance objective in terms of tracking error minimization into an optimization problem. The optimization aims at maximizing a utility function that characterizes the relationship between the sampling rate and the capability of disturbance rejection of the control system (i.e., minimizing the discretization-induced tracking error); and the constraints of the sampling rate come from the wireless network capacity and the requirement of packet (i.e., bounding the network-induced tracking errors). The solution to this optimization problem leads to a cross-layer design of control system sampling rate adaptation and network scheduling, where the sampling rate adaptation determines the bandwidth demands of the network, and the scheduling at the media access control layer resolves the location-dependent interference and determines the available resource capacity of each wireless link.

This sample rate optimization problem, however, is non-trivial to solve. The tight coupling of the sampling rate and the required delay bound of the control system (i.e., the delay needs to be less than the sampling time) poses a nonlinear

constraint, which has never been addressed in the existing rate optimization solutions ([10], [11]). To solve this problem, we present a coupled-loop approach. In the inner loop, a relaxed problem, where the delay bound is fixed and independent of the sampling rate, is solved via dual decomposition. In particular, a double-price scheme is employed to regulate the sampling rate traffic demand and the wireless capacity supply. The capacity price regulates the resource usage at the wireless link level, and the delay price regulates the relationship between the achieved packet delay and the required delay bound at the end-to-end flow level. The control system then adapts its sampling rate based on its utility function so that its net profit, which is the difference between the utility and the cost (product of price and rate), is maximized. The convergence and optimality of this algorithm is proven. The outer loop determines the optimal delay bounds progressively based on the converged sampling rate from the inner loop. The proposed algorithm naturally leads to a distributed cross-layer implementation.

The main contributions of this paper are summarized as follows. First, we present a new formulation for NCS performance optimization by decoupling its performance metric (tracking error) into two parts – discretization and network effect, which are formulated into the objective and the constraints of an optimization problem respectively. This formulation leads to a cross-layer joint design of sampling rate adaptation and network scheduling which can be easily deployed on existing control systems and networks. We employ a control design approach based on passivity, and we formally prove that the stability and the performance optimality of NCS can be simultaneously achieved. Second, we present a distributed algorithm that solves the NCS performance optimization problem and resolves the complex interdependency between delay and sampling rate. By introducing a novel Virtual Link Capacity Margin (VLCM) parameter that can be adjusted to control the delay and the rate over a wireless link, our solution does not depend on a specific model of packet arrival processes and is suitable for NCS systems with packet arrivals that are not characterized as Poisson process (which is an assumption usually used in networking delay analysis). Third, our solution is evaluated in an integrated simulation environment [12] that consists of Matlab and ns-2. Using ns-2 – a packet-level network simulator that implements all the details of the network protocol stack, allows highly accurate evaluation of network effects on the NCS performance, which is impossible by using Matlab/Simulink alone.

The remainder of this paper is organized as follows. Sec. II briefly reviews the related works. In Sec. III, we present the control system model and the wireless network model. In Sec. IV and V, we formulate the problem of optimal rate allocation and derive the double-price-based rate adaptation algorithm. We evaluate the algorithm in different multi-hop scenarios using our Networked Control System Wind Tunnel (NCSWT) simulation tool in Sec. VI. We conclude the paper in Sec. VII.

II. Related Work

Closely related work include the approach presented in [13] which focuses on compensating for exponentially bounded long dropout bursts in the network by reconfiguring the controller or the network to guarantee stability. In contrast, our approach assumes a passivity-based framework by design which inherently guarantees stability and focuses on the performance optimization with an integrated design of the controller and the network. The framework in [14] obtains the controller behavior from the aggregate computation of different nodes in the network and it presents a new paradigm that is different from the classical NCS model, which is considered in our work. A methodology for determining optimal sampling rates for feedback loops focusing on WirelessHart networks is presented in [15] .

Utility functions have been used in [16] to capture the relationship between the sampling rate and control performance. However, the optimization problem formulation in [16] leads to an offline solution which only deals with fixed computing resources, while our solution is fully distributed and can handle dynamic wireless bandwidth resource. Further, our work considers the interaction of the sampling rate and the end-to-end delay experienced by the control systems, which has not been addressed in the existing rate optimization solutions [10], [11], [17], [18], [8].

III. Problem Description

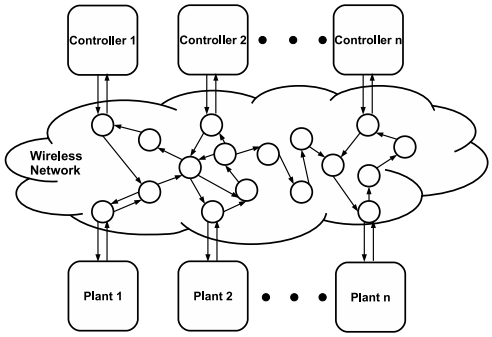

Fig. 1. NCS over multi-hop wireless networks

We consider NCS consisting of multiple plants and digital controllers communicating via a multi-hop wireless network, as shown in Fig. 1. The objective of the control system is that the plants follow the reference trajectories provided by the controllers to complete certain tasks. For example, in a manufacturing factory, a group of robotic operators perform the task of moving objects from one place to another. The network controllers receive desired reference trajectory from the operators and are responsible for ensuring the movement of each robot tracks the desired trajectory.

A. Control System Model

A continuous-time plant is described by

$$\dot{x}_p(t) = A_p x_p(t) + B_p u_p(t) + B_w w(t) \quad (1)$$

$$y_p(t) = C_p x_p(t) \quad (2)$$

where $x_p(t) \in \Re^n$ denotes the plant state, $u_p(t) \in \Re^m$ denotes the control input, $w(t) \in \Re^m$ is the disturbance input, and $y_p(t) \in \Re^m$ is the plant output. A_p, B_p, and B_w define the plant state matrices and C_p defines the plant output matrix.

The state-space representation of the continuous-time controller is

$$\dot{x}_c(t) = A_c x_c(t) + B_c u_c(t) \quad (3)$$

$$y_c(t) = C_c x_c(t) + D_c u_c(t) \quad (4)$$

where $x_c(t) \in \Re^n$ denotes the controller state, and $u_c \in \Re^m$ denotes the error signal, or the difference between the plant output $y_c(t) \in \Re^m$ and the reference signal input $r(t) \in \Re^m$. A_c and B_c define the controller state matrices, while C_c and D_c define the controller output matrices. Let the reference signal denote by $r(t)$. The tracking error of the system is

$$e_{rr}(t) = r(t) - y(t) \quad (5)$$

The controller is implemented as a discrete-time control system. We consider sampling instants $t_k \in \mathbb{R}$, $k = 0, 1...$, with $t_{k+1} > t_k, t_0 = 0$ and we define the sampling interval as $T_k = t_{k+1} - t_k$. In order to simplify the notations, let $x(k+1)$ represent $x(t_{k+1})$, the signal $x(t)$ sampled at time instant t_{k+1}.

B. Wireless Network Model

We model a multi-hop wireless network as a directed graph $G = (V, L)$, where V is the set of wireless nodes in the network. The nodes communicate with each other via directed wireless links $l \in L$. Such a network supports a set of control systems H. For each NCS $h \in H$, its plant and controller are deployed on two different nodes in the network. The traffic from the controller to the plant and the traffic backwards generate two end-to-end flows denoted as $F(h)$. We collect all end-to-end flows in the network into set F. An end-to-end flow f may go through multiple hops in the network and traverse a sequence of links defined by the routing policy. We use set $L(f)$ to represent all the links along the route of flow f and $F(l)$ to denote all the flows that traverse link l.

C. NCS Performance Optimization

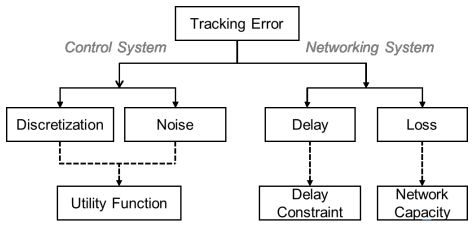

Fig. 2. Decompose tracking error based on its source

The NCS performance can be characterized by the tracking error of the plant systems. The main focus of this paper is to minimize the tracking error of the NCS deployed over the multi-hop wireless network while maintaining certain level of fairness among the plant-controller pairs. As shown in Fig. 2, there are two main sources of error. When a continuous-time control system is discretized, its response to environmental disturbances degrades compared to the response of the idealized continuous system. The level of the degradation depends on the sampling rate, which determines how well the digital controller approximates the continuous controller. High sampling rate allows frequent state updates and thus provides better capability to reduce the effect of environmental disturbances and minimize the tracking error. Packet loss and delay also deteriorate the tracking error. We focus on the congestion-induced packet loss and delay. Network congestion appears when the traffic demand overwhelms the capacity supply. While the sampling rate determines the network traffic demand, the network resource management mechanisms such as media access control scheduling allocate appropriate capacity to each wireless links.

Optimizing the NCS performance requires the coordination between the control system and the networking system. The control system needs to have the capability to adapt its sampling rate based on the resource utilization information from the network. The networking system should schedule its wireless transmission to meet the resource needs from the control system. This paper studies how to minimize the NCS tracking error via joint sampling rate adaptation and networking scheduling[1].

IV. Optimization Framework For Tracking Error Minimization

In this section, we present our control system design and formulate the problem of NCS tracking error minimization as a sampling rate optimization problem. We first show that our passivity-based control system design is able to ensure system stability with time-varying sampling time. Then we define the optimization objective through a utility function which characterizes the relationship between the sampling rate and the capability of disturbance rejection of the control system (*i.e.*, minimizing the discretionarily-induced tracking errors). The optimization constraints are based on the wireless network schedulability and the NCS delay requirement.

A. Passivity-based control system – ensuring system stability with time-varying sampling time

Fig. 3 shows our passivity-based control system design. A passive system is defined as a system with bounded output energy such that the system does not produce more energy than what was initially stored. We assume the plant system is either passive. A large class of systems can be "passified" by adding local control and filter components [19][20]. The controller $G_c(s)$ is designed so that the plant tracks the reference $r(k)$ and is also assumed to be passive. The control architecture uses (1) a discretization approach defined by the Inner Product Equivalent Sampling and Hold (IPESH) transform, which is composed by the Inner Product Equivalent Sampling (IPES) and Zero Order Hold (ZOH) blocks and (2)

[1]This paper assumes fixed network routing, which is known a priori.

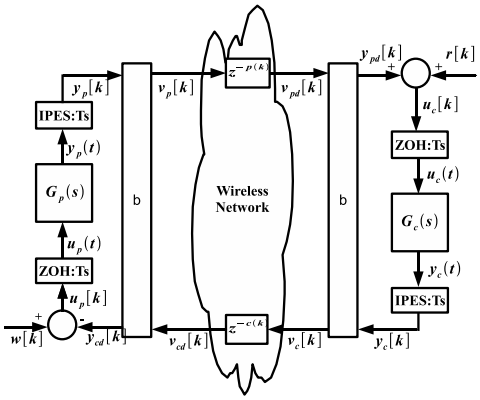

Fig. 3. Passivity Based Control Architecture Over Wireless Networks

a bilinear transform b for converting the control signals into wave variables for communication over a wireless network. These transformations ensure that the NCS is passive and stable in the presence of time-varying delays and packet loss[2]. In order to Next, we show that the NCS is ensured with time varying sampling time, which allows us to use sampling rate adaptation.

A passive continuous-time linear time invariant (LTI) system can be converted to a discrete-time passive system at a varying sampling time, T_k, with the discrete-time state space equations described as

$$x_p(k+1) = \Phi_k x(k) + \Gamma_k u(k) \quad (6)$$

$$y_p(k) = C_{dk} x(k) + D_{dk} u(k) \quad (7)$$

In [22][23], it is shown that in order to obtain a passive discrete-time equivalent of a LTI passive continuous-time system for a given fixed sampling time T, the IPESH is used to compute the system coefficients, Φ_k, Γ_k, C_{dk} and D_{dk} to preserve passivity.

The case of discretization with time-varying sampling time can be deduced by applying the IPESH for each resulting sampling time, T_k, hence ensuring passivity of the discretization at each sampling instant and thus the overall passivity of the discrete-time system for a given time interval. This implies that the new system coefficients are redefined as $\Phi_k = \Phi(T_k)$, $\Gamma_k = \Gamma(T_k)$, $C_{dk} = C_d(T_k)$ and $D_{dk} = D_d(T_k)$. By ensuring the passivity of the discrete-time system, the stability of the resulting discrete-time system is guaranteed.

B. Utility function - modeling error from discretization

To characterize the impact of sampling rate on tracking error, we first introduce a utility function based on the comparison of the disturbance rejection capability of the discrete-time system with its continuous-time counterpart.

1) Continuous-time control system: By [24], the covariance matrix of the zero-mean white noise process of the continuous-time system can be defined as

$$E[w(t)w^T(t+\tau)] = Q\delta(\tau) \quad (8)$$

[2]We refer readers to [21], [20] for a detailed description and proofs.

where E denotes the expected value and Q represents the power spectral density of w, or the continuous-time noise covariance matrix. The power spectral density can also be referred to as the "white noise intensity" or mean-square spectral density. The continuous-time state covariance matrix P_c can be described by

$$P_c(t) = E[x(t)x^T(t)] \quad (9)$$

Based on the knowledge of Q, the steady state value of the state covariance can be obtained by the equation [25]

$$A_{cl}P_c + P_c A_{cl} + B_{wcl} Q B_{wcl}^T = 0 \quad (10)$$

where the matrices A_{cl} and B_{wcl} denote the closed loop matrices of the continuous-time system, or the coefficients of $x(t)$ and $w(t)$ respectively. From the resulting state covariance matrix, the root mean square of a state can then be determined. The Root-Mean-Square (RMS) of the plant states is equivalent to the standard deviation of one of the plant states. For example, if a system has only one plant state variable x_p, and its plant state covariance is $v(x_p)$, the RMS of the plant state is equal to $\sqrt{v(x_p)}$. When a system has several plant state variables, we can use the plant state covariance from one of them to calculate the RMS of all the plant states.

2) Discrete-time control system: Based on the knowledge of the continuous-time noise covariance matrix Q, the discrete-time noise covariance matrix Q_d can be obtained using the Van Loan's algorithm [24] and can be defined as

$$Q_d = \int_0^{T_f} \Phi(\tau) B_{wcl} Q B_{wcl}^T \Phi^T(\tau) d\tau \quad (11)$$

where Φ is the closed loop matrix, or the discrete-time state coefficient of $x(k)$, and B_{wcl} denote the closed loop matrix of the continuous-time system, or the coefficient of $x(t)$.

The steady state discrete-time state covariance matrix can then be obtained from the following equation

$$\Phi P_d \Phi^T + Q_d = P_d \quad (12)$$

From the resulting state covariance matrix, the discrete RMS of the plant state can then be determined in a similar way as the continuous-time case.

3) Utility function formulation: We now define the utility function of a control system as a function of its sampling rate $1/T$ using the ratio of RMS between the discrete-time system with the continuous-time counterpart. Thus, the utility function reflects the amount of degradation of the system response to the white noise compared to the continuous closed loop system.

$$U(1/T) = \frac{RMS_{\texttt{continuous}}}{RMS_{\texttt{discrete}}(T)} \quad (13)$$

To demonstrate the definition of our utility function, we consider a single-input-single-output (SISO) linear-time invariant (LTI) system, without loss of generality and show its utility function in Fig. 4. As we could see, the utility function is a strictly concave function of the sampling rate. The concavity of the utility function reflects the marginal return on the control performance when its sampling rate increases.

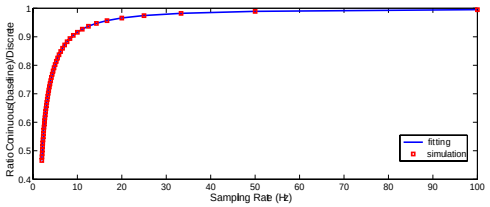

Fig. 4. Example utility function for the control system where the transfer function of the plant is $G_p(s) = \frac{1}{Js}$, the transfer function of the controller is $G_c(s) = \frac{K_p + K_d s}{s}$, where $J = 2.93$, $K_d = 32.1$ and $K_p = 8.2$.

4) Relationship between utility function and tracking error: In the continuous-time system, considering the closed loop equations, the system response of the plant can be described as

$$x(t) = e^{A_{cl}t}x(0) + e^{A_{cl}t}\int_0^t e^{-A_{cl}\tau}B_{cl}r(\tau)d\tau + e^{A_{cl}t}\int_0^t e^{-A_{cl}\tau}B_{wcl}w(\tau)d\tau \quad (14)$$

$$y(t) = C_{cl}e^{A_{cl}t}x(0) + C_{cl}e^{A_{cl}t}\int_0^t e^{-A_{cl}\tau}B_{cl}r(\tau)d\tau - C_{cl}e^{A_{cl}t}\int_0^t e^{-A_{cl}\tau}B_{wcl}w(\tau)d\tau \quad (15)$$

Recall that the tracking error of the system $e_{rr}(t) = r(t) - y(t)$. From (15), the output response of the plant has two main components that contribute towards the tracking error. The first component is the plant response to the reference input, $r(t)$, and the other is the plant response to the disturbance input, $w(t)$. The passive controller is designed to ensure the plant's response to the reference input minimizes the tracking error. The system achieves a certain level of disturbance rejection. The contribution of the input disturbance can be characterized by covariance of the tracking error.

From (15), and the fact that $r(t)$ is not stochastic we have that $E(r(t)y^T(t)) = E(r(t)y^T(t)) = 0$ and $E(r(t)r^T(t)) = 0$. The covariance of the tracking error can be described by

$$C_e(t) = E[e(t)e^T(t)] = E(y(t)y^T(t)) \quad (16)$$

This essentially implies that the covariance of the error is equal to the output covariance. Based on the knowledge of Q, the steady state value of the output covariance can be obtained by the equation [25]

$$C_e = C_{cl}P_cC_{cl}^T \quad (17)$$

C. Delay and capacity constraints – bounding error from network

1) Capacity constraint: To limit the effect of packet loss caused by network congestion on the tracking error, we need to limit the network load within its capacity. Wireless network communication is subject to location dependent interference. Thus the achievable capacity of each wireless link is related to the scheduling algorithm. We adopt the conflict graph [26] concept to model wireless interference. Each vertex in the conflict graph represents a wireless link of the original network and there is an edge between two vertices if their corresponding wireless links interfere with each other. The communications along wireless links are scheduled on a slotted time basis. In each time slot, one independent set[3] I of the conflict graph is selected and only the links corresponding to the vertices in I are allowed to transmit because they are interference free. Let c_l be the channel capacity. A L-dimension column vector r^I is used to represent the capacity vector of I, where $r_l^I = c_l$ if $l \in I$, and $r_l^I = 0$ otherwise. We adopt the model of feasible capacity region Λ as introduced in [27] to model the feasible link capacity allocation. The feasible capacity region is a convex hull, which is defined as $\Lambda := \sum_I \alpha_I r^I$, where $\sum_I \alpha_I = 1$ and $\alpha_I \geq 0$. *Scheduling* essentially determines the capacity allocation $\hat{\boldsymbol{c}} = (\hat{c}_l, l \in L)$ of the links, where $\hat{c}_l$ is the average capacity over time based on the scheduling. Obviously, $\hat{\boldsymbol{c}} \in \Lambda$. To limit the packet congestion losses, the aggregated traffic load on any wireless link $l \in L$ should be no more than its achievable capacity $\hat{c}_l$.

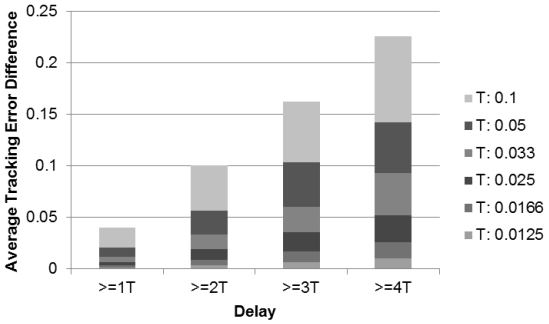

Fig. 5. Impact of Delay On the NCS Average Tracking Error

2) Delay effect on tracking error: To determine the effect of delay on the tracking error, we perform a set of simulation studies using NCSWT [12] over a NCS with one pair of plant and controller. Based on the assumption that the discrete plant/controller systems update and process data received only at sampling instants, the delays viewed from the control systems' perspective are integral multiple of the sampling interval. We vary the sampling time and manually introduce delays which are integral multiples of the sampling time. Then we evaluate the average tracking error difference, which is the difference between the time-averaged tracking error with delay introduced and that without any delay.

From the experiment, we observe that when the delay is within one sampling time, the tracking error difference remains zero. Fig. 5 shows the effect of delay on the tracking error difference when it is larger than the sampling time. We observe that the error increases superlinearly when the delay increases beyond one sampling time. Based on this observation, we bound the average end-to-end delay of control system flows to their system sampling time.

[3]The independent set of a graph is a set of vertices within which no edge exists between any two vertices.

3) Controlling delay with $VLCM$: Providing delay assurance is notoriously difficult in wireless networks. The main difficulty comes from the complex interactions between traffic arrival and departure, which is shaped by the network scheduling. Most of the existing works on delay analysis make explicit assumptions on the packet arrival process (*e.g.*, Poisson arrivals) [28], which do not reflect the NCS traffic characteristics. Here we look for a general delay control method which is not limited to a predefined packet arrival process. In particular, we introduce a parameter *Virtual Link Capacity Margin* ($VLCM$) σ_l of link l defined as follows, to limit the maximum allowable rate m_l.

$$\sigma_l = \hat{c}_l - m_l, \text{ with } m_l < \hat{c}_l, \forall l \in L \tag{18}$$

We regard the link delay (*i.e.*, average packet delay along the link) as a function of the virtual link capacity margin $\varphi(\sigma_l)$. Then the average delay of flow f is the sum of all link delays along its route.

D. Optimization Framework

Recall that each control system is associated with two flows. Let $z_h = \frac{sample\_size}{T_h}$ be the traffic rate of one flow for the control system h, where $sample\_size$ is the size of the sample and T_h is its sampling time. And $T_{h(f)}$ is the sampling time of control system h which flow f is associated with. Thus, the maximum allowable rate satisfies $m_l \geq \sum_{h\in H: f\in F(h)\&f\in F(l)} z_h$. We overload U_h as a function of traffic rate for control system h, as defined by Eq. (13). Now we formulate the optimal sampling rate allocation problem as follows:

$$W: \quad \max \quad \sum_{h\in H} U_h(z_h) \tag{19}$$

$$\text{s.t.} \quad \sum_{h\in H: f\in F(h)\cap F(l)} z_h \leq \hat{c}_l - \sigma_l, \forall l \in L \tag{20}$$

$$\sum_{l\in L(f)} \varphi(\sigma_l) \leq T_{h(f)}, \forall f \in F \tag{21}$$

$$\text{over} \quad \hat{\boldsymbol{c}} \in \Lambda \tag{22}$$

The objective of the nonlinear problem is to maximize the aggregate utility of all control systems in the network. This objective minimizes the tracking error induced by discretization and maintains certain fairness among all the plant-controller pairs [29]. Inequality (20) represents the wireless capacity constraint for each wireless link. Note that the $VLCM$ σ_l is introduced here to control the link delay. Inequality (22) defines the scheduling feasibility. Inequality (21) is the flow delay constraint where the average flow delay is bounded by the sampling interval of its control system. It is important to note that there is a possibility that the optimal solution of sampling time that minimizes the tracking error may fall below the delay bound. We choose to incorporate this delay bound (21) in our problem formulation for two reasons. First, from Fig. 5, we observe that the tracking error increases super-linearly with respect to delay when the delay goes beyond the sampling time; while the utility only increases sub-linearly with respect to sampling rate. Intuitively, this implies the marginal benefit of increasing the sampling rate is overweighed by the marginal penalty of pushing the delay beyond the sampling time. Based on this intuition, we bound the average delay by the sampling period. On the other hand, without this delay bound constraint, providing a formulation that fully captures the complex interaction among sampling time/rate, delay, delay-introduced error, and discretization-introduced error will lead to an intractable optimization problem, where identifying a distributed solution is even harder.

V. Distributed Cross-Layer Algorithm

A. Solution overview

Problem W is non-trivial due to the complicated interactions between the $VLCM$s, the sampling rates and the end-to-end delays. The tight coupling of the sampling rate and the required delay bound of the control system (i.e., the delay needs to be less than the sampling time) poses a nonlinear constraint, which has never been addressed in the existing rate optimization solutions ([10], [11], [30]). To solve this problem, we first relax the delay constraint and consider the optimization problem with a fixed delay requirement. Then we show how to adjust the delay requirement to achieve the optimal solution of the original problem W.

B. Cross-layer algorithm with fixed delay

The optimization framework with a fixed delay requirement can be written as

$$W1: \quad \max \quad \sum_{h\in H} U_h(z_h) \tag{23}$$

$$\text{s.t.} \quad \sum_{h\in H: f\in F(h)\cap F(l)} z_h \leq \hat{c}_l - \sigma_l, \forall l \in L \tag{24}$$

$$\sum_{l\in L(f)} \varphi(\sigma_l) \leq D_{h(f)}, \forall f \in F \tag{25}$$

$$\text{over} \quad \hat{\boldsymbol{c}} \in \Lambda \tag{26}$$

where the constraint (21) is replaced by (25), in which $D_{h(f)}$ is the delay requirement of control system h.

1) Double-Price Algorithm: Direct solution to $W1$ requires global coordination of all network components, such as flows, and links, which is computationally expensive. We consider its dual decomposition. Let $\boldsymbol{\nu} = \{\nu_l, l \in L\}$ and $\boldsymbol{\mu} = \{\mu_f, f \in F\}$ be the Lagrange multipliers with respect to constraints (24) and (25) respectively. The Lagrangian of (23) is:

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{z}, \boldsymbol{\nu}, \boldsymbol{\sigma}, \boldsymbol{\mu}, \hat{\boldsymbol{c}}) \\
&= \sum_{h\in H} U_h(z_h) - \sum_{l\in L}[\nu_l\sigma_l + \sum_{f\in F(l)} \varphi(\sigma_l)\mu_f] \\
&\quad - \sum_{h\in H} z_h(\sum_{l\in L(f)\&f\in F(h)} \nu_l) + \sum_{f\in F} \mu_f D_{h(f)} + \sum_{l\in L} \nu_l \hat{c}_l
\end{aligned}$$

The dual of $W1$ is

$$\bar{D}(\boldsymbol{\nu}, \boldsymbol{\mu}) = \min_{\boldsymbol{\nu}\geq 0, \boldsymbol{\mu}\geq 0} D(\boldsymbol{\nu}, \boldsymbol{\mu}) \tag{27}$$

where

$$
\begin{aligned}
& D(\boldsymbol{\nu},\boldsymbol{\mu}) \qquad (28)\\
&= \max_{\boldsymbol{z},\boldsymbol{\sigma},\hat{\boldsymbol{c}}} L(\boldsymbol{z},\boldsymbol{\nu},\boldsymbol{\sigma},\boldsymbol{\mu},\hat{\boldsymbol{c}})\\
&= \max_{\boldsymbol{\sigma}} \left\{ -\sum_{l\in L}[\nu_l\sigma_l + \sum_{f\in F(l)} \varphi(\sigma_l)\mu_f] \right\}\\
&\quad + \max_{\boldsymbol{z}} \left\{ \sum_{h\in H}[U_h(z_h) - z_h \sum_{l\in L(f)\&f\in F(h)} \nu_l] \right\}\\
&\quad + \max_{\hat{\boldsymbol{c}}} \left\{ \sum_{l\in L} \nu_l\hat{c}_l \right\} + \sum_{f\in F} \mu_f D_{h(f)}
\end{aligned}
$$

The solution $(\boldsymbol{z}^*,\boldsymbol{\sigma}^*,\hat{\boldsymbol{c}}^*)$ to (28) should satisfy:

$$
z_h^* = \arg\max_{z_h} \left\{ \sum_{h\in H} \left(U_h(z_h) - z_h \sum_{l\in L(f)\&f\in F(h)} \nu_l \right) \right\} \qquad (29)
$$

$$
\sigma_l^* = \arg\max_{\sigma_l} \left\{ -\sum_{l\in L} \left(\sum_{f\in F(l)} \varphi(\sigma_l)\mu_f + \nu_l\sigma_l \right) \right\} \qquad (30)
$$

$$
\hat{c}_l^* = \arg\max_{\hat{c}_l\in\Lambda} (\sum_{l\in L} \nu_l\hat{c}_l) \qquad (31)
$$

Here the multiplier ν_l can be seen as the implicit congestion price [30] of link l, which represents the cost of delivering a unit of data through link l. The multiplier μ_f can be interpreted as the implicit delay price of flow f, which represents the cost of imposing a unit of delay on flow f. If $\boldsymbol{\nu}$ and $\boldsymbol{\mu}$ are given, we can obtain the maximizers z_h^* and σ_l^* by taking the derivative with respect to z_h and σ_l respectively.

$$
z_h^*(\kappa_h) = U_h^{'-1}(\kappa_h), \text{ with } \kappa_h = \sum_{l\in L(f)\&f\in F(h)} \nu_l, \forall h \in H \qquad (32)
$$

$$
\sigma_l^*(\lambda_l,\nu_l) = \varphi_l^{'-1}(\frac{-\nu_l}{\lambda_l}), \text{ with } \lambda_l = \sum_{f\in F(l)} \mu_f, \forall l \in L \qquad (33)
$$

(32) implies that the optimal sampling rate of a control system h is determined by its price κ_h, which is the aggregated price of the link along its flow routes. (33) implies that the optimal $VLCM$ of a link is relevant to its congestion price ν_l and link margin price λ_l. The intuition is: 1) the congestion price determines the available capacity margin that can be used for $VLCM$ adjustment; and 2) the link margin price implicitly reflects the overall delay requirement (from all of its supporting flow delay requirement) on its $VLCM$. The maximizer $\hat{c}_l^*$ can be generated from a maximum weight based scheduling policy, which will be discussed later.

Now $W1$ is converted into three sub-problems: the sampling rate adaptation problem (29), the *VLCM* assignment problem (30) and the scheduling problem (31). The link congestion price $\boldsymbol{\nu}$ and the flow delay price $\boldsymbol{\mu}$ can be computed iteratively, from the opposite direction to the gradient $\nabla(L(\boldsymbol{\nu},\boldsymbol{\mu}))$ [31].

This adaptation approach is called *double-price* scheme. Based on the information of two price signals, the algorithm iteratively reach a global optimum. The property of this algorithm is formally characterized in Proposition 1 and Proposition 2.

Proposition 1 There is no duality gap between (23) and (27). For any $(\boldsymbol{\nu}^*,\boldsymbol{\mu}^*)$ that minimizes (28), if $(\boldsymbol{z}^*,\boldsymbol{\sigma}^*,\hat{\boldsymbol{c}}^*)$ solves (29), then $(\boldsymbol{z}^*,\boldsymbol{\sigma}^*,\hat{\boldsymbol{c}}^*)$ is the unique maximizer of (19).

Proposition 2 If $||\boldsymbol{\beta}||_2$ and $||\boldsymbol{\gamma}||_2$ are sufficiently small, starting from any initial values $\boldsymbol{z}(0)$, $\boldsymbol{\sigma}(0)$, $\hat{\boldsymbol{c}}(0)$ and prices $\boldsymbol{\nu}(0) \geq 0$, $\boldsymbol{\mu}(0) \geq 0$, the cross-layer algorithm converges to the optimal solution $(\boldsymbol{z}^*,\boldsymbol{\sigma}^*,\hat{\boldsymbol{c}}^*,\boldsymbol{\nu}^*,\boldsymbol{\mu}^*)$. [4].

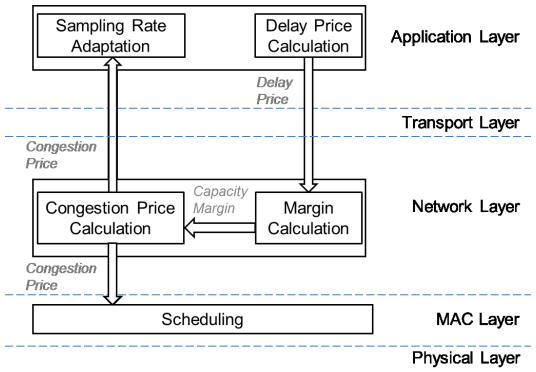

Fig. 6. NCS over multi-hop wireless networks

2) Cross-Layer Rate Allocation Implementation: Our algorithm naturally leads to a cross-layer implementation via joint *VLCM* assignment, sampling rate adaptation and scheduling, as shown in Fig. 6. Scheduling is performed at the MAC layer. At the network layer, the margin calculation generates the optimal *VLCM*s for a wireless interface queue; the congestion price calculation provides per-hop congestion price, which reflects the level of congestion at this queue. They can be implemented as part of the queue management mechanism. At the application layer, the per-hop congestion price is aggregated to calculate the sampling rate; the end-to-end delay is measured to calculate the delay price.

Our algorithm implementation only requires the knowledge of the first order derivative of the link delay with respect to the capacity margin $\frac{\partial\varphi(\sigma_l)}{\sigma_l}$, based on (33), rather than some statistical characteristics, like mean or variance of the packet arrival rate. The derivative of link delay can be profiled online. According to (31), we need to find a scheduling policy so that the aggregate link weight $\sum_{l\in L} z_l\hat{c}_l$ could be maximized. We achieve this by using a maximum matching based scheduling policy [30].

[4]Due to spacelimitations, the proof of these two propositions are provided in our report [32]

C. Solution with Delay Bound Tuning

With the optimal sampling rate adaptation solution to the problem $W1$ with fixed delay requirement obtained, we now solve the original optimal problem W by determining the optimal delay requirements for all control systems. We proceed in two steps. First we will determine the ranges of delay requirements. Then, we adjust the delay requirements to find the optimal ones which yield the optimal sampling rate allocation within the range.

1) Determine range of delay requirement: The lower bound $\boldsymbol{D}^l = (D_h, h \in H)$ of the delay range can be computed via the optimization problem of

$$W^l: \quad \max \quad \sum_{h \in H} U_h(z_h) \tag{34}$$

$$\text{s.t.} \quad \sum_{h \in H: f \in F(l) \& f \in F(h)} z_h \leq \hat{c}_l, \forall l \in L \tag{35}$$

$$\text{over} \quad \hat{\boldsymbol{c}} \in \Lambda \tag{36}$$

This is a simplified form of W, with the $VLCM$ $\sigma_l = 0$ and without the delay constraints. The solution of this problem $\boldsymbol{z}^l$ is the maximum achievable sampling rate considering only the network capacity constraint. This maximum achievable rate corresponds to the minimum sampling time of the control system $T_h(z_h^l)$. As our delay constraint in the original problem W is that the flow delay should not exceed one sampling time, we can treat the minimum sampling time as the lower delay bound $\boldsymbol{D}^l = (T_h(z_h^l), h \in H)$.

With the sampling rate of $\boldsymbol{z}^l$, the maximum amount of traffic with only network capacity constraint is injected into the network. Thus the measured end-to-end delay $\boldsymbol{d}^l = (d_f^l, f \in F)$ is the upper bound of the end-to-end delay. If $d_f^l \leq T_h(z_h^l), \forall f \in F(h), \forall h \in H$, then $\boldsymbol{z}^l$ will also be the optimal sampling rate for the original problem W. If there exists $d_f^l > T_h(z_h^l)$, then we set the upper bound of the delay requirement to $\boldsymbol{D}^u = \boldsymbol{d}^l$.

2) Optimal delay requirement adjustment: Starting from the lower bound of the delay requirement, we adjust the delay requirement of each control system based on the algorithm shown in Table I. In the algorithm, we gradually increase the delay requirement of each system h from its lower bound until 1) it is smaller than the corresponding optimal sampling time based on problem $W1$ but within a constant bound ϵ; or 2) it exceeds the corresponding optimal sampling time. In the latter case, we restore the delay requirement to its last value and reduce the adjustment size from a_h/m_h to $a_h/(m_h+1)$, where m_h is initialized to 1.

TABLE I
DELAY REQUIREMENT ADJUSTMENT

Adjustment of Delay Requirement D_h
0) *initialization*
$\forall h, m_h = 1, D_h = D_h^l$
let a_h be the initial adjustment size, ϵ be a sufficiently small constant;
1) compute z_h by solving $W1$ where the delay requirements are D_h;
derive the corresponding sampling time $T_h(z_h)$;
If $\forall h, 0 \leq T_h(z_h) - D_h \leq \epsilon$, stop;
$(D_h, h \in H)$ is the optimal delay requirements.
2) If $\exists h, D_h < T_h(z_h) - \epsilon$, increase its delay requirement:
$D_h = D_h + a_h/m_h$
3) If $\exists h, D_h > T_h(z_h)$, decrease the delay requirement:
$D_h = D_h - a_h/m_h$
and reduce the adjustment size:
$m_h = m_h + 1$
repeat 1) to 3)

VI. PERFORMANCE EVALUATION

In this section, we evaluate our cross-layer sampling rate adaptation and network scheduling algorithm using an integrated simulation tool named Networked Control System Wind-Tunnel (NCSWT) [12]. NCSWT integrates two simulators Matlab and ns-2. Based on the HLA standard, the tool allows us to simulate the control system models in Matlab/Simlink and the networking systems in ns-2. Using ns-2, a packet-level network simulator that implements all the details of the network protocol stack, we can perform highly accurate evaluation of the network effects on the NCS performance, including queueing delays and network scheduling, which is impossible by using Matlab/Simulink alone.

A. Simulation Setup

In our experiments, the control systems consist of three pairs of plants and controllers. Each of the three plant systems used in the experiments is the model of a single joint of a robotic arm. They are described by the continuous time state space representation as defined in (1) and (2), with the parameters $A_p = 0, B_p = 1, C_p = 0.3413$. Each of the controllers can be defined by the continuous time state space representation as defined in (3) and (4) with the parameters $A_c = 0, B_c = 1, C_c = 32.1, D_c = 8.2$. The plants and controllers are discretized based on the sampling interval, T_h defined by the network, to obtain the discrete time equivalent of the systems for each sampling interval. The utility function used in the experiments is the same as the function presented in Section III. The objective of the systems is for the joint velocity of each robotic arm to track a sinusoidal reference input $r[k] = sin(\omega k)$ for $k = 0, 1, 2, \cdots$ with $\omega = \frac{2\pi}{80}$. The disturbance inputs for Plant2 and Plant3 are white noise with the power spectral density of 1. For comparison, Plant 1 does not have any white noise input. In the wireless network, the interference range and the transmission range are set to 250m. The capacity of the wireless channel is $2Mbps$. The packet size is 260 bytes. Each simulation runs for 180 seconds.

Four aspects of the system are evaluated after the first period of the reference signal when the adaptation converges:

1) The average tracking error $\bar{e}_{rr}$, which is the average absolute difference between the plant output and its reference signal.
2) The converged sampling time T_h.
3) The end-to-end delays of the flows associated with system h.
4) The channel utilization, which is the ratio of the total network load to the channel capacity.

The first two metrics evaluate the performance of the NCS, and the last two evaluate the performance of the networking system.

B. Simulation Results

1) Single-hop Scenario: In the first experiment, there are six nodes in the wireless network, each hosting either a plant or a controller. All the nodes are within the transmission range of each other, forming a single-hop network topology.

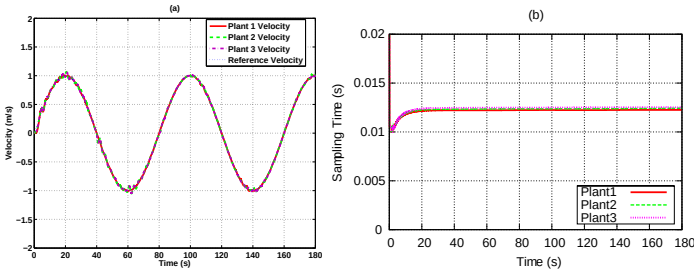

Fig. 7. Velocity Outputs with the Optimal Delay Requirement in Single-hop

Fig. 7 presents the simulation results with the optimal delay requirement derived from the delay requirement adjustment algorithm. Fig. 7(a) shows the plant outputs, and Fig. 7(b) illustrates the sampling time convergence of the three plant-controller pairs. The sampling time quickly converges, and the plant outputs closely follow the reference trajectory. In Plant2 and Plant3, white noise appears every 15 seconds. The outputs deviate from the reference trajectory, while their differences quickly diminish after a short period of time.

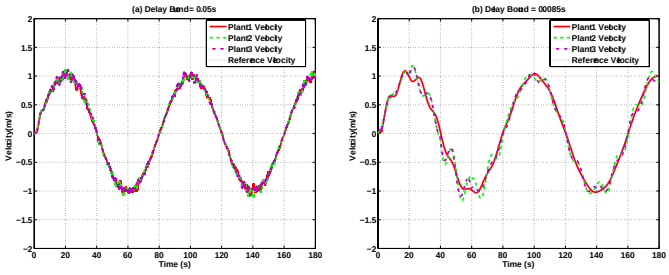

Fig. 8. Velocity Outputs with Different Delay Requirements in Single-hop

TABLE II
PERFORMANCE METRICS FOR DIFFERENT DELAY REQUIREMENTS

Delay Requirements	Average Track Error	Sampling Time	Average Delay	Channel Utilization
Optimal	0.0069	0.0125	0.0121	50 %
0.05	0.0251	0.0087	0.0481	72 %
0.0085	0.0428	0.2013	0.0087	3 %

Next we compare the performance of the NCS with fixed delay requirements, which are different from the optimal one. Tab. II lists the values of the performance metrics. Fig. 8(a) illustrates the plant outputs using the delay requirement of $0.05s$, and Fig. 8(b) demonstrates the plant outputs using the delay requirement of $0.0085s$. We observe that both outputs are much worse than those in Fig. 7. In addition in Fig. 8(b), Plant2 and Plant3 suffer from larger oscillations than Plant1, and cannot track the reference trajectory closely. With a larger delay requirement, the control systems are allowed to send packets with a larger sampling rate, which increases the traffic load of the networks. The average end-to-end delay experienced by the control systems is more than 5 times of the sampling time. Thus, the outputs exhibit a lot of oscillation. On the other hand, a small delay requirement leads to small sampling rates, which degrades the system capability of white noise rejection. Thus, the controller cannot be notified in time about the occurrence of the white noise disturbance.

2) Multi-hop Scenario: Direct communication in wireless networks requires two nodes within the transmission range of each other. When they are out of range, intermediate nodes can provide relays to route packets. We evaluate our solution over a multi-hop wireless network with 12 nodes organized in a grid topology. The plants and controllers are deployed on nodes at the network edges. Plant2 resides in the middle of the network. The paths of all the control system pairs are set up using the shortest-path routing algorithm.

TABLE III
PERFORMANCE METRICS WITH OPTIMAL REQUIREMENTS

	Delay Requirements	Average Track Error	Sampling Time	Average Delay
Plant1	0.024	0.0020	0.026	0.031
Plant2	0.035	0.0186	0.038	0.048
Plant3	0.024	0.0120	0.026	0.023

Fig. 9(a) shows the velocity outputs of the three plants with the optimal delay requirements, and Tab. III compares their performance metrics. Compared with the single hop case, the plants experience larger oscillation at the beginning of the simulation. Because it takes longer time to setup the routes between the plant and controller pairs. Plant1 does not have white noise disturbance, so after convergence its velocity output exactly follows the reference signal. Although Plant2 and Plant3 have the same amount of white noise input, Plant2 has larger oscillation than Plant3. As the flow of Plant2 experiences more interference than those of Plant3, it has a larger delay requirement and is more vulnerable to noise with a larger sampling time.

TABLE IV
PERFORMANCE METRICS OF THE NCS WITH FIXED RATES

	Average Track Error	Sampling Time	Average Delay
Plant1	0.0132	0.0117	0.0807
Plant2	0.0223	0.0176	0.2044
Plant3	0.0139	0.0117	0.0519

We further run the experiment with fixed sampling time for the three control systems. In Tab. IV, we compare the average tracking error with fixed sampling time and that with the optimal delay requirements. We observe that the three plants experience larger tracking error. Their sampling rates are about twice of those under the optimal delay requirements, which leads to much longer delay in a multi-hop network. When

the average delay exceeds the sampling time by a number of times, the tracking error increases significantly.

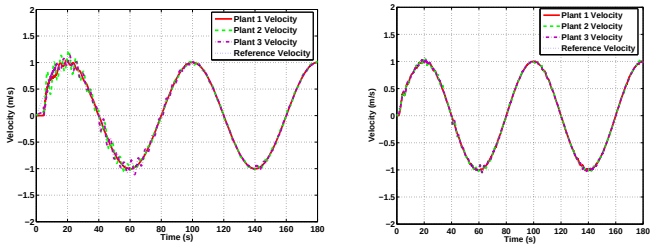

Fig. 9. (a) Multi-hop Wireless Network (b) Single-hop wireless network with 10% packet loss

3) With wireless random packet loss: We now set up a single-hop wireless network with 10% random packet loss that may be caused by wireless interference or noise. In Fig. 9(b), we show the velocity outputs of the three plants with the optimal delay requirements. Compared with the no loss case, the plant outputs experience slightly larger oscillation, but still are able to track the reference closely.

VII. Conclusion

This paper investigates the problem of NCS performance optimization in terms of tracking error minimization. It presents an optimization formulation where the objective is to maximize a utility function that characterizes the relationship between the sampling rate and the capability of disturbance rejection of the control system and the constraints come from the wireless network capacity and the requirement of packet. A distributed double-price-based algorithm is presented to solve the problem. Our solution has desired properties from both theoretical and practical aspects. From theoretical perspective, it is shown to achieve both system stability and performance optimality. From the view of practice, it can be naturally deployed over the existing layered networking systems with well-defined cross-layer interactions. Simulation studies conducted in an integrated simulation environment consisting of Matlab/Simulink and ns-2 demonstrate that our algorithm is able to provide agile and stable sampling rate adaptation and achieve optimal NCS performance.

References

[1] R. Gupta and M.-Y. Chow, "Networked control system: Overview and research trends," *Industrial Electronics, IEEE Trans. on*, vol. 57, no. 7, pp. 2527–2535, July 2010.

[2] G. Irwin, J. Colandairaj, and W. Scanlon, "An overview of wireless networks in control and monitoring," in *Computational Intelligence*, ser. Lecture Notes in Computer Science, D.-S. Huang, K. Li, and G. Irwin, Eds. Springer Berlin / Heidelberg, 2006, vol. 4114, pp. 1061–1072.

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

[4] L. Montestruque and P. Antsaklis, "Stability of model-based networked control systems with time-varying transmission times," *Automatic Control, IEEE Trans. on*, vol. 49, no. 9, pp. 1562–1572, Sep. 2004.

[5] M. Branicky, S. Phillips, and W. Zhang, "Stability of networked control systems: Explicit analysis of delay," in *Proc. of the American Control Conference*, 2000, pp. 2352–2357.

[6] M. S. Branicky, S. M. Phillips, and W. Zhang, "Scheduling and feedback co-design for networked control systems," in *Decision and Control, Proc. of the 41st IEEE Conf. on*, vol. 2, Dec. 2002, pp. 1211–1217.

[7] X. Liu and A. Goldsmith, "Wireless network design for distributed control," in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 3, 2004, pp. 2823 – 2829.

[8] U. Ojha and M. Chow, "Behavioral control based adaptive bandwidth allocation in a system of unmanned ground vehicles," in *IECON*, Nov. 2010, pp. 3123–3128.

[9] F.-L. Lian, J. Yook, D. Tilbury, and J. Moyne, "Network architecture and communication modules for guaranteeing acceptable control and communication performance for networked multi-agent systems," *Industrial Informatics, IEEE Transactions on*, vol. 2, no. 1, pp. 12 – 24, feb. 2006.

[10] A. Al-Hammouri, M. Branicky, V. Liberatore, and S. Phillips, "Decentralized and dynamic bandwidth allocation in networked control systems," in *IPDPS*, Apr. 2006, p. 8.

[11] J. Colandairaj, G. W. Irwin, and W. G. Scanlon, "Wireless networked control systems with qos-based sampling," *Control Theory Applications, IET*, vol. 1, no. 1, pp. 430–438, january 2007.

[12] D. Riley, E. Eyisi, J. Bai, X. Koutsoukos, Y. Xue, and J. Sztipanovits, "Networked Control System Wind Tunnel (NCSWT) - An evaluation tool for networked multi-agent systems," in *Proc. of SIMUTools*, 2011.

[13] M. Lemmon and X. S. Hu, "Almost sure stability of networked control systems under exponentially bounded bursts of dropouts," in *Proceedings of the 14th international conference on Hybrid systems: computation and control*, 2011, pp. 301–310.

[14] M. Pajic, S. Sundaram, G. Pappas, and R. Mangharam, "The wireless control network: A new approach for control over networks," *Automatic Control, IEEE Transactions on*, vol. 56, no. 10, pp. 2305 –2318, 2011.

[15] A. Saifullah, C. Wu, P. Tiwari, Y. Xu, Y. Fu, C. Lu, and Y. Chen, "Near optimal rate selection for wireless control systems," in *IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'12)*, April 2012.

[16] D. Seto, J. Lehoczky, L. Sha, and K. Shin, "On task schedulability in real-time control systems," in *Real-Time Systems Symposium, 1996., 17th IEEE*, 1996, pp. 13 –21.

[17] L. Bao, M. Skoglund, C. Fischione, and K. Johansson, "Rate allocation for quantized control over noisy channels," in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, 2009. WiOPT 2009. 7th International Symposium on*, 2009, pp. 1 –9.

[18] Z. Li and M.-Y. Chow, "Adaptive multiple sampling rate scheduling of real-time networked supervisory control system - part ii," in *IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference on*, nov. 2006, pp. 4615 –4620.

[19] A. Kelkar and S. Joshi, "Robust control of non-passive systems via passification [for passification read passivation]," in *American Control Conference, 1997. Proceedings of the 1997*, vol. 5, 1997, pp. 2657 – 2661.

[20] N. Kottenstette, J. Hall, X. koutsoukos, J. Sztipanovits, and P. Antsaklis, "Design of networked control systems using passivity," *IEEE Trans. Control Systems Technlology (Accepted for publication)*, 2012.

[21] N. Kottenstette, X. Koutsoukos, J. Hall, J. Sztipanovits, and P. Antsaklis, "Passivity-based design of wireless networked control systems for robustness to time-varying delays," in *Proc. of RTSS*, 2008, pp. 15–24.

[22] S. Stramigioli, C. Secchi, A. van der Schaft, and C. Fantuzzi, "A novel theory for sampled data system passivity," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 2, 2002, pp. 1936 – 1941 vol.2.

[23] N. Kottenstette, "Control of passive plants with memoryless nonlinearitites over wireless networks," Ph.D. dissertation, University of Notre Dame, 2007.

[24] G. F. Franklin, J. D. Powel, and M. L. Workman, *Digital Control of Dynamic Systems*. Addison Wesley, 1997.

[25] W. S. Levine, *Control System Funcamentals*. CRC Press, 2000.

[26] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," *Wirel. Netw.*, vol. 11, pp. 471–487, July 2005.

[27] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, april 2006, pp. 1 –13.

[28] G. R. Gupta and N. B. Shroff, "Delay analysis and optimality of scheduling policies for multihop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 19, pp. 129–141, February 2011.

[29] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *The Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237–252, 1998.

[30] S. Low and D. Lapsley, "Optimization flow control. i. basic algorithm and convergence," *Networking, IEEE/ACM Transactions on*, vol. 7, no. 6, pp. 861 –874, dec. 1999.

[31] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1989.

[32] F. Qiu, J. Bai, and Y. Xue, "Optimal rate allocation in multi-hop wireless networks with delay constraints," Vanderbilt University, Tech. Rep., 2011.